Change Detection Using Directional Statistics

Tsuyoshi Idé **Dzung T. Phan** **Jayant Kalagnanam**
IBM Research, T. J. Watson Research Center
1101 Kitchawan Rd., Yorktown Heights, NY 10598, USA
{tide,phandu,jayant}@us.ibm.com

Abstract

This paper addresses the task of change detection from noisy multivariate time-series data. One major feature of our approach is to leverage directional statistics as the noise-robust signature of time-series data. To capture major patterns, we introduce a regularized maximum likelihood equation for the von Mises-Fisher distribution, which simultaneously learns directional statistics and sample weights to filter out unwanted samples contaminated by the noise. We show that the optimization problem is reduced to the trust region subproblem in a certain limit, where global optimality is guaranteed. To evaluate the amount of changes, we introduce a novel distance measure on the Stiefel manifold. The method is validated with real-world data from an ore mining system.[1]

1 Introduction

The problem we wish to solve is change detection of multivariate time-series data. Figure 1 shows a typical setting, where our task is to compute the degree of change, or the *change score*, of the data within the test window taken at time t in comparison to the training window.

The task of change detection has a long history in statistics. The standard strategy is to use a parametric model for probability density and compute the likelihood ratio to quantify the degree of change between fitted distributions [Chen and Gupta, 2012]. For a concise review from a statistical machine learning perspective, see [Yamada *et al.*, 2013].

When applying a change detection method to real-world problems, the major requirements are interpretability and robustness to nuisance noise variables. To validate detected changes with domain knowledge, it is almost always required to explicitly present statistics (or *feature*) for the parametric model, such as the mean for Gaussian. Although it is possible to design an algorithm that skips the explicit step of feature extraction and jumps directly into score calculation [Liu *et al.*, 2013], such an approach is not highly appreciated in practice due to the lack of interpretability. In the multivariate setting, the robustness to noise variables is the other critical requirements since changes may not always occur in all of the variables simultaneously. In fact, under the existence of nuisance noise variables, the performance of direct density-ratio estimation approaches is known to significantly degrade [Yamada *et al.*, 2013].

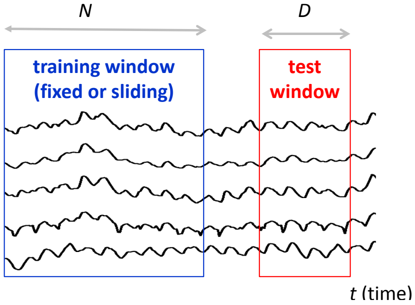

Figure 1: Change detection problem.

Considering these two requirements, we focus on change detection approaches having explicit two steps of feature extraction and score calculation. There are two important decision points here: (1) Parametric model for probability density function, and (2) scoring model for the change score.

In this paper, we propose a novel framework for change detection for multivariate sensor data. Our contribution is threefold:

- We develop a novel feature extraction method based on regularized maximum likelihood of the von Mises-Fisher (vMF) distribution.
- We show that the feature extraction method is reduced to an optimization problem called the trust-region subproblem [Tao and An, 1998; Hager, 2001].
- We propose a novel scoring method based on a parametrized Kullback-Leibler (KL) divergence.

Implications of these contributions are as follows. *First*, our feature extraction method is the first proposal to efficiently remove the multiplicative noise that is ubiquitous in many physical systems. Thanks to an ℓ_1 regularization scheme, it is also capable of automatically removing samples

[1] The final version with corrections of T. Idé *et al.*, "Change Detection using Directional Statistics," Proceedings of the Twenty-Fifrth International Joint Conference on Artificial Intelligence (IJCAI 16, July 9-15, 2016, New York, USA), pp.1613-1619.

contaminated by the unwanted noise. *Second*, the trust-region subproblem guarantees the global optimality in a certain limit, which is especially important for noisy data. *Third*, the parametrized Kullback-Leibler divergence for scoring provides us with a trustworthy way to quantify the discrepancy between different subspaces with different dimensionalities.

2 Extracting feature matrix from noisy data

The proposed method consists of two steps. The first step computes an orthonormal matrix as the signature of the fluctuation patterns of multivariate data. The second step computes the difference between two data sets in the past and present through the computed orthonormal matrices. This section explains the first step.

2.1 The von Mises-Fisher distribution

Assume we are monitoring a system with M sensors, and we are given N measurements $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)} \in \mathbb{R}^M\}$, which may correspond to either the training or the test window in Fig. 1. Our principal probabilistic model is the von Mises-Fisher (vMF) distribution [Mardia *et al.*, 1980]:

$$\mathcal{M}(\boldsymbol{z} \mid \boldsymbol{u}, \kappa) \equiv c_M(\kappa) \exp\left(\kappa \boldsymbol{u}^\top \boldsymbol{z}\right) \quad (1)$$

$$c_M(\kappa) \equiv \frac{\kappa^{M/2-1}}{(2\pi)^{M/2} I_{M/2-1}(\kappa)}, \quad (2)$$

where $I_{M/2-1}(\cdot)$ denotes the modified Bessel function of the first kind with the order $\frac{M}{2} - 1$. The random variable $\boldsymbol{z}$ is assumed to be normalized in the sense $\boldsymbol{z}^\top \boldsymbol{z} = 1$, where $^\top$ represents transpose. The vMF distribution has two parameters: the mean direction $\boldsymbol{u}$ and the concentration parameter κ. As these names suggest, the vMF distribution describes random variability of the *direction* around the mean vector.

