Pattern Change Discovery between High Dimensional Data Sets

Yi Xu, Zhongfei Zhang
Computer Science Department
Binghamton University
{yxu,zhongfei}@cs.binghamton.edu

Bo Long
Yahoo! Inc.
bolong@yahoo-inc.com

Philip S. Yu
Computer Science Department
University of Illinois at Chicago
psyu@cs.uic.edu

ABSTRACT

This paper investigates the general problem of pattern change discovery between high-dimensional data sets. Current methods either mainly focus on magnitude change detection of low-dimensional data sets or are under supervised frameworks. In this paper, the notion of the principal angles between the subspaces is introduced to measure the subspace difference between two high-dimensional data sets. Principal angles bear a property to isolate subspace change from the magnitude change. To address the challenge of directly computing the principal angles, we elect to use matrix factorization to serve as a statistical framework and develop the principle of the dominant subspace mapping to transfer the principal angle based detection to a matrix factorization problem. We show how matrix factorization can be naturally embedded into the likelihood ratio test based on the linear models. The proposed method is of an unsupervised nature and addresses the statistical significance of the pattern changes between high-dimensional data sets. We have showcased the different applications of this solution in several specific real-world applications to demonstrate the power and effectiveness of this method.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation, Measurements

Keywords

Unsupervised Learning, Pattern Change Detection, Principal Angles, Principle of Dominant Subspace Mapping, Matrix Factorization

1. INTRODUCTION

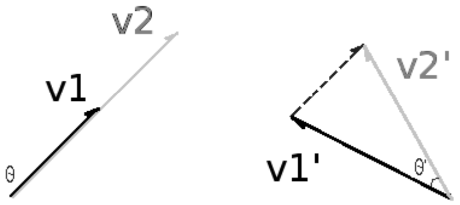


Figure 1: The Euclidean metric fails to differentiate the length difference from the direction difference

High dimensional data exist everywhere in our life and in all the sectors of our society in every modality of the data we live with today, including text, imagery, audio, video, and graphics. Pattern change discovery from high dimensional data sets is a general problem that arises in almost every application in the real-world; examples of such applications include concept drift mining in text data, event discovery in surveillance video data, event discovery in news data, hot topic discovery in the literature, image pattern change detection, as well as genome sequence change detection in bioinformatics, to just name a few.

In each of the above applications, we formulate the problem as follows. Given two typically high-dimensional data sets, we intend to determine whether there is a significant pattern change between the two data sets. In different applications, the physical interpretation of the two data sets may be different. For example, in detecting any topic change between two text documents, the two high-dimensional data sets may be the two text documents; in detecting any concept drift among a text stream, any pair of two neighboring snapshots of the text collections in the timeline may be considered as the two high-dimensional data sets; in detecting any pattern change between two images or two collections of images, the two high dimensional data sets may be the two corresponding images or the two collections of the images; in detecting any event occurred in a surveillance video camera, the two high-dimensional data sets may be any pair of two neighboring video frames or groups of video frames in the video stream; in detecting any hot topics in a news data stream, the two high-dimensional data sets may be two neighboring sample windows of the news text data within the stream.

One may wonder what makes high-dimensional data dif-

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

ferent when it comes to change detection. For almost all the magnitude change detection methods, an invisible pitfall arises with the increase of data's dimensionality. The tricky conflict between Euclidean distance and dimensionality is illustrated in Fig 1. Here we use Euclidean distance because it is the most intuitive and popular metric. Moreover, many commonly used metrics, such as L-norms, K-L divergence, or more generally, Bregman divergence, are defined based on the Euclidean distance. Fig 1 gives two pairs of vectors (v_1, v_2) and (v_1', v_2'), and the angles, θ, θ' between each pair, respectively. Under Euclidean distance, $\|v_1 - v_2\|$, and $\|v_1' - v_2'\|$ are the same. In other words, Euclidean distance fails to detect $\theta \neq \theta'$, and therefore, is unable to differentiate the length difference from the direction difference introduced by the dimensionality.

In fact in quite a few real-world applications, high dimensional data per se do not contribute to the data vectors' magnitude change, but to a new combination of a certain subset of the features. For example, we do not intend to conclude that the difference between a human baby and an adult is the same as that between the baby and a little monkey; a banker is not interested in the volume of the financial news but the newly emerged key words; to examine the mutation of a DNA sequence, a biologist needs to find the new combination of Adenine and Guanine instead of the DNA data size change. In these cases, the change of feature subspace should not be confused with the change of data's magnitude. One may argue that we still could round up all the vectors into the same length and then apply the Euclidean distance to avoid the confusion with the magnitude. Such a manipulation theoretically works only when the subspace dimension spanned by data is one (to compare only two vectors). Moreover, the round-up errors and the change of the original data structure may lead to unmanageable consequences.

Based on the above fact, our first motivation is to find a metric that is invariant under data's magnitude change and only characterize the subspace change introduced by dimensionality. Further, we require that such a metric is in a form suitable for computation and manipulation. We would like to clarify that, it is not our intention to underestimate the significance of detecting data's magnitude change. Our standpoint is that to detect the subspace change between high-dimensional data sets through a magnitude-based metric is inaccurate and conceptually confusing. In the rest of the paper, when we say pattern change between high-dimensional data sets, we refer to the subspace change, not the magnitude change.

In order to identify the appropriate subspace for discovering the pattern change between the data sets, we introduce the concept of dominant subspace based on the principal angles [7]. The notion of principal angels between two subspace has a nice property of invariant under an isomorphism, thus is independent of data's magnitude change. The challenge then is to compute the principal angles. To address this challenge, we elect to use matrix factorization to serve as a statistical framework for computing the principal angles. We develop the principle of dominant subspace mapping and show how matrix factorization can be naturally embedded into the likelihood ratio test based on the principle. The proposed method is of an unsupervised nature and addresses the statistical significance of the pattern changes between high-dimensional data sets.

The contribution of this work is highlighted as follows. First, we have studied the very general problem of pattern change discovery among different high dimensional data sets. Second, we have introduced the notion of principal angles between subspaces as a metric for pattern change. Third, we have introduced the principle of the dominant subspace mapping to transfer the principal angle based detection to a matrix factorization problem. Fourth, we have showcased the different applications of this solution in several specific real-world applications to demonstrate the power and effectiveness of this method.

2. RELATED WORK

The classic paradigm for magnitude-based change detection between two data sets is through parameter estimations based on established distribution models. More recent work in this direction [25, 12, 27] attempts to avoid the parametric dependency and to define alternative distance measures between the two distributions. In [25], Song et al. developed a Monte-Carlo framework to detect distribution changes for low dimensional data. In [12], Kifer et al. defined the A-distance to measure the non-parametric distribution change. In [27], Leeuwen and Siebes described the data distributions using their compressed code table and defined the Code-Table-Difference to capture the distribution difference between data sets. The limitation of the low-dimensional distribution models, as Vapnik pointed out at the beginning of his book [28], is that they do not reflect the singularities of the high-dimensional cases, and consequently cannot grasp the change of the subspace.

Based on Vapnik's statistic supervised learning theory, the pattern change detection problem, also called *concept drift* in several specific applications, has been attracted great effort [26, 31, 30, 29, 21]. Classifiers are trained to capture the subspace structures of the high-dimensional data sets via support vectors. The pattern changes can be indirectly reflected through evaluating the classification errors on the data sets. We refer to the survey by Tsymbal [26] for an overview of the important literature on this topic. The main categories of the methods to address the concept drift analysis problem include the instance selection and weighting [13, 11], the ensemble learning [30, 31, 2], and the two-samples hypothesis test [5, 10, 21]. Although supervised learning techniques have the capacity to detect structural changes between high-dimensional data sets, they require labels to train and validate the classifiers. Most of the real-world data sets, however, typically lack sufficient labels that can be used to train the classifiers. In [5], Dries and Rückert proposed a trade-off strategy. Without using real labels, they constructed two virtual classifiers by giving two different types of the labels to the two data sets, respectively, and then proposed three two-sample test methods based on the quality of the classifiers; a good quality indicates a concept drift between the two data sets. Using one classifier to describe the whole dataset, however, oversimplifies the mixture structures of the data sets, and the detection performance is expected to be impaired (see Sec. 5).

