Neurocomputing 130 (2014) 63–72

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A novelty detection machine and its application to bank failure prediction

Shukai Li [a,*], Whye Loon Tung [b,c], Wee Keong Ng [b]

[a] Institute for Infocomm Research, A*Star, Singapore
[b] School of Computer Engineering, Nanyang Technological University, Singapore
[c] Service Platform Lab, HP Labs, Singapore

ARTICLE INFO

Article history:
Received 15 January 2012
Received in revised form
8 February 2013
Accepted 27 February 2013
Available online 27 July 2013

Keywords:
Novelty detection
Cluster assumption
Bank failure prediction

ABSTRACT

Novelty detection has been well-studied for many years and has found a wide range of applications, but correctly identifying the outliers is still a hard problem because of the diverse variation and the small quantity of such outliers. We address the problem using several distinct characteristics of the outliers and the normal patterns. First, normal patterns are usually grouped together, forming clusters in the high density regions of the data space. Second, outliers are characteristically very different from the normal patterns, and hence tend to be located far away from the normal patterns in the data space. Third, the number of outliers is generally very small in a given dataset. Based on these observations, we can envisage that the appropriate decision boundary segregating the outliers and the normal patterns usually lies in some low density regions of the data space. This is referred to as cluster assumption. The resultant optimization problem to learn the decision function can be solved using the mixed integer programming approach. Following that, we present a cutting plane algorithm together with a multiple kernel learning technique to solve the convex relaxation of the optimization problem. Specifically, we make use of the scarcity of the outliers to find a violating solution to the cutting plane algorithm. Experimental results with several benchmark datasets show that our proposed novelty detection method outperforms existing hyperplane and density estimation-based novelty detection techniques. We subsequently apply our method to the prediction of banking failures to identify potential bank failures or high risk banks through the traits of financial distress.

1. Introduction

Novelty detection, also known as outlier detection, anomaly detection, or one-class classification, is an important problem in data mining and machine learning. The primary task of novelty detection is to differentiate the known objects (normal patterns) from the deviant samples (outliers) [1–3]. The typical assumption of novelty detection is that the normal patterns are in abundance and frequently observed; while the outliers are very rare or may even be previously unseen samples that are characteristically very different from the normal patterns. Novelty detection has found many real-world applications; for instance, in mechanical diagnosis to isolate the faulty jet engines [4], to perform intrusion detection in network systems [5], and to detect fraudulent credit card transactions [3], etc. In finance, bank failure is an important issue in credit risk management, and the number of failing banks is small within the entire banking system. However, the collapse and failure of a bank could have devastating consequences to the entire banking system and an adverse repercussion effect on other banks and financial institutions. Some of the negative impacts are the massive bail out cost for a failing bank and the negative sentiments and loss of confidence developed by investors and depositors. Hence, bank failure prediction is an important issue for regulators of the banking industries. In this paper, we will apply our proposed technique of novelty detection to predict banking failures.

Traditionally, outliers are often detected by estimating the density functions of some pre-defined models (e.g. multivariate Gaussian (MVG), Gaussian mixtures, kernel density estimation (KDE) [6], etc.) to fit the normal patterns. However, such methods may fail miserably when the pre-defined models cannot capture the true distribution of the normal patterns. Alternatively, several k nearest neighbors (kNN) based outlier detection methods such as local outlier factor (LOF) [7] and prototype-based domain description (PDD) [8] have been proposed, and they showed encouraging results for detecting outliers. However, these kNN based methods cannot handle previously unseen data and may scale poorly on large and high dimensional datasets.

Instead of estimating the density distribution of the normal patterns or using the computationally expensive k nearest neighbor approaches, a simpler and more intuitive methodology is to directly model the support of the normal pattern distribution.

* Corresponding author. Tel.: +65 8178 6016; fax: +65 6792 6559.
E-mail addresses: lisk@i2r.a-star.edu.sg (S. Li), wltung@pmail.ntu.edu.sg (W.L. Tung), AWKNG@ntu.edu.sg (W.K. Ng).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.02.043

Tax and Duin proposed the support vector data description (SVDD) technique [9], which uses a small hypersphere (enclosing ball) to enclose most of the normal patterns. Computationally, this leads to a convex quadratic programming (QP) problem, which has the important feature that the solution obtained is always globally optimal. Moreover, as with other kernel methods, SVDD works well with high-dimensional datasets and can be easily extended to nonlinear generalization by replacing the dot products between the data patterns with the corresponding *kernel* evaluations. Besides the enclosing ball methodology, Schölkopf et al. [10] proposed a one-class support vector machine (SVM) that uses a hyperplane to separate the normal patterns from the outliers with a large margin. Again, this leads to a QP problem. Moreover, when a Gaussian kernel is used, the solution of the one-class SVM is equivalent to that of SVDD. For further details, the interested reader can refer to [11,2,3] for several comprehensive surveys of existing novelty detection algorithms.

In this paper, we employ the hyperplane approach for novel detection as it can handle high dimensional datasets as well as achieve good generalization performance on previously unseen data. Moreover, it is easy to incorporate constraints to encode prior knowledge of the outliers into the training process of the hyperplane approach. Several empirical observations provide the motivation for our proposed novelty detection method. First, normal patterns are usually grouped together, forming clusters in the high density regions of the data space. Second, the outliers are characteristically very different from the normal patterns, and hence far away from the normal patterns. Third, the number of outliers is generally small for a given dataset. Based on these observations, we envisage that the decision boundary between the outliers and the normal patterns lies in some low density regions of the data space. To the best of our knowledge, such empirical knowledge has not been explicitly exploited for novelty detection. Moreover, the scarcity of the outliers may serve as a constraint to help identify such outliers from the normal patterns.

The major contributions of this paper are outlined as follows:

(1) We observe that the decision function to segregate the normal patterns and the outliers lies in some low density regions and can be employed for novelty detection. This is known as cluster assumption. Hence, we explicitly define a constraint based on this empirical observation in the corresponding optimization problem to facilitate novelty detection. The resultant optimization process can learn the decision function (decision boundary) and the labels of the data samples simultaneously.
(2) Due to the combinatorial nature of the problem, the resultant optimization process is NP hard. Hence, we introduce a convex relaxation to this non-convex optimization problem, and subsequently proposed an efficient cutting plane algorithm to solve this convex relaxation.
(3) By exploiting the scarcity of the outliers, we have devised an efficient method to find an approximation to the most violating labeling for our proposed cutting plane algorithm.
(4) Comprehensive experimental results on several benchmark datasets demonstrated that our proposed novelty detection technique outperforms the existing hyperplane-based novelty detection approaches.

The rest of this paper is organized as follows. Section 2 gives a brief review on the one-class SVM and the notion of cluster assumption. Section 3 describes our proposed methodology for novelty detection, and the corresponding experimental results are presented in Section 4. Section 5 concludes the paper.

For simplicity, the transpose of a vector/matrix will be denoted by the superscript $'$, and $\mathbf{0}, \mathbf{1} \in \mathbb{R}^n$ denote the zero vector and the vector of all ones, respectively. In addition, the inequality $\mathbf{v} = [v_1, \ldots, v_k]' \geq \mathbf{0}$ means that $v_i \geq 0$ for $i = 1, \ldots, k$. tr($\mathbf{X}$) is short for tr($\mathbf{X}$) which means the sum of the diagonal elements of the matrix $\mathbf{X}$.

