The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Latent Discriminant Subspace Representations for Multi-View Outlier Detection

Kai Li,[†] Sheng Li,[⋆] Zhengming Ding,[†] Weidong Zhang,[⋆⋆] Yun Fu[†‡]
[†]Department of Electrical & Computer Engineering, Northeastern University, Boston, USA
[⋆]Adobe Research, USA
[⋆⋆]AI & Big Data Division, JD.COM American Technologies Corporation, USA
[‡]College of Computer & Information Science, Northeastern University, Boston, USA
kaili@ece.neu.edu, sheli@adobe.com, allanding@ece.neu.edu, weidong.zhang@jd.com, yunfu@ece.neu.edu

Abstract

Identifying multi-view outliers is challenging because of the complex data distributions across different views. Existing methods cope this problem by exploiting pairwise constraints across different views to obtain new feature representations, based on which certain outlier score measurements are defined. Due to the use of pairwise constraint, it is complicated and time-consuming for existing methods to detect outliers from three or more views. In this paper, we propose a novel method capable of detecting outliers from any number of data views. Our method first learns latent discriminant representations for all view data and defines a novel outlier score function based on the latent discriminant representations. Specifically, we represent multi-view data by a global low-rank representation shared by all views and residual representations specific to each view. Through analyzing the view-specific residual representations of all views, we can get the outlier score for every sample. Moreover, we raise the problem of detecting a third type of multi-view outliers which are neglected by existing methods. Experiments on six datasets show our method outperforms the existing ones in identifying all types of multi-view outliers, often by large margins.

Introduction

Outlier detection, or anomaly detection, is a basic data analysis technique which aims to identify the abnormal objects in a dataset. It is widely applied in many fields, such as web spam detection (Spirin and Han 2012), information disparity management (Duh et al. 2013), network failure detection (Ding et al. 2012). Many outlier detection methods have been proposed over the past decades (Zhou, Yang, and Yu 2012; Schubert, Zimek, and Kriegel 2014). These methods first analyze the distribution of a dataset, and then define certain criteria to identify outliers within it. However, it is worthy to notice that these methods are targeted for data of only one source, i.e., single-view data.

On the other hand, in numerous practical scenarios, data are from various sources or acquired by different feature extractors and shows heterogeneous characteristics. For example, a person can be uniquely identified by his or her face, fingerprint, iris or signature; while an image could be described by the color or texture. These multi-view features represent the same instance from different perspectives, thereby providing complementary information for the instance. Taking advantages of these complementary information across various views, multi-view based algorithms often reach better performances than their single-view counterparts (Ding and Fu 2014; 2016). However, detecting outliers from multi-view data remains challenging, due to the complex distributions and inconsistent behaviors of multi-view data samples.

A number of multi-view outlier detection methods have been proposed to detect outliers that have abnormal behaviors in each view (Muller et al. 2012; Das et al. 2010) or have inconsistent behaviors across different views (Gao et al. 2011; Liu and Lam 2012; Iwata and Yamada 2016; Li, Shao, and Fu 2015; Zhao and Fu 2015). These methods usually compose of two parts: multi-view model formulation and outlier score definition. In the multi-view model formulation stage, the goal is to exploit the consensus nature of multi-view data and uncover the subtle cross-view differences. The outlier score definition stage is to define some outlier measurements based on the cross-view differences revealed in the previous stage. In both stages, existing methods only consider pairwise relationship between views, making them hard to be extended into three or more views.

Two types of multi-view outliers have been addressed by existing methods. They are attribute outliers and class outliers, following the terminology in (Zhao and Fu 2015). Attribute outliers are samples which have abnormal behaviors in each view, and they will be considered as outliers in every view. The red triangles in Figure 1 represent this type of outliers. Class outliers are the data samples which exhibit inconsistent characteristics (mainly referring to cluster memberships) across different views. When considered in each view individually, this kind of samples would not be identified as outliers because they have normal characteristics as other samples within each view. But when considering their mutual behaviors across views, we will identify them as anomalies because they do not behave consistently across views, not like the inlier samples. The green circles in Figure 1 illustrate this type of outliers. It is easy to find that there is a third type of outliers neglected by existing methods. It is a mix of the above two types of outliers: the samples exhibit class outlier characteristics in some views, while shows attribute outlier properties in the other views.

In other words, they are the samples which exhibit normal within-view behaviors whereas inconsistent cross-view behaviors in some views, but consistently behave abnormally in the other views. For terminological convenience, we call this type of outlier as *class-attribute outlier*. Illustration of this type of outliers is shown in Figure 1, indicated by the blue squares.

In this paper, we propose a new multi-view outlier detection algorithm which is able to detect all the three types of outliers simultaneously, and is essentially extensible to deal with any number of views. We achieve this by learning a latent discriminant representation for each view data and defining a novel outlier score function based on the latent discriminant representations of all views. Specifically, we represent the input multi-view data by a global low-rank representation shared by all views and view-specific residual representations specific to each view. Through analyzing the view-specific residual representations for all views, we can get the outlier scores for the samples. Our major contributions are outlined as:

- We develop a new model for outlier detection from multi-view data. Our model represents each view data by a global low-rank subspace representation and a latent view-specific subspace representation. The global low-rank subspace representation encodes the consensus information shared by all views, while the latent subspace representation encodes discriminant information specific to each view. In this way, we do not need to encode cross-consistency in a pairwise fashion, allowing our model to conveniently handle three or more view data.
- We define an outlier score measurement by using the latent discriminant representations of all views. Thanks to the avoidance of calculating outlier score by permuting view pairs, our outlier score measurement can better handle the heterogeneity of cross-view data, thus being more reliable for data of three or more sources. Moreover, our outlier score measurement can easily handle multi-view data of different dimensions and is totally unsupervised.
- To the best our knowledge, we are the first to raise the problem of detecting the third type of outlier, as a complementation of the existing two types. We believe this complementation is beneficial for the community to develop more complete and reliable multi-view outlier detection algorithms and systems.

