JMLR: Workshop and Conference Proceedings 20 (2011) 47–62 Asian Conference on Machine Learning

Learning to Locate Relative Outliers

Shukai Li SKLI@NTU.EDU.SG and **Ivor W. Tsang** IVORTSANG@NTU.EDU.SG
School of Computer Engineering
Nanyang Technological University
Singapore 639798

Editor: Chun-Nan Hsu and Wee Sun Lee

Abstract

Outliers usually spread across regions of low density. However, due to the absence or scarcity of outliers, designing a robust detector to sift outliers from a given dataset is still very challenging. In this paper, we consider to identify *relative outliers* from the target dataset with respect to another reference dataset of normal data. Particularly, we employ Maximum Mean Discrepancy (MMD) for matching the distribution between these two datasets and present a novel learning framework to learn a relative outlier detector. The learning task is formulated as a Mixed Integer Programming (MIP) problem, which is computationally hard. To this end, we propose an effective procedure to find a largely violated labeling vector for identifying relative outliers from abundant normal patterns, and its convergence is also presented. Then, a set of largely violated labeling vectors are combined by multiple kernel learning methods to robustly locate relative outliers. Comprehensive empirical studies on real-world datasets verify that our proposed relative outlier detection outperforms existing methods.

Keywords: Relative novelty detection; Maximum Mean Discrepancy; Mixed Integer Programming; Multiple Kernel Learning

1. Introduction

Outlier detection refers to some observations that appear to be inconsistent with most of the set of data. These inconsistent observations are often regarded as novelties, outliers, anomalies, exceptions, contaminants, aberrations or contaminant in different applications (Chandola et al., 2009; Markou and Singh, 2003). Outlier detection has been extensively used in a wide variety of applications, such as in bioinformatics, the outliers come from abnormal patient conditions, instrumentation errors, and disease outbreaks in a specific area. Due to its popularity, this line of research has attracted lots of attentions.

In practice, obtaining labeled data of diversified novel behaviors is often prohibitively expensive. For example in the fault detection for aircraft engine, it is too costly to get outlier instances. Due to lack of prior knowledge for various novel behaviors, traditional outlier detection approaches fall into the category of unsupervised learning, such as the well-known One-Class Support Vector Machine (OCSVM) (Schölkopf et al., 2000). But this kind of methods fail to make use of the labeled data, especially large amount of normal instances. Considering this, several works (Gao et al., 2006; Blanchard et al., 2010; Wu and Ye, 2009) proposed semi-supervised and supervised algorithms for outlier detection. Because of making use of labeled instances, this kind of semi-supervised/supervised outlier

detection techniques often outperform the traditional unsupervised methods. However, the novel instances are quite limited, and the novel behavior is often dynamic in nature, e.g., new types of novelties might arise, for which there is no available labeled outliers for training. For instance, in the network intrusion detection, the nature of outliers keeps changing over time as the intruders adapt their network attacks to evade the existing intrusion detection solutions. Thus, the semi-supervised models, which learn from known kinds of outliers, do not fit for the practical requirements, and also fail to detect unknown kinds of outliers.

Considering that abundant normal instances are often easy to get, Smola et al. (2009), Kanamori et al. (2009) and Hido et al. (2008) proposed to use both the labeled normal instances (source domain) and unlabeled set (target domain) to predict the outliers in the target domain. This setting is more practical than the former unsupervised and semi-supervised outlier detection. In order to avoid the density estimation of normal instances, which is a well-known challenging problem in outlier detection, these models estimate the density ratio instead of density. However, estimating density ratio is still difficult, especially on high dimensional problems. As aforementioned, outliers are diversified and dynamic, and these outliers can still easily contaminate the estimate of density ratio. To completely avoid density or density-ratio estimation, learning a hyperplane classifier, such as OCSVM, which explicitly separates the outliers from normal instances, is more preferred. But this kind of methods usually does not consider the information of available normal instances.

Based on above considerations, this work examines a new learning paradigm for hyperplane based outlier detection that we only have examples of normal observations in a source domain, and have no labeled examples in the target domain. So in the source domain it provides knowledge for normal instances, and in the target domain it becomes unsupervised learning. In what follows, we transfer the knowledge from the source domain to the target domain by identifying normal instances and outliers from the target unlabeled dataset with respect to the reference dataset. Specifically, we use Maximum Mean Discrepancy (MMD) (Borgwardt et al., 2006) as the matching criterion so as to minimize the distribution difference between the two domains. Then, the selected normal instances and outliers from unlabeled target dataset are used for training the outlier detector. These two processes are seamlessly combined together to form a Mixed Integer Programming (MIP) problem. To this end, we present an efficient algorithm, namely Maximum Mean Discrepancy based Relative Outlier Detection (MMD-ROD), to solve a convex relaxation of this MIP problem. Comprehensive empirical studies on real-world datasets verify that our proposed MMD-ROD outperforms existing relative outlier detection methods.

2. Preliminaries and Related Work

In this paper, the transpose of vector/matrix is denoted by the superscript $'$. $\mathbf{I}_m \in \mathbb{R}^{m\times m}$ is the identity matrix, and $\mathbf{0}_m, \mathbf{1}_m \in \mathbb{R}^m$ denote the zero vector and the vector of all ones, respectively. The operator $\odot$ denotes the element-wise product between two vectors/matrices.

We suppose that a set of normal instances $\{\mathbf{x}_i^{sr}\}_{i=1}^m$ in the source domain D^{sr} under a reference distribution $q(\mathbf{x})$ (q for short) and another set of unlabeled instances $\{\mathbf{x}_i^{ta}\}_{i=1}^n$ in the target domain D^{ta} under the target distribution $p(\mathbf{x})$ (p for short) are given together with their corresponding label vectors $\mathbf{y}^{sr} = [y_1^{sr}, \ldots, y_m^{sr}]'$ and $\mathbf{y}^{ta} = [y_1^{ta}, \ldots, y_n^{ta}]'$, where

$\mathbf{x}_i^{sr}, \mathbf{x}_i^{ta} \in \mathcal{X}$ and $y_i^{sr}, y_i^{ta} \in \{\pm 1\}$. Without loss of generality, we assume the class label for normal instances is 1 and the class label for outliers is -1. We further denote $\mathbf{y} = [\mathbf{y}^{sr\prime}\ \mathbf{y}^{ta\prime}]'$.

2.1. Maximum Mean Discrepancy

Many parametric criteria (*e.g.* Kullback-Leibler (KL) divergence) have been used to measure the distance between data distributions. However, tedious density estimation is usually required as a prelude to estimate these criteria. To avoid such a non-trivial task, we first review an effective nonparametric criterion, namely Maximum Mean Discrepancy (MMD) (Borgwardt et al., 2006), which is used to measure the difference between two data distributions based on the distance between the means of samples from the source domain D^{sr} and the target domain D^{ta} in the Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$, namely:

$$\begin{aligned} \mathrm{MMD}_k(D^{sr}, D^{ta}) &= \sup_{\|h\|_{\mathcal{H}} \le 1} \left(\mathrm{E}_q[h(\mathbf{x}^{sr})] - \mathrm{E}_p[h(\mathbf{x}^{ta})]\right) \qquad (1) \\ &= \sup_{\|h\|_{\mathcal{H}} \le 1} \left\langle h, \left(\mathrm{E}_q[\phi(\mathbf{x}^{sr})] - \mathrm{E}_p[\phi(\mathbf{x}^{ta})]\right)\right\rangle_{\mathcal{H}} \\ &= \left\|\mathrm{E}_q[\phi(\mathbf{x}^{sr})] - \mathrm{E}_p[\phi(\mathbf{x}^{ta})]\right\|_{\mathcal{H}} \qquad (2) \end{aligned}$$

where $\mathrm{E}_u[\cdot]$ denotes the expectation operator under a distribution u, and $h(\mathbf{x})$ is any function in $\mathcal{H}$. The second equality holds as $h(\mathbf{x}) = \langle h, \phi(\mathbf{x})\rangle_{\mathcal{H}}$ by the property of RKHS (Schölkopf and Smola, 2002), where $\phi(\cdot)$ is a nonlinear feature mapping of a kernel k, *i.e.*, $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)$. Asymptotically, the empirical measure of MMD in (2) is well-estimated by:

$$\mathrm{MMD}_k(D^{sr}, D^{ta}) = \left\| \frac{1}{m}\sum_{i=1}^{m} \varphi(\mathbf{x}_i^{sr}) - \frac{1}{n}\sum_{i=1}^{n} \varphi(\mathbf{x}_i^{ta}) \right\|_{\mathcal{H}}. \qquad (3)$$