2. Review of related works

2.1. One-class support vector machine (SVM)

We first review the one-class SVM. Given a set of unlabeled patterns $\{\mathbf{x}_i\}_{i=1}^n$ where $\mathbf{x}_i \in \mathcal{X}$ is the input, these patterns are first mapped to the feature space $\mathcal{H}$ via a nonlinear mapping function ϕ induced by a kernel k. Next, we assume that the outliers generally lie in the low density regions of the data space. In the one-class SVM, the outliers are presumed to be close to the origin [10], and a decision function $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x})$ is found to separate the majority of the data (the normal patterns) from the origin (the reference for the outliers) with a large margin $\rho/\|\mathbf{w}\|$ by solving the following structural risk functional:

$$\min_{\mathbf{w},\rho,\xi_i} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i^p - \rho : \mathbf{w}'\phi(\mathbf{x}_i) \geq \rho - \xi_i, \quad \xi_i \geq 0 \ \ \forall i = 1, \ldots, n, \tag{1}$$

where C is a parameter that trades off the empirical risk $\sum_{i=1}^n \xi_i^p$ (ξ_i is the slack variable) and the model complexity $\|\mathbf{w}\|^2$, and $p = 1$ or 2 corresponds to the hinge loss and the squared hinge loss, respectively.

This constrained optimization problem (for $p = 1$ or 2) is usually solved using its dual form. For simplicity, we just present the case of $p = 2$ in this paper, i.e.

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j\left(k(\mathbf{x}_i, \mathbf{x}_j) + \frac{1}{C}\delta_{ij}\right) : \quad \alpha_i \geq 0, \quad \sum_{i=1}^{n} \alpha_i = 1, \tag{2}$$

where α_i is a dual variable for each inequality constraint in (1), and δ_{ij} is an indicator function (i.e., $\delta_{ij} = 1$ if $i = j$; and 0 otherwise). Let $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]'$ be the vector of dual variables, and $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)] \in \mathbb{R}^{n\times n}$ be the kernel matrix, and $\mathcal{A} = \{\boldsymbol{\alpha} | \boldsymbol{\alpha} \geq \mathbf{0}, \boldsymbol{\alpha}'\mathbf{1} = 1\}$. Then the QP in (2) can be re-expressed as

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}'\left(\mathbf{K} + \frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}. \tag{3}$$

When the Gaussian kernel is used, this QP problem is equivalent to the dual of SVDD, and hence they share the same solution. Moreover, the decision function to identify the outliers can be expressed as

$$f(\mathbf{x}) = \sum_{i:\alpha_i > 0} \alpha_i k(\mathbf{x}_i, \mathbf{x}), \tag{4}$$

which is an expansion of the kernel evaluations on the support vectors only. Thus, identifying the outliers can be very efficient when the number of support vectors is small.

2.2. Cluster assumption

It may be too restrictive to simply assume that the outliers are located close to the origin. As discussed in [11,3,7,12], outliers generally are located far away from the majority of the data (i.e. the normal patterns). These normal patterns often lie in the high density regions of the data space forming clusters. Intuitively, the corresponding decision boundary to separate the normal patterns from the outliers should lie in some low density regions of the data space. This is referred to as cluster assumption [13]. Such a notion has been widely used in semi-supervised learning (SSL) algorithms [13,14] such as transductive SVM (TSVM) to perform text categorization [15]. Besides SSL, cluster assumption has also been employed in the development of unsupervised learning models such as maximum margin clustering [16].

In practice, cluster assumption is generally realized using transductive learning. Transductive learning is employed to learn

the decision function and the labels of the data samples simultaneously, such that the boundary of the decision function is located in some low density regions between the two different classes/clusters representing the normal patterns and the outliers. To the best of our knowledge, explicitly enforcing the decision boundary to lie in the low density regions of the data space for the purpose of identifying the outliers has not been widely explored in novelty detection research.

3. Methodology

In this paper, we suppose that a set of unlabeled patterns $\{\mathbf{x}_i\}_{i=1}^n$ is given. We further assume that the majority of the patterns are normal patterns, and we know the fraction of outliers to be ν. The task of novelty detection is thus to learn a decision function $f(\mathbf{x})$ that assigns a label y_i for the pattern $\mathbf{x}_i$ (that is, classify $\mathbf{x}_i$ as a normal pattern or an outlier). Without loss of generality, we assume that the class label for the normal patterns is +1 and the class label for the outliers is -1, respectively.

3.1. Proposed formulation

As discussed in Section 2.2, the notion of cluster assumption is inherent in novelty detection but has not been explicitly exploited during the learning process to derive the decision function. Motivated by the success of the transductive SVM (TSVM) [15] and the maximum margin clustering [16] techniques, we consider the search for a decision function $f(\mathbf{x})=\mathbf{w}'\phi(\mathbf{x})$ that minimizes the following structural risk functional:

$$\min_{\mathbf{y}\in\mathcal{Y}}\min_{\mathbf{w},\rho,\xi_i}\frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{n}\xi_i^p-\rho,:\quad y_i\mathbf{w}'\phi(\mathbf{x}_i)\geq\rho-\xi_i,\ \ \xi_i\geq 0 \tag{5}$$

which is very similar to the objective function of the one-class SVM in (1), except that we have also introduced the labels $\mathbf{y}\in\mathcal{Y}$ as variables in the optimization problem, where

$$\mathcal{Y}=\left\{\mathbf{y}|y_i\in\{\pm 1\},\ \sum_{i=1}^{n}y_i=n(1-2\nu)\right\} \tag{6}$$

and ν is the fraction of outliers in the given dataset. Hence, for a small ν, most of the y_i's are +1. Note that (5) learns the decision function $f(\mathbf{x})$ and the corresponding labels of all the data patterns simultaneously.

By replacing the inner minimization problem with its dual, and considering only the scenario of the squared hinge loss,[1] (5) can be rewritten as

$$\min_{\mathbf{y}\in\mathcal{Y}}\max_{\boldsymbol{\alpha}\in\mathcal{A}}-\frac{1}{2}\boldsymbol{\alpha}'\left(\mathbf{K}\odot\mathbf{y}\mathbf{y}'+\frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}, \tag{7}$$

where $\odot$ denotes the element-wise product. Note that the resultant decision function can be expressed as

$$f(\mathbf{x})=\sum_{i:\alpha_i>0}\alpha_i y_i k(\mathbf{x}_i,\mathbf{x})=\underbrace{\sum_{i:\alpha_i>0,y_i=1}\alpha_i k(\mathbf{x}_i,\mathbf{x})}_{\text{normal patterns}}-\underbrace{\sum_{i:\alpha_i>0,y_i=-1}\alpha_i k(\mathbf{x}_i,\mathbf{x})}_{\text{outliers}}. \tag{8}$$

Suppose that $k(\mathbf{x}_i,\mathbf{x})\geq 0$ for all $\mathbf{x}_i$'s (e.g. using the Gaussian kernel), we can observe that (4) is always positive. On the other hand, (8) evaluates as either positive or negative, and hence is more discriminative in differentiating the outliers from the normal patterns in a given dataset.

To make the optimization process more tractable, one can adopt the concept of convex relaxation as in the maximum margin clustering algorithm [16]. This eventually leads to a convex semidefinite programming (SDP) problem. However, as this will involve $O(n^2)$ optimization variables, it is computationally expensive even for small datasets.

Recently, Li et al. [17] proposed an efficient convex optimization method for the mixed integer programming problem of the maximum margin clustering (MMC) algorithm. Referred to as LG-MMC, the method maximizes the margin separation between two opposite clusters via a multiple label-kernel learning and "Label-Generation" strategy. It has been shown that the proposed strategy achieved a tighter convex relaxation than the SDP relaxation with respect to MMC and is very scalable on large datasets [17]. The basic idea involves the generation of a set of active constraints (indexed by the label vectors) to solve the corresponding mixed integer programming problem. Inspired by LG-MMC, we extend the algorithm to solve the mixed integer programming problem of (7). Since each label vector is generated to identify the novel (deviant) patterns from the normal ones, our proposed method is intuitively referred to as a "Novelty Detection Machine" (NDM).