Related Works

In this section, we introduce two most relevant research topics to our approach, including multi-view outlier detection and multi-view subspace learning.

Multi-view Outlier Detection. Traditional multi-view outlier detection approaches focus on detecting outliers that exhibit abnormal behaviors in each view (Das et al. 2010; Gao et al. 2010; Janeja and Palanisamy 2013). Recently, a new branch of multi-view outlier detection methods have been proposed (Gao et al. 2011; Liu and Lam 2012; Marcos Alvarez et al. 2013). These methods try to find the samples that have inconsistent cross-view cluster memberships. HOrizontal Anomaly Detection (HOAD) (Gao et al. 2011) pioneers this branch of methods. It firstly computes spectral embeddings with an ensemble similarity matrix, and then calculates the outlier score with the cosine distance between different embeddings. Subsequent works utilize sophisticated machine learning algorithms to detect inconsistent characteristics for each object, e.g., consensus clustering (Liu and Lam 2012), affinity propagation (Marcos Alvarez et al. 2013), and probabilistic latent variable models (Iwata and Yamada 2016). To identify two types of outliers simultaneously, $l_{2,1}$-norm induced error terms are integrated into low-rank subspace learning (Li, Shao, and Fu 2015) and K-means clustering (Zhao and Fu 2015). Our method detects the three types of outliers (see the introduction above) simultaneously and solves the problem of existing methods in dealing with three or more view data by learning latent discriminative subspace representations for multi-view data.

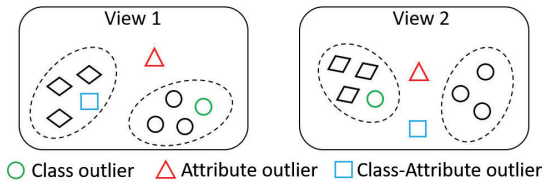


Figure 1: Illustration of three types of outliers.

Multi-view Subspace Clustering. Subspace clustering methods generally hold the assumption that data samples are drawn from multiple subspaces corresponding to different clusters. Recently, the subspace clustering based on self representation has been proposed, where each data point in a dataset can be expressed as a linear combination of the samples within the dataset (Liu, Lin, and Yu 2010; Elhamifar and Vidal 2013). Multi-view subspace clustering methods take advantages of the rich and complementary information of multi-view data for clustering task. (Guo 2013) formulates the multi-view subspace learning as a joint optimization for a consensus subspace representation matrix and a group sparsity inducing norm. (White et al. 2012) provide a convex reformulation of two-view subspace learning. Some methods tackle this problem from the view of dimensionality reduction, which typically learn a low-dimensional subspace from the multi-view data and employ existing clustering methods to get the results. The representatives of this line include (Chaudhuri et al. 2009; Blaschko and Lampert 2008), which project the multi-view high dimensional data onto a low dimensional subspace by exploiting canonical correlation analysis. These existing multi-view clustering methods have limitations of either targeting only from two-view cases, or are quite susceptible with the quality of original features, especially under the condition that the observations are insufficient and/or grossly corrupted (Cao et al. 2015). Our proposed multi-view subspace discovery model is specially designed to deal with corrupted data and is not limited by the number of views. We achieve this by expressing multi-view data with a common low-rank representation and view-specific representations where view-discriminant information (including outlier information) are encoded.

Algorithm

In this section, prior to presenting the proposed method, we introduce the preliminary knowledge about low-rank subspace analysis for the single-view case. Based on that, we introduce the proposed multi-view outlier detection algorithm.

Low-Rank Subspace Analysis

A dataset usually lies in an underlying low-dimensional subspace, rather than distributing uniformly in the entire space (Vidal and Favaro 2014). Thus, the data points can be represented by a low-dimensional subspace. Specifically, given a dataset $X = \{x_1, x_2, \ldots, x_v, \ldots, x_n\} \in \mathbb{R}^{d\times n}$, by exploiting the self-expressiveness property, the dataset can be represented as:

$$X = XZ + E, \tag{1}$$

where $Z = \{z_1, z_2, \ldots, z_i, \ldots, z_n\} \in \mathbb{R}^{n\times n}$ is the subspace representation matrix; each z_i is the representation of the original data point $x_i \in X$ based on the subspace. $E \in \mathbb{R}^{d\times n}$ is the error matrix. By assuming the samples in the same cluster could be drawn from the same subspace, Z should be a low-rank coefficient matrix that has the block-diagonal structure. In this way, we can learn a compact coefficient representation for the original data by solving the following problem:

$$\begin{aligned} \min_{Z,E} \quad & \|Z\|_* + f(E) \\ \text{s.t.} \quad & X = XZ + E, \end{aligned} \tag{2}$$

where $\|\cdot\|_*$ represents the trace norm (Candès et al. 2011). Trace norm is a commonly-used approximation of the non-convex rank$(\cdot)$ function. $f(E)$ is some regularization function of E, for example, $f(E) = \|E\|_{2,1}$ (Liu, Lin, and Yu 2010).

We extend this single-view low-rank analysis method to multi-view cases, by assuming that multi-view data can be represented by a global low-rank coefficient matrix and view-specific coefficient matrices.