2.2. Relative Novelty Detection

To learn a decision function $h(\mathbf{x})$ for detecting novel instances in the unlabeled dataset relative to the reference dataset, referred to as *relative outliers*, Smola et al. (2009) proposed to estimate a truncation of log density ratio $\frac{p(\mathbf{x})}{q(\mathbf{x})}$ by introducing a transfer function $T_{nv}(\cdot)$ on the target unlabeled data:

$$T_{nv}(\zeta) = \begin{cases} \infty, & \text{if } \zeta > 0 \\ \zeta e^{\rho}, & \text{if } \zeta \in [-e^{-\rho}, 0] \\ -1 - \rho - \log(-\zeta), & \text{if } \zeta < -e^{-\rho} \end{cases} \qquad (4)$$

for the f-divergence (Nguyen et al., 2008) to measure the difference between p and q:

$$D_\phi(p, q) = \sup_h \left(\mathrm{E}_q[h(\mathbf{x}^{sr})] - \mathrm{E}_p[T_{nv}(h(\mathbf{x}^{ta}))]\right) \qquad (5)$$

where ρ is a threshold, and T_{nv} in (4) is the Fenchel-Legendre dual of a truncated log function ϕ. Essentially, they alter the symmetric MMD in (1) for measuring discrimination between the two datasets under an asymmetric setting. Moreover, the norm constraint $\|h\|_{\mathcal{H}} \le 1$ in (1) is replaced by a regularizer $\Omega(\|h\|_{\mathcal{H}})$ to control the complexity of $h(\mathbf{x})$. By replacing the expectation operator in (5) with the empirical mean from the samples, the decision function $h(\mathbf{x})$ is learned by minimizing the following structured functional:

$$\min_h \frac{1}{n}\sum_{i=1}^{n} T_{nv}(h(\mathbf{x}_i^{ta})) - \frac{1}{m}\sum_{i=1}^{m} h(\mathbf{x}_i^{sr}) + \Omega(\|h\|_{\mathcal{H}}). \qquad (6)$$

This is referred to as Truncated Kullback Leibler (TKL) divergence. However, as discussed in (Smola et al., 2009), the resultant problem (6) is solved by expensive non-convex optimization methods. Moreover, the estimation of h is still heavily affected by diverse outliers.

3. Learning to Locate Relative Outliers

In this section, we introduce our proposed Relative Outlier Detection method using Maximum Mean Discrepancy criterion, namely MMD-ROD, which identifies normal instances and outliers from the target domains using a given reference dataset of normal instances; meanwhile, it learns a decision function $f(\mathbf{x})$ for outlier detection using the chosen normal instances and outliers. And an efficient algorithm is also presented for MMD-ROD.

In Section 3.1, we first present an effective method, namely Normal Instance Matching (NIM), to identify normal instances from an unlabeled dataset such that the distribution of these normal instances becomes closer to that of another reference dataset of normal instances. In Section 3.2, we employ NIM as the criterion to learn a hyperplane classifier for detecting outliers with respect to a reference dataset of normal instances. Details of the algorithm to solve this learning problem are depicted in Section 3.3 and Section 3.4. Section 3.5 gives the prediction function of the relative outlier detector on unseen data. We study the complexity of the proposed method in Section 3.6.

3.1. Normal Instance Matching via Maximum Mean Discrepancy

Since we have little knowledge on the outliers, and even if we have some outliers in our training process, we may encounter other kinds of outliers, which are difficult and costly to collect. On the other hand, normal instances are abundant and easily annotated without much human effort, and it is relatively cheap to collect them. Thus, the setting with a given reference dataset of these normal instances fits to practice. In what follows, we use this reference dataset as the source to identify normal instances and outliers from an unlabeled data in the target domain. To this purpose, we have to define a criterion for matching the distribution between the two domains. As mentioned in Section 2.2, Maximum Mean Discrepancy (MMD) in an asymmetric setting has demonstrated promising results to detect relative outliers (Smola et al., 2009). However, their approach involves complicated and non-convex procedures to estimate the density ratio of the target domain to a source domain. In addition, density ratio estimation can be highly affected by diverse outliers, especially on high dimensional problems. Instead of estimating the density ratio, here we explicitly locate the relative outliers by introducing a vector of control variables $\mathbf{d} = [d_1, \ldots, d_n]'$ where $d_i \in \{0, 1\}$ to indicate normal instances by 1 and outliers by 0. Assuming there are ν fraction of outliers, from (3), this vector $\mathbf{d}$ can be obtained by minimizing:

$$\mathrm{MMD}_k(D^{sr}, D^{ta}; \mathbf{d}) = \Big\| \frac{1}{m} \sum_{i=1}^{m} \varphi(\mathbf{x}_i^{sr}) - \frac{1}{n(1-\nu)} \sum_{i=1}^{n} \varphi(\mathbf{x}_i^{ta}) d_j \Big\|_{\mathcal{H}}. \tag{7}$$

Note, ν can be deemed as a hyperparameter as in OCSVM (Schölkopf et al., 2000) and TKL (Smola et al., 2009), which can be determined by validation procedures using artificial outliers (Abe et al., 2006). In addition, when the actual outliers are chosen (*i.e.*, $d_i = 0$) in the unlabeled dataset, these outliers will be absent during computing the mean in (7). Thus, the estimation of (7) is more robust than the traditional density or density ratio estimation, in which their estimates are fairly sensitive to diverse outliers. Furthermore,

when this measure tends to zero, this indicates that the distribution of the chosen normal instances (*i.e.*, $d_i = 1$) from the target domain would match that of the normal instances in the source domain. Hence, this process is called Normal Instance Matching (NIM).

Here, we define a column vector $\mathbf{s}$ with $N = m + n$ entries, in which the first m entries are set as $1/m$ and the remaining entries are set as $-1/(n(1-\nu))$, respectively, and we also let $\tilde{\mathbf{d}} = [\mathbf{1}'_m \ \mathbf{d}']'$. Thus, the square of the MMD criterion in (7) can be rewritten as:

$$\mathrm{MMD}_k^2(D^{sr}, D^{ta}; \mathbf{d}) = \mathrm{trace}((\mathbf{K} \odot \tilde{\mathbf{d}}\tilde{\mathbf{d}}')\mathbf{S}) = \tilde{\mathbf{d}}'(\mathbf{K} \odot \mathbf{S})\tilde{\mathbf{d}}, \tag{8}$$

where $\mathbf{S} = \mathbf{s}\mathbf{s}' \in \mathbb{R}^{N\times N}$, and $\mathbf{K} = \begin{bmatrix} \mathbf{K}^{sr,sr} & \mathbf{K}^{sr,ta} \\ \mathbf{K}^{ta,sr} & \mathbf{K}^{ta,ta} \end{bmatrix} \in \mathbb{R}^{N\times N}$, and $\mathbf{K}^{sr,sr} \in \mathbb{R}^{m\times m}$, $\mathbf{K}^{ta,ta} \in \mathbb{R}^{n\times n}$ and $\mathbf{K}^{sr,ta} \in \mathbb{R}^{m\times n}$ are the kernel matrices defined for the source domain, the target domain and the cross-domain from the source domain to the target domain, respectively. And each entry inside these kernel matrices is equal to $k(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)'\varphi(\mathbf{x}_j)$.