As an unsupervised paradigm, matrix factorization is recently considered for subspace analysis of high-dimensional data sets. The theory and applications of matrix factorization have been intensively developed during the last decade. In [16], Lee and Seung developed the breakthrough of the multiplicative updating rules for solving matrix factorization, extending the classical vector quantization and principal components analysis to a new horizon. In [8], Gordon unified the matrix factorization literature with the generalized linear models, strengthening the statistical foundation for matrix factorization. As for the applications, Ding et al. [4] applied non-negative matrix factorization to spectral clustering, graph matching, and clique finding. Long et al. [19, 18] used matrix factorization for relational clustering. Miettinen [20] developed factorization algorithms for binary data sets. In this paper, we use matrix factorization and the notion of the principal angles between subspaces to capture the structural difference between the high-dimensional data sets. We address the statistical significance of the difference through a likelihood hypothesis test based on the linear model. To the best of our knowledge, this is the first time when matrix factorization is used to develop a statistical framework for the pattern change detection.

Other recent efforts on the magnitude-based change detection for specific applications include the event detection from time series data [15, 22] focusing on discovering a significant magnitude change and its duration on a particular feature, word bursts tracking[6, 9], and trend analysis in blogosphere by tracking singular values [3].

3. PRELIMINARIES

3.1 Notations

In this paper, a matrix is denoted as a capital letter in boldface such as $\mathbf{X}$. $\mathbf{X}_{ij}$ is the entry in the ith row and the jth column. $\mathbf{X}_{i\cdot}$ stands for the ith row of $\mathbf{X}$ and $\mathbf{X}_{\cdot j}$ stands for the jth column of $\mathbf{X}$. A vector is a lowercase letter in boldface such as $\mathbf{x}$. A scalar variable is denoted as a lowercase letter such as x. $\mathbf{U}^T$ stands for the transpose of the matrix $\mathbf{U}$. $\mathbf{X}_{m\times n}$ stands for a matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$. $span(\mathbf{A})$ stands for the subspace spanned by the column vectors of the matrix $\mathbf{A}$. $\|\cdot\|$ by default is the Frobenius norm for a matrix; $\|\cdot\|_2$ is the 2-norm [7] for a matrix. $diag(\{x_i\})$ stands for a diagonal matrix with x_i as its ith diagonal entry.

3.2 Principal Angles and Dominant Subspace

In this section, we introduce the *principal angels* between subspaces to measure the subspace difference between data sets of high dimensions. We have already addressed the pitfall of the popular distance metrics and now we explain why the principal angles can avoid it. We start with the same example in Fig 1. We have already shown that Euclidean distance fails to detect $\theta \neq \theta'$, and therefore, is unable to differentiate the length difference from the direction difference introduced by the dimensionality. On the other hand, in this specific example, the principal angle between $span(v_1)$ and $span(v_2)$ is actually θ, and that between $span(v'_1)$ and $span(v'_2)$ is θ'. One may notice that we here use $span(v)$ instead of just v. This indicates that θ and θ' are invariant under the length shrinking or stretching for the corresponding vectors. Now one can reasonably understand the notion of principal angles between two subspaces as a generalization of an angle between two vectors as the dimensionality goes from one (as for $span(v_1)$) to n where $n \geq 1$. The principal angles have a very important property that all the Euclidean-based metrics do not have – Invariant under an isomorphism and thus independent of the magnitude change (e.g., invariant under scalar multiplication when the dimensionality is one).

Without loss of generality, assume two vector sets $\{\mathbf{x}_i\}_{i=1}^m$ and $\{\mathbf{y}_i\}_{i=1}^l$, $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{R}^n$. Golub and Loan proposed in [7] the definition of principal angles as to measure the structural difference between the two subspaces $S_1 = span(\{\mathbf{x}_i\}_{i=1}^m)$ and $S_2 = span(\{\mathbf{y}_i\}_{i=1}^l)$: An increasing sequence of principal angles $\{\theta_k\}_{k=1}^q$ is defined between two arbitrary subspaces S_1 and S_2 using their orthonormal basis: ([7] Page 602):

Definition 1. *Let S_1 and S_2 be subspaces in $\mathbb{R}^n$ whose dimensions satisfy*

$$p = dim(S_1) \geq dim(S_2) = q \geq 1$$

The principal angles $\theta_k \in [0, \pi/2]$, $k = 1, \ldots, q$, *between S_1 and S_2 are defined recursively as*

$$\cos(\theta_k) = \max_{\mathbf{u}\in S_1, \mathbf{v}\in S_2} \mathbf{u}^T\mathbf{v} = \mathbf{u}_k^T\mathbf{v}_k$$

when $k = 1$, $\|\mathbf{u}_1\| = \|\mathbf{v}_1\| = 1$; *when* $k \geq 2$, $\|\mathbf{u}_k\| = \|\mathbf{v}_k\| = 1$; $\mathbf{u}_k^T\mathbf{u}_i = 0$; $\mathbf{v}_k^T\mathbf{v}_i = 0$ *where* $i = 1, \ldots, k-1$.

In this definition, vectors $\{\mathbf{u}_k\}_{k=1}^q$ and $\{\mathbf{v}_k\}_{k=1}^q$ are actually part of the orthonormal basis for S_1 and S_2; the inner products of each pair $\mathbf{u}_k$ and $\mathbf{v}_k$ form a unique increasing sequence of angles. These angles explicitly give the difference of the subspace structure between S_1 and S_2. The algorithm given in [7] to compute the principal angles takes $\mathcal{O}(4n(q^2 + 2p^2) + 2pq(n+q) + 12q^3)$ in time complexity.

The leading largest principal angles depict the most noticeable structural difference between S_1 and S_2. The corresponding dimensions responsible for the largest principal angles are of great interest as they reflect the major pattern change. We call the subspace formed by these dimensions the *dominant subspace*. Now in order to measure the structural difference between $\{\mathbf{x}_i\}_{i=1}^m$ and $\{\mathbf{y}_i\}_{i=1}^l$, one may resort to directly computing the principal angles between S_1 and S_2, and then obtain the dominant subspace based on the largest principal angles. In practice, however, this is not an optimal solution. First, the values of n, p, and q in Definition 1 can be very large in real-world data sets, resulting in a high complexity to compute the principal angles. Second, since the real-world data sets typically contain noise and outliers, the principal angles directly computed from the raw data may not reflect the true situation. Third, in many applications, given a large amount of samples, one is only interested in the most frequent pattern changes in the majority of the data set and does not care of the principal angles for all the samples. All these issues require to developing an alternative solution to directly computing the principal angles. On the other hand, matrix factorization [16, 1, 4, 24] has been used extensively for reducing dimensionality and extracting collective patterns from noisy data in a form of a linear model. In the next section, we develop the principle of dominant subspace mapping through matrix factorization as the alternative to obtain the dominant subspace.