Proposed Model

Denoted by $\mathcal{X} = \{X^1, X^2, \ldots, X^v, \ldots, X^V\}$ the collection of V view data, where $X^v \in \mathbb{R}^{d_v\times n}$ denotes the n samples of dimension d_v from the v-th view. In each view, we can learn a compact representation for the original data through the above low-rank subspace analysis. From the perspective of data representation, since the new representation of a data sample can be seen as the coefficient when the sample is expressed by all the samples, it is therefore reasonable to expect the expression coefficients for the same sample to be consistent across all views. This property can be exploited to guide the multi-view subspace discovery process to learn consistent cross-view representations.

However, when the data are corrupted by outliers, the cross-view consistency property of multi-view coefficient matrices may not be well preserved, especially when the data are corrupted by class or class-attribute outliers because they behave normally as inliers in some views and are hard to be excluded for data representation. To cope this problem, we formulate a robust multi-view subspace discovery model as:

$$\begin{aligned} \min_{Z_c, Z_r^v, E^v} \quad & \|Z_c\|_* + \alpha \sum_{v=1}^{V} ||Z_r^v||_{2,1} + \beta \sum_{v=1}^{V} ||E^v||_{2,1} \\ \text{s.t.} \quad & X^v = X^v Z_c + X^v Z_r^v + E^v, \\ & \forall v = \{1, 2, \ldots, V\}, \end{aligned} \tag{3}$$

where Z_c is the coefficient matrix shared by all views, and Z_r^v is the residual coefficient matrix specific to the v-th view. E^v is the error matrix for the v-th view.

We constrain the view-invariant coefficient $\|Z_c\|_*$ to be low-rank, following the traditional single-view low-rank subspace clustering algorithms. By taking advantages of the good power of $l_{2,1}$-norm in feature selection and error modeling (Nie et al. 2010), we add $\|E^v\|_{2,1}$ in our objective function. In fact, Z_r^v can be viewed as the error component for the coefficient matrix of the v-th view data. So, we constrain it with $\|Z_r^v\|_{2,1}$ in our objective function as well. Through representing the multi-view data with common coefficient matrix and view-specific matrices, and constrain them with low-rank and $l_{2,1}$-norm, respectively, we avoid to encode the consistency of multi-view data in a pairwise manner, making our model extensible to any number of views.

One can observe that we represent each view data X_v by a common coefficient matrix Z_c shared by all views and a latent view-specific coefficient matrix Z_r^v in a self-expression fashion, plus a view-specific error matrix E^v. The common coefficient matrix Z_c encodes the information sharable across all views, while the view-specific coefficient matrix Z_r^v represents the discriminant information related only to the current view. As discussed above, class outliers and class-attribute outliers have inconsistent cluster membership in different views. This inconsistency will be reflected in their corresponding view-specific coefficient matrices, thus facilitating us to figure out the two types of outliers. The view-specific error matrix E^v encodes gross errors in each view data, from which we identify the attribute outliers and class-attribute outliers.

Outlier Score Measurement

As analyzed above, the view-specific coefficient matrices $\{Z_r^v\}_{v=1}^V$ encode the discriminant information for the multi-view data, and the cross-view inconsistency can be obtained by analyzing them. Therefore, class outliers and class-attribute outliers can be identify from $\{Z_r^v\}_{v=1}^V$. Meanwhile, $\{E_r^v\}_{v=1}^V$ encode the gross outliers, from which, we can find out attribute outliers and class-attribute outliers. Denoted by $s(i)$ the outlier score for the i-th sample, we define a novel outlier score function as:

$$s(i) = \sum_{v=1}^{V} \|Z_r^{v,i}\|_2^2 + \lambda \|E^{v,i}\|_2^2, \tag{4}$$

where $Z_r^{v,i}$ and $E^{v,i}$ are the i-th columns Z_r^v and E^v, respectively; λ is a balancing parameter.

Since the new representations of the i-th sample from all views are $\{Z_c^i + Z_r^{v,i}\}_{v=1}^V$, the first term $\sum_{k=1}^V \|Z_r^{v,i}\|_2^2$ can

be viewed as the squared error of the sample's new representations. Similarly, the second term $\sum_{k=1}^{V} \|E^{v,i}\|_2^2$ is the squared error of the error vectors for the i-th sample from all view, under the zero-mean assumption of the gross outliers. Squared error measures the distribution compactness of a group of data points, so that the smaller $s(i)$ is, the less likely the i-th sample is an outlier.

Our outlier score measurement avoids a series of problems existing in the previous methods. First, our measurement can easily evaluate outlier scores of samples with more than two views. Previous methods (Li, Shao, and Fu 2015; Zhao and Fu 2015) evaluate outlier scores of samples in pairwise manners, so that it is cumbersome to permute all view pairs and calculate the outlier scores. Second, our outlier score function is completely unsupervised, unlike some existing methods (Li, Shao, and Fu 2015; Zhao and Fu 2015) which utilize class information to boost the outlier detection performances. Third, some existing methods use element-wise multiplication of cross-view data vectors to calculate outlier scores, which limits their methods to be applicable to only multi-view data of the same same dimension (Li, Shao, and Fu 2015; Gao et al. 2011). In contrast, our outlier score measurement does not have such a limitation.

Discussion

Our proposed method inherits some ideas from two recent multi-view outlier detection methods, MLRA (Li, Shao, and Fu 2015) and DMOD (Zhao and Fu 2015). We all target to detect multiple types of outliers simultaneously from the perspective of feature representation: both MLRA and our method learn subspace feature representations, while DMOD takes the cluster indicator matrices as the new feature representations. All three methods use $l_{2,1}$-norm to constrain the error matrices. Our method and MLRA both use low-rank constraint to reveal the intrinsic structure of data.