The intuition behind the use of vMF distribution is as follows. When describing the measurements using the vMF distribution, we look only at the direction, disregarding fluctuations along the direction. This automatically gives the robustness to *multiplicative* noise, which is quite common in practice in systems with redundancy. As an example, we will look at an ore transfer system, where two belt conveyors are operated by the same electronic system. In this system, major fluctuations in one system due to e.g. dynamic load changes are shared by the other system. In systems having strongly correlated variables, the vMF distribution can be a more natural tool for system monitoring than the Gaussian.

2.2 Weighted joint maximum likelihood

To simplify the notation, we introduce the data matrix

$$\mathsf{X} \equiv [\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}] = [b^{(1)}\boldsymbol{z}^{(1)}, \ldots, b^{(N)}\boldsymbol{z}^{(N)}], \quad (3)$$

where $\|\boldsymbol{z}^{(n)}\|_2 = 1$ and $b^{(n)} \equiv \|\boldsymbol{x}^{(n)}\|_2$ for $n = 1, \ldots, N$ and $\|\cdot\|_p$ being the p-norm. The parameters of the vMF distribution may be inferred by maximizing a likelihood function:

$$L(\boldsymbol{u}, \boldsymbol{w}|\mathsf{X}) \equiv \sum_{n=1}^{N} w^{(n)} b^{(n)} \ln \mathcal{M}(\boldsymbol{z}^{(n)}|\boldsymbol{u}, \kappa), \quad (4)$$

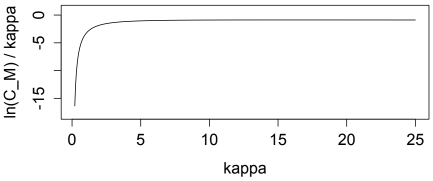

Figure 2: γ/κ as a function of κ $(M = 10)$.

where we introduce sample weights $\boldsymbol{w} \equiv (w^{(1)}, \ldots, w^{(N)})^\top$. We wish to capture major patterns represented as the directional data by optimally choosing the weights so that less trustworthy samples are down-weighted.

The weighted likelihood $L(\boldsymbol{u}, \boldsymbol{w}|\mathsf{X})$ has only a single pattern $\boldsymbol{u}$, and naively maximizing $L(\boldsymbol{u}, \boldsymbol{w}|\mathsf{X})$ produces only the single direction. To capture multiple patterns of the change, we jointly fit m different distributions while keeping the overlap minimal by imposing the orthogonality between different patterns:

$$\max_{\mathsf{U},\mathsf{W}} \sum_{i=1}^{m} \{L(\boldsymbol{u}_i, \kappa, \boldsymbol{w}_i|\mathsf{X}) - R(\boldsymbol{w}_i)\} \text{ s.t. } \mathsf{U}^\top\mathsf{U} = \mathsf{I}_m, \quad (5)$$

where I_m is the m-dimensional identity matrix, $\mathsf{U} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$, and $\mathsf{W} \equiv [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m]$. The term $R(\boldsymbol{w}_i)$ is a regularizer to remove the trivial solution on the sample weights. Here we consider the elastic net regularization [Zou and Hastie, 2005]:

$$R(\boldsymbol{w}_i) = \lambda \left(\frac{1}{2}\|\boldsymbol{w}_i\|_2^2 + \nu\|\boldsymbol{w}_i\|_1 \right), \quad (6)$$

where λ and ν are given constants, typically determined by cross-validation.

The first term of the objective function is given by

$$\sum_{i=1}^{m} L(\boldsymbol{u}_i, \kappa, \boldsymbol{w}_i|\mathsf{X}) = \kappa \mathrm{Tr}\left(\mathsf{XWU}^\top\right) + \gamma \mathbf{1}^\top \mathsf{W}^\top \boldsymbol{b}, \quad (7)$$

where we defined $\gamma \equiv \ln c_M(\kappa)$, $\boldsymbol{b} \equiv [b^{(1)}, \ldots, b^{(N)}]^\top$, and $\mathbf{1}$ is a column vector of all ones. Throughout the paper we will treat κ as a given constant. Fortunately, γ/κ is quite insensitive to κ, as shown in Fig. 2. One useful heuristic is simply to set $\kappa = M$. Otherwise, we can use known approximation algorithms given in [Sra, 2012].