3.2. Convex relaxation

With respect to the works reported in [17–19], we introduce a mild convex relaxation for our proposed novelty detection machine. We first consider interchanging the order of $\max_{\boldsymbol{\alpha}\in\mathcal{A}}$ and $\min_{\mathbf{y}\in\mathcal{Y}}$ in (7), leading to

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}}\min_{\mathbf{y}\in\mathcal{Y}}-\frac{1}{2}\left(\boldsymbol{\alpha}'\left(\mathbf{K}\odot\mathbf{y}\mathbf{y}'+\frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}\right). \tag{9}$$

According to the minmax theorem [20], the optimal objective of (7) is an upper bound to that of (9). This can be further re-expressed as

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}}\left\{\max_{\theta}-\theta:S(\boldsymbol{\alpha},\mathbf{y}^t)\geq-\theta\quad\forall\mathbf{y}^t\in\mathcal{Y}\right\}, \tag{10}$$

where $S(\boldsymbol{\alpha},\mathbf{y}^t)=-\frac{1}{2}(\boldsymbol{\alpha}'(\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}+(1/C)\mathbf{I})\boldsymbol{\alpha})$.

This is a convex quadratic constrained quadratic programming (QCQP) problem. For the inner optimization subproblem, let $\mu_t\geq 0$ be the dual variable for each of the constraints. The corresponding Lagrangian can be obtained as

$$-\theta+\sum_{t:\mathbf{y}^t\in\mathcal{Y}}\mu_t\left(\theta-\frac{1}{2}\left(\boldsymbol{\alpha}'\left(\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}+\frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}\right)\right).$$

By setting its partial derivative with respect to θ to zero, we have $\sum\mu_t=1$. Let $\boldsymbol{\mu}$ be the vector of μ_t's, and $\mathcal{M}$ be the simplex $\{\boldsymbol{\mu}|\sum\mu_t=1,\mu_t\geq 0\}$. We can then replace the inner optimization subproblem with its dual and (10) becomes

$$\begin{aligned}&\max_{\boldsymbol{\alpha}\in\mathcal{A}}\min_{\boldsymbol{\mu}\in\mathcal{M}}-\frac{1}{2}\sum_{t:\mathbf{y}^t\in\mathcal{Y}}\mu_t\left(\boldsymbol{\alpha}'\left(\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}+\frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}\right)\\&=\min_{\boldsymbol{\mu}\in\mathcal{M}}\max_{\boldsymbol{\alpha}\in\mathcal{A}}-\frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{t:\mathbf{y}^t\in\mathcal{Y}}\mu_t\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}+\frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}.\end{aligned} \tag{11}$$

In (11), the equality holds as the objective function is concave in $\boldsymbol{\alpha}$ and convex in $\boldsymbol{\mu}$. Hence, (11) can be regarded as a form of multiple kernel learning (MKL) [21], where the target kernel matrix is a convex combination of $|\mathcal{B}|$ base kernel matrices $\{\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}\}$, each of which is constructed from a feasible label vector $\mathbf{y}^t\in\mathcal{Y}$. Since each base kernel $\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}$ is indexed by a label vector $\mathbf{y}^t$, we refer to it as a label kernel. Hence, this form of MKL is referred to as multiple label kernel learning (MLKL). Details on how to solve this MLKL problem will be described in the following subsections.

[1] The dual expression can be derived for $p=1$ or 2. For simplicity, we just present the formulation for $p=2$ (squared hinge loss) in this paper.

3.3. Cutting plane algorithm

Recall that (11) can be regarded as a MKL problem. However, due to the exponential number of possible labels $\mathbf{y}^t \in \mathcal{Y}$, the set of base kernels is also large. Hence, it is computationally intractable to solve (11) by existing MKL techniques [21,22]. Fortunately, not all the constraints in (10) are active at optimality, and including only a subset of these constraints can usually lead to a good approximation of the original optimization problem. Therefore, similar to [17–19], we apply the cutting-plane method to iteratively generate a pool of labels $\mathbf{y}^t$s to construct the quadratic inequality constraints in (10).

It is interesting to note that a similar strategy has been suggested in the recently proposed Infinite Kernel Learning (IKL) method [23], in which the kernel is learned from an infinite set of general kernel parameters. Hence, our MLKL formulation (with the kernel $\sum_{t:\mathbf{y}^t\in\mathcal{Y}}\mu_t \mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}$) can also be regarded as a variant of IKL. Following that, our proposed cutting plane algorithm enjoys the same convergence property as IKL [23].

The proposed cutting-plane algorithm is presented as Algorithm 1. We denote the subset of constraints by $\mathcal{C}\subset\mathcal{Y}$. First, we initialize the vector of Lagrangian multipliers $\boldsymbol{\alpha}$ to $(1/n)\mathbf{1}$, and set the working set $\mathcal{C}=\{\mathbf{y}^1\}$ so that we have one base kernel to begin with. Since the working set $\mathcal{C}$ is now a subset of $\mathcal{Y}$ (and thus the number of base kernel matrices is no longer exponential in size), one can perform MKL and obtain $\boldsymbol{\alpha}$ from (11). The most violated label vector $\mathbf{y}^t$ is added to $\mathcal{C}$, and the process is repeated until convergence.

Algorithm 1. Cutting plane algorithm for NDM.

1: Initialize $\boldsymbol{\alpha}=(1/n)\mathbf{1}$. Find the most violated $\mathbf{y}^1$ (Algorithm 2) and set $\mathcal{C}=\{\mathbf{y}^1\}$.
2: Run MKL for the subset of kernel matrices selected in $\mathcal{C}$ and obtain $\boldsymbol{\alpha}$ from (11).
3: Find the most violated $\mathbf{y}^t$ (Algorithm 2) and set $\mathcal{C}=\mathbf{y}^t\bigcup\mathcal{C}$.
4: Repeat steps 2–3 until convergence.

3.4. MLKL with a subset of kernel matrices

For simplicity, we use an adaptation of the SimpleMKL algorithm [22] to solve the multiple label-kernel learning (MLKL) problem defined on the current working set of label kernel matrices in $\mathcal{C}=\{\mathbf{y}^1,\ldots,\mathbf{y}^t,\ldots,\mathbf{y}^T\}$. Note that the feature map corresponding to the base kernel matrix $\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}$ is $y_i^t\phi(\mathbf{x}_i)$. The MKL problem in (11) thus corresponds to the following primal optimization problem:

$$\min_{\mu\in\mathcal{M},\mathbf{w},\xi} \quad \frac{1}{2}\sum_{t=1}^{T}\frac{1}{\mu_t}\|\mathbf{w}_t\|^2-\rho+\frac{C}{2}\sum_{i=1}^{n}\xi_i^2$$

$$\text{s.t.} \quad \sum_{t=1}^{T} y_i^t\mathbf{w}_t'\phi(\mathbf{x}_i)\geq\rho-\xi_i \quad \forall i=1,\ldots,n. \tag{12}$$

It is easy to show that its dual can be written as

$$\max_{\alpha\in\mathcal{A},\theta} -\theta : \theta\geq -S(\boldsymbol{\alpha},\mathbf{y}^t), \quad t=1,\ldots,T.$$

This is identical to (10). Following SimpleMKL, we solve (11) (or equivalently (12)) iteratively. First, we fix the mixing coefficients $\boldsymbol{\mu}$ of the base kernel matrices and solve the SVM's dual

$$\max_{\alpha\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{t=1}^{T}\mu_t\mathbf{K}\odot\mathbf{y}^t\mathbf{y}^{t\prime}+\frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}.$$

Then, we hold $\boldsymbol{\alpha}$ and use the reduced gradient method to update $\boldsymbol{\mu}$. These two steps are repeated interleavely until convergence.