However, our method differs from MLRA and DMOD in the following aspects: (1) Our model is essentially extensible to deal with any number of data views, while it is hard for MLRA and DMOD to do so. This is because we encode cross-view consistency via learning a common representation shared by all view data. In contrast, the cross-view consistency is guaranteed by enforcing pairwise similarity in MLRA and DMOD. (2) Our outlier score function is able to evaluate three or more view data of different dimensions and is totally unsupervised, while those of MLRA and DMOD are not. Our novel outlier score function does not rely on pairwise analysis of the new representations as what MLRA and DMOD do, so that it can calculate the outlier score for each sample of any number of data sources. Meanwhile, MLRA uses dot production to evaluate the correlation of a pair of feature vectors, which requires the feature vectors being of the same dimension. So, MLRA is unable to handle multi-view data of different dimensions. We instead calculate outlier score for each sample simply by calculating the square errors of the new representations and the error vectors of samples, so that our method is able to handle multi-view data of various dimensions. Furthermore, class information are both utilized in MLRA and DMOD for boosting the outlier detection performance, which makes their methods not fully unsupervised. Our outlier score function on the other hand is totally unsupervised, which fits better for practical applications. (3) MLRA and DMOD is designed to detect the two types of multi-view outliers, while our method targets for detecting the three types of outliers simultaneously.

Optimization

We have presented above the proposed multi-view outlier detection model and the corresponding outlier score measurement. In this part, we introduce the details about how to optimize the model and analyze the time complexity.

Our model in (3) is not jointly convex with respect to all the variables, so that it is hard to global optimizer for it. So, we adopt the famous inexact augmented Lagrange multiplier (ALM) algorithm for efficiently optimizing it (Lin, Chen, and Ma 2010). By introducing a relaxation variable J, we rewrite our objective function as:

$$\begin{aligned} \min_{Z_c, Z_r^v, J, E^v} \quad & \|J\|_* + \alpha \sum_{v=1}^{V} ||Z_r^v||_{2,1} + \beta \sum_{v=1}^{V} ||E^v||_{2,1} \\ \text{s.t.} \quad & X^v = X^v Z_c + X^v Z_r^v + E^v, \\ & \forall v = \{1, 2, \ldots, V\}, \\ & Z_c = J. \end{aligned} \tag{5}$$

To solve (5), we introduce Lagrange multipliers P and Q^v ($v = 1, 2, \ldots, V$), and formulate our objective function as:

$$\begin{aligned} \Phi = \|J\|_* + \langle P, Z_c - J \rangle + \tfrac{\mu}{2}||Z_c - J||_{\mathrm{F}}^2 + \\ \alpha \sum_{v=1}^{V} ||Z_r^v||_{2,1} + \beta \sum_{v=1}^{V} ||E^v||_{2,1} \\ + \sum_{v=1}^{V} (h(Z_c, Z_r^v, E^v, Q^v) - \tfrac{1}{\mu}\|Q^v\|_{\mathrm{F}}^2), \end{aligned} \tag{6}$$

where $h(Z_c, Z_r^v, E^v, Q^v) = \frac{\mu}{2}\|X^v - X^v Z_c - X^v Z_r^v - E^v + \frac{Q^v}{\mu}\|_{\mathrm{F}}^2)$, and $\mu > 0$ is a penalty parameter. $\langle \cdot \rangle$ represents the inner product of two matrices, i.e. $\langle A, B \rangle = \mathrm{tr}(A^\top \mathrm{B})$.

The variables in (6) can be alternatively optimized by fixing the others when optimizing one of them. The step-by-step optimization procedures are as follows.

Update J: By keeping only the terms relevant to J, we obtain

$$J = \arg\min_{J} \frac{1}{\mu}\|J\|_* + \frac{1}{2}\|J - (Z_c + \frac{P}{\mu})\|_{\mathrm{F}}^2. \tag{7}$$

The singular value thresholding (SVT) algorithm (Cai, Candès, and Shen 2010) can be employed to get optimal solution to this problem.

Update Z_c: Ignoring irrelevant terms with respect to Z_c in (6), we obtain

$$\begin{aligned} Z_c = \arg\min_{Z_c} \quad & \langle P, Z_c - J \rangle + \tfrac{\mu}{2}||Z_c - J||_{\mathrm{F}}^2 \\ & + \sum_{v=1}^{V} h(Z_c, Z_r^v, E^v, Q^v). \end{aligned} \tag{8}$$

Setting the derivative w.r.t. Z_c to be 0, we obtain the solutions as follows:

$$\begin{aligned} Z_c = \quad & \tfrac{1}{\mu}(\mathrm{I} + \sum_{v=1}^{V} X^{v\top} X^v)^{-1}(-P + \mu J + \\ & \mu \sum_{v=1}^{V} X^{v\top}(X^v - X^v Z_r^v - E^v + \tfrac{Q^v}{\mu})), \end{aligned} \tag{9}$$

Algorithm 1. *Optimization of* (3)

Input: Multi-view data $\mathcal{X} = \{X^1, X^2, \ldots, X^v, \ldots, X^V\}$, parameters α and β.