2.3 Iterative sequential algorithm

The optimization problem (5) can be sequentially solved for each of $(\boldsymbol{u}_i, \boldsymbol{w}_i)$. Imagine that the sample weight $\boldsymbol{w}$ is initialized to a vector. Given $\boldsymbol{w}$, Eq. (5) is reduced to

$$\max_{\boldsymbol{u}} \left\{\kappa \boldsymbol{u}^\top \mathsf{X} \boldsymbol{w}\right\} \quad \text{s.t. } \boldsymbol{u}^\top \boldsymbol{u} = 1 \quad (8)$$

for the first $\boldsymbol{u}$. The solution is readily obtained as

$$\boldsymbol{u} = \frac{\mathsf{X}\boldsymbol{w}}{\|\mathsf{X}\boldsymbol{w}\|_2}. \quad (9)$$

Given this solution, the problem (5) for $\boldsymbol{w}$ is now written as

$$\begin{aligned}\arg\max_{\boldsymbol{w}} & \left\{ \boldsymbol{w}^\top \boldsymbol{q} - \frac{\lambda}{2}\boldsymbol{w}^\top \boldsymbol{w} - \lambda\nu \|\boldsymbol{w}\|_1 \right\} \\ &= \arg\min_{\boldsymbol{w}} \left\{ \frac{1}{2}\|\boldsymbol{w} - \frac{\boldsymbol{q}}{\lambda}\|_2^2 + \nu\|\boldsymbol{w}\|_1 \right\}, \qquad (10)\end{aligned}$$

where $\boldsymbol{q}$ is defined by

$$\boldsymbol{q} \equiv \gamma \boldsymbol{b} + \kappa \mathsf{X}^\top \boldsymbol{u}. \qquad (11)$$

This problem is a special case of LASSO regression, and has a closed-form solution [Wen *et al.*, 2010] as

$$\boldsymbol{w} = \mathrm{sign}(\boldsymbol{q}) \odot \max\left\{ \frac{|\boldsymbol{q}|}{\lambda} - \nu\mathbf{1}, \mathbf{0} \right\}, \qquad (12)$$

where $\odot$ denotes the componentwise product, $\mathbf{0}$ is the zero vector, and $|\boldsymbol{q}|$ is an N-dimensional vector whose n-th entry is $|q_n|$. With this new $\boldsymbol{w}$, we can solve Eq. (8) again. We repeat solving Eqs. (8) and (10) alternatingly until convergence.

Once we get the first solution $(\boldsymbol{u}_1, \boldsymbol{w}_1)$, we move on to the next solution. We again start with initialized $\boldsymbol{w}$ and solve the following problem instead of Eq. (8):

$$\max_{\boldsymbol{u}} \left\{ \kappa \boldsymbol{u}^\top \mathsf{X} \boldsymbol{w} \right\} \quad \text{s.t.} \ \ \boldsymbol{u}^\top \boldsymbol{u} = 1, \ \boldsymbol{u}_1^\top \boldsymbol{u} = 0 \ . \qquad (13)$$

By introducing Lagrange multipliers α, β_1 for the two constraints, respectively, we have the condition of optimality as

$$\begin{aligned}\mathbf{0} &= \frac{\partial}{\partial \boldsymbol{u}} \left[\kappa \boldsymbol{u}^\top \mathsf{X} \boldsymbol{w} - \frac{\alpha}{2} \boldsymbol{u}^\top \boldsymbol{u} - \beta_1 \boldsymbol{u}^\top \boldsymbol{u}_1 \right] \\ &= \kappa \mathsf{X} \boldsymbol{w} - \alpha \boldsymbol{u} - \beta_1 \boldsymbol{u}_1. \qquad (14)\end{aligned}$$

Using the constraints, the candidates for the solution are

$$\boldsymbol{u} \leftarrow \kappa(\mathsf{X}\boldsymbol{w} - \boldsymbol{u}_1 \boldsymbol{u}_1^\top \mathsf{X} \boldsymbol{w}) \qquad (15)$$

$$\boldsymbol{u} \leftarrow \pm \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|_2} \qquad (16)$$

By plugging these two stationary points into the objective function in Eq. (13), we can get the maximizer $\boldsymbol{u}^*$. This solution is inserted into Eq. (10) to get a new $\boldsymbol{w}$. These steps are repeated until convergence.

It is straightforward to generalize the above procedure for $j = 2, \ldots, m$. We summarize the procedure in Algorithm 1, which we call the REgularized Directional feature extraction (RED) algorithm:

In Eq. (17), we define $\mathsf{U}_{j-1} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{j-1}]$. I_M is the M-dimensional identity matrix. The complexity of RED algorithm for each while loop iteration is $O(NM)$.

3 Theoretical analysis

3.1 Convergence of RED algorithm

We prove the following theorem:

Theorem 1. *Define* $\tilde{\mathsf{X}} \equiv [\mathsf{I}_M - \mathsf{U}_{j-1}\mathsf{U}_{j-1}^\top]\mathsf{X}$ *and* $g(\boldsymbol{w}_j, \boldsymbol{u}_j) \equiv \kappa \boldsymbol{u}_j^\top \tilde{\mathsf{X}} \boldsymbol{w}_j + \gamma \boldsymbol{w}_j^\top \boldsymbol{b} - R(\boldsymbol{w}_j)$ *in the notation of Algorithm 1. For a fixed* U_{j-1}*, the sequence* $\{g(\boldsymbol{w}_j^t, \boldsymbol{u}_j^t)\}_{t=0,1,\ldots}$ *generated Eqs.* (17)-(20) *in the* j*-th while-loop has a finite limit.*