3.2. Proposed Formulation

As mentioned in Section 1, to achieve good generalization on unseen data, a hyperplane based classifier $f(\mathbf{x}) = \mathbf{w}'\varphi(\mathbf{x})$ that separates outliers from normal data is desirable. Moreover, as shown in (Nie et al., 2011), learning the label of unlabeled data and the hyperplane classifier together can achieve better generalization performance. In this subsection, we present the proposed formulation for learning $f(\mathbf{x})$ to detect relative outliers from an unlabeled dataset D^{ta} using a reference dataset D^{sr} of normal instances as follows:

$$\min_{\mathbf{d}} \min_{\mathbf{w}} \quad \eta \mathrm{MMD}_k^2(D^{sr}, D^{ta}; \mathbf{d}) + \Omega(\|\mathbf{w}\|) + C\ell(D, \mathbf{y}; \mathbf{w}) \tag{9}$$

where $\eta > 0$ and $C > 0$ are tradeoff parameters. $\Omega(\cdot)$ is any increasing function for regularization, and $\ell(\cdot)$ is any convex loss function (*e.g.*, (square) hinge loss, logistic loss, etc). Moreover, D is a labeled dataset from the both domains (*i.e.*, $D = D^{sr} \cup D^{ta}$), and we also let $y_i = 2d_i - 1$ such that $y_i = 1$ represents a normal instance and $y_i = -1$ corresponds to an outlier. In particular, the resultant problem in (9) combines the objective of MMD and the structural risk minimization so as to learn $f(\mathbf{x})$ and $\mathbf{d}$ simultaneously. Furthermore, we can use any convex loss function in (9), for simplicity, we just present the case of square hinge loss here:

$$\min_{\mathbf{d}} \min_{\mathbf{w},\rho,\xi_i} \eta \mathrm{MMD}_k^2(D^{sr}, D^{ta}; \mathbf{d}) + \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{2}\sum_{i=1}^{N}\xi_i^2 - \rho \quad : \quad y_i\mathbf{w}'\varphi(\mathbf{x}_i) \geq \rho - \xi_i, \tag{10}$$

where $2\rho/\|\mathbf{w}\|$ represents the margin separating the normal instances and outliers. By replacing the inner minimization in (10) by its dual, and using (8) with $y_i = 2d_i - 1$, we have

$$\min_{\mathbf{y}\in\mathcal{Y}} \max_{\boldsymbol{\alpha}\in\mathcal{A}} \quad \frac{\eta}{4}(\mathbf{y} + \mathbf{1}_N)'(\mathbf{K} \odot \mathbf{S})(\mathbf{y} + \mathbf{1}_N) - \frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K} \odot \mathbf{y}\mathbf{y}' + \frac{1}{C}\mathbf{I}_N\Big)\boldsymbol{\alpha}. \tag{11}$$

where $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_N]'$, α_i is the dual variable for each inequality constraint in (10) and $\mathcal{A} = \{\boldsymbol{\alpha} | \alpha_i \geq 0, \ \boldsymbol{\alpha}'\mathbf{1}_N = 1\}$ is the feasible set of $\boldsymbol{\alpha}$ and

$$\mathcal{Y} = \{\mathbf{y} | y_i^{sr} = 1, \ y_i^{ta} \in \{\pm 1\}, \ \mathbf{y}^{ta\prime}\mathbf{1}_n = n(1-2\nu)\} \tag{12}$$

is the feasible set of $\mathbf{y}$. For a small ν, most of y_i^{ta}'s are $+1$, and so the outliers are rare.

However, due to integer variables of $\mathbf{y} \in \mathcal{Y}$, (11) is a MIP problem, which is computationally hard to be solved. To make it more tractable, one can follow the transductive learning strategy proposed in (Joachims, 1999) to learn $f(\mathbf{x})$ and $\mathbf{y}$ iteratively, resulting in local solutions. Alternatively, one can use a semidefinite relaxation used in (Xu et al., 2005) for approximating $\mathbf{y}\mathbf{y}'$, leading to a Semi-Definite Programming (SDP) problem with $O(n^2)$ optimization variables, which is computationally expensive even for small datasets.

3.3. Convex Relaxation and Cutting Set Algorithm

To avoid local minima or expensive SDP problems in solving the MIP problem in (11), we resort to the recently developed label generating technique (Li et al., 2009b,a; Yang and Tsang, 2011), which demonstrates superior performance and scalability of learning the label on large-scale unsupervised datasets (Li et al., 2009b). We first present a convex relaxation for (11) based on the minimax theorem in (Kim and Boyd, 2008), we arrive at:

$$\min_{\boldsymbol{\mu}\in\mathcal{M}}\max_{\boldsymbol{\alpha}\in\mathcal{A}}\frac{\eta}{4}\sum_{t:\mathbf{y}_t\in\mathcal{Y}}\mu_t(\mathbf{y}_t+\mathbf{1}_N)'(\mathbf{K}\odot\mathbf{S})(\mathbf{y}_t+\mathbf{1}_N)-\frac{1}{2}\boldsymbol{\alpha}'\Big(\sum_{t:\mathbf{y}_t\in\mathcal{Y}}\mu_t\mathbf{K}\odot\mathbf{y}_t\mathbf{y}_t'+\frac{1}{C}\mathbf{I}_N\Big)\boldsymbol{\alpha}. \tag{13}$$

The details of the above derivation are similar to the mixed integer problems in (Li et al., 2009b,a; Yang and Tsang, 2011), and so are omitted here. This optimization can be deemed as a Multiple Kernel Learning (MKL) problem (Rakotomamonjy et al., 2008), and the target kernel matrix is a combination of $|\mathcal{Y}|$ base kernel matrices $\{\mathbf{K}\odot\mathbf{y}_t\mathbf{y}_t'\}$, each of which is constructed from a feasible label vector $\mathbf{y}_t\in\mathcal{Y}$.

Due to exponential number of possible labelings $\mathbf{y}_t\in\mathcal{Y}$, the set of base kernels is also exponential in size and so (13) is computationally intractable to be solved by existing MKL techniques. Fortunately, not all the base kernels in (13) are necessarily active at optimality. Only using a subset $\mathcal{C}\subset\mathcal{Y}$ of them can well approximate the original optimization problem. Therefore, we can apply the cutting set method (Mutapcic and Boyd, 2009) to seek active base kernels in (13) and solve the reduced problem. Similar strategies have also been employed in infinite kernel learning (IKL) (Gehler and Nowozin, 2008), in which the kernel is learned from an infinite set of general kernel parameters, and so, MKL (with the kernel $\sum_{t:\mathbf{y}^t\in\mathcal{Y}}\mu_t\mathbf{K}\odot\mathbf{y}_t\mathbf{y}_t'$) can be deemed as a variant of IKL. As a result, our algorithm enjoys the same convergence. The whole algorithm is depicted in Algorithm 1.

Algorithm 1: Cutting set algorithm for MMD-ROD

Initialize $\boldsymbol{\alpha}=\frac{1}{N}\mathbf{1}_N$. Find the most violated $\mathbf{y}$ and set $\mathcal{C}=\{\mathbf{y}\}$.
while *not convergent* **do**
  Run MKL for the subset of kernel matrices selected in $\mathcal{C}$ and obtain $\boldsymbol{\alpha}$ from (13).
  Find the most violated $\mathbf{y}$ and set $\mathcal{C}=\mathbf{y}\bigcup\mathcal{C}$.
end

3.4. Finding a Violated y

Notice that, similar to IKL, finding the most violated base kernel (indexed by the labeling $\mathbf{y}_t$) in the proposed MKL is problem specific and the most challenging part in cutting set algorithms. Here, we will discuss how to explore the most violated base kernel to satisfy the constraint of relative novelty detection in (12).