Initialize $Z_c = J = P = 0$,
$\{Z_r^v\}_{v=1}^V = \{E^v\}_{v=1}^V = \{Q^v\}_{v=1}^V = 0$,
$\rho = 1.3$, $\mu = 10^{-4}$, $\mu_{\max} = 10^{10}$, $\sigma = 10^{-6}$.

while *not converged* **do**

1: update J using (7).
2: update Z_c using (9).
3: update $\{Z_r^v\}_{v=1}^V$ using (11).
4: update $\{E^v\}_{v=1}^V$ using (13).
5: Update P and $\{Q^v\}_{v=1}^V$ using (14) and (15), respectively.
6: Update the penalty parameter μ by
$\mu = \min(\mu_{\max}, \rho\mu)$
7: Check the convergence conditions:
$||X^v - X^v Z_c - X^v Z_r^v - E^v||_\infty < \sigma$ and
$||Z_c - J||_\infty < \sigma$

end while

Output: Z_r^v, E^v

Update Z_r^v: Keeping the terms relevant only to Z_r^v, we have

$$Z_r^v = \arg\min_{Z_r^v} = \alpha\|Z_r^v\|_{2,1} + \tfrac{\mu}{2}h(Z_c, Z_r^v, E^v, Q^v). \tag{10}$$

Following (Nie et al. 2010), we can get the solution as follows

$$\begin{aligned} Z_r^v = \quad & -(2\alpha R_z + \mu X^{v\top} X^v)^{-1} \\ & \big(\mu X^{v\top}(X^v Z_c + E^v - X^v - \tfrac{Q^v}{\mu})\big). \end{aligned} \tag{11}$$

where R_z is a diagonal matrix, *i.e.*, $R_z = \mathrm{diag}(r_z^1, r_z^2, \cdots, r_z^n)$, with $r_z^i = \frac{1}{2\sqrt{\|Z_r^{v,i}\|_2^2 + \epsilon}}$. ϵ is a small constant used to avoid trivial solution, and $Z_r^{v,i}$ is the i-th row of Z_r^v.

Update E^v: Similarly, ignoring terms independent of E^v, we have

$$\begin{aligned} E^v = \arg\min_{E^v} \ & \tfrac{\beta}{\mu}||E^v||_{2,1} + \tfrac{1}{2}||E^v - (X^v - X^v Z_c \\ & - X^v Z_r^v + \tfrac{Q^v}{\mu})||_{\mathrm{F}}^2. \end{aligned} \tag{12}$$

The solution of this type of problems has been discussed in (Liu, Lin, and Yu 2010). Specifically, let $\Omega = X^v - X^v Z_c - X^v Z_r^v + \frac{Q^v}{\mu}$, the solution of E^v then has the following form:

$$e_i^v = \begin{cases} \frac{\|\Omega_i\| - \beta}{\|\Omega_i\|}\Omega_i, & \text{if } \beta < \|\Omega_i\|, \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

where e_i^v is the i-th column is of E^v.

Update P **and** Q^v: P and Q^v are multipliers, we update them as follows:

$$P = P + \mu(Z_c - J) \tag{14}$$

$$Q^v = Q^v + \mu(X^v - X^v Z_c - X^v Z_r^v - E^v). \tag{15}$$

The complete optimization procedures are outlined in **Algorithm 1**.

Table 1: Basic information of five datasets from UCI Machine Learning Repository.

	zoo	*letter*	*wine*	*wdbc*	*pima*
# class	7	26	3	2	2
# sample	101	1300	178	569	768
# feature	16	16	12	30	8

Complexity Study

In this part, we study the complexity of our model. There are three dominant time-cost components, i.e., low-rank optimization on J and matrix multiplication and matrix inverse on Z_c and Z_r^v. Specifically, low-rank optimization on $J \in \mathbb{R}^{n\times n}$ would cost $\mathcal{O}(n^3)$. When n is very large, this step would be very expensive. Fortunately, according to Theorem 4.3 of (Liu et al. 2013), the SVD for J could be sped up to $\mathcal{O}(rn^2)$ where r is the rank of J. Each of the general multiplication takes $\mathcal{O}(n^3)$. The inverse operators also cost $\mathcal{O}(n^3)$. Thus, the time complexity of the steps 2-3 (Algorithm 1) is $\mathcal{O}(n^3)$.

Experimental Results

We employ six datasets for performance evaluation. Among them, five come from UCI Machine Learning Repository[1], i.e., *zoo*, *letter*, *wine*, *wdbc*, and *pima*. Table 1 shows the basic information about the five datasets. One thing worth to be noted is that there are totally 20000 samples for the 26 letters in the *letter* dataset, with each letter containing 700∼900 samples. To save evaluation time, following the strategy of (Li, Shao, and Fu 2015), we randomly select 50 samples for each letter, producing a subset of 1300 samples for our experiments. The last dataset is *BUAA VisNir*, which comprises of facial images of 150 persons, with 9 visual and 9 near infrared images for each person. The faces for the same identity can be considered as two different views, as they were collected in different conditions. Following previous methods, we vectorize the image pixel intensity values and project the feature vectors into a 100-dimensional latent space by PCA (Zhao and Fu 2015).

We employ four multi-view outlier detection algorithms for comparison: HOrizontal Anomaly Detection (HOAD) (Gao et al. 2011), anomaly detection via Affinity Propagation (AP) (Marcos Alvarez et al. 2013), Multi-view Low-Rank Analysis (MLRA) (Li, Shao, and Fu 2015), and Dual regularized Multi-view Outlier Detection (DMOD) (Zhao and Fu 2015). For AP, we utilize the l_2 distance with HSIC to yield better performance. For all methods, we carefully tune the parameters and report the best results.