4. MODEL FORMULATION

Given two data sets $\mathbf{X}' = \{\mathbf{x}'_i\}_{i=1}^{n'}$ and $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$, we present a model to detect the pattern changes between $\mathbf{X}'$ and $\mathbf{X}$. Instead of computing the principal angles directly, we provide a more practical strategy involving three steps: To establish a null-hypothesis on pattern change, to extract a set of basis vectors from $span(\{\mathbf{x}_i\}_{i=1}^{n})$ under a null and its alternative hypothesis, and a statistical test to confirm these changes. We will show how principal angles, matrix factorization and linear models work together to serve the purpose.

4.1 Matrix Factorization

Learning mixture patterns from data can be formulated as *generalized*^{2} *linear*^{2} *models* [8, 24] using the following matrix factorization term:

$$\mathbf{X} \approx \mathbf{PS}^T \tag{1}$$

where the matrix $\mathbf{X}_{m\times n} = [\mathbf{x}_1, \mathbf{x}_2 \cdots \mathbf{x}_n], \mathbf{x} \in \mathbb{R}^m$, consist of n data samples represented as the n column vectors. Matrices $\mathbf{P}_{m\times k}$ and $\mathbf{S}_{n\times k}, k \ll \min(m, n)$, are two lower-dimension factors whose product approximates the original data set $\mathbf{X}$. The k column vectors of $\mathbf{P}$ are prototype patterns learned from $\mathbf{X}$; the ith row of $\mathbf{S}$ is a soft indicator using k prototypes to restore the ith sample. Thus, the columns of $\mathbf{P}$ can also be considered as an approximate generating set for the subspace containing samples $\{\mathbf{x}_i\}_{i=1}^{n}$. In this modeling, we concentrate on $\mathbf{P}$, the prototype patterns, and its changing behavior. $\mathbf{S}$ describes how the k prototypes are distributed among the n samples and may also contains useful information to characterize the dataset; yet in this paper we do not discuss its behavior and leave it as an open question.

Another advantage for matrix factorization is its form as a linear model under which a hypothesis test can be developed. More specifically, given a linear model with additive Gaussian noise $G : \mathbf{X} = \mathbf{PS}^T + \varepsilon$, where $\varepsilon_{\cdot j} \sim N_{m\times 1}(\mathbf{0}, \sigma^2\mathbf{I}_{m\times m})$, our strategy is to check the pattern change in $\mathbf{P}$ by properly constructing a hypothesis on $\mathbf{P}$ and then applying the standard likelihood ratio test.

4.2 Principle of Dominant Subspace Mapping

In order to extract the plausible pattern changes, instead of directly computing the principal angles we develop the principle of dominant subspace mapping through constructing a hypothesis testing as follows. We first establish a hypothesis on the pattern matrix $\mathbf{P}$. Assuming $\mathbf{P}'$ and $\mathbf{P}$ from the two data sets $\mathbf{X}'$ and $\mathbf{X}$, since the principal angles $\{\theta_i\}_{i=1}^{k}$ between $span(\mathbf{P}')$ and $span(\mathbf{P})$ indicate the scale of pattern changes, it is straightforward to set up the hypothesis on the principal angles to begin with. Now we have two options for the null-hypothesis: To assume no pattern change or to assume an obvious pattern change. If we choose the former, there are two concerns. First, the possibility that two data sets obtained from different times or locations share the same subspace is almost zero, resulting in a hypothesis on an almost impossible event. Second, as shown in definition 1, the principal angles are computed via *cos*; the null-hypothesis of no pattern change gives $H_o : \|diag(\{\cos\theta_i\}_{i=1}^{k})\| = k$, indicating that every principal angle is zero; such a setting is vulnerable due to different k value in different applications and we have no prior knowledge about the specific value of k. On the other hand, if we set the hypothesis as an obvious pattern change, it serves both purposes of detecting pattern change and a convenient form of $H_o : \|diag(\{\cos\theta_i\}_{i=1}^{k})\| = 0$. If the hypothesis is true, the values of $\{\theta_i\}_{i=1}^{k}$ are large, indicating the large pattern change between $span(\mathbf{P}) \subseteq \{\mathbf{x}_i\}_{i=1}^{n}$ and $span(\mathbf{P}') \subseteq \{\mathbf{x}'_i\}_{i=1}^{n'}$. More importantly, this hypothesis is independent of the value k, making the detection more robust despite the possible information loss caused by matrix factorization.

While the hypothesis $H_o : \|diag(\{\cos\theta_i\}_{i=1}^{k})\| = 0$ is straightforward, in order to construct a simple statistic test, we elect to use a hypothesis cast directly on $\mathbf{P}$ and $\mathbf{P}'$. For this purpose, we first introduce the following lemma:

LEMMA 1. *Given that $\mathbf{P}' \in \mathbb{R}^{m\times p}$ and $\mathbf{P} \in \mathbb{R}^{m\times q}$, each with linearly independent columns, and that each column is normalized into the same 2-norm length L, and further given the QR factorizations $\mathbf{P} = \mathbf{QR}$ and $\mathbf{P}' = \mathbf{Q}'\mathbf{R}'$,the principal angles $\{\cos\theta_i\}_{i=1}^{k}$ between span$(\mathbf{P})$ and span$(\mathbf{P}')$ satisfy inequality:*

$$\frac{1}{pqL^2}\|\mathbf{P}'^T\mathbf{P}\| \leq \|diag(\{\cos\theta_i\}_{i=1}^{k})\| \leq \frac{a}{|\sigma_1\sigma_2|}\|\mathbf{P}'^T\mathbf{P}\| \tag{2}$$

where $a \leq pq$ is a constant, and σ_1 and σ_2 are the smallest eigenvalues of $\mathbf{R}'$ and $\mathbf{R}$, respectively.

The proof of Lemma 1 is in Appendix A. Lemma 1 gives the upper and lower bounds of $\|diag(\{\cos\theta_i\}_{i=1}^{k})\|$ in terms of $\|\mathbf{P}'^T\mathbf{P}\|$. More importantly, due to the Sandwich Theorem, $\|\mathbf{P}'^T\mathbf{P}\|$ and $\|diag(\{\cos\theta_i\}_{i=1}^{k})\|$ are asymptotically equivalent as $\|diag(\{\cos\theta_i\}_{i=1}^{k})\|$ is close to zero. Therefore, we establish a hypothesis using $\mathbf{P}$ and $\mathbf{P}'$directly, as shown in the following corollary:

COROLLARY 1. *The null-hypothesis*

$$H_o : \|diag(\{\cos\theta_i\}_{i=1}^{k})\| = 0$$

has its equivalent form of

$$H_o : \mathbf{P}'^T\mathbf{P} = \mathbf{0} \tag{3}$$

PROOF. It is a direct result due to Lemma 1 and the Sandwich Theorem. □

4.3 Likelihood Ratio Test

Given the simple form of null-hypothesis $H_o : \mathbf{P}'^T\mathbf{P} = \mathbf{0}$ on linear model $G : \mathbf{X} = \mathbf{PS}^T + \varepsilon$, where $\varepsilon_{\cdot j} \sim N_{m\times 1}(\mathbf{0}, \sigma^2\mathbf{I}_{m\times m})$, one can use the standard likelihood ratio test for verification ([23] Page 98): First, estimate $\mathbf{P}$ only based on linear model G. Second, estimate $\mathbf{P}$ under constraint H_o. Finally, compute the likelihood ratio between the two cases.