Referring to (11), to find the most violated $\mathbf{y}$, we have to solve the following problem:

$$\max_{\mathbf{y}\in\mathcal{Y}}\mathbf{y}'\Big(\mathbf{K}\odot\boldsymbol{\alpha}\boldsymbol{\alpha}'-\frac{\eta}{2}\mathbf{K}\odot\mathbf{S}\Big)\mathbf{y}-\eta\mathbf{1}_N'(\mathbf{K}\odot\mathbf{S})\mathbf{y}. \tag{14}$$

We first define $\mathbf{G}=\mathbf{K}\odot\boldsymbol{\alpha}\boldsymbol{\alpha}'-\frac{\eta}{2}(\mathbf{K}\odot\mathbf{S})$ and $\mathbf{h}=-\eta\mathbf{1}_N'(\mathbf{K}\odot\mathbf{S})$, (14) is simplified as

$$\max_{\mathbf{y}\in\mathcal{Y}}\mathbf{y}'\mathbf{G}\mathbf{y}+\mathbf{h}\mathbf{y}. \tag{15}$$

We further partition $\mathbf{G}$ and $\mathbf{h}$ as $\mathbf{G} = \begin{bmatrix} \mathbf{G}_{sr,sr} & \mathbf{G}_{sr,ta} \\ \mathbf{G}_{ta,sr} & \mathbf{G}_{ta,ta} \end{bmatrix}$ and $\mathbf{h} = [\mathbf{h}_{sr}\ \mathbf{h}_{ta}]$, respectively, where sr and ta indicate the set of indices of data in the source and target domains, respectively. By ignoring some constant terms, then (15) is rewritten as:

$$\max_{\mathbf{y}^{ta} \in \mathcal{Y}_2} \ \mathbf{y}^{ta\prime}\mathbf{G}_{ta,ta}\mathbf{y}^{ta} + \mathbf{b}\mathbf{y}^{ta}, \tag{16}$$

where $\mathbf{b} = 2\mathbf{y}^{sr\prime}\mathbf{G}_{sr,ta} + \mathbf{h}_{ta}$ and $\mathcal{Y}_2 = \{\mathbf{y}^{ta} | y_i^{ta} \in \{\pm 1\},\ \mathbf{y}^{ta\prime}\mathbf{1}_n = n(1-2\nu)$.

However, this is a concave QP and so cannot be solved efficiently. As discussed in (Gehler and Nowozin, 2008), while the use of the most violated constraint may lead to faster convergence, the cutting plane/cutting set algorithm only requires the addition of a violated constraint at each iteration. Note, the approximation method used in (Li et al., 2009b) cannot be applied here as (16) has an additional linear term. Hence, we propose an efficient method in the following for finding a good approximation of the most violated $\mathbf{y}^{ta}$.

3.4.1. Successive Approximation Method

Since $y_i^{ta} \in \{\pm 1\}$, the constraint $\mathbf{y}^{ta\prime}\mathbf{y}^{ta} = n$ is implicitly enforced. Thus, $\mathbf{y}^{ta\prime}(\mathbf{G}_{ta,ta} + \varpi\mathbf{I}_n)\mathbf{y}^{ta} = \mathbf{y}^{ta\prime}\mathbf{G}_{ta,ta}\mathbf{y}^{ta} + \varpi n$. We let $n_\nu = n(1-2\nu)$ and $\mathbf{A} = \mathbf{G}_{ta,ta} + \varpi\mathbf{I}_n$ where ϖ is sufficiently large such that $\mathbf{A} \succ 0$ is positive definite. Similar to spectral relaxation (Ng et al., 2001), we relax the integer constraints on y_i^{ta}'s, then (16) is reformulated as follows,

$$\max_{\mathbf{y}^{ta}} \mathbf{y}^{ta\prime}\mathbf{A}\mathbf{y}^{ta} + \mathbf{b}\mathbf{y}^{ta} \quad : \quad \mathbf{y}^{ta\prime}\mathbf{y}^{ta} = n,\ \mathbf{1}_n'\mathbf{y}^{ta} = n_\nu. \tag{17}$$

For $\mathbf{1}_n'$, we define its null space as S ($Proj_S = \mathbf{M} = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n'$), and its range space as $S^\perp$ ($Proj_{S^\perp} = \mathbf{I}_n - \mathbf{M} = \frac{1}{n}\mathbf{1}_n\mathbf{1}_n'$), where $\mathbf{M} = \mathbf{M}' = \mathbf{M}^2$. Both S and $S^\perp$ are finite-dimensional subspaces of an inner product space $V \subset \mathbb{R}^n$. According to (Hogben, 2007), each $\mathbf{y}^{ta} \in V$ can be written uniquely as

$$\mathbf{y}^{ta} = \mathbf{y}_\parallel + \mathbf{y}_\perp, \tag{18}$$

where $\mathbf{y}_\parallel \in S$ and $\mathbf{y}_\perp = \frac{n_\nu}{n}\mathbf{1}_n \in S^\perp$ such that $\mathbf{1}_n'\mathbf{y}_\perp = n_\nu$ and $\mathbf{1}_n'\mathbf{y}_\parallel = 0$. In addition, we have $\mathbf{y}_\parallel{}'\mathbf{y}_\parallel = n - \mathbf{y}_\perp{}'\mathbf{y}_\perp = n - n(1-2\nu)^2 = \beta^2$, where $\beta = \sqrt{n(1-(1-2\nu)^2)}$ and $\nu \in [0, 1)$.

Since (17) is still a non-convex QP problem, it is computationally hard to directly obtain or even update the solution in the space V. Fortunately, we observed it is easier to update the solution in the space S. Based on this notion, we present an iterative algorithm to solve (17). The idea is precisely the method of successive approximations (Horst and Tuy, 1996; Xu et al., 2009), which is an optimization tool to progressively solve a linearly constrained eigenvalue decomposition problem, and our method is the first work to consider the quadratic function with a linear term in the objective function.

For a fixed $\mathbf{y}_k^{ta}$, we update $\mathbf{y}_\parallel$ in space S using the first order Taylor approximation of the original objective. After that, $\mathbf{y}_\parallel$ is mapped to the original space V, and then we get a better $\mathbf{y}_{k+1}^{ta}$. Specifically, for a fixed $\mathbf{y}_k^{ta}$, the approximation of (17) is:

$$\max_{\mathbf{y}_{k+1}^{ta}} \mathbf{y}_{k+1}^{ta}{}'(2\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{b}') - \mathbf{y}_k^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta} \quad : \quad \mathbf{y}_{k+1}^{ta}{}'\mathbf{y}_{k+1}^{ta} = n,\ \mathbf{1}_n'\mathbf{y}_{k+1}^{ta} = n_\nu. \tag{19}$$

From (18), we can further simplify (19) as

$$\max_{\mathbf{y}_\parallel} \mathbf{y}_\parallel{}'(2\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{b}') \quad : \quad \mathbf{y}_\parallel{}'\mathbf{y}_\parallel = \beta^2,\ \mathbf{1}_n'\mathbf{y}_\parallel = 0. \tag{20}$$

Lemma 1 *In each step, suppose* $\mathbf{y}_{k+1}^{ta} = \mathbf{y}_{\|_{k+1}} + \mathbf{y}_{\perp}$. *Then* $\mathbf{y}_{\|_{k+1}} = \beta \frac{2\mathbf{M}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{M}\mathbf{b}'}{\|2\mathbf{M}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{M}\mathbf{b}'\|}$ *is the solution of (20).*

Proof When $\mathbf{1}_n'\mathbf{y}_{\|} = 0$, which is equivalent to $\mathbf{M}\mathbf{y}_{\|} = \mathbf{y}_{\|}$. Therefore, (20) can be reduced to $\max_{\mathbf{y}_{\|}} \mathbf{y}_{\|}'\mathbf{M}(2\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{b}') : \mathbf{y}_{\|}'\mathbf{y}_{\|} = \beta^2$, and its solution is $\mathbf{y}_{\|} = \beta \frac{2\mathbf{M}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{M}\mathbf{b}'}{\|2\mathbf{M}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{M}\mathbf{b}'\|}$. ■

Hence, $\mathbf{y}_{k+1}^{ta}$ can be updated. This process is repeated until convergence. The whole algorithm is depicted in Algorithm 2.