3.5. Finding a violated label vector $\mathbf{y}^t$

Similar to IKL, finding the most violated constraint (indexed by the label $\mathbf{y}^t$) in MLKL is problem specific, and it constitutes the hardest challenge to existing cutting plane algorithms. In this subsection, we will describe how to search for the most violated constraint for novelty detection with respect to (6). Looking at (7), to find the most violated $\mathbf{y}^t$, we have to identify the $\mathbf{y}$ that maximizes

$$\max_{\mathbf{y}\in\mathcal{Y}} \mathbf{y}'(\mathbf{K}\odot\boldsymbol{\alpha}\boldsymbol{\alpha}')\mathbf{y}. \tag{13}$$

However, this is a concave QP and hence cannot be solved efficiently. As reported in [23], while the use of the most violated constraint may lead to a faster convergence, the cutting plane algorithm essentially only require the addition of a violated constraint at each iteration. Hence, we propose a simple and efficient method to find an approximation to the most violated $\mathbf{y}^t$. Let $\mathbf{G}=\mathbf{K}\odot\boldsymbol{\alpha}\boldsymbol{\alpha}'$, and (13) can be re-expressed as

$$\max_{\mathbf{y}\in\mathcal{Y}} \quad \mathbf{y}'\mathbf{G}\mathbf{y}. \tag{14}$$

Algorithm 2. Finding a violated $\mathbf{y}^t$ for NDM.

1: Initialize $\mathbf{y}_{diff}=\mathbf{y}_{sort}-\mathbf{y}_{cons}$, $\mathbf{b}=\mathbf{G}*\mathbf{y}_{cons}$.
2: Compute the value of the expression in (16) as follows.
$index_{nonzero}=find(\mathbf{y}_{diff}\neq 0)$;
$obj_{linear}=\mathbf{b}(index_{nonzero})'*\mathbf{y}_{diff}(index_{nonzero})$;
$obj_{sq}=\mathrm{tr}(\mathbf{G}(index_{nonzero},index_{nonzero})*\mathbf{y}_{diff}(index_{nonzero})*\mathbf{y}_{diff}(index_{nonzero})')$;
$obj=2*obj_{linear}+obj_{sq}$.
3: Randomly select a positive y_i and a negative y_j in $\mathbf{y}$, and exchange their signs. Let $\mathbf{y}_{diff}=\mathbf{y}-\mathbf{y}_{cons}$.
4: Recompute the value of the expression in (16) using step 2. If this value increases, we keep the change made to $\mathbf{y}$. Otherwise, we discard the change to $\mathbf{y}$ and $index_{nonzero}$.
5: Repeat steps 3–4 until convergence. The most violated $\mathbf{y}^t$ is obtained by $\mathbf{y}^t(index_{nonzero})=-\mathbf{1}$ and the other y_i^t's are +1.

In novelty detection, most data samples are the normal patterns; i.e., $y_i=+1$. Based on the empirical observation that outliers are scarce, we devise the following strategy to find a violated $\mathbf{y}^t$. First, we find an initial solution using

$$\max_{\mathbf{y}\in\mathcal{Y}} \quad \mathbf{y}'\mathbf{G}\mathbf{1}.$$

We sort the elements in the vector $\mathbf{G1}$ in ascending order, and assign $y_i=+1$ to the patterns corresponding to the large sorted values, and assign $y_i=-1$ to the last $n\nu$ patterns. This gives the initial label $\mathbf{y}_{sort}$, and we further define $\mathbf{y}_{cons}=\mathbf{1}$. From (14), we can deduce that

$$\begin{aligned}\max_{\mathbf{y}\in\mathcal{Y}} &(\mathbf{y}-\mathbf{y}_{cons}+\mathbf{y}_{cons})'\mathbf{G}(\mathbf{y}-\mathbf{y}_{cons}+\mathbf{y}_{cons})\\ &=\max_{\mathbf{y}\in\mathcal{Y}}\mathrm{tr}(\mathbf{G}\mathbf{y}_{cons}\mathbf{y}_{cons}')+2\,\mathrm{tr}(\mathbf{G}\mathbf{y}_{cons}(\mathbf{y}-\mathbf{y}_{cons})')\\ &\quad+\mathrm{tr}(\mathbf{G}(\mathbf{y}-\mathbf{y}_{cons})(\mathbf{y}-\mathbf{y}_{cons})').\end{aligned} \tag{15}$$

Since $\mathrm{tr}(\mathbf{G}\mathbf{y}_{cons}\mathbf{y}_{cons}')$ is a constant, we can simplify the right-sided expression as

$$\max_{\mathbf{y}\in\mathcal{Y}} \quad 2\,\mathrm{tr}(\mathbf{G}\mathbf{y}_{cons}(\mathbf{y}-\mathbf{y}_{cons})')+\mathrm{tr}(\mathbf{G}(\mathbf{y}-\mathbf{y}_{cons})(\mathbf{y}-\mathbf{y}_{cons})'). \tag{16}$$

As the majority of the data samples has label $y_i=+1$, it follows that most $y_i-y_{0i}=0$. Hence, we only need to consider the cases where $y_i-y_{0i}\neq 0$ and the corresponding part in the matrix $\mathbf{G}$. This significantly reduce the computational cost required to update $\mathbf{y}$. In the proposed strategy, we randomly choose a positive y_i and

a negative y_j, and switch their signs (i.e. the labels). We then recompute the value of the expression in (16). If the increase in this value is above a pre-defined threshold, we keep the change made to $\mathbf{y}$. Otherwise, we discard the change in the labeling and repeat the aforementioned steps. When the steps are repeated several times with no significant change in value to the expression in (16), the algorithm terminates. We summarize the proposed strategy as Algorithm 2. For the new incoming patterns, we can use a predicting function to obtain their labels such that

$$y^{pre} = \mathbf{w}'\varphi(\mathbf{x}) = \sum_t \mu_t \sum_i \alpha_i y_i^t k(\mathbf{x}_i, \mathbf{x}). \tag{17}$$

4. Experiments

In this section, we first use a synthetic dataset (referred to as *Banana*) to illustrate the deficiencies of several well-established novelty detection algorithms. Following that, comprehensive evaluations of these algorithms and our proposed Novelty Detection Machine (NDM) using a collection of the UCI benchmark datasets will be presented. The descriptions of the UCI datasets are listed as Table 1. The benchmarking algorithms are: (1) the Multi-Variate-Gaussian (MVG) method; (2) the Kernel Density Estimator (KDE) method; and (3) the One-class Support Vector Machine (OSVM), respectively.

Table 1
UCI datasets used in the experiments.

Name	# Instances	# Features
Delft pump	720	160
Breast cancer Wisconsin	699	9
Ball-bearing	4150	32
Waveform	5000	21
Splice	2991	60
Diabetes	768	8

4.1. Experimental setup

For the OSVM and NDM methods, the C parameter is selected from the set {0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000}; and a Gaussian kernel is used for KDE, OSVM and NDM. The width parameter σ of the Gaussian kernel $\exp(-\|z\|^2/2\sigma^2)$ is selected from the set $\{0.25\sqrt{\gamma}, 0.5\sqrt{\gamma}, \sqrt{\gamma}, 2\sqrt{\gamma}, 4\sqrt{\gamma}\}$, where γ is the average distance between all pairs of data samples in a given dataset.

In the experiments, half of the positive instances (normal patterns) are used to learn the decision functions for the various techniques (referred to as Stage I, the in-sample stage), while the remaining positive instances are partitioned equally for validation of the models and out-of-sample evaluation (denoted as Stage II), respectively. The various parameters are empirically determined using the validation set. For the negative instances (outliers), we set the outlier ratio ν at the 2%, 4% and 8% level for Stage I, while the remaining negative instances are equally divided for validation and out-of-sample evaluation. During Stage I, all instances (data samples) are treated as unlabeled data, and only the outlier ratio is known. With respect to novelty detection, discerning the positive and negative instances (i.e. the normal patterns and the outliers) is equally important. Hence, we adopt the Balanced Accuracy (BA) metric as the performance measure:

$$\mathrm{BA} = \frac{\mathrm{TP} + \mathrm{TN}}{2}, \tag{18}$$

where TP denotes the true positive ratio and TN is the true negative ratio, respectively. Each dataset is randomly partitioned into the in-sample, validation and out-of-sample subsets; and all the aforementioned novelty detection techniques are evaluated using the same subsets. The experiments are subsequently repeated 20 times and the various techniques are benchmarked based on their average performance.