All the six datasets are naturally outlier-free. To generate the three types of outliers, we follow (Gao et al. 2011) and pre-process the data as follows: First, we split the feature vectors into V subsets ($V \geqslant 2$); each subset is considered as one data view. This feature splitting procedure is not necessary for the *BUAA VisNir* dataset, because it naturally composes of data from two sources. Second, we generate the three types of outliers: For the class outlier, we

[1] http://archive.ics.uci.edu/ml/

Table 2: AUC values (mean ± standard deviation) on five datasets with outlier level **5%** for each of the three types. The best and second best results are in red and blue, respectively.

	Two views					Three views				
	zoo	*wine*	*wdbc*	*pima*	*letter*	*zoo*	*wine*	*wdbc*	*pima*	*letter*
HOAD	0.63 ± 0.05	0.74 ± 0.08	0.67 ± 0. 07	0. 67 ± 0.10	0.31 ± 0.05	0. 58 ± 0.08	0.62 ± 0.10	0.66 ± 0.07	0.61 ± 0.19	0.28 ± 0.09
AP	0.85 ± 0.03	0.79 ± 0.03	0.98 ± 0.01	0. 74 ± 0.02	0.89 ± 0.01	0.78 ± 0.06	0.73 ± 0.03	0.91 ± 0.01	0.52 ± 0.02	0.75 ± 0.02
DMOD	0.76 ± 0.11	0.84 ± 0.15	0.86 ± 0.04	0.77 ± 0.07	0.85 ± 0.01	0.75 ± 0.06	0.84 ± 0.04	0.84 ± 0.04	0.80 ± 0.02	0.79 ± 0.01
MLRA	0.74 ± 0.05	0.83 ± 0.04	0.88 ± 0.02	0.77 ± 0.04	0.80 ± 0.02	0.74 ± 0.08	0.84 ± 0.03	0.84 ± 0.02	0.76 ± 0.02	0.79 ± 0.02
Ours	0.89 ± 0.04	0.89 ± 0.03	0.98 ± 0.01	0.85 ± 0.01	0.91 ± 0.01	0.86 ± 0.04	0.88 ± 0.03	0.97 ± 0.01	0.83 ± 0.02	0.87 ± 0.02

Table 3: AUC values (mean ± standard deviation) on the *BUAA VisNir* dataset with outlier level **5%** for each of the three types.

	AUC (mean ± standard deviation)
HOAD	0.71 ± 0.02
AP	0.96 ± 0.01
DMOD	0.89 ± 0.01
MLRA	0.90 ± 0.09
Ours	0.99 ± 0.01

Table 4: Average AUC values with two types of outliers. The outlier level is **10%** for each type.

	HOAD	AP	DMOD	MLRA	Ours
zoo	0.63	0.80	0.71	0.72	0.85
wine	0.49	0.74	0.78	0.77	0.80
wdbc	0.57	0.97	0.76	0.76	0.97
pima	0.60	0.68	0.77	0.73	0.80
letter	0.38	0.86	0.81	0.77	0.88
BUAA-VisNir	0.71	0.91	0.78	0.86	0.96

randomly select two objects from two different classes and swap their feature vectors in $\lfloor \frac{V}{2} \rfloor$ view(s) but not in the other view(s). For attribute outlier, we randomly choose a sample, and replace its feature in all views by random values. For class-attribute outlier, we take two objects from two different classes and swap the feature vectors in $\lfloor \frac{V}{2} \rfloor$ view(s), while replace the feature vectors of the two classes with random values in the other view(s).

We follow previous methods (Li, Shao, and Fu 2015) and use AUC (area under ROC curve) as performance evaluation metric. The same evaluation procedure are also adopted, by randomly generating outliers for 50 times, evaluating each method on the 50 sets, and reporting the average AUC value.

UCI Datasets

Table 2 shows the AUC values (mean ± standard deviation) on the five UCI datasets. The two-view cases and three-view cases correspond to two and three splittings of the original features, respectively. Please note that the implementations of all baseline methods are designed for two-view data. Their AUC values for three-view data are generated by averaging the AUC values from all three pairs of views.

From Table 2, we can observe that the proposed method consistently outperforms all the baseline methods, often by large margins, in both two- and three-view cases. Some baseline methods can reach comparable performances with ours in some datasets, but perform much worse than ours in the other datasets. A typical example is AP, which reaches the same AUC value as ours in the *wdbc* dataset for the two-view case, but its AUC value is more than 30 percents inferior to ours in the *pima* dataset for the three-view case.

Comparing the AUC values for the two-view cases and three-views cases, we observe that there are performance drops for most methods in the majority of the datasets. The reasons could be as follows: (1) The feature dimension in each view for the three-view case is smaller than that in the two-view case, leading to vaguer data grouping structure, thereby inferior outlier assignment results. (2) The ratio of abnormal features versus overall feature in the three-view case is lower than that in the two-view case. For the class and class-attribute outliers, half features are swapped in the two-view case, while only 1/3 features are swapped for the three-view case. The higher abnormal feature rate makes it is easier to detection outliers from the two-view data.

In some datasets, some methods perform even better in the three-view case than that in the two-view case. Apart from the reasonable fluctuations caused by AUC value averaging, we speculate another reason may be that these methods favor the feature structures for two-view case than that in the three-view case.