Algorithm 2: Successive Approximation of $\mathbf{y}$

Initialize $\mathbf{M} = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n'$, $\mathbf{y}_{\perp} = \frac{n_\nu}{n}\mathbf{1}_n$, $\beta = \sqrt{n(1-(1-2\nu)^2)}$, $k = 1$ and set $\mathbf{y}_1^{ta}$ to a proper initial solution.
while *not convergent* **do**
  $\mathbf{y}_{\|_{k+1}} = \beta \frac{2\mathbf{M}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{M}\mathbf{b}'}{\|2\mathbf{M}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{M}\mathbf{b}'\|}$, $\mathbf{y}_{k+1}^{ta} = \mathbf{y}_{\|_{k+1}} + \mathbf{y}_{\perp}$;
  $k = k + 1$;
end

3.4.2. Convergence Analysis

Here, we will study the convergence behavior for Algorithm 2. We first show that the update procedures in Algorithm 2 converges in a finite iterations. Then we verify the solution satisfying the optimal condition of (17).

Lemma 2 *The objective (17) is bounded by* $n\lambda_{max} + \sqrt{n}\|\mathbf{b}\|$, *where* λ_{max} *is the largest eigenvalues of* $\mathbf{A}$.

Proof By the eigen-decomposition, we have $\mathbf{y}^{ta\prime}\mathbf{A}\mathbf{y}^{ta} = \mathbf{y}^{ta\prime}\mathbf{U}\Lambda\mathbf{U}'\mathbf{y}^{ta} = \sum_{i=1}^n \lambda_i |(\mathbf{U}'\mathbf{y}^{ta})_i|^2$, where λ_i' are the eigenvalues, and λ_{max} is the largest one among them. Because $\mathbf{A}$ is symmetric, $\mathbf{U}$ is an orthonormal matrix, i.e. $\mathbf{U}'\mathbf{U} = \mathbf{I}_n$. Then, we have $\mathbf{y}^{ta\prime}\mathbf{A}\mathbf{y}^{ta} \leq \lambda_{max}\sum_{i=1}^n |(U'\mathbf{y}^{ta})_i|^2 = n\lambda_{max}$. Besides this, since $\mathbf{y}^{ta\prime}\mathbf{y}^{ta} = n$, we have $\mathbf{b}\mathbf{y}^{ta} \leq \mathbf{b}\frac{\sqrt{n}\mathbf{b}}{\|\mathbf{b}\|}$. So the proof is completed. ■

Theorem 1 *Given the projection matrix* $\mathbf{M}$ *and the update rule, Algorithm 2 converges.*

Proof From (19), we know that

$$2\mathbf{y}_{k+1}^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{b}\mathbf{y}_{k+1}^{ta} \geq 2\mathbf{y}_k^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta} + \mathbf{b}\mathbf{y}_k^{ta} \;\Rightarrow\; \mathbf{y}_{k+1}^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta} \geq \mathbf{y}_k^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta} + \frac{\mathbf{b}\mathbf{y}_k^{ta} - \mathbf{b}\mathbf{y}_{k+1}^{ta}}{2}. \quad (21)$$

Furthermore, with the positive semidefinite property of $\mathbf{A}$ we have $(\mathbf{y}_{k+1}^{ta} - \mathbf{y}_k^{ta})'\mathbf{A}(\mathbf{y}_{k+1}^{ta} - \mathbf{y}_k^{ta}) \geq 0$. So it is obvious that

$$\mathbf{y}_{k+1}^{ta\prime}\mathbf{A}\mathbf{y}_{k+1}^{ta} \geq -\mathbf{y}_k^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta} + 2\mathbf{y}_{k+1}^{ta\prime}\mathbf{A}\mathbf{y}_k^{ta}. \quad (22)$$

Combining the inequality in (21), we obtain

$$\mathbf{y}_{k+1}^{ta\prime}\mathbf{A}\mathbf{y}_{k+1}^{ta} \geq -\mathbf{y}_{k}^{ta\prime}\mathbf{A}\mathbf{y}_{k}^{ta} + 2\mathbf{y}_{k}^{ta\prime}\mathbf{A}\mathbf{y}_{k}^{ta} + (\mathbf{b}\mathbf{y}_{k}^{ta} - \mathbf{b}\mathbf{y}_{k+1}^{ta}).$$

Based on above the deduction, we obtain

$$\mathbf{y}_{k+1}^{ta\prime}\mathbf{A}\mathbf{y}_{k+1}^{ta} + \mathbf{b}\mathbf{y}_{k+1}^{ta} \geq \mathbf{y}_{k}^{ta\prime}\mathbf{A}\mathbf{y}_{k}^{ta} + \mathbf{b}\mathbf{y}_{k}^{ta}. \tag{23}$$

Then we know that the algorithm is monotonically increasing. And from Lemma 2, the objective of problem (17) is bounded, so Algorithm 2 can surely converge. ■

Thus, the intermediate solutions $\mathbf{y}_{k+1}^{ta}$ in Algorithm 2 can progressively improve until Algorithm 2 terminates.

Theorem 2 *Algorithm 2 converges to a critical point of (17).*

Proof We know that $\mathbf{M} = \mathbf{M}' = \mathbf{M}^2$, $\mathbf{1}'\mathbf{M} = \mathbf{0}'$ and $\mathbf{y}^{ta} = \mathbf{y}_\perp + \mathbf{M}\mathbf{y}^{ta} \Leftrightarrow \mathbf{1}_n'\mathbf{y}^{ta} = n_\nu$. If we let $\mathbf{y}^{ta} = \mathbf{y}_\perp + \mathbf{M}\mathbf{y}^{ta}$, the constraint $\mathbf{1}_n'\mathbf{y}^{ta} = n_\nu$ can be ignored. Then (17) becomes

$$\max_{\mathbf{y}^{ta}} (\mathbf{y}_\perp + \mathbf{M}\mathbf{y}^{ta})'\mathbf{A}(\mathbf{y}_\perp + \mathbf{M}\mathbf{y}^{ta}) + \mathbf{b}(\mathbf{y}_\perp + \mathbf{M}\mathbf{y}^{ta}) \quad : \quad \mathbf{y}^{ta\prime}\mathbf{M}'\mathbf{M}\mathbf{y}^{ta} = \beta^2. \tag{24}$$

By omitting the constant terms in (24), we have

$$\max_{\mathbf{y}^{ta}} \mathbf{y}^{ta\prime}\mathbf{M}\mathbf{A}\mathbf{M}\mathbf{y}^{ta} + 2\mathbf{y}^{ta\prime}\mathbf{M}\mathbf{A}\mathbf{y}_\perp + \mathbf{b}\mathbf{M}\mathbf{y}^{ta} \quad : \quad \mathbf{y}^{ta\prime}\mathbf{M}\mathbf{y}^{ta} = \beta^2. \tag{25}$$

The Lagrangian function of problem (25) is $\mathbf{y}^{ta\prime}\mathbf{M}'\mathbf{A}\mathbf{M}\mathbf{y}^{ta} + 2\mathbf{y}^{ta\prime}\mathbf{M}\mathbf{A}\mathbf{y}_\perp + \mathbf{b}\mathbf{M}\mathbf{y}^{ta} - \lambda(\mathbf{y}^{ta\prime}\mathbf{M}\mathbf{y}^{ta} - \beta^2)$, where λ is the Lagrangian multiplier. Then the KKT conditions of (25) becomes

$$\begin{aligned} 2\mathbf{M}\mathbf{A}\mathbf{M}\mathbf{y}^{ta} + 2\mathbf{M}\mathbf{A}\mathbf{y}_\perp + \mathbf{M}\mathbf{b}' - 2\lambda\mathbf{M}\mathbf{y}^{ta} &= \mathbf{0} \\ \|\mathbf{M}\mathbf{y}^{ta}\| &= \beta \end{aligned} \tag{26}$$