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.
```

Proof. First, we note that $g(\boldsymbol{w}_j, \boldsymbol{u}_j)$ is the objective function to be maximized for each j under the constraint $\|\boldsymbol{u}_j\|_2 = 1$. The sequence $\{g(\boldsymbol{w}_j^t, \boldsymbol{u}_j^t)\}$ is bounded above since

$$\begin{aligned} g(\boldsymbol{w}_j, \boldsymbol{u}_j) &\leq \kappa \boldsymbol{u}_j^\top \tilde{\mathsf{X}} \boldsymbol{w}_j + \gamma \boldsymbol{w}_j^\top \boldsymbol{b} - \frac{\lambda}{2}\|\boldsymbol{w}_j\|_2^2 \qquad (21) \\ &= -\frac{\lambda}{2} \left\| \boldsymbol{w}_j - \frac{\kappa \tilde{\mathsf{X}}^\top \boldsymbol{u}_j + \gamma \boldsymbol{b}}{\lambda} \right\|_2^2 + \frac{\|\kappa \tilde{\mathsf{X}}^\top \boldsymbol{u}_j + \gamma \boldsymbol{b}\|_2^2}{2\lambda} \\ &\leq \frac{1}{2\lambda} \left\{ \|\kappa \tilde{\mathsf{X}}^\top \boldsymbol{u}_j\|_2^2 + \|\gamma \boldsymbol{b}\|_2^2 \right\} \\ &\leq \frac{1}{2\lambda} \left\{ \kappa^2 \sigma_{\max}(\tilde{\mathsf{X}}\tilde{\mathsf{X}}^\top) + \|\gamma \boldsymbol{b}\|_2^2 \right\}, \qquad (22) \end{aligned}$$

where $\sigma_{\max}$ is the nonnegative maximum eigenvalue of $\tilde{\mathsf{X}}\tilde{\mathsf{X}}^\top$. The inequality (21) is due to λ and ν being positive, the last inequality (22) is derived from $\|\boldsymbol{u}_j\|_2 = 1$.

For a j, we have

$$g(\boldsymbol{w}_j^t, \boldsymbol{u}_j^t) \leq g(\boldsymbol{w}_j^t, \boldsymbol{u}_j^{t+1}) \leq g(\boldsymbol{w}_j^{t+1}, \boldsymbol{u}_j^{t+1}),$$

where the first inequality comes from the fact that $\boldsymbol{u}_j^{t+1}$ is the maximizer given $\boldsymbol{w}_j^t$, and the last one is because $\boldsymbol{w}_j^{t+1}$ is the maximizer given $\boldsymbol{u}_j^{t+1}$. The boundedness and monotonically increasing of the sequence $\{g(\boldsymbol{w}_j^t, \boldsymbol{u}_j^t)\}$ complete the proof. □

3.2 Global optimality in $\nu \to 0$

Here we look at the following theorem:

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

$$\min_{\boldsymbol{u}} \left\{ \boldsymbol{u}^\top \mathsf{Q} \boldsymbol{u} + \boldsymbol{c}^\top \boldsymbol{u} \right\} \quad s.t. \ \ \boldsymbol{u}^\top \boldsymbol{u} = 1, \qquad (23)$$

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.

When $\nu = 0$, the objective function (5) leads to the optimality condition w.r.t. $\boldsymbol{w}$ as

$$\boldsymbol{0} = \frac{\partial}{\partial \boldsymbol{w}} \left\{ \boldsymbol{w}^\top \boldsymbol{q} - \frac{\lambda}{2} \boldsymbol{w}^\top \boldsymbol{w} \right\},$$

which immediately gives an analytic solution as $\boldsymbol{w} = \boldsymbol{q}/\lambda$. By inserting this solution into the objective function in Eq. (5), we have an optimization problem only for $\boldsymbol{u}$:

$$\max_{\boldsymbol{u}} \left\{ \kappa \boldsymbol{u}^\top \mathsf{X}\mathsf{X}^\top \boldsymbol{u} + 2\gamma \boldsymbol{u}^\top \mathsf{X} \boldsymbol{b} \right\} \quad \text{s.t.} \;\; \boldsymbol{u}^\top \boldsymbol{u} = 1,$$

which is obviously equivalent to Eq. (23) with $\mathsf{Q} \equiv \kappa \mathsf{X}\mathsf{X}^\top \in \mathbb{R}^{M \times M}$ and $\boldsymbol{c} \equiv 2\gamma \mathsf{X} \boldsymbol{b} \in \mathbb{R}^M$.

Let $\boldsymbol{u}_1$ be the solution of this problem. Once $\boldsymbol{u}_1$ is obtained, we again solve another trust-region subproblem of the form (23) but in the $\mathbb{R}^{M-1}$ space by adding an orthogonality condition $\boldsymbol{u}_1^\top \boldsymbol{u} = 0$. In particular, without loss of generality, we assume the last component of $\boldsymbol{u}_1$ is nonzero, i.e., $u_{1,M} \neq 0$. Define $\tilde{\boldsymbol{u}} \equiv (u_1, \ldots, u_{M-1})^\top$ using the first $M-1$ dimensions of $\boldsymbol{u}$, and $\boldsymbol{a} \equiv (u_{1,1}, \ldots, u_{1,M-1})^\top / u_{1,M}$ in place of $\boldsymbol{u}_1$. The problem (23) with the additional linear equality constraint is equivalent to

$$\min_{\tilde{\boldsymbol{u}}} \left\{ \left(\tilde{\boldsymbol{u}}^\top, -\boldsymbol{a}^\top \tilde{\boldsymbol{u}}\right) \mathsf{Q} \begin{pmatrix} \tilde{\boldsymbol{u}} \\ -\boldsymbol{a}^\top \tilde{\boldsymbol{u}} \end{pmatrix} + \sum_{i=1}^{M-1} c_i u_i - c_M \boldsymbol{a}^\top \tilde{\boldsymbol{u}} \right\}$$
$$\text{s.t.} \;\; \tilde{\boldsymbol{u}}^\top (\mathsf{I}_{M-1} + \boldsymbol{a}\boldsymbol{a}^\top) \tilde{\boldsymbol{u}} = 1. \tag{24}$$

Consider the Cholesky decomposition for rank-one update

$$\mathsf{I}_{M-1} + \boldsymbol{a}\boldsymbol{a}^\top = \mathsf{L}\mathsf{L}^\top$$