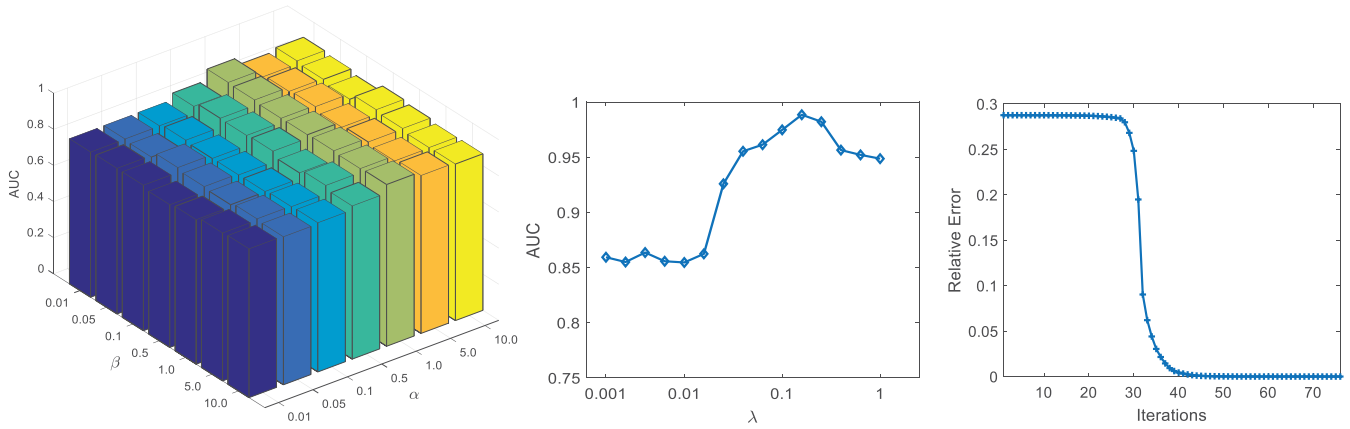
It is noticed that our method is much more stable than the baseline methods when extended from two views to three views. This is within our expectation because our method is essentially extensible to any number of data views.

BUAA-VisNir Dataset

Table 3 shows the experimental results on the *BUAA VisNir* dataset. We can see that our proposed method identifies nearly all the outliers and gains AUC value of 99.4%. One possible reason for the near-perfect performance is that both the visual data and near infrared data are of favorable grouping structures, so that the outliers can be easily identified.

Analytical Experiments

Impact of Outlier Rate. We show in Table 2 the results when each of the three types of outliers account for 5% of the total samples. To further evaluate the robustness of our method to outliers, we experiment on data corrupted by higher percentages of outliers. As shown in Figure 3, we report the results on the *zoo* dataset with outlier rate of 5%,



(a) (b) (c)

Figure 2: Analytical experiments on the *zoo* dataset. (a): The AUC values with different values of α and β. (b): The AUC values with different values of λ. (c): Convergence curve of the proposed method.

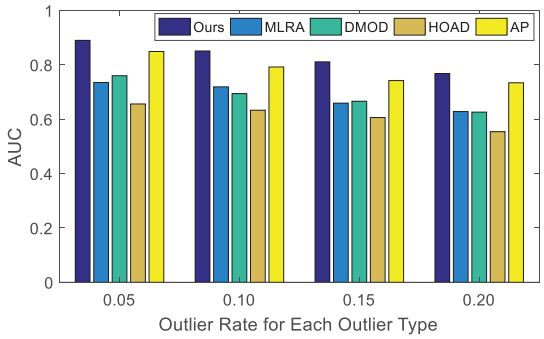


Figure 3: The AUC values with different percentages of outliers for each outlier type.

10%, 15% and 20% for each of the three types of outliers. We can observe that our proposed method consistently outperforms the others with all outlier rates. This substantiates the robustness of the proposed method.

Performance with Two Types of Outliers. The above experiments show that our method outperforms existing methods in detecting three types of outliers simultaneously. Since the third type of outliers are introduced firstly in this paper and are not what existing methods target for, we conduct an additional set of experiments to show our superior performances are not merely due to the introduction of the third type of outliers. We corrupt the six datasets only by class and attribute outliers and test if our method has advantages over existing ones in their favored setting. Table 4 shows the results with outlier level is 10% for both outlier types. We can see that our method still outperforms the existing methods in this setting. This further proves the superiority of our method in detecting multi-view outliers.

Parameter Analysis. Our method has three major parameters, α, β, and λ, where α and β are used to balance different terms of the objective function. λ is utilized in our outlier score function to weight different components. We first fix λ and evaluate the impact of α and β. Figure 2(a) shows the AUCs when we permute the combinations of α and β in $\{0.01, 0.05, 0.1, 0.5, 1.0, 5.0, 10\}$ on the *zoo* dataset. We can observe that our method is fairly robust with various values of α and β. We set $\alpha = 1$ and $\beta = 1$ as default.

We further evaluate the impact of λ on the performance with fixed α and β. Figure 2(b) shows the change of AUC with respect to different values of λ. We can see that the proposed method maintains good performances within a wide range for the value of λ. In practice, we choose $\lambda = 0.1$ as default. Note although we show here only the parameter analysis results on the *zoo* dataset, similar results can be obtained in the other datasets as well.

Convergence Analysis. To analyze the convergence property of the proposed model, we calculate the relative errors of the model on every view $\{\|X^v - X^v Z_c + X^v Z_r^v + E^v\|_F / \|X^v\|_F\}_{v=1}^V$ in each iteration. The change of the maximal relative error w.r.t. iteration among all views on the *zoo* dataset is shown in Figure 2(c). We can see that the maximal relative error decreases quickly first and remains stable. This shows the good convergence property of our model.

Running Time *w.r.t.* View Number. In the case where there are three or more views, our method does not rely on permuting all pairs of views. So the running time of our method would not increase quadratically w.r.t. the increase of view number. This is an appealing benefit of our method over the existing ones. To verify this, we test in the *zoo* dataset by splitting it into two and four subsets, and the corresponding running times of our method are 0.59 and 0.73, respectively. We can see that there is only 23% increase for the running time when the view number is doubled. This indicates the benefit of our method in dealing with data with many views.