The update rule $\mathbf{y}_{k+1}^{ta} = \frac{\beta(2\mathbf{M}\mathbf{A}(\mathbf{y}_k^{ta}) + \mathbf{M}\mathbf{b}')}{\|2\mathbf{M}\mathbf{A}(\mathbf{y}^{ta}{}_k) + \mathbf{M}\mathbf{b}'\|} + \mathbf{y}_\perp$ in Algorithm 1 is a continuous map on $\mathbf{y}^{ta}$. With Brouwer fixed point theorem, we know that there must exist some $\mathbf{y}^{ta}$ such that $\mathbf{y}^{ta}{}_{k+1} = \mathbf{y}^{ta}{}_k$. Thus, after convergence, the fixed point $\mathbf{y}^{ta}$ must satisfy

$$\mathbf{y}^{ta} = \frac{\beta(2\mathbf{M}\mathbf{A}\mathbf{y}^{ta} + \mathbf{M}\mathbf{b}')}{\|2\mathbf{M}\mathbf{A}\mathbf{y}^{ta} + \mathbf{M}\mathbf{b}'\|} + \mathbf{y}_\perp.$$

On the other hand, with condition $\|\mathbf{M}\mathbf{y}^{ta}\| = \beta$ and considering $\mathbf{y}^{ta} = \mathbf{y}_\perp + \mathbf{M}\mathbf{y}^{ta}$, we know that $\mathbf{M}\mathbf{y}^{ta} = \mathbf{y}^{ta} - \mathbf{y}_\perp = \frac{\beta(2\mathbf{M}\mathbf{A}\mathbf{y}^{ta} + \mathbf{M}\mathbf{b}')}{\|2\mathbf{M}\mathbf{A}\mathbf{y}^{ta} + \mathbf{M}\mathbf{b}'\|} = \frac{1}{2\lambda}(2\mathbf{M}\mathbf{A}\mathbf{y}^{ta} + \mathbf{M}\mathbf{b}')$ when $\lambda = \frac{\|2\mathbf{M}\mathbf{A}\mathbf{y}^{ta} + \mathbf{M}\mathbf{b}'\|}{2\beta}$. In addition, $\mathbf{M}\mathbf{y}^{ta} = \frac{1}{2\lambda}(2\mathbf{M}\mathbf{A}(\mathbf{M}\mathbf{y}^{ta} + \mathbf{y}_\perp) + \mathbf{M}\mathbf{b}')$. Then we obtain $2\mathbf{M}\mathbf{A}\mathbf{M}\mathbf{y}^{ta} + 2\mathbf{M}\mathbf{A}\mathbf{y}_\perp + \mathbf{M}\mathbf{b}' - 2\lambda\mathbf{M}\mathbf{y}^{ta} = \mathbf{0}$, which completes the proof. ■

From this theorem, we know the final solution $\mathbf{y}_{k+1}^{ta}$ in Algorithm 2 will satisfy the optimality condition of (17). Empirically, Algorithm 2 converges in a few iterations. Finally, similar to spectral clustering (Ng et al., 2001), we can apply standard rounding techniques to the solution of (17) and get back an integer solution to (16).

3.5. Prediction

It is worth noting that, the solution $\mathbf{y}\mathbf{y}'$ in (11) is relaxed to $\mathbf{M} = \sum_{t:\mathbf{y}_t \in \mathcal{Y}} \mu_t \mathbf{y}_t \mathbf{y}_t'$ in (13). When the optimal kernel in (13) is spanned by one base kernel, we can solve the exact solution of (11). However, in practice, existing MKL algorithms return the solution spanned by a few base kernels. Since the sign of labeling $\mathbf{y}_t$ of MMC (Li et al., 2009b) is ambiguous, the final cluster assignment can only be determined by using eigendecomposition on $\mathbf{M}$. However, in relative outlier detection, since most y_i's are $+1$, for new-coming instances, we can use a predicting function as follows,

$$y^{pre} = \mathbf{w}'\varphi(\mathbf{x}) = \sum_t \mu_t \sum_i \alpha_i y_{ti} k(\mathbf{x}_i, \mathbf{x}) = \sum_i \alpha_i \big(\sum_t \mu_t y_{ti} \big) k(\mathbf{x}_i, \mathbf{x}),$$

which is the ensemble output from a set of largely violated $\mathbf{y}_t$'s. This prediction is more robust than a single hypothesis for detecting relative outliers.

3.6. Complexity Analysis

Since the computations of MMD-ROD is dominated by solving MKL problems and finding the largely violated labeling $\mathbf{y}$, in this subsection, we mainly discuss the complexity of these two components. Nowadays MKL techniques usually involve a series of SVM training, and usually converges in a few iterations. Empirically, SVM takes $O(n^{2.3})$ time and $O(n)$ space complexities, so the SVM training can scale for large datasets. Thus, solving MKL problems is still very efficient. For finding the largely violated labeling problem, it involves a few iterations of matrix vector product operations, which takes $O(n^2)$ time. So MMD-ROD can work medium sized datasets. In order to improve the efficiency and deal with large datasets, we can also use random feature mapping (Rahimi and Recht, 2007) instead of computing kernel functions.

4. Experiments

In this section, we perform comprehensive empirical studies to evaluate several state-of-the-art relative outlier detection methods including our proposed method in identifying relative outliers on a collection of real-world datasets, which covers various characteristics and application domains. The statistics of the real-world datasets are shown in Table 1.

Besides the USPS handwritten dataset, the other seven datasets are from UCI machine learning repository. The first dataset is concerning credit card applications. The normal instances are the approved customers. Due to various reasons, some customers are rejected, which are set as outliers. In the DNA dataset, it is to detect splice junctions as exon/intron boundaries (referred to as EI sites), intron/exon boundaries (IE sites) and neither of them. We set the EI as the normal class, and the others are outliers. In the Ionosphere dataset, radar signals pass through the ionosphere to detect free electrons in the ionosphere. The normal instances are those showing evidence of some types of structure, and outliers returned are those that do not. In the Segment dataset, the instances were drawn randomly from a database of seven outdoor images. The images were hand-segmented to create a classification task. We set the first class "brickface" and second class "sky" as normal instances, and the other five classes as outliers. In the Ball-bearing dataset, the task is to

distinguish new ball-bearings from bearings where abnormal race includes completely broken, broken cage, damaged case and worn bearings. The USPS dataset is a collection of images for handwritten digit 1-9 and 0. We suppose the first five digits as normal instances, and the last five as outliers. According to the Arrhythmia dataset, its aim is to distinguish between the presence and absence of cardiac arrhythmia and to classify the ECG in one of the 16 groups. Class 1 refers to normal ECG, and classes 2 to 15 refer to different classes of arrhythmia and class 16 refers to the rest of unclassified ones. We set class 1 as the normal class, classes 2-15 as novelties. For the Scene dataset, the images consist of six classes, like Corel stock photo library and personal images. We use class '0' and '1' as normal instances, and others as novelties.

Since the normal instances in some datasets, such as the USPS dataset, come from several major classes, which are considered as multi-modal normal instances, and the other classes as different outliers, which are multi-modal outliers. We also have some datasets use one class as the normal instances. In this way, our experiments include different settings such as one class vs several classes and several classes vs several classes. Hence, we can evaluate the performances of different relative outlier detection methods under different characteristics of datasets.

Table 1: Datasets used in the experiments.

ID	Data	# instances	# Features
1	*Australian*	690	14
2	*DNA*	3186	180
3	*Ionosphere*	351	34
4	*Segment*	2310	19
5	*Ball-bearing*	4150	32
6	*USPS*	7291	256
7	*Arrhythmia*	420	287
8	*Scene*	1211	294

4.1. Compared Methods

We compare our proposed MMD-ROD with the following cutting edge methods in (relative) outlier detection: 1) Least-Squares Outlier Detection (LSOD) method (Hido et al., 2008) [1]; 2) Maximum Likelihood Outlier Detection (MLOD) method (Kanamori et al., 2009) [2]; 3)Truncated Kullback Leibler (TKL) divergence estimation (Smola et al., 2009); 4)Kullback Leibler(KL)divergence estimation (Smola et al., 2009) [3]. Note that TKL and KL are implemented in C++; while the other three methods including MMD-ROD are implemented by Matlab. All experiments are conducted in a PC with 16GB memory and the Intel(R) Core(TM) i7 CPU (2.80GHz).