and the change of variable $\boldsymbol{y} \equiv \mathsf{L}^\top \tilde{\boldsymbol{u}}$. The decomposition can be done efficiently. See, e.g., [Gill *et al.*, 1974]. It is straightforward to see that the problem (24) is rewritten as

$$\min_{\boldsymbol{y} \in \mathbb{R}^{M-1}} \left\{ \boldsymbol{y}^\top \bar{\mathsf{Q}} \boldsymbol{y} + \bar{\boldsymbol{c}}^\top \boldsymbol{y} \right\} \quad \text{s.t.} \;\; \boldsymbol{y}^\top \boldsymbol{y} = 1$$

for a certain $(\bar{\mathsf{Q}}, \bar{\boldsymbol{c}})$, which has exactly the same format with (23).

When more than two orthonormal vectors $\boldsymbol{u}_i$ are needed, a set of linear equations is additionally considered

$$\boldsymbol{u}^\top \boldsymbol{u}_i = 0, \quad i = 1, \ldots, j-1.$$

A variable elimination method can be used in order to work on a reduced space. Note that an eigenvalue decomposition for a matrix of dimension $M-j$ is needed to transform it into a $(M-j)$-dimensional trust region subproblem. In this way, we have a set of orthonormal vectors $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m\}$, where m is an input parameter representing the number of major directional patterns. □

Although the global optimality is no longer guaranteed when $\nu > 0$, we can take advantage of the global solution at $\nu = 0$ to initialize $\boldsymbol{w}$ in Algorithm 1. In practice, if there is a concern about the quality of the solution, we can gradually increase the value of ν, and use the obtained $\{\boldsymbol{w}_j\}$'s for the next trial for a larger ν. Although mathematically the RED algorithm is an iterative algorithm that may be trapped by sub-optimality, we can loosely say that it is an "almost guaranteed" algorithm in practice.

4 Parameterized KL divergence for scoring

Solving the optimization problem for the training and the test windows, we obtain two sets of orthonormal vectors $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m\}$ and $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r\}$ where m and r are the number of vectors given as input parameters. Computing the change score amounts to evaluating the dissimilarity between the two vector spaces specified by orthonormal matrices

$$\mathsf{U} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m], \quad \mathsf{U}^{(t)} \equiv [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r]. \tag{26}$$

Now our problem is to compute the dissimilarity on a *Stiefel manifold*, which is the space spanned by orthogonal matrices.

As illustrated in Fig. 1, in practice, the window size D should be chosen as small as possible to minimize the time lag in change detection, while the number of samples N in the training data should be large enough to make sure to capture major patterns. Depending on the nature of data, it makes sense to assume $m \neq r$.

To handle this general situation, we consider linear combinations of $\boldsymbol{u}_i$'s and $\boldsymbol{v}_i$'s and define the change score as a parameterized version of KL divergence:

$$a^{(t)} = \min_{\boldsymbol{f}, \boldsymbol{g}} \int \mathrm{d}\boldsymbol{x}\, \mathcal{M}(\boldsymbol{x} | \mathsf{U}\boldsymbol{f}, \kappa) \ln \frac{\mathcal{M}(\boldsymbol{x} | \mathsf{U}\boldsymbol{f}, \kappa)}{\mathcal{M}(\boldsymbol{x} | \mathsf{U}^{(t)}\boldsymbol{g}, \kappa)} \tag{27}$$

under the constraint of $\boldsymbol{f}^\top \boldsymbol{f} = 1$, $\boldsymbol{g}^\top \boldsymbol{g} = 1$. This can be also viewed as an averaged version of the log likelihood ratio, which is the standard measure of change detection [Chen and Gupta, 2012].

By inserting the definition of the vMF distribution in Eq. (1), we have

$$a^{(t)} = \kappa \min_{\boldsymbol{f}, \boldsymbol{g}} \left\{ \langle \boldsymbol{x} \rangle^\top \left(\mathsf{U}\boldsymbol{f} - \mathsf{U}^{(t)} \boldsymbol{g} \right) \right\}, \tag{28}$$

where $\langle \cdot \rangle$ is the expectation w.r.t. $\mathcal{M}(\boldsymbol{x} | \mathsf{U}\boldsymbol{f}, \kappa)$. Since $\langle \boldsymbol{x} \rangle \propto \mathsf{U}\boldsymbol{f}$ follows from the basic property of the vMF distribution [Mardia *et al.*, 1980], we have

$$a^{(t)} = 1 - \max_{\boldsymbol{f}, \boldsymbol{g}} \left\{ \boldsymbol{f}^\top \mathsf{U}^\top \mathsf{U}^{(t)} \boldsymbol{g} \right\}$$
$$\text{s.t.} \;\; \boldsymbol{f}^\top \boldsymbol{f} = 1, \; \boldsymbol{g}^\top \boldsymbol{g} = 1, \tag{29}$$

where we dropped an unimportant prefactor.