To estimate $\mathbf{P}$ only based on the given linear model $G : \mathbf{X} = \mathbf{PS}^T + \varepsilon$, where $\varepsilon_{\cdot j} \sim N_{m\times 1}(\mathbf{0}, \sigma^2\mathbf{I}_{m\times m})$, the likelihood function for G is [23]

$$L(\mathbf{P}, \mathbf{S}) = (2\pi\sigma^2)^{-mn} exp[-\frac{1}{2\sigma^2}\|\mathbf{X} - \mathbf{PS}^T\|^2]. \tag{4}$$

Algorithm 1 LRatio

Input: data sets $\mathbf{X}, \mathbf{X}'$, , and threshold h.
Output: Feature basis $\mathbf{P}_H$, indicator matrix $\mathbf{S}_H$, the likelihood ratio test statistic Λ, and the testing result.
Method:

```
1: Initialize $\mathbf{P}', \mathbf{S}'$,$\hat{\mathbf{P}}$, $\hat{\mathbf{S}}$,$\hat{\mathbf{P}}_H$ and $\hat{\mathbf{S}}_H$, and $\lambda$ randomly.
2: Iteratively update $\mathbf{P}'$ and $\mathbf{S}'$ using (5) until convergence
3: Iteratively update $\hat{\mathbf{P}}$ and $\hat{\mathbf{S}}$ using (5) until convergence
4: Iteratively update $\hat{\mathbf{P}}_H$ and $\hat{\mathbf{S}}_H$ using (7) until convergence
5: Compute $\Lambda$ using (8)
6: Reject $H_o$ if $\Lambda$ is smaller than $h$.
```

Name	Part 1	Part 2	Sample no.	Dim.
sys	comp.sys.ibm.pc	comp.sys.mac	400 × 2	1558
ossys	comp.os.ms-windows.misc, comp.sys.ibm.pc	comp.sys.mac comp.windows.x	200 × 4	2261
computer	comp.graphics, comp.os.ms-windows.misc, comp.sys.ibm.pc	comp.sys.ibm.pc, comp.sys.mac, sci.electronics	100 × 6	1606
socialtalk	talk.politics.guns, talk.politics.mideast, talk.politics.misc, talk.religion.misc	alt.atheism, soc.religion.christian, talk.politics.misc, talk.religion.misc,	100 × 8	3312
sci	sci.crypt	sci.med	400 × 2	2870
rec-sci	rec.sport.baseball rec.sport.hockey	sci.electronics, sci.space	200 × 4	2800
comp-sci	comp.graphics, comp.os.ms-windows.misc, comp.sys.ibm.pc	sci.electronics, sci.med, sci.space	100 × 6	1864
rec-talk	rec.autos, rec.motorcycles, rec.sport.baseball rec.sport.hockey	talk.politics.guns, talk.politics.mideast, talk.politics.misc, talk.religion.misc	100 × 8	2992

Table 1: Configuration of the pattern change data sets.

Maximizing the likelihood function (4) is equivalent to estimating the factors $\mathbf{P}$ and $\mathbf{S}$ that minimize $\|\mathbf{X} - \mathbf{P}\mathbf{S}^T\|^2$. This normal-distribution-based matrix factorization can be efficiently solved via the multiplicative iteration algorithm proposed by Lee and Seung in [16, 17]:

$$\mathbf{P}_{ij}^{update} = \mathbf{P}_{ij}\frac{(\mathbf{P}^T\mathbf{X})_{ij}}{(\mathbf{P}\mathbf{S}^T\mathbf{S})_{ij}};$$

$$\mathbf{S}_{ij}^{update} = \mathbf{S}_{ij}\frac{(\mathbf{X}^T\mathbf{P})_{ij}}{(\mathbf{S}\mathbf{P}^T\mathbf{P})_{ij}} \tag{5}$$

The proof of the convergence of the updating rule can be found in [17]. This updating rule generates the estimation $\hat{\mathbf{P}}$ and $\hat{\mathbf{S}}$.

Finding the maximum likelihood estimates subject to the constraint (3) gives the following log likelihood function ([23] Page 98):

$$\begin{aligned}\mathcal{L}(\mathbf{P}, \mathbf{S}) &= -\log L(\mathbf{P}, \mathbf{S}) + \lambda\|\mathbf{P}^T\mathbf{P}'\|^2 \\ &= constant + \frac{np}{2} + \frac{1}{2\sigma^2}\|\mathbf{X} - \mathbf{P}\mathbf{S}^T\|^2 + \lambda\|\mathbf{P}^T\mathbf{P}'\|^2\end{aligned}$$

which is equivalent to minimizing:

$$\mathcal{L}(\mathbf{P}, \mathbf{S}) = \|\mathbf{X} - \mathbf{P}\mathbf{S}^T\|^2 + \lambda\|\mathbf{P}^T\mathbf{P}'\|^2 \tag{6}$$

where $\lambda > 0$ is the Lagrange multiplier. To solve this constrained optimization problem, we give the non-increasing updating rule through the following Lemma:

LEMMA 2. *The loss function* (6) *is non-increasing under the updating rule:*

$$\mathbf{P}_{ij}^{update} = \mathbf{P}_{ij}\frac{(\mathbf{X}\mathbf{S})_{ij}}{(\mathbf{P}\mathbf{S}^T\mathbf{S} + \lambda\mathbf{P}'\mathbf{P}'^T\mathbf{P})_{ij}}$$

$$\mathbf{S}_{ij}^{update} = \mathbf{S}_{ij}\frac{(\mathbf{X}^T\mathbf{P})_{ij}}{(\mathbf{S}\mathbf{P}^T\mathbf{P})_{ij}}$$

$$\lambda^{update} = \frac{1}{mk}\sum_{ij}\frac{(\mathbf{X}\mathbf{S} - \mathbf{P}\mathbf{S}^T\mathbf{S})_{ij}}{(\mathbf{P}'\mathbf{P}'^T\mathbf{P})_{ij}} \tag{7}$$

The loss function (6) *is invariant under this rule if and only if* $\mathbf{P}$ *and* $\mathbf{S}$ *are at a stationary point of the loss function.*

The proof of the lemma is in Appendix B. Using updating rule (7), we obtain estimation $\hat{\mathbf{P}}_H$ and $\hat{\mathbf{S}}_H$ under the null-hypothesis. The time complexity of the updating rule (7) for each iteration is $\mathcal{O}(mnk + k^2m)$, where m, n, and k are defined at the beginning of Section 4.1.

After we obtain the estimation of $\hat{\mathbf{P}}$, $\hat{\mathbf{S}}$ and $\hat{\mathbf{P}}_H$, $\hat{\mathbf{S}}_H$, the likelihood ratio statistic is given by ([23] Page 99):

$$\Lambda = \frac{\|\mathbf{X} - \hat{\mathbf{P}}\hat{\mathbf{S}}^T\|^2}{\|\mathbf{X} - \hat{\mathbf{P}}_H\hat{\mathbf{S}}_H^T\|^2} \tag{8}$$

According to the likelihood principle, a small Λ indicates a bad estimation of $\hat{\mathbf{P}}_H$ and $\hat{\mathbf{S}}_H$, and we then reject H_o. On the other hand, a large value of Λ suggests a pattern change detected in $\hat{\mathbf{P}}_H$. The algorithm, called LRatio, is summarized in Algorithm 1.

The time complexity of the updating rule (5) is $\mathcal{O}(mnk)$ in each iteration, and that of (7) is $\mathcal{O}(mnk + k^2m)$ in each iteration; the complexity to compute Λ is $\mathcal{O}(mnk)$, where m, n, and k are defined in Section 4.1. Thus, the total time complexity is $\mathcal{O}(mnk + k^2m)$ for each iteration, which is much lower than that of directly computing the principal angles between $\mathbf{X}$ and $\mathbf{X}'$.

5. EXPERIMENTS

In order to demonstrate the power and promise of LRatio as well as its superiority to the existing literature in discovering significant pattern changes in different applications in the real-world, we have applied LRatio to several different real-world problems in comparison with the existing methods in the related literature in these different applications.