For MMD-ROD, the C parameter is selected in a range of {0.001 0.01 0.1 1 10 100 1000} and the η parameter is selected from {0 0.01 1 10 100 1000}. Based on the suggestion of (Smola et al., 2009), the weight of regularization term λ is {0.001 0.01 1 10 100 1000} in TKL and KL. Particular in TKL, the thresholded level ρ is selected from the range of {0.001 0.01 0.1 1 10 100 1000}. According to (Kanamori et al., 2009; Hido et al., 2008), the

1. The software is available at http://sugiyama-www.cs.titech.ac.jp/∼sugi/software/LSOD/index.html
2. The software is available at http://sugiyama-www.cs.titech.ac.jp/∼sugi/software/MLOD/index.html
3. The source codes of KL and TKL are obtained from the authors.

weight of regularization term λ is chosen from {0.001 0.01 1 10 100 1000} in LSOD, and the number of kernels are among {10 50 100 150 200}.

Gaussian kernel is used in the experiments. In particular, the width σ of the Gaussian kernel $\exp(-||z||^2/2\sigma^2)$ is picked via the range $\{0.25\sqrt{\gamma}, 0.5\sqrt{\gamma}, \sqrt{\gamma}, 2\sqrt{\gamma}, 4\sqrt{\gamma}\}$ where γ is the average distance from all pairs of instances. Following (Smola et al., 2009), we use random feature mapping (Rahimi and Recht, 2007) to approximate Gaussian kernel to speed up the training process of TKL and KL. From our experimental results, the performances of LSOD and MLOD using the original leave-one-out cross-validation (LOOCV) are inferior to those adopting artificial outliers for validation. For fair comparisons, we use artificial outliers in validation for all comparing methods.

4.2. Experimental Setup

Here, 1/4 of the positive instances is for the target and source domains respectively, and 1/4 is for validation and testing sets respectively. To illustrate the robustness of different relative outlier detection methods with varying ν, following the setup from (Smola et al., 2009), we also set the outlier ratio as 2%, 4% and 8% for negative instances in the target domain respectively, and the rest negative instances are used in the testing set. Moreover, the same amount of artificial outliers as the normal instances are added into the validation set. The artificial outliers here are created by using a uniform distribution U that is defined within a bounded subspace whose minimum and maximum are limited to be 10% beyond the observed minimum and maximum from the training data (Abe et al., 2006).

To evaluate the performance in detecting relative outliers, we follow (Smola et al., 2009) to choose the Average Precision@k (AP@k) (Joachims, 2005) as the evaluation criterion for performance measures. Note, AP@k means the mean of the precision scores obtained after the k novelty instances are retrieved. Thus, AP@k also considers ranking information compared with the traditional accuracy. The training, validation and testing set are randomly generated and all the methods are evaluated by the average performance of 20 repetitions.

We evaluate the performances of all methods in two stages. We first remove the labels for the target dataset, and use the validation dataset to choose parameters for each method. After that, the prediction function is used to decide the labels for both the target and test datasets (in the first and second stages, respectively). Then we measure the outlier detection performances according to the true labels. Results for the target and unseen test set are shown in Table 2 and Table 3 respectively. The best performance is listed in bold.

4.3. Analysis of Experimental Results

In the target datasets (TABLE 2), most of the instances are normal and only a few percentages are novel instances. Based on this setting, we compare the performance of different methods. From TABLE 2, we observe that hyperplane-based outlier detection methods (MMD-ROD) outperform density estimation methods (LSOD, MLOD, TKL and KL). MMD-ROD achieves 100% accuracy in all the eight datasets. Meanwhile, LSOD and MLOD gain 100% accuracy in three datasets; while TKL and KL achieve this in one dataset only. We also observe that MMD-ROD significantly outperforms other density-ratio based methods on the USPS dataset, in which there are multi-modal normal instances. It is possibly

Table 2: AP@k (%) on various target datasets

Dataset	Novelty ratio(%)	MMD-ROD	LSOD	MLOD	TKL	KL
Australian	2	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	83.48±19.22	86.08±12.55
	4	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	72.38±17.14	74.61±21.11
	8	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	66.64±19.23	70.02±14.82
DNA	2	**100.00±0.00**	90.11±16.37	82.01±24.37	**100.00±0.00**	**100.00±0.00**
	4	**100.00±0.00**	87.33±12.06	91.77±7.73	**100.00±0.00**	**100.00±0.00**
	8	**100.00±0.00**	91.76±6.17	91.18±6.36	**100.00±0.00**	**100.00±0.00**
Ionosphere	2	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	97.50±11.18	97.50±11.18
	4	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	92.71±18.23	**100.00±0.00**
	8	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	97.33±5.25	94.90±12.56
Segment	2	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	96.56±10.70	97.92±6.55
	4	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	96.19±9.22	97.94±9.23
	8	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	99.07±2.35	95.75±6.58
Ball-bearing	2	**100.00±0.00**	85.13±10.04	88.14±11.36	99.47±1.77	97.04±6.80
	4	**100.00±0.00**	92.15±4.72	90.70±7.10	98.97±1.83	98.65±2.55
	8	**100.00±0.00**	93.72±3.60	93.09±3.29	98.41±2.66	98.63±2.30
USPS	2	**100.00±0.00**	71.47±19.72	66.81±25.12	83.54±20.83	82.13±17.24
	4	**100.00±0.00**	63.53±21.14	66.35±15.29	78.68±21.52	62.81±14.16
	8	**100.00±0.00**	74.43±8.37	72.95±10.83	68.87±12.00	66.29±10.57
Arrhythmia	2	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
	4	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	99.17±3.73
	8	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	96.67±4.72	96.34±6.53
Scene	2	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
	4	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	99.17±3.73
	8	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	99.63±1.64	97.16±5.67

because estimating the density ratio of normal instances is more challenging on multi-modal problems.

In the second stage (TABLE 3), using the learned classifier from stage I, we detect the new coming instance, and again, our MMD-ROD performs the best on seven out of eight datasets. Moreover, we also observe that the generalization performance of MMD-ROD is very stable across different outlier ratios. While the generalization performances of TKL and KL degrade on four and five datasets respectively with decreasing outlier ratio on the training sets. It is possibly due to the fact that the density ratio estimation in TKL and KL is still sensitive to diverse outliers.

Here, we also provide an empirical assessment of the time complexity of different relative novelty detection methods in Figure 1. There are the eight datasets sorted by their size in the x axis, and the average training CPU time of the five methods is presented. The time complexity of MMD-ROD and MLOD are the highest. Because MMD-ROD needs to solve MKL problems, involving a series of SVM training ($O(n^{2.3})$), and the MLOD is also time consuming. However, MMD-ROD is a kernel method, which is only sensitive to the number of instance, but not to the number of features. The time complexity of LSOD is better than above two methods; and TKL and KL are the best, which is due to the use of random feature mapping. Moreover, TKL and KL are implemented in C++, but the other three methods are implemented by Matlab. So the comparison between (T)KL methods and other three methods are just for reference.