4.2. Experimental results using synthetic and UCI datasets

The *Banana* dataset is a synthetic dataset used to study the preliminary performances of the four benchmarked novelty detection

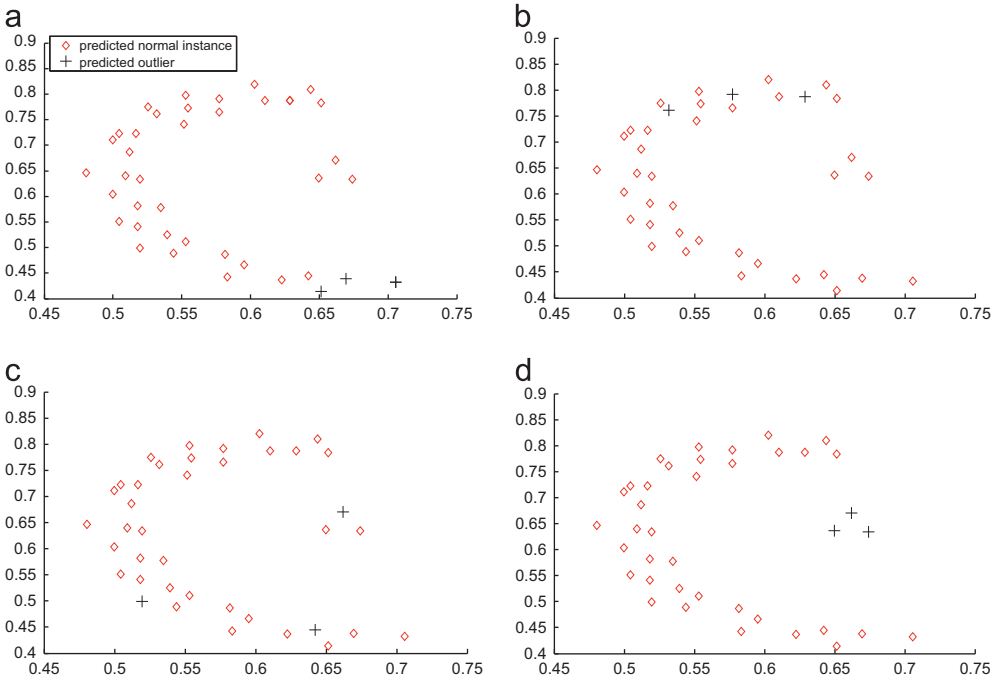

Fig. 1. Performances of the various benchmarked techniques on the *Banana* dataset. (a) MVG, (b) KDE, (c) OSVM and (d) NDM.

techniques. With reference to Fig. 1, the dataset consists of two distinct clusters, where the smaller cluster of three samples constitutes the outliers. The classification performances of the various techniques are subsequently plotted; and from Fig. 1, it is clear that only our proposed NDM algorithm can correctly detect all the outliers (novel points). On the other hand, both KDE and MVG fail to correctly identify any of the three outliers, while OSVM only manages to recognize one of the three novel points.

Following that, we evaluate the four benchmarked novelty detection techniques using six different UCI datasets. The classification results over 20 repetitions for Stage I (in-sample testing) and Stage II (out-of-sample testing) are tabulated as Tables 2 and 3, respectively. In Stage I, the in-sample data subsets are used to learn the decision functions for the various benchmarked techniques. The quality of the corresponding decision boundaries is evaluated by performing a classification of the data patterns in the respective in-sample subsets. From Table 2, we observe that the hyperplane-based novelty detection methods, namely OSVM and NDM, outperform the density estimation methods (i.e. MVG and KDE). In addition, NDM outperforms OSVM on five out of the six UCI datasets based on the performance measure defined in (18). This demonstrates that it is effective to use the notion of cluster assumption to identify the outliers.

Table 2
In-sample balanced accuracy measure (%) for the UCI datasets.

Dataset	ν (%)	MVG	KDE	OSVM	NDM
Delft	2	51.44 ± 0.10	91.06 ± 12.49	80.13 ± 11.91	75.74 ± 17.52
Pump	4	52.24 ± 1.14	95.44 ± 7.65	91.49 ± 0.00	81.10 ± 7.88
	8	53.02 ± 2.43	91.86 ± 5.65	81.32 ± 4.73	83.05 ± 8.38
Breast	2	47.60 ± 0.36	72.04 ± 14.04	92.33 ± 9.18	93.65 ± 11.29
Cancer	4	43.92 ± 0.63	79.69 ± 9.24	90.90 ± 8.16	97.40 ± 4.63
Wisconsin	8	38.85 ± 0.68	82.13 ± 5.86	79.92 ± 8.87	97.56 ± 2.76
Ball-bearing	2	51.24 ± 0.03	83.57 ± 5.48	78.03 ± 7.20	71.67 ± 6.63
	4	52.42 ± 0.06	83.83 ± 5.65	77.04 ± 3.82	77.48 ± 5.82
	8	54.71 ± 0.11	87.04 ± 3.12	76.28 ± 3.00	80.44 ± 3.75
Waveform	2	75.47 ± 3.60	78.18 ± 6.31	70.76 ± 4.75	77.22 ± 4.65
	4	77.78 ± 3.54	79.20 ± 4.01	67.51 ± 2.86	79.04 ± 4.28
	8	76.43 ± 2.53	78.07 ± 3.30	72.10 ± 0.37	81.91 ± 2.82
Splice	2	59.03 ± 6.03	58.05 ± 4.78	58.92 ± 4.95	65.11 ± 2.98
	4	59.74 ± 3.74	59.36 ± 4.05	58.5 ± 4.42	66.95 ± 2.81
	8	62.49 ± 3.04	62.39 ± 2.64	74.62 ± 0.00	69.59 ± 2.26
Diabetes	2	49.73 ± 3.81	50.58 ± 5.24	65.52 ± 9.13	66.55 ± 8.70
	4	49.17 ± 3.19	49.69 ± 3.80	63.34 ± 7.52	68.17 ± 3.19
	8	47.37 ± 2.31	49.09 ± 4.59	56.64 ± 4.38	60.16 ± 5.95

Table 3
Out-of-sample balanced accuracy measure (%) for the UCI datasets.

Dataset	ν (%)	MVG	KDE	OSVM	NDM
Delft	2	59.40 ± 5.31	77.07 ± 7.64	85.05 ± 5.15	90.32 ± 2.91
Pump	4	59.79 ± 4.91	77.18 ± 7.43	84.20 ± 5.10	89.04 ± 2.85
	8	59.59 ± 6.28	83.30 ± 5.09	79.79 ± 4.43	89.52 ± 2.77
Breast	2	46.05 ± 2.12	81.29 ± 9.46	84.63 ± 11.45	93.25 ± 2.43
Cancer	4	42.83 ± 4.24	78.00 ± 6.50	89.33 ± 7.65	94.46 ± 1.95
Wisconsin	8	36.64 ± 4.44	80.21 ± 7.57	76.71 ± 9.36	94.42 ± 2.13
Ball-bearing	2	51.50 ± 0.56	71.74 ± 2.50	82.76 ± 3.25	88.36 ± 1.57
	4	52.48 ± 0.78	75.26 ± 1.95	81.02 ± 2.21	88.70 ± 1.45
	8	54.68 ± 0.86	78.56 ± 1.95	81.52 ± 1.85	87.70 ± 2.03
Waveform	2	76.84 ± 1.48	76.94 ± 1.41	77.68 ± 1.15	81.65 ± 1.10
	4	76.97 ± 1.22	78.66 ± 1.21	74.47 ± 2.16	82.28 ± 1.35
	8	76.92 ± 1.28	78.57 ± 1.23	69.07 ± 2.39	82.58 ± 1.10
Splice	2	61.07 ± 1.24	55.10 ± 0.75	58.13 ± 1.37	67.83 ± 1.81
	4	62.55 ± 1.05	56.27 ± 0.91	58.34 ± 1.27	68.20 ± 1.76
	8	64.19 ± 1.39	58.54 ± 1.43	58.39 ± 1.76	67.43 ± 1.32
Diabetes	2	49.07 ± 1.92	49.93 ± 0.54	54.10 ± 4.68	54.78 ± 3.77
	4	49.66 ± 1.52	49.37 ± 1.17	54.44 ± 7.00	53.81 ± 3.70
	8	47.39 ± 2.25	49.89 ± 0.50	53.58 ± 4.79	53.81 ± 3.57