5.1 Topic Change Detection among Documents

The goal of the first application is to verify the performance of LRatio test using collections of text documents. In this application, we use the standard 20-newsgroup data sets [14], the dimension scale of which is of thousands. As listed in Table 1, we construct eight scenarios using different topic combinations. For each scenario, two parts are set up (Part 1 and Part 2). Each part contains articles evenly mixed from one or more topics. Under each scenario, if the two data sets are sampled from the same part, they should bear similar subspace structure; while if the two data sets are from

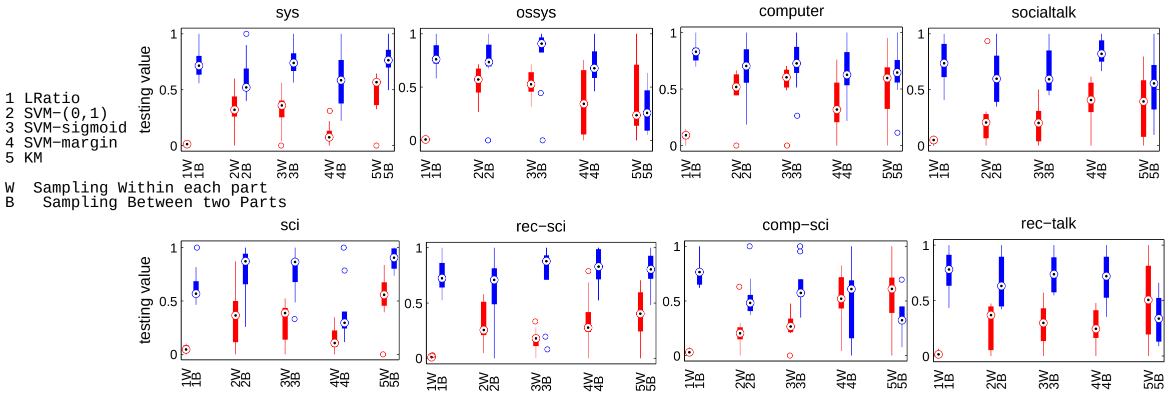


Figure 2: The detection performance of LRatio and 4 comparison methods. For each pair of W and B, a smaller overlap between W and B indicates a better performance.

different parts, their subspace structures are different and LRatio test should be able to reflect this difference through the testing statistic. These eight scenarios are constructed to showcase data sets with different structural complexities and/or pattern change strengths. The first four scenarios intend to imitate moderate pattern change by electing similar topics between the two parts. The next four scenarios imitate strong pattern change by setting different topics between the two parts. We compare the performance of LRatio with the following methods.

Baseline. We apply the standard K-means to each of the two data sets to obtain the data matrices composed of the K centroids, respectively, and then compute the subspaces distance between the pair of the K centroids based on Definition (1). Intuitively, a pattern change shall results in a large distance. We use this distance as a statistic to indicate the pattern change, and compare its sensitivity with LRatio test. For the reference purpose, we call this baseline method as KM.

Peers. We elect to use three different concept drift detection methods in the recent literature [5] by Dries and Rückert for a peer comparison. They are PCA-Bayesian Margin Test and two other error rate based test methods. For the reference purpose, we call them SVM-margin, SVM-sigmoid, and SVM-(0,1), respectively. The reasons why we select these comparing methods are the following. First, under the framework of support vector machine (SVM), their methods are suitable for high-dimensional data sets. Second, although based on supervised techniques, the model does not require real labels and therefore can be used in unsupervised applications. Third, these methods are in a similar two-sample statistical test framework to that of LRatio, resulting in a fair comparison environment.

In order to verify the detection sensitivity, we compare the testing statistics of the data set pair having no pattern change in between with the testing statistics of the data set pair involving a pattern change in between. The evaluation protocol is defined as follows.

For each scenario,

1. Obtain testing statistic from data set pair with no pattern change in between:

i). Constructing two data sets by randomly sampling 200 articles, each dataset with 100 samples, only from Part 1 (or Part 2).

ii). Applying LRatio and the four comparison methods on the two data sets.

iii). Repeating i) and ii) 20 times.

2. Obtain testing statistic from data set pair with pattern changes in between:

i). Constructing the first data set by randomly sampling 100 articles form Part 1; constructing the second data set by randomly sampling 100 articles from Part 2.

ii). Applying LRatio and the four comparison methods on the two data sets.

iii). Repeating i) and ii) 20 times.

3. For each method, normalize the 40 testing statistics to the range of $[0, 1]$ for easy comparison.

Ideally, there should be a big gap between the first 20 testing statistics and the last 20 testing statistics, because the first 20 tests are from the dataset pair that has no pattern change, and the last 20 tests are from the data set pair with pattern changes. Fig. 2 documents all the results of this experiment, where a boxplot is used to represent the numerical distribution of the statistics obtained from the sampling within each part (red boxplots) and sampling between two parts (blue boxplots). In each boxplot, the median(in $\odot$), the 25th percentile(in bars), the 75th percentiles(in whiskers) and the outliers(in $\circ$) of the distribution are drawn. Consequently, for each method and for each of the eight collections, there is a corresponding pair of boxplots representing the statistic distributions for sampling within each part (red boxplot labeled with letter W) and for sampling between two parts (blue boxplot labeled with letter B), respectively. Clearly, more overlap between the pair of boxplots indicates the worse performance in discovering the pattern change for the method. From the figure, all the four comparing methods have the overlaps in the majority of the eight collections; in comparison, LRatio is the only method that has no overlap at all for all the eight collections; further, for the first four scenarios where there expects to be only moderate topic changes between the two parts, LRatio still clearly stands out with no overlap at all between the two boxplots. This demonstrates that LRatio is not only powerful in discovering pattern changes, but also very sensitive to the pattern changes.

5.2 Event Detection from News Streams

While the 20 newsgroups data experiment is for systematic evaluations of the pattern change discovery capabilities and sensitivities, the next experiment is an application scenario of event detection in a news stream data set. We have manually collected Google news data everyday from 23. Oct. to 22. Nov. for the year of 2008 for four specific tracks: political news, financial news, sports news, and entertainment news. To form the news stream data for each of the four tracks, we group the news documents and time-stamp these documents in a unit of every three neighboring days. Since all the five methods we used in the previous experiments are for pattern change discovery between two data sets, we apply each of them to each pair of the neighboring units of the news stream in each track to obtain the statistic value. Consequently, for the whole month news data in each track, each method generates a statistic sequence, which is called the test sequence for the track of the news for the corresponding method.

Fig. 3 documents the political news test sequences within the window between October 23, 2008, and November 22, 2008 for LRatio, KM, and SVM-margin. Since the three methods from [5] are very close in performance, for the clarity purpose in the figure, we only show the test sequence of SVM-margin in this figure. Presumably in the figure for each method a significant peak in the test sequence means a significant pattern change, indicating that significant news events are detected by this method on that day. We manually examined everyday's news data within this whole month to provide the ground truth regarding whether there are any significant news events on everyday of the month, and annotated the specific events.