Solving the optimization problem (29) is easy. Introducing Lagrange multipliers π_1, π_2 for the two constraints, it is straightforward to obtain optimality conditions as

$$\mathsf{U}^\top \mathsf{U}^{(t)} \boldsymbol{g} = \pi_1 \boldsymbol{f}, \quad {\mathsf{U}^{(t)}}^\top \mathsf{U} \boldsymbol{f} = \pi_2 \boldsymbol{g}.$$

This means that $\boldsymbol{f}$ and $\boldsymbol{g}$ are found via singular-value decomposition of $\mathsf{U}^\top \mathsf{U}^{(t)}$, which is a small matrix of size $m \times r$. The maximum of the objective function is simply given by the maximum singular value, $\sigma_1^{(t)}$; thereby we reach the final formula for the change score:

$$a^{(t)} = 1 - \sigma_1^{(t)}. \tag{30}$$

The maximum singular vector is efficiently computed by the power method [Golub and Loan, 1996]. In most practical situations, it requires only iterations as many as $\min\{m, r\}$. The complexity to compute the KL-based change score is $(\min\{m, r\})^3$. Since m and r can be just several in most applications, it is negligible in practice.

5 Related work

As described so far, the proposed method extensively uses the vMF distribution. For change or anomaly detection, little work has focused on the vMF distribution. [Idé and Kashima, 2004] seems to be one of the earliest pieces of work that explicitly leverages the vMF distribution for anomaly detection, but it does not discuss the particular task of change detection and sample regularization.

In statistics, a lot of efforts have focused on asymptotic analysis of the likelihood ratio [Chen and Gupta, 2012]. Recently, [Kuncheva, 2013] compares various change detection approaches and concludes that a model combining Gaussian mixture and Hotelling's T^2 statistic works best. However, it is widely known that stably learning Gaussian mixture is hard for physical sensor data we are interested in, which is quite noisy and includes many outliers. Also, in general, accurately estimating densities itself is challenging when dimensionality is high ($M \gtrsim 10$).

To address the challenge of density estimation, [Kawahara and Sugiyama, 2009; Liu *et al.*, 2013] proposed an interesting technique of direct density-ratio estimation, which integrates the two steps of parametric model estimation and scoring into a single step of density-ratio estimation. Thanks to the integration of the two steps, their approach is generally better than those estimates two densities individually. However, due to the very same reason, they lack practical interpretability, which is of critical importance in practice. Also, it has been pointed out that the performance significantly degrades when some of the variables are just non-informative nuisance features [Yamada *et al.*, 2013].

Recently, to extract the most informative features from multivariate time-series data, [Blythe *et al.*, 2012] proposed an interesting approach called stationary subspace analysis (SSA). Although SSA proposed for the task of time-series segmentation, which is different from the on-line change-detection we are dealing with, we experimentally compare it with the proposed method in the next section.

6 Experimental results

6.1 Methods compared

We compare the proposed method denoted by `RED+KL`, which uses Algorithm 1 for computing U and $\mathsf{U}^{(t)}$ in Eqs. (26) and (30) for the change score, with the following alternative methods:

- `RED+tr`: Use Algorithm 1 for U but replace Eq. (30) with the trace norm where $r = m$:

$$a^{(t)} = 1 - \frac{1}{m}\mathrm{Tr}(\mathsf{U}^\top \mathsf{U}^{(t)}). \tag{31}$$

- `SSA`: Use SSA for U. First identify the most stationary subspace by solving

$$\min_{\mathsf{V}^\top \mathsf{V} = \mathsf{I}_{M-m}} \sum_{i=1}^{E} \{-\ln|\mathsf{V}^\top \Sigma_i \mathsf{V}| + \|\mathsf{V}^\top \boldsymbol{\mu}_i\|^2\}, \tag{32}$$

where E is the number of epochs defined by a non-overlapping time window of size D, and $\{\boldsymbol{\mu}_i, \Sigma_i\}$ are the sample mean and covariance matrix computed in the i-th epoch. $|\cdot|$ represents the determinant in this case. Once V is found, U is obtained as the complement space of V. The change score is computed by

$$a^{(t)} = -\ln|\mathsf{U}^\top \Sigma^{(t)} \mathsf{U}| + \|\mathsf{U}^\top \boldsymbol{\mu}^{(t)}\|^2 + \mathrm{Tr}(\mathsf{U}^\top \Sigma^{(t)} \mathsf{U}),$$

where $\boldsymbol{\mu}^{(t)}$ and $\Sigma^{(t)}$ are the mean and the covariance matrix over $\mathcal{D}^{(t)}$, which is defined as the set of the samples in the sliding window at time t. Before computing these, the data is whitened with the same pooled mean and covariance of the training data.