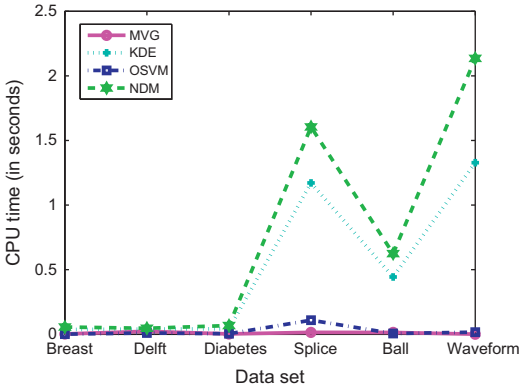

Fig. 2. Computation time comparison.

In Stage II, the generalization capability of the learned decision functions and the corresponding classifiers are evaluated using previously unseen data samples. The out-of-sample classification results are presented as Table 3. Again, one can easily observe that our proposed NDM algorithm has the most superior performance amongst the benchmarked techniques, with the highest accuracy measure reported across the different outlier ratios.

In Fig. 2, we depict the computation time of the various techniques for each of the datasets averaged over 60 independent experiments (i.e., $20 \times 3 = 60$ where the experiment for a given dataset was repeated 20 times for each of the three outlier ratios of 2%, 4% and 8%, respectively). All the methods except OSVM were implemented using MatLAB, while OSVM was coded in *C++*. Owing to this difference in computing platform, the computation time of OSVM is included for reference only. The plots show that MVG has the lowest computation time across the different UCI datasets, as it is only sensitive to the number of features in a dataset. KDE and NDM, on the other hand, are similar in their time complexity profiles as their computation times are proportional to the number of instances in a dataset. This is because KDE and NDM are both kernel-based methods, and hence they are indifferent to the number of features defining the data samples. Nevertheless, all the four methods are observed to be able to efficiently handle large datasets such as the *Ball-bearing* and *Waveform* datasets.

4.3. Case study: American bank failure prediction

Bank failure prediction is an important issue for the regulators of the banking industries. The collapse and failure of a bank could trigger an adverse financial repercussion and generate negative impacts such as a massive bail out cost for the failing bank and loss of confidence from the investors and depositors. Very often, bank failures are due to financial distress. Hence, it is desirable to have a system that can accurately identify potential bank failure or high-risk banks through the traits of financial distress. Such a system could serve as a early warning system (EWS) to aid bank regulation [24].

In the study of bank failures, there are different concepts of failure—economic, business and official—and there are further distinctions within each of these concepts [25]. Here, regulatory closure is the defining event of failure, because the event of regulatory closure is unambiguous and is more important and consistent than the straight-forward identification of problem banks. Such banks might come good in the future, given time or financial assistance or

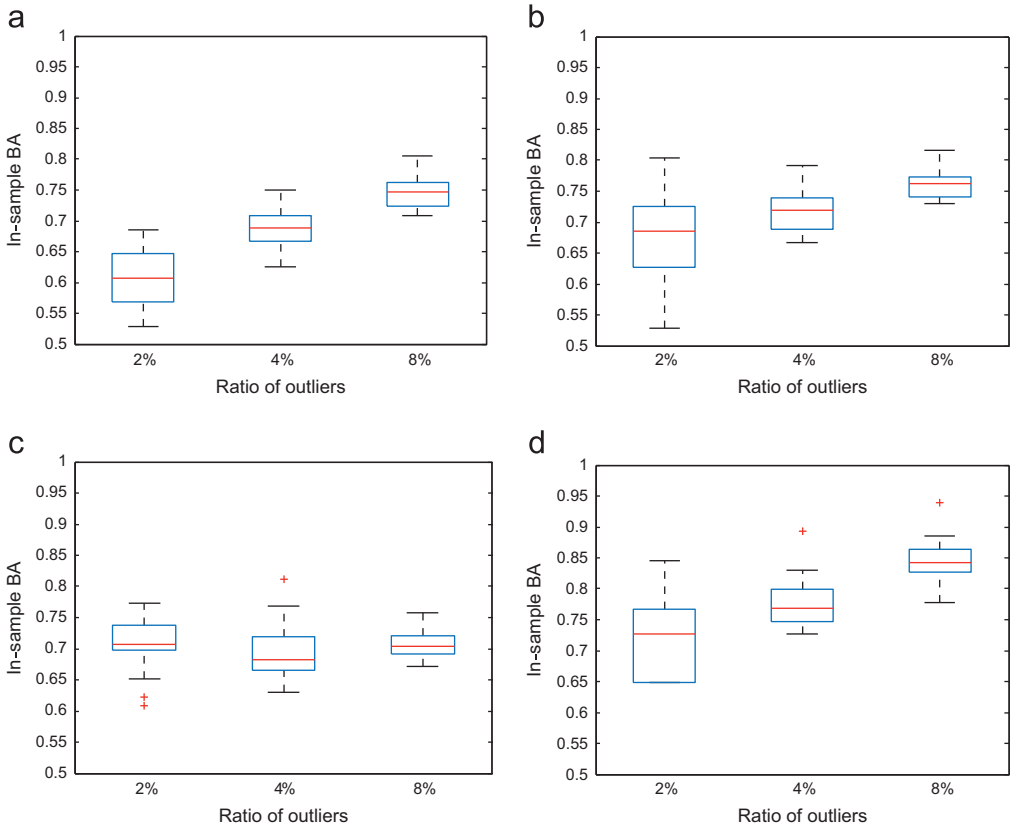

Fig. 3. In-sample detection results of the various banchmarked techniques for the bank failure dataset. (a) MVG, (b) KDE, (c) OSVM and (d) NDM.