Table 3: AP@k (%) on various unseen test datasets

Dataset	Novelty ratio(%)	MMD-ROD	LSOD	MLOD	TKL	KL
Australian	2	**89.39±3.00**	86.16±2.57	84.07±5.38	87.94±4.43	86.60±4.55
	4	**90.99±4.47**	87.01±4.25	85.17±3.73	87.57±3.64	88.79±2.91
	8	89.19±3.67	86.38±2.06	85.03±5.33	88.60±2.84	**90.11±2.93**
DNA	2	**94.33±1.74**	86.79±2.46	86.91±2.96	93.47±1.63	88.90±3.14
	4	91.93±3.30	86.96±2.80	88.10±1.78	**94.30±1.51**	89.75±3.59
	8	92.76±2.66	87.93±2.19	88.18±3.08	**93.81±1.71**	92.78±2.47
Ionosphere	2	**97.41±1.38**	91.03±7.17	94.97±4.15	84.33±2.53	83.32±4.27
	4	**98.72±1.12**	93.70±5.50	95.37±4.99	83.71±5.29	83.35±5.48
	8	**97.18±1.33**	91.62±7.50	93.76±7.74	87.52±4.21	84.86±5.12
Segment	2	**98.98±1.09**	97.24±3.61	97.53±6.07	81.11±6.78	85.52±8.43
	4	98.22±3.98	96.78±4.11	**98.65±3.68**	89.06±6.43	86.43±7.58
	8	**99.43±0.43**	96.36±3.94	97.29±4.43	94.11±3.38	89.58±6.23
Ball-bearing	2	**98.62±0.38**	97.36±1.38	89.62±3.59	98.25±1.08	97.73±2.06
	4	**98.95±0.35**	96.97±1.29	90.78±3.90	98.90±0.67	98.59±0.98
	8	**99.25±0.39**	96.68±1.55	90.11±4.19	98.95±0.73	99.09±0.64
USPS	2	**91.88±2.54**	87.22±2.82	85.12±3.28	82.39±2.90	83.57±3.09
	4	**93.69±1.99**	88.47±1.86	84.51±2.55	83.72±3.44	83.72±3.73
	8	**94.82±1.74**	87.35±2.32	83.66±3.24	87.24±2.37	86.45±2.72
Arrhythmia	2	**86.59±3.47**	85.22±2.26	79.76±9.50	75.60±6.31	72.33±6.46
	4	**85.52±4.65**	84.47±2.69	79.74±9.61	74.90±5.33	73.59±4.90
	8	**84.69±4.29**	83.61±2.23	82.00±8.58	73.38±5.95	74.86±5.42
Scene	2	**85.75±3.75**	80.61±4.24	80.90±3.95	81.54±9.41	80.67±3.96
	4	**87.07±4.92**	79.81±4.42	80.83±3.61	83.60±8.09	82.52±8.59
	8	85.75±4.37	81.52±3.27	81.90±3.33	**86.63±4.47**	85.21±5.96

5. Summary and Discussion

In this paper, we have cast the relative outlier detection problem as measuring the distribution difference between the target and source datasets. Particularly, we employ Maximum Mean Discrepancy (MMD) in an asymmetric setting for matching the distribution between these two datasets and present a novel learning framework to learn a relative outlier detector. Since the resultant problem is in the form of a Mixed Integer Programming (MIP) problem, which is computationally hard. To reduce this computational burden, we develop an effective procedure to find a largely violated labeling vector for identifying relative outliers from abundant normal patterns. Then, these largely violated labeling vectors are combined by multiple kernel learning methods to robustly locate relative outliers. We also present the convergence analysis for finding the largely violated labeling vectors. From our empirical studies on real-world datasets, we show that our proposed relative outlier detection outperforms state-of-the-art relative outlier detection methods.

Acknowledgments

This research is suppoted by the Singapore National Research Foundation Interactive Digital Media R&D Program, under research Grant NRF2008IDM-IDM004-018.

References

N. Abe, B. Zadrozny, and J. Langford. Outlier detection by active learning. In *KDD*, pages 504–509, 2006.

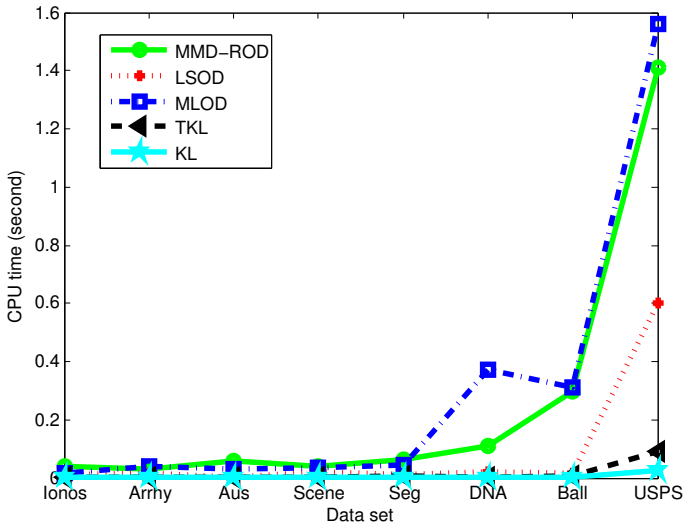

Figure 1: Computation Time Comparison

G. Blanchard, G. Lee, and C. Scott. Semi-supervised novelty detection. *JMLR*, 11:2973–3009, 2010.

K. M. Borgwardt, A. Gretton, M. J. Rasch, H.-P. Kriegel, B. Schölkopf, and A. J. Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(4):49–57, 2006.

V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3), July 2009.

J. Gao, H. Cheng, and P.-N. Tan. Semi-supervised outlier detection. In *ACM Symposium on Applied Computing*, pages 635–636, 2006.

P. Gehler and S. Nowozin. Infinite kernel learning. Technical Report TR-178, Max Planck Institute for Biological Cybernetics, 2008.

S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori. Inlier-based outlier detection via direct density ratio estimation. In *ICDM*, 2008.

L. Hogben. *Handbook of Linear Algebra*. Chapman and Hall, 2007.

R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. Springer, 1996.

T. Joachims. Transductive inference for text classification using support vector machines. In *ICML*, 1999.

T. Joachims. A support vector method for multivariate performance measures. In *ICML*, pages 377–384, 2005.

T. Kanamori, S. Hido, and M. Sugiyama. A least-squares approach to direct importance estimation. *JMLR*, 10:1391–1445, 2009.

S.-J. Kim and S. Boyd. A minimax theorem with applications to machine learning, signal processing, and finance. *SIAM Journal on Optimization*, 19(3):1344–1367, 2008.

Y.F. Li, J.T. Kwok, I.W. Tsang, and Z.H. Zhou. A convex method for locating regions of interest with multi-instance learning. In *ECML*, pages 15–30, 2009a.

Y.F. Li, I.W. Tsang, J.T. Kwok, and Z.H. Zhou. Tighter and convex maximum margin clustering. In *AISTATS*, 2009b.

M. Markou and S. Singh. Novelty detection: A review, Part I: Statistical approaches. *Signal Processing*, 83(12):2481–2497, 2003.

A. Mutapcic and S. Boyd. Cutting-set methods for robust convex optimization with pessimizing oracles. *Optimization Methods and Software*, 24(3):381–406, 2009.

A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, 2001.

X.L. Nguyen, M. Wainwright, and M. Jordan. Estimating divergence functionals and the likelihood ratio by penalized convex risk minimization. In *NIPS*, 2008.

F. Nie, Z. Zeng, I. W. Tsang, D. Xu, and C. Zhang. Spectral embedded clustering: A framework for in-sample and out-of-sample spectral clustering. *To appear in IEEE Transactions on Neural Networks*, 2011.

A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *NIPS*, 2007.

A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet. SimpleMKL. *J. of Mach. Learn. Res.*, 9:2491–2521, 2008.

B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press., 2002.

B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt. Support vector method for novelty detection. pages 582–588, San Mateo, CA, 2000. Morgan Kaufmann.

A. Smola, L. Song, and C. H. Teo. Relative novelty detection. In *AISTATS: Artificial Intelligence and Statistics, JMLR Proceedings Track*, volume 5, pages 536–543, 2009.

M. Wu and J. Ye. A small sphere and large margin approach for novelty detection using training data with outliers. *IEEE TPAMI*, 31(11):2088–2092, 2009.

L. Xu, J. Neufeld, B. Larson, and D. Schuurmans. Maximum margin clustering. In *NIPS*, pages 1537–1544. MIT Press, Cambridge, MA, 2005.

L. Xu, Li. W., and D. Schuurmans. Fast normalized cut with linear constraints. In *CVPR*, 2009.

J. Yang and I. W. Tsang. Hierarchical maximum margin learning for multi-class classification. In *UAI*, pages 753–760, 2011.