- `PCA`: Use the principal component analysis for U. Employ the mean reconstruction error for scoring [Papadimitriou and Yu, 2006] :

$$a^{(t)} = \frac{1}{D} \sum_{n \in \mathcal{D}^{(t)}} \|(\mathsf{I}_m - \mathsf{U}\mathsf{U}^\top)\boldsymbol{x}^{(n)}\|^2. \tag{33}$$

- `T2`: Use the mean Hotelling's T^2 statistic

$$a^{(t)} = \frac{1}{D} \sum_{n \in \mathcal{D}^{(t)}} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu})^\top \Sigma^{-1} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}). \tag{34}$$

for scoring. Here $\boldsymbol{\mu}$ and Σ are the mean and the covariance matrix of the training data. `T2` is compared only in change detection since it has no explicit feature extraction step.

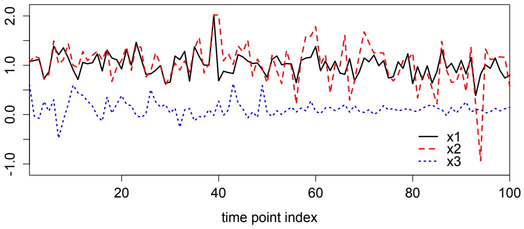

Figure 3: Synthetic three-dimensional time-series data.

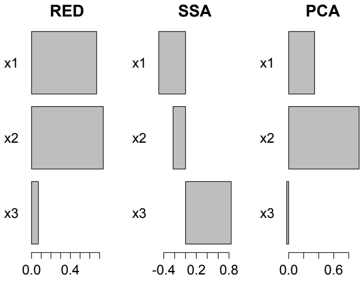

Figure 4: Comparison of $\boldsymbol{u}_1$ (Synthetic data).

6.2 Synthetic data: comparison of feature extraction methods

Figure 3 shows three-dimensional synthetic time-series data we generated. In this data, x_1 and x_2 are quite noisy but significantly correlated, while x_3 is uncorrelated to the others. Thus we expect the principal direction would point to the 45° line between x_1 and x_2.

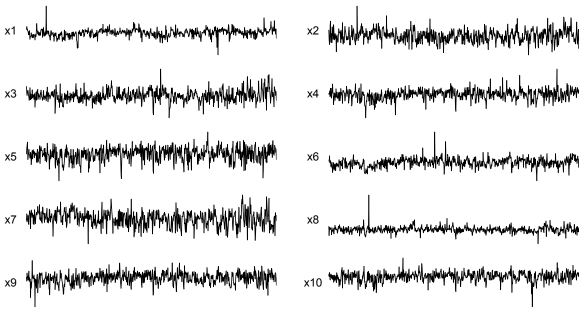

Figure 5: Ore transfer data under the normal operation.

We compared the proposed algorithm (denoted simply by RED) with two alternatives SSA and PCA. All methods take the data matrix X as the input, and return an orthonormal matrix U as the output. Figure 4 shows the coefficients of the first component $\boldsymbol{u}_1$. For RED, we used $(\lambda, \nu) = (1, 4)$, while for SSA we used $D = 25$.

We see that RED successfully captures the expected 45° direction as the principal direction. PCA has a similar trend, but due to the major outlier in x_2 at 94 (see Fig. 3), it has more weight in x_2. Close inspection shows that RED automatically removes this outlier by putting zero weight, demonstrating the automated noise-filtering capability.

The basic assumption of SSA is that the stationary components are most likely noise, and the non-stationary components are more informative for change detection. In this particular example, the major pattern is the correlation structure between x_1 and x_2, and the other variable x_3 is the noise. However, due to the heavy noise especially in the first half of x_3, SSA fails to disregard the noise variable.

6.3 Real-world data: comparison of on-line change detection performance

We applied the proposed method to a real-world on-line change detection task. Figure 5 shows time-series data from an ore transfer system being monitored by ten sensors measuring physical quantities such as speed, current, load, temperature, and displacement. The data itself was generated by a testbed system to simulate the normal operating condition and thus used as the training data. As introduced in Sec. 2.1, the system consists of two almost equivalent subsystems, and some of the variables are significantly correlated, and incur the multiplicative noise. As seen, the data is extremely noisy and sometimes exhibits impulse-like noise due to the physical operating condition of the outdoor ore transfer system.

For this training data, we applied the RED algorithm to find U and W. Figure 6 shows $\mathsf{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2]$, where we used $(\lambda, \nu) = (2, 9)$ that maximized the F score for the test data (see the next paragraph). We see that many samples are driven to zero. In fact, 43 and 39 percents of the samples have exact zero in $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$, respectively, out of which about 16 percent are zero in both. Close inspections show that almost all of "out-of-context" outliers are cut off, while informative outliers that are consistent to the major correlation structure survive. This is exactly what we expected.

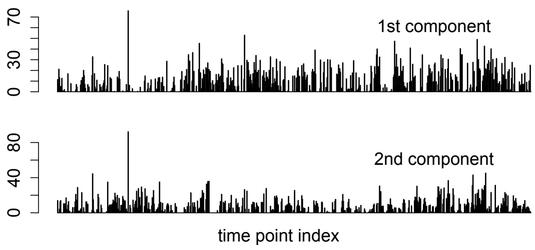

Figure 6: The magnitude of the sample weights $\{w_i^{(n)}\}$ for the first and second components (should be aligned with Fig. 5).

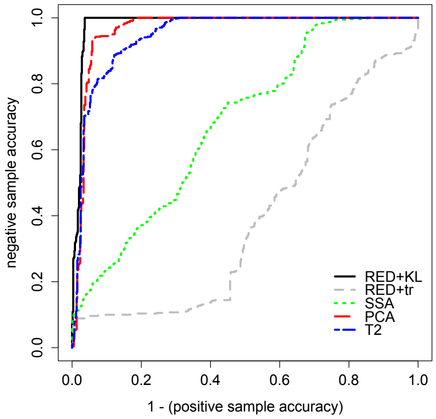

Figure 7: Comparison of the ROC curve.