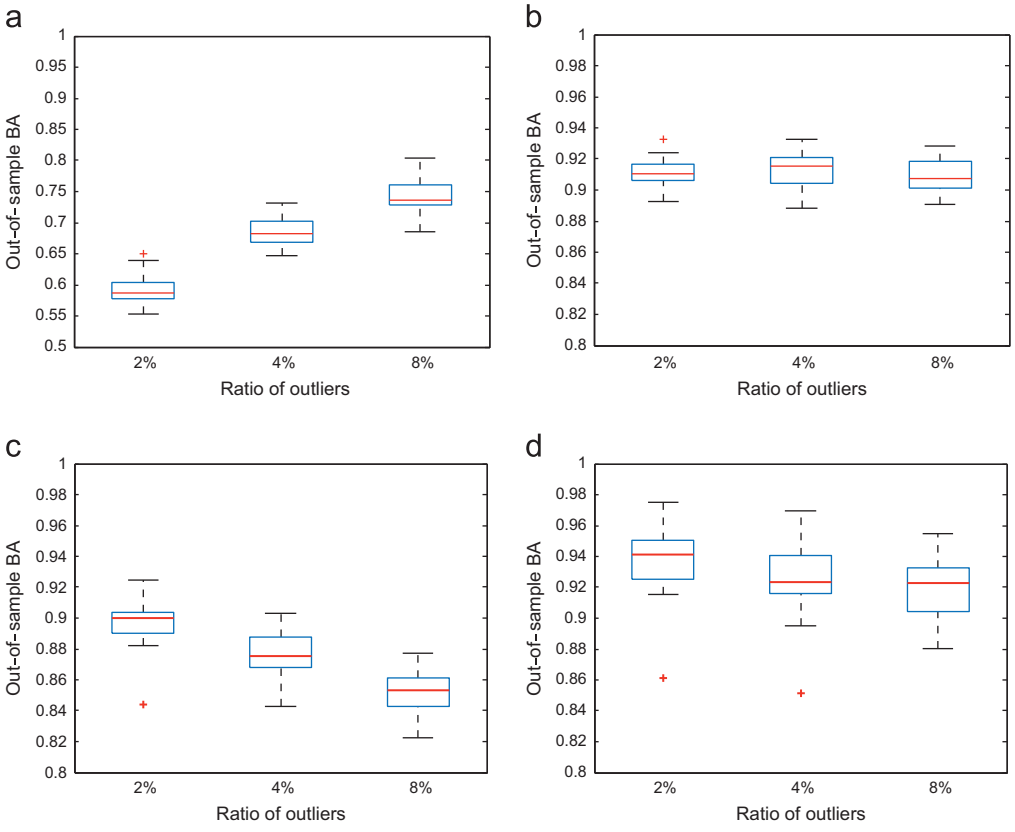

Fig. 4. Out-of-sample detection results of the various banchmarked techniques for the bank failure dataset. (a) MVG, (b) KDE, (c) OSVM and (d) NDM.

both. Besides, only the regulatory authorities can revoke or remove a bank charter to operate under existing ownership.

In this paper, the solvency of the banks that are tracked is characterized by the financial covariates (features) derived from the last publicly available annual financial statements of the observation period. The observation period of the survived (non-failing) banks consists of 21 years from January 1980 to December 2000 inclusively. For consistency, the data for the failed and survived banks have the same balance sheet dates. The financial variables (covariates) used in this bank failure prediction study are extracted from the Call Reports, which are downloaded from the website of the Federal Reserve Bank, Chicago [26]. The expected impacts of the variables on bank failures are explained in [25]. Normality plots of these variables indicate that the variables are not normally distributed. The statistical significance of the variables is investigated by best score selection, stepwise selection and purposeful selection [25]. Based on the findings of these selection procedures and an analysis of the correlations between the variables, only nine selected variables are used for the classification and prediction of banking failures [24]. The interested reader can refer to Table A1 in Appendix for the definitions of these selected financial features and their expected impacts on banking failure.

The raw dataset has been preprocessed to filter out the last available financial statement for each of the banks during the observation period. For the failed banks, it would be the records prior to failure while the records for the surviving banks are those of year 2000 (last year of the observation period). From the filtered financial statements, the nine selected variables (financial covariates) are extracted. These covariates provide a consistent measure to the financial health of the observed banking institutions. The interim dataset consists of 702 failed banks (with failure dates spreading across the entire observation period) and 2933 banks that survived the observation period, leading to a total of 3635 observed banks. However, banks whose record has missing fields are removed leading to the final dataset of 548 failed and 2555 survived (non-failing) banks. Hence, there are a total of 3103 observed banks. The failed banks constituted approximately 17.7% of the dataset while the survived banks made up the remaining

Table A1
Definition of the nine selected financial covariates and their expected impact on bank failure (numbers in bracket are the identification of the data elements from the downloaded Call Reports).

CAMEL category	Covariates	Impact
Capital adequacy	CAPADE Avg total equity capital (3210)/avg total assets (2170) (higher is the ratio, greater is the capacity to absorb losses, smaller is the probability of failure)	−ve
Asset (loan) quality	OLAQLY Avg (accumulated) loan loss allowance (3123)/ avg total loans & leases, gross (1400) (smaller is the ratio, better is the loan quality, smaller is the probability of failure)	−ve
	PROBLO Avg (accumulated) loans 90 + days late (1407)/ avg total loans & leases, gross (1400) (higher is the ratio, poorer is the loan quality, higher is the probability of failure)	+ve
	PLAQLY (Annual) loan loss provisions (4230)/avg total loans & leases, gross (1400) (higher is the ratio, poorer is the loan quality expected to be, higher is the probability of failure)	+ve
Management	NIEOIN Non-interest expense (4093)/operating income (4000) (higher is the ratio, less operationally efficient and profitable is the bank, higher is the probability of failure)	+ve
Earnings	NINMAR Total interest income (4107)− interest expense (4073)/ avg total assets (2170) (higher is the net interest margin, more profitable is the bank, smaller is the probability of failure)	−ve
	ROE Net income (after tax) (4340)+applicable income taxes (4302)/avg total equity capital (3210) (higher is return on equity before tax, smaller is the probability of failure)	−ve
Liquidity	LIQUID Avg cash (0010)+avg federal funds sold (1350)/ avg total deposits (2200)+avg fed funds purchased (2800)+avg banks liability on acceptances (2920)+ avg other liabilities (2930) (higher liquidity indicates inefficient utilization of resources; it can also reflect an expectation of unfavorable events (runs on deposits for example). Overall, higher liquidity suggests a higher probability of failure)	+ve
Miscellaneous	GROWLA Total loans & leases, gross $(1400)_t$−total loans & leases, gross $(1400)_{t-1}$/total loans & leases, gross $(1400)_{t-1}$ (with appropriate credit control and adequate loan loss provisions, a bank with higher loan growth rate would have better profitability and smaller probability of failure)	−ve

82.3%. The experimental setup for this study is similar to the procedure described in Section 4.1. The objective is to differentiate the failed banks from the surviving ones using the nine selected financial features. The classification results for the various benchmarked novelty detection algorithms for both Stage I (in-sample testing) and Stage II (out-of-sample testing) evaluation are depicted as Figs. 3 and 4, respectively. These results are derived across 20 repetitions of the experiments for each of the three outlier ratios.

For the in-sample testing, our proposed NDM technique is observed to have the best detection accuracies against the other three novelty detection algorithms for the different outlier ratio ν (see Fig. 3). This clearly demonstrates that the use of the notion of cluster assumption and the proposed optimization approach to learn the underlying decision function is effective in discerning the failed banks from the surviving ones. We subsequently extend the study to analyze the performances of the benchmarked algorithms on previously unseen data samples (i.e. Stage II—out-of-sample testing). The corresponding classification results are presented as Fig. 4. Similar to the scenario of in-sample testing, our proposed NDM algorithm has consistently outperformed the other benchmarked techniques in differentiating the failed and surviving banks, even though the test samples are previously unknown. From these results, we can see that the proposed NDM classifier can generalize well to new unseen data patterns. In addition, the proposed method has been evaluated in a diverse set of experimental data and has reported encouraging results. Moreover, NDM can handle mixed data attributes which are numeric, ordinal, categorical and symbolic. From this, we can infer that the proposed model has competent performance on a wide range of datasets.

For all the experiments reported in this paper, we assume that the outlier ratio ν is known. In practice, this parameter is often derived from the prior knowledge of the domain experts. For instance, credit card companies are familiar with the rate of fraudulent card usage. In cases where domain knowledge may not be readily available, we can obtain this ratio through random sampling methods. In comparison, other well-established outlier detection models such as OSVM [10] and SVDD [9] also have parameters that need to be empirically determined. To enhance our proposed model, we are currently looking to develop a technique to judiciously estimate the outlier ratio ν via Bayesian methods with maximum aposteriori probability (MAP) approximation [27].