Since both LRatio and KM conduct the pattern change discovery through the clustering manner, they are both further able to detect the specific events through key words in each cluster, and are able to rank the "significance" of each detected event based on the number of samples in each cluster. Fig. 4 documents the top five detected significant events on Nov. 7, Nov. 13, and Nov. 19 for both methods, respectively, where each event is represented using a bar with the length proportional to the significance of the event, and the same event is ground-truthed with the same color. From the figure, there are three observations. First, for each of the three days, all the five detected events by LRatio are unique and distinct, while there are many duplications of the five top events detected by KM when compared with the ground truth; this is particularly true for Nov. 7 where all the five top events detected by KM are about the *Election Campaign*. Second, as is also observed in Fig. 3, for many events KM is unable to detect them "in time" but rather with a delay; in other words, for many of the news events KM detected are actually old news events. For example, the event that Obama was elected as the US President occurred on Nov. 7 (which LRatio corrected detected in time) is declared as the number one detected event for KM both on Nov. 13 and on Nov. 19, but not on Nov. 7. In fact, for all the three days' top five events detected by KM, only the event *Senator Clinton Became Secretary of State* on Nov. 13 and the event *North Korea Nuclear Crisis* on Nov. 19 are caught by KM in time, whereas all others are actually old news events. Third, the specific event data reported in Fig. 4 coincide with the holistic event detection results reported in Fig. 3 very well. Of all the three days, KM essentially only detected old news events, and that is why in Fig. 3 on these three days there is no peak in the KM test sequence curve, indicating KM fails to detect any significant events for these three days. While SVM-margin does not have the capability to do the clustering analysis to report the specific events detected on each day as LRatio and KM do, it detects the events based on a holistic analysis reported in the test sequence shown in Fig. 3, from which it is clear that SVM-margin still fails to detect any significant events on Nov. 13 and Nov. 19 with an exception on Nov. 7 because the event *Obama Became US President* was such an obvious significant event that SVM-margin did not miss.

The difference in performance between LRatio and the comparing methods is obvious. LRatio aims at discovering pattern changes regardless of whether the pattern changes come from a completely new topic or a new direction of an existing topic. KM, on the other hand, aims at discovering major clusters from the data; thus, new topics need time to "accumulate" to form clusters in order to become significant topics, while new directions of an existing topic are likely to be absorbed into the clusters and would never show up until they eventually dominate the clusters. That is why KM always misses many significant events and often detects an event with a delay in time. For SVM-margin, SVM-(0,1), and SVM-(sigmoid), although they also aim at discovering pattern changes, they work well only when the data have a simple structure and the majority of the samples bear a similar pattern change, which also explains why they only provide a holistic statistic on event detection with no capability for the specific pattern changes.

5.3 Event Detection in Surveillance Video

Finally, we also showcase the application of LRatio to the surveillance video stream data to detect events. In this context, each frame of the video stream is considered as a sample vector. Like the news data stream in the previous experiment, here again we apply LRatio to each pair of neighboring video segments (each segment has 100 frames) to see whether there is any event occurred. To demonstrate the power of the surveillance event detection capability, we apply LRatio to several different video surveillance data sets collected at different specific surveillance applications. Figs. 5 to 7 showcase three different tests of using LRatio for surveillance event detection, where in each of the figures the left panel is a snapshot of the surveillance video stream and the right panel indicates the test sequence of LRatio along the timeline.

6. CONCLUSION

We have studied the very general problem of pattern change discovery among different high dimensional data sets which exist everywhere in almost every application in the real-world, and have identified an approach based on the principal angles to discover the pattern change. We have introduced the principle of the dominant subspace mapping to

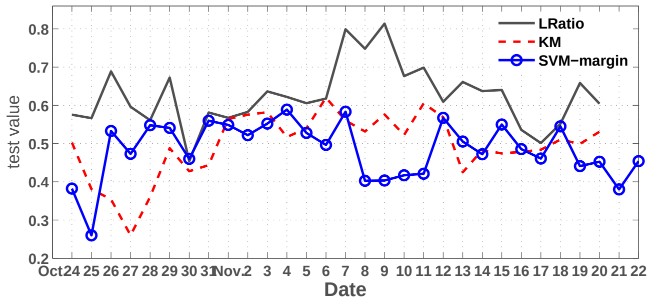


Figure 3: Test sequences for Google political news stream

transfer the principal angle based detection to a matrix factorization problem through a hypothesis testing. Finally, we have showcased the different applications of this solution in several specific real-world applications to demonstrate the power and effectiveness of this method.

7. APPENDIX

7.1 Poof of Lemma 1

The proof of Lemma 1 uses the method in [7] for computing the principal angles. We first give this method as follows.

Given $\mathbf{A} \in \mathbb{R}^{m\times p}$ and $\mathbf{B} \in \mathbb{R}^{m\times q}$ ($p \geq q$), each with linearly independent columns, the principal angles between subspaces $span(\mathbf{A})$ and $span(\mathbf{B})$ can be computed as follows . First, compute the QR factorizations for $\mathbf{A}$ and $\mathbf{B}$, respectively

$$\begin{aligned} &\mathbf{A} = \mathbf{Q_A R_A} && \mathbf{Q_A}^T\mathbf{Q_A} = \mathbf{I}_p, && \mathbf{R_A} \in \mathbb{R}^{p\times p} \\ &\mathbf{B} = \mathbf{Q_B R_B} && \mathbf{Q_B}^T\mathbf{Q_B} = \mathbf{I}_q, && \mathbf{R_B} \in \mathbb{R}^{q\times q} \end{aligned}$$

Then, let $\mathbf{C} = \mathbf{Q_A}^T\mathbf{Q_B}$ and compute the SVD (singular value decomposition) of $\mathbf{C}$ such that $\mathbf{Y}^T\mathbf{CZ} = diag(\cos\boldsymbol{\theta})$, where $diag(\cos\boldsymbol{\theta})$ is short for the diagonal matrix with the cosines of the principal angles $\{\cos\theta_1, \cos\theta_2 \ldots \cos\theta_q\}$ as the diagonal elements.

Proof. Since

$$\|diag(\cos\boldsymbol{\theta})\|^2 = \|\mathbf{Y}(\mathbf{Q}^T\mathbf{Q}')\mathbf{Z}\|^2 = \|\mathbf{Q}^T\mathbf{Q}'\|^2$$

where $diag(\cos\boldsymbol{\theta}) = \mathbf{Y}(\mathbf{Q}^T\mathbf{Q}')\mathbf{Z}$ is the SVD of $\mathbf{Q}^T\mathbf{Q}'$. The inequality we are to prove can now be re-written as:

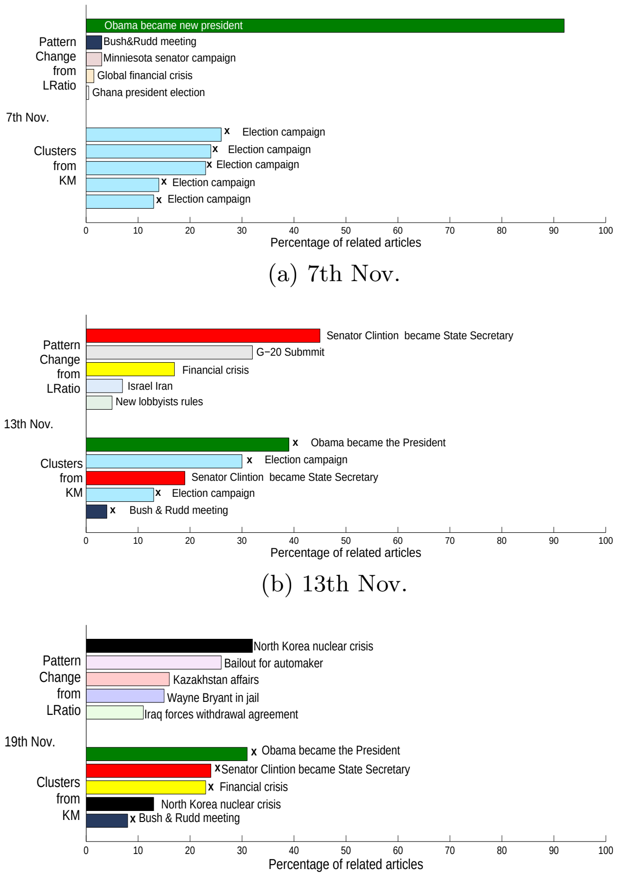
$$\frac{1}{pqL^2}\|\mathbf{P'}^T\mathbf{P}\| \leq \|\mathbf{Q}^T\mathbf{Q}'\|^2 \leq \frac{a}{|\sigma_1\sigma_2|}\|\mathbf{P'}^T\mathbf{P}\| \tag{9}$$