Conclusions

We have presented in this paper a new multi-view outlier detection method, which avoids the limitation of the existing ones that is difficult to extend from two views to three or

more views. We achieve this by learning latent discriminant subspace representations for all view data and defining a novel outlier score function based on the latent subspace representations. Moreover, we raise the problem of detecting a new type of multi-view outliers neglected by existing methods. Experiments on six datasets show the proposed method has superior performances in detecting all three types of outliers, has good convergence property and performs more robustly under high outlier levels.

Acknowledgment

This research is supported in part by the NSF IIS award 1651902, ONR Young Investigator Award N00014-14-1-0484, and U.S. Army Research Office Award W911NF-17-1-0367. The majority of work was done when Kai Li was an intern at JD.COM American Technologies Corporation, USA.

References

Blaschko, M. B., and Lampert, C. H. 2008. Correlational spectral clustering. In *Proc. of CVPR*. IEEE.

Cai, J.-F.; Candès, E. J.; and Shen, Z. 2010. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization* 20(4):1956–1982.

Candès, E. J.; Li, X.; Ma, Y.; and Wright, J. 2011. Robust principal component analysis? *Journal of the ACM (JACM)* 58(3):11.

Cao, X.; Zhang, C.; Fu, H.; Liu, S.; and Zhang, H. 2015. Diversity-induced multi-view subspace clustering. In *Proc. of CVPR*.

Chaudhuri, K.; Kakade, S. M.; Livescu, K.; and Sridharan, K. 2009. Multi-view clustering via canonical correlation analysis. In *Proc. of ICML*. ACM.

Das, S.; Matthews, B. L.; Srivastava, A. N.; and Oza, N. C. 2010. Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study. In *Proc. of KDD*.

Ding, Z., and Fu, Y. 2014. Low-rank common subspace for multi-view learning. In *Proc. of ICDM*. IEEE.

Ding, Z., and Fu, Y. 2016. Robust multi-view subspace learning through dual low-rank decompositions. In *Proc. of AAAI*.

Ding, Q.; Katenka, N.; Barford, P.; Kolaczyk, E.; and Crovella, M. 2012. Intrusion as (anti) social communication: characterization and detection. In *Proc. of KDD*.

Duh, K.; Yeung, C.-M. A.; Iwata, T.; and Nagata, M. 2013. Managing information disparity in multilingual document collections. *ACM Transactions on Speech and Language Processing (TSLP)* 10(1):1.

Elhamifar, E., and Vidal, R. 2013. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(11).

Gao, J.; Liang, F.; Fan, W.; Wang, C.; Sun, Y.; and Han, J. 2010. On community outliers and their efficient detection in information networks. In *Proc. of KDD*. ACM.

Gao, J.; Fan, W.; Turaga, D.; Parthasarathy, S.; and Han, J. 2011. A spectral framework for detecting inconsistency across multi-source object relationships. In *Proc. of ICDM*. IEEE.

Guo, Y. 2013. Convex subspace representation learning from multi-view data. In *Proc. of AAAI*.

Iwata, T., and Yamada, M. 2016. Multi-view anomaly detection via robust probabilistic latent variable models. In *Proc. of NIPS*.

Janeja, V. P., and Palanisamy, R. 2013. Multi-domain anomaly detection in spatial datasets. *Knowledge and information systems* 36(3):749–788.

Li, S.; Shao, M.; and Fu, Y. 2015. Multi-view low-rank analysis for outlier detection. In *Proc. of SDM*.

Lin, Z.; Chen, M.; and Ma, Y. 2010. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*.

Liu, A. Y., and Lam, D. N. 2012. Using consensus clustering for multi-view anomaly detection. In *Proc. of IEEE Symposium on Security and Privacy Workshops*.

Liu, G.; Lin, Z.; Yan, S.; Sun, J.; Yu, Y.; and Ma, Y. 2013. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(1):171–184.

Liu, G.; Lin, Z.; and Yu, Y. 2010. Robust subspace segmentation by low-rank representation. In *Proc. of ICML*.

Marcos Alvarez, A.; Yamada, M.; Kimura, A.; and Iwata, T. 2013. Clustering-based anomaly detection in multi-view data. In *Proc. of CIKM*. ACM.

Muller, E.; Assent, I.; Iglesias, P.; Mulle, Y.; and Bohm, K. 2012. Outlier ranking via subspace analysis in multiple views of the data. In *Proc. of ICDM*.

Nie, F.; Huang, H.; Cai, X.; and Ding, C. H. 2010. Efficient and robust feature selection via joint 2, 1-norms minimization. In *Proc. of NIPS*.

Schubert, E.; Zimek, A.; and Kriegel, H.-P. 2014. Generalized outlier detection with flexible kernel density estimates. In *Proc. of SDM*. SIAM.

Spirin, N., and Han, J. 2012. Survey on web spam detection: principles and algorithms. *ACM SIGKDD Explorations Newsletter* 13(2):50–64.

Vidal, R., and Favaro, P. 2014. Low rank subspace clustering (lrsc). *Pattern Recognition Letters* 43:47–61.

White, M.; Zhang, X.; Schuurmans, D.; and Yu, Y.-l. 2012. Convex multi-view subspace learning. In *Proc. of NIPS*.

Zhao, H., and Fu, Y. 2015. Dual-regularized multi-view outlier detection. In *Proc. of IJCAI*.

Zhou, X.; Yang, C.; and Yu, W. 2012. Automatic mitral leaflet tracking in echocardiography by outlier detection in the low-rank representation. In *Proc. of CVPR*. IEEE.