5. Discussion and conclusions

In this paper, we have proposed a novelty detection technique based on several empirical observations about the differing characteristics of the normal patterns and the outliers that we want to segregate. These observations culminated to the notion of cluster assumption that has been studied in the semi-supervised learning paradigm. Using such an approach, we have formulated the learning of a decision function to separate the normal patterns and the outliers as a mixed integer programming optimization problem. Next, we present a cutting plane algorithm to solve the convex relaxation of this optimization problem. Specifically, we make use of the scarcity of the outliers to find a violating solution to the proposed cutting plane algorithm. The resultant classifier is thus intuitively named as *Novelty Detection Machine*. Comprehensive experiments using a collection of synthetic and UCI datasets have demonstrated that our proposed novelty detection method outperformed the existing hyperplane and density estimation-based novelty detection methods. We then apply the proposed technique to construct a classifier to perform banking failure prediction so as to identify potential bank failures or high risk banks through the traits of financial distress as captured by the financial covariates used to characterize the financial health of a set of observed banks. As future work, we will attempt to develop an efficient algorithm to reduce the time complexity and judiciously estimate the outlier ratio ν in the proposed NDM model.

Appendix A

See Table A1.

References

[1] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection for discrete sequences: a survey, IEEE Transactions on Knowledge and Data Engineering 24 (5) (2012) 823–839.

[2] S. Marsland, Novelty detection in learning systems, Neural Computing Surveys 3 (2003) 157–195.

[3] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey, ACM Computing Surveys 41 (3) (2009) 1–58.

[4] N. Meskin, A multiple model-based approach for fault diagnosis of jet engines, IEEE Transactions on Control Systems Technology 21 (1) (2013) 254–262.

[5] Á. Herrero, E. Corchado, M.A. Pellicer, A. Abraham, Movih-ids: a mobile-visualization hybrid intrusion detection system, Neurocomputing 72 (13–15) (2009) 2775–2784.

[6] L.J. Latecki, A. Lazarevic, D. Pokrajac, Outlier detection with kernel density functions, in: International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM), Leipzig, Germany, 2007, pp. 61–75.

[7] M.M. Breunig, H.-P. Kriegel, R.T. Ng, J. Sander, Lof: identifying density-based local outliers, in: ACM SIGMOD International Conference on Management of Data, Dallas, Texas, USA, 2000, pp. 93–104.

[8] F. Angiulli, Prototype-based domain description for one-class classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (6) (2012) 1131–1144.

[9] D.M.J. Tax, R.P.W. Duin, Support vector data description, Machine Learning 54 (1) (2004) 45–66.

[10] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, J. Platt, Support vector method for novelty detection, in: Advances in Neural Information Processing Systems 12, Denver, Colorado, USA, 2000, pp. 582–588.

[11] M. Markou, S. Singh, Novelty detection: a review, Part II: neural network based approaches, Signal Processing 83 (12) (2003) 2499–2521.

[12] S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets, in: ACM SIGMOD International Conference on Management of Data, Dallas, Texas, USA, 2000, pp. 427–438.

[13] O. Chapelle, A. Zien, Semi-supervised classification by low density separation, in: International Workshop on Artificial Intelligence and Statistics (AISTATS), Barbados, 2005, pp. 57–64.

[14] X. Zhu, Semi-supervised learning literature survey, Technical report, Computer Sciences Technique Report 1530, University of Wisconsin-Madison, 2009.

[15] T. Joachims, Transductive inference for text classification using support vector machines, in: International Conference on Machine Learning (ICML), Bled, Slovenia, 1999, pp. 200–209.

[16] L. Xu, J. Neufeld, B. Larson, D. Schuurmans, Maximum margin clustering, in: Advances in Neural Information Processing Systems, vol. 17, MIT Press, Vancouver, British Columbia, Canada, 2005, pp. 1537–1544.

[17] Y.-F. Li, I.W. Tsang, J.T.-Y. Kwok, Z.-H. Zhou, Tighter and convex maximum margin clustering, Journal of Machine Learning Research – Proceedings Track 5 (2009) 344–351.

[18] Y.-F. Li, J. Kwok, I. Tsang, Z.-H. Zhou, A convex method for locating regions of interest with multi-instance learning, in: European Conference on Machine Learning (ECML), Bled, Slovenia, 2009, pp. 15–30.

[19] Y.-F. Li, J. Kwok, Z.-H. Zhou, Semi-supervised learning using label mean, in: International Conference on Machine Learning (ICML), Montreal, Quebec, Canada, 2009, pp. 633–640.

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

[21] G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L.E. Ghaoui, M.I. Jordan, Learning the kernel matrix with semidefinite programming, Journal of Machine Learning Research 5 (2004) 27–72.

[22] A. Rakotomamonjy, F.R. Bach, S. Canu, Y. Grandvalet, Simple MKL, Journal of Machine Learning Research 9 (2008) 2491–2521.

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

[24] W.L. Tung, C. Quek, P.Y.K. Cheng, GenSo-EWS: a novel neural-fuzzy based early warning system for predicting bank failures, Neural Networks 17 (4) (2004) 567–587.

[25] P. Cheng, Predicting bank failures: a comparison of the cox proportional hazards model and the time varying covariates model, Doctoral dissertation, Nanyang Technological University, Singapore, 2002.

[26] Repository for bank data (online). Available: Federal Reserve Bank of Chicago. URL ⟨http://www.chicagofed.org⟩.

[27] B. Cseke, T. Heskes, Improving posterior marginal approximations in latent gaussian models, Journal of Machine Learning Research – Proceedings Track 9 (2010) 121–128.

Shukai Li is received his Ph.D. at the School of Computer Engineering, Nanyang Technological University, Singapore. He does research in outlier detection with kernel methods, credit risk management and ensemble learning. For the 23rd IEEE International Conference on Tools with Artificial Intelligence (ICTAI) 2011, in Florida, USA, he obtained the Best Paper Award. He also won the Best Student Paper Award (Gold Prize) from the Pattern Recognition and Machine Intelligence Association (PREMIA). He currently works as a research associate in the INSEAD Business School.

Whye Loon Tung received the B.A.Sc (first-class honors) and Ph.D.degrees in computer engineering (computational intelligence) from Nanyang Technological University (NTU) Singapore in 2000 and 2004, respectively. He is also a recipient of the prestigious 2004 Singapore Millennium Foundation (SMF) Postdoctoral Research Scholarship and the 2006 Lee Kuan Yew Postdoctoral Research Fellowship. His current research interests include neuro-cognitive science and the study of brain-inspired learning memory systems, artificial neural networks, fuzzy rule-based systems, computational finance and evolutionary computing techniques. He is currently a senior manager in the Measurement Science (Methods) team at Nielsen Singapore, working on statistical data modeling and data analytics problems.

Wee Keong Ng received his Ph.D. from the University of Michigan at Ann Arbor and is currently Associate Professor and Associate Chair (Research) at the School of Computer Engineering, Nanyang Technological University, Singapore. He works in privacy-preserving data analytics, encrypted data storage for cloud computing, data mining, and database systems with more than 200 refereed journal and conference publications in these areas. He is currently Associate Editor of the International Journal of Artificial Intelligence, Member of Editorial Review Boards of the International Journal of Applied Decision Sciences (IJADS), the International Journal of Intelligent Data Analysis (IJIDA), the International Journal of Intelligent Information and Database Systems (IJIIDS), and the International Journal of Intelligent Information Technologies (IJIIT). In recent years, he was on the International Advisory Committee of the 7th International Conference on Knowledge Information and Creativity Support Systems (KICSS2012), Senior PC Member of PAKDD2012, Program Committee Vice Chair of PRICAI2010, Asia Pacific Liaison Chair of the 7th IFIP WG6.11 Conference on e-Commerce, e-Business, and e-Government (I3E 2008), Program Vice Chair of PRICAI2008, Area Chair of PAKDD2008, and Program Co-Chair of ICSSSM'08. He is a member of IEEE, IEEE Computer Society and ACM.