For the left hand side inequality, since $\|\mathbf{P}\| = \|\mathbf{QR}\| = \|\mathbf{R}\| = pL$ and similarly $\|\mathbf{P}'\| = \|\mathbf{R}'\| = qL$, we have

$$\begin{aligned} \|\mathbf{P}^T\mathbf{P}'\| &= \|\mathbf{R}^T\mathbf{Q}^T\mathbf{Q}'\mathbf{R}'\| \\ &\leq \|\mathbf{R}\|\|\mathbf{Q}^T\mathbf{Q}'\|\|\mathbf{R}'\| = pqL^2\|\mathbf{Q}^T\mathbf{Q}'\| \end{aligned}$$

For the right hand side inequality,

$$\begin{aligned} \|\mathbf{Q}^T\mathbf{Q}'\| &= \|(\mathbf{RR}^{-1})\mathbf{Q}^T\mathbf{Q}'(\mathbf{R}'\mathbf{R}'^{-1})\| \\ &= \|\mathbf{R}^{-1}\mathbf{P}^T\mathbf{P}'\mathbf{R}'^{-1}\| \\ &\leq \|\mathbf{R}^{-1}\|\|\mathbf{P}^T\mathbf{P}'\|\|\mathbf{R}'^{-1}\| \end{aligned} \tag{10}$$



(a) 7th Nov.

(b) 13th Nov.

(c) 19th Nov.

Figure 4: Pattern changes detected by LRatio v.s. Clusters found by KM. '×' marks the old news topics that have been detected in the previous days. Boxes with the same colors are related to the same news topics

Since $\mathbf{R}$ and $\mathbf{R}'$ are upper triangular, the inverses $\mathbf{R}^{-1}$ and $\mathbf{R}'^{-1}$ are also upper triangular. Therefore, the eigenvalues of $\mathbf{R}$ are $\{(\mathbf{R})_{ii}|i = 1,\ldots,p\}$, the diagonal entries of $\mathbf{R}$. Hence, the eigenvalues of $\mathbf{R}^{-1}$ are $\{\frac{1}{(\mathbf{R})_{ii}}\}$, the inverse of the diagonal entries of $\mathbf{R}$. The same conclusion also holds true for $\mathbf{R}'^{-1}$. Thus, we have

$$\begin{aligned} \|\mathbf{R}^{-1}\| &\leq \|diag((\mathbf{R})_{ii}^{-1})\| \leq \frac{p}{|\sigma|} \\ \|\mathbf{R}'^{-1}\| &\leq \|diag((\mathbf{R}')_{ii}^{-1})\| \leq \frac{q}{|\sigma'|} \end{aligned} \tag{11}$$

Combining (10) and (11) we obtain

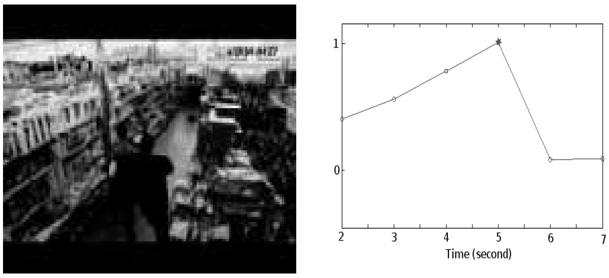
$$\|\mathbf{Q}^T\mathbf{Q}'\| \leq \frac{a}{|\sigma\sigma'|}\|\mathbf{P}^T\mathbf{P}'\|$$

where $a \leq pq$.

This completes the proof of the Lemma. □

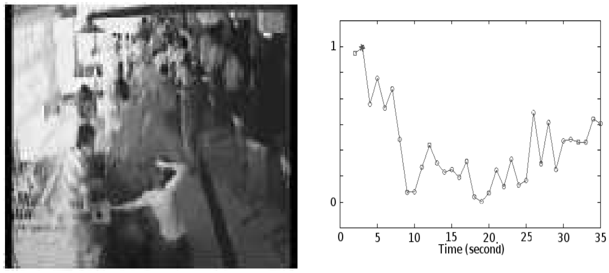
7.2 Proof of Lemma 2

We first prove the convergence of the updating rules for $\mathbf{P}$ and $\mathbf{S}$, then determin the value of λ.To prove the updating rules for $\mathbf{P}$ and $\mathbf{S}$, we make use of an auxiliary function similar to that used in the Expectation-Maximization algorithm [17].



(a) (b)

Figure 5: detection result for video 1; (a) rank 1 event: a man is running towards a cart; (b) the test sequence; the star marks the time when this man begins running.



(a) (b)

Figure 6: detection result for video 2; (a) rank 1 event: an earthquake occurs and people are running out; (b) the test sequence; the star marks the time when the earthquake occurs.

Definition 2. *$G(u,u')$ is an auxiliary function for $F(u)$ if the conditions*

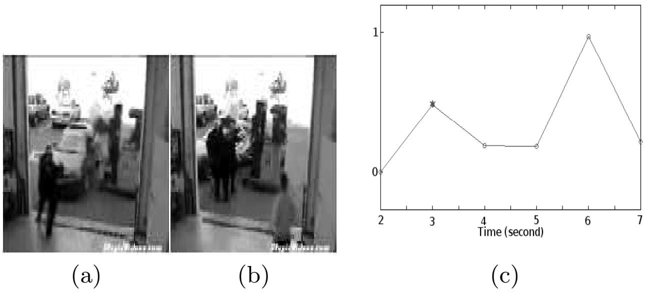
$$G(u,u') \geq F(u), G(u,u) = F(u) \tag{12}$$

are satisfied.

The auxiliary function is a useful concept due to the following lemma:

Lemma 3. *If G is an auxiliary function, then F is non-increasing under the update:*

$$u^{t+1} = \arg\min_u G(u,u^t) \tag{13}$$



(a) (b) (c)

Figure 7: detection result for video 3; (a) rank 2 event: the car collision occurs;(b) rank 1 event: a man is running towards the accident scene (c) the test sequence; the star marks the time of the collision.

Proof. $F(u^{t+1}) \leq G(u^{t+1},u^t) \leq G(u^t,u^t) = F(u^t)$ □

We show that by defining the appropriate auxiliary function $G(u,u^t)$ for (6), the update rule in Lemma 2 easily follows from (13). Now let $u = \mathbf{P}_{i\cdot}^T$, $u' = \mathbf{P}_{i\cdot}'^T$.