We evaluated the performance of on-line change detection using another data set, where changes due to system malfunctions are recorded. Most of the failure symptoms are associated with unreasonable changes in the correlation structure between the variables. We simply use the negative-sample and positive-sample accuracies as the performance metric. Here, the negative samples are defined by those not belonging to change-points, while the positive samples are those belonging to change-points. We used the harmonic average (F score) of them to determine the (λ, ν) values. We use the window of $D = 60$ over about 1200 samples. Figure 7 compares the ROC curve. Clearly, RED-KL with $(m, r) = (2, 3)$ outperforms the alternatives. It is interesting to see RED-tr gives the worst, which demonstrates the importance the proposed subspace comparison technique of Eq. (30). For this data set, SSA fails to capture change points. The main reason is that SSA is not necessarily robust to spiky outliers as seen in Fig. 5. Since the correlational structure is critical to detect change points in this data, it makes sense that the PCA and T2 do a good job.

7 Conclusion

We have proposed a new on-line change detection algorithm for multivariate time-series data. Our algorithm extracts major directional patterns while automatically disregarding less informative samples. We proved the convergence of the algorithm, and showed that the quality of the solution is supported by the global optimality of the trust-region subproblem. We also proposed a new method of scoring the change based on a parameterized KL divergence. We validated the algorithm using real-world data.

References

[Blythe *et al.*, 2012] D.A.J. Blythe, P. von Bunau, F.C. Meinecke, and K.-R.Muller. Feature extraction for change-point detection using stationary subspace analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 23(4):631–643, 2012.

[Chen and Gupta, 2012] Jie Chen and Arjun K. Gupta. *Parametric Statistical Change Point Analysis*. Birkh'auser Applied Probability and Statistics. Springer Verlag, 2012.

[Gill *et al.*, 1974] P. E. Gill, G. H. Golub, W. Murray, and M. A. Saunders. Methods for modifying matrix factorizations. *Mathematics of Computation*, 28(126):505–535, 1974.

[Golub and Loan, 1996] Gene H. Golub and Charles F. Van Loan. *Matrix computations (3rd ed.)*. Johns Hopkins University Press, Baltimore, MD, 1996.

[Hager, 2001] W. W. Hager. Minimizing a quadratic over a sphere. *SIAM Journal on Computing*, 12:188–208, 2001.

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

[Kawahara and Sugiyama, 2009] Yoshinobu Kawahara and Masashi Sugiyama. Change-point detection in time-series data by direct density-ratio estimation. In *Proc. of 2009 SIAM Intl. Conf. on Data Mining (SDM 09)*, 2009.

[Kuncheva, 2013] Ludmila I. Kuncheva. Change detection in streaming multivariate data using likelihood detectors. *IEEE Transactions on Knowledge and Data Engineering*, 25(5):1175–1180, 2013.

[Liu *et al.*, 2013] S. Liu, M. Yamada, N. Collier, and M. Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.

[Mardia *et al.*, 1980] K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis*. Academic Press, 1980.

[Papadimitriou and Yu, 2006] Spiros Papadimitriou and Philip Yu. Optimal multi-scale patterns in time series streams. In *Proc. 2006 ACM SIGMOD Intl. Conf. Management of Data*, pages 647–658, 2006.

[Sorensen, 1997] D. C. Sorensen. Minimization of a large-scale quadratic function subject to a spherical constraint. *SIAM Journal on Optimization*, 7(1):141–161, 1997.

[Sra, 2012] Suvrit Sra. A short note on parameter approximation for von Mises-Fisher distributions: and a fast implementation of $\mathrm{I}_s(x)$. *Computational Statistics*, 27(1):177–190, March 2012.

[Tao and An, 1998] Pham Dinh Tao and Le Thi Hoai An. A D.C. optimization algorithm for solving the trust-region subproblem. *SIAM J. Optimization*, 8:476–505, 1998.

[Toint *et al.*, 2009] P. L. Toint, D. Tomanos, and M. Weber-Mendonca. A multilevel algorithm for solving the trust-region subproblem. *Optimization Methods and Software*, 24(2):299–311, 2009.

[Wen *et al.*, 2010] Zaiwen Wen, Wotao Yin, Donald Goldfarb, and Yin Zhang. A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation. *SIAM Journal on Scientific Computing*, 32(4):1832–1857, 2010.

[Yamada *et al.*, 2013] M. Yamada, A. Kimura, F. Naya, and H. Sawada. Change-point detection with feature selection in high-dimensional time-series data. In *Proc. Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI 13, pages 1827–1833, 2013.

[Zou and Hastie, 2005] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society*, 67:301–320, 2005.