Lemma 4. *Function*

$$G(u,u^t) = F(u^t) + (u-u^t)^T \nabla F(u^t) + \frac{1}{2}(u-u^t)^T K(u^T)(u-u^t) \tag{14}$$

is an auxiliary function for

$$F(u) = \frac{1}{2}\sum_i (x_i - \sum_a \mathbf{S}_{ia}u_a)^2 + \frac{1}{2}\sum_a (u'_a u_a)^2 \tag{15}$$

where $K(u^t)$ is a diagonal matrix defined as

$$K_{ab}(u^t) = \delta_{ab}(\mathbf{S}^T\mathbf{S}u + \lambda u'^T u'\mathbf{I}u)_a / u_a^t \tag{16}$$

Proof. Since $G(u,u) = F(u)$ is obvious, we only need to show that $G(u,u^t) \geq F(u)$. To do this, we compare

$$F(u) = F(u^t) + (u-u^t)^T \nabla F(u^t) + \frac{1}{2}(u-u^t)^T(\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})(u-u^t) \tag{17}$$

with (14) to find that $G(u,u^t) \geq F(u)$ is equivalent to

$$0 \leq (u-u^t)^T[K(u^t) - (\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})](u-u^t) \tag{18}$$

To prove the positive semidefiniteness, consider the matrix

$$M_{ab}(u^t) = u_a^t(K(u^t) - (\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I}))_{ab}u_b^t \tag{19}$$

which is a rescaling of the components of $K(u^t) - (\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})$. Then, $K(u^t) - (\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})$ is positive semidefinite if and only if M is, and

$$\begin{aligned} v^T M v &= \sum_{ab} v_a M_{ab} v_b \\ &= \sum_{ab} u_a^t(\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})_{ab}u_b^t v_a^2 \\ &- v_a u_a^t(\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})_{ab}u_b^t v_b \\ &= \frac{1}{2}\sum_{ab}(\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})_{ab}u_a^t u_b^t[v_a^2 + v_b^2 - 2v_a v_b] \\ &= \frac{1}{2}\sum_{ab}(\mathbf{S}^T\mathbf{S} + \lambda u'^T u'\mathbf{I})_{ab}u_a^t u_b^t(v_a - v_b)^2 \\ &\geq 0 \end{aligned} \tag{20}$$

□

Now we are ready for the proof of the updating rule for $\mathbf{P}$ in Lemma 2.

Proof. Applying update rule (13) to the auxiliary function (14) results in:

$$u^{t+1} = u^t - K(u^t)^{-1}\nabla F(u^t) \tag{21}$$

Lemma 3 guarantees that F is non-increasing under this update rule. Writing the component of this equation explicitly, we obtain

$$\mathbf{P}_{ij}^{t+1} = \mathbf{P}_{ij}^t \frac{(\mathbf{XS})_{ij}}{(\mathbf{PS}^T\mathbf{S} + \lambda\mathbf{P'P'}^T\mathbf{P})_{ij}} \tag{22}$$

The proof of update rule for $\mathbf{S}$ is the same as that in [17]. □

The approach to determine λ is more straightforward. Since λ only controls the convergence rate of $\mathbf{P}$ and $\mathbf{S}$, theoretically its value does not influence the final convergence of $\mathbf{P}$ and $\mathbf{S}$. We set the course to regularize λ by using standard Lagrange multiplier procedure. Let

$$\frac{\partial \mathcal{L}(\mathbf{P},\mathbf{S})}{\partial \mathbf{P}} = \mathbf{P}\mathbf{S}^T\mathbf{S} - \mathbf{X}\mathbf{S} + \lambda \mathbf{P}'\mathbf{P}'^T\mathbf{P} = 0$$

We then get $\lambda = \frac{(\mathbf{XS}-\mathbf{PS}^T\mathbf{S})_{ij}}{(\mathbf{P}'\mathbf{P}'^T\mathbf{P})_{ij}}$. During the update, each entry of $\mathbf{P}$ and $\mathbf{S}$ may have different gradient speed, we therefore set λ to be the average

$$\lambda = \frac{1}{mk}\sum_{ij}\frac{(\mathbf{XS}-\mathbf{PS}^T\mathbf{S})_{ij}}{(\mathbf{P}'\mathbf{P}'^T\mathbf{P})_{ij}}$$

as given in Lemma 7

8. REFERENCES

[1] A. Banerjee, S. Merugu, I.S.Dhillon, and J.Ghosh. Clustering with Bregman divergence. *Journal of Machine Learning Research*, 6:1705–1749, 2005.

[2] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavaldà. New ensemble methods for evolving data streams. In *KDD '09*, pages 139–148, 2009.

[3] Y. Chi, B. Tseng, and J. Tatemura. Eigen-trend: trend analysis in the blogosphere based on singular value decomposition. In *CIKM*, pages 68–77, 2006.

[4] C. Ding, T. Li, and M. Jordan. Nonnegative matrix factorization for combinatorial optimization: Spectral clustering, graph matching, and clique finding. In *ICDM*, pages 183–192, 2008.

[5] A. Dries and U. Rückert. Adaptive concept drift detection. *Statistical Analysis and Data Mining*, 2(5-6):311–327, 2009.

[6] R. Ge, M. Ester, B. Gao, Z. Hu, B. Bhattacharya, and B. Ben-Moshe. Joint cluster analysis of attribute data and relationship data: The connected k-center problem algorithms and applications. *Trans. on Knowledge discovery from Data*, 2:1–35, 2008.

[7] G. H. Golub and C. F. V. Loan. *Matrix computation.* The Johns Hopkins University Press, Baltimore and London, 1996.

[8] G. Gordon. Generalized^2 linear^2 models. In *NIPS*, 2002.

[9] D. He and D.Parker. Topic dynamics: An alternative model of 'bursts' in streams of topics. In *KDD*, pages 443–452, 2010.

[10] S. Hido, T. Ide, H. Kashima, H. Kubo, and H. Matsuzawa. Unsupervised change analysis using supervised learning. In *Advances in Knowledge Discovery and Data Mining*, pages 148–159. Springer Berlin / Heidelberg, 2008.

[11] G. Hulten, L. Spencer, and P. Domingos. Mining time-changing data streams. In *KDD*, pages 97–106, 2001.

[12] D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. In *VLDB*, pages 180–191, 2004.

[13] R. Klinkenberg. Learning drifting concepts: Example selection vs. example weighting. *Intelligent Data Analysis*, 8(3):697–717, 2004.

[14] K. Lang. http://people.csail.mit.edu/jrennie/20newsgroups/.

[15] L.Chen and A.Roy. Event detection from flickr data through wavelet-based spatial analysis. In *CIKM*, pages 523–532, 2009.

[16] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[17] D. Lee and H. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[18] B. Long, Z. Zhang, and P. Yu. Co-clustering by block value decomposition. In *KDD*, 2005.

[19] B. Long, Z. Zhang, and P. Yu. Unsupervised learning on k-partite graphs. In *KDD*, pages 317–326, 2006.

[20] P. Miettinen. *Matrix Decomposition Methods for Data Mining: Computational Complexity and Algorithms.* Helsinki University Print, 2009.

[21] K. Nishida and K. Yamauchi. Detecting concept drift using statistical testing. In *Proceedings of the 10th international conference on Discovery science*, DS'07, pages 264–269, Berlin, Heidelberg, 2007. Springer-Verlag.

[22] D. Preston, P. Protopapas, and C. Brodley. Event discovery in time series. In *SDM*, pages 61–72, 2009.

[23] G. Seber and A. Lee. *Linear Regression Analysis.* Wiley, 2003.

[24] A. P. Singh and G. Gordon. A unified view of matrix factorization models. In *ECML PKDD*, 2008.

[25] X. Song, M. Wu, C. Jermaine, and S. Ranka. Statistical change detection for multi-dimensional data. In *KDD*, 2007.

[26] A. Tsymbal. The problem of concept drift: Definitions and related work. Technical report, 2004.

[27] M. van Leeuwen and A. Siebes. Streamkrimp: Detecting change in data streams. In *ECML*, 2008.

[28] V. N. Vapnik. *Statistical Learning Theory.* John Wiley & sons, 1998.

[29] J. Vreeken, M. Leeuwen, and A. Siebes. Characterising the difference. In *KDD '07*, pages 226–235, 2007.

[30] H. Wang, W. Fan, P. Yu, and J. Han. Mining concept-drifting data streams using ensemble classifiers. In *KDD '03*, pages 226–235, 2003.

[31] P. Zhang, X. Zhu, and Y. Shi. Categorizing and mining concept drifting data streams. In *KDD '08*, pages 812–820, 2008.