OR Spectrum (2024) 46:1213–1239
https://doi.org/10.1007/s00291-024-00751-5

ORIGINAL ARTICLE

Convex and nonconvex nonparametric frontier-based classification methods for anomaly detection

Qianying Jin[1] · Kristiaan Kerstens[2] · Ignace Van de Woestyne[3]

Received: 16 February 2023 / Accepted: 23 January 2024 / Published online: 29 March 2024

Abstract

Effective methods for determining the boundary of the normal class are very useful for detecting anomalies in commercial or security applications—a problem known as anomaly detection. This contribution proposes a nonparametric frontier-based classification (NPFC) method for anomaly detection. By relaxing the commonly used convexity assumption in the literature, a nonconvex-NPFC method is constructed and the nonconvex nonparametric frontier turns out to provide a more conservative boundary enveloping the normal class. By reflecting on the monotonic relation between the characteristic variables and the membership, the proposed NPFC method is in a more general form since both input-like and output-like characteristic variables are incorporated. In addition, by allowing some of the training observations to be misclassified, the convex- and nonconvex-NPFC methods are extended from a hard nonparametric frontier to a soft one, which also provides a more conservative boundary enclosing the normal class. Both simulation studies and a real-life data set are used to evaluate and compare the proposed NPFC methods to some well-established methods in the literature. The results show that the proposed NPFC methods have competitive classification performance and have consistent advantages in detecting abnormal samples, especially the nonconvex-NPFC methods.

✉ Qianying Jin
qianying.jin@nuaa.edu.cn

Kristiaan Kerstens
k.kerstens@ieseg.fr

Ignace Van de Woestyne
ignace.vandewoestyne@kuleuven.be

[1] College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

[2] Univ. Lille, CNRS, IESEG School of Management, UMR 9221 - LEM - Lille Économie Management, F-59000 Lille, France

[3] KU Leuven, Research Center for Operations Research and Statistics (ORSTAT), Brussels Campus, Warmoesberg 26, B-1000 Brussels, Belgium

Keywords Nonparametric frontier · Convex · Nonconvex · Anomaly detection

1 Introduction

Anomaly detection can be defined as the task of detecting anomalous data that differ in some aspects from the normal data which is known during training. The practical use and challenging nature of anomaly detection have gained considerable research attention and led to many methods being proposed. The applications of anomaly detection methods settle across sectors and disciplines, such as in medical diagnosis (Park et al. 2010; da Silva et al. 2021), faults and failure detection in complex systems (Clifton et al. 2014; Jiang et al. 2020), and credit card or telecom fraud detection (Bhattacharyya et al. 2011; Ahmed et al. 2016; Al-Hashedi and Magalingam 2021) to name but a few. In all of these commercial or security applications, detecting potential anomalies are of crucial importance to prevent from some catastrophic outcome. For comprehensive and structured investigations of anomaly detection methods and their application domains, one may refer to the surveys of, e.g., Markou and Singh (2003a, 2003b); Ruff et al. (2021); Pang et al. (2021).

In practice, it is difficult and costly to collect large-scale labeled anomaly data. Thus, a typical data set observed in anomaly detection is extremely unbalanced, with a large number of normal data, but insufficient abnormal data to describe the anomaly or say novelty. Therefore, classical binary classification methods may not be applicable for anomaly detection since they normally require the two classes to be more or less balanced. To address the anomaly detection problem from a discriminative perspective, one-class classification, occasionally also called single-class classification, is introduced, see Moya et al. (1993); El-Yaniv and Nisenson (2006), or Khan and Madden (2014). The one-class classification anomaly detection methods are particularly based on the single class of normal data.

The fundamental idea of the one-class classification methods aims at finding a boundary around the normal class to describe the domain containing normal data only. If a new data point is located within the boundary, then it is regarded as normal; otherwise, it is an anomaly. Among the various one-class classification methods, one-class support vector classifiers (OC-SVCs) determine the boundary using only the normal data closest to it, i.e., the support vectors, not relying on any property of the distribution of the data. Over the past decades, OC-SVCs have been extensively studied and found suitable for anomaly detection in numerous applications, e.g., Alam et al. (2020).

Two evolutionary methods of OC-SVCs are the support vector data description (SVDD) method and the one-class support vector machine (OC-SVM) method. First, the SVDD method proposed by Tax and Duin (1999) defines a hypersphere with minimum radius that encloses the normal class. It gives a satisfactory performance but leads to a loose boundary for multivariate data sets, see Tax and Juszczak (2003). Second, the OC-SVM method proposed by Schölkopf et al. (1999) constructs a hyperplane to separate the normal class with the maximal margin from the origin in some feature space. In the OC-SVM method, all anomalies are assumed to be close to the origin, while the normal data points are far from the origin. In this respect, the OC-SVM method is not purely based on the normal class. Moreover, both the SVDD and OC-SVM methods have fixed their general shape of the boundary a priori: the former defines a hypersphere, and the latter describes a hyperplane. For cases where the normal class cannot be directly described by a hypersphere or a hyperplane, the introduction of kernel functions can bring some flexibility (Noble 2006).

In line with the fundamental idea of OC-SVCs, this contribution is also interested in determining the boundary of the normal class using only part of the normal data. Moreover, the boundary is expected to be around the data set without pre-determining the exact shape. To meet this, the Data Envelopment Analysis (DEA) method, which floats a piecewise linear boundary enveloping the observed data set, becomes of interest. It is a linear programming model proposed by Banker et al. (1984) and widely applied in production economics and finance, see the surveys and historical developments in Emrouznejad and Yang (2018); Emrouznejad et al. (2019); Kaffash et al. (2020). The piecewise linear boundary generated from the DEA model is commonly termed as a nonparametric frontier. The shape of the nonparametric frontier is determined by the originally observed data and the imposition of some weak maintained axioms, not necessarily in any feature space.

The main inspiration of introducing the nonparametric frontier to anomaly detection stems from the earliest work of Troutt et al. (1996) and a modified version by Seiford and Zhu (1998). They propose to develop an acceptance frontier with DEA method for case-based computer systems. Based on their work, theoretical extensions include characterizing data with non-discretionary characteristics (Leon and Palacios 2009), incorporating importance measures of characteristics (Yan and Wei 2011), etc. A series of empirical and experimental studies with the acceptance boundary were contemporaneously conducted by Pendharkar and coauthors in various domains, e.g., bankruptcy prediction (Pendharkar 2002), mining breast cancer patterns (Pendharkar et al. 2000), etc.

In the above classification methods, a convex (C) nonparametric frontier is constructed based on a certain group of training data. Then, this C nonparametric frontier is used to predict the group membership of test data. If a test data point is located within the C nonparametric frontier, then it has the same group membership as the training data which generate the C nonparametric frontier. Otherwise, it should be assigned to another group. We refer to these classification methods as C-nonparametric frontier-based classification (C-NPFC) methods.

These C-NPFC methods are all constructed from a single group of data. In this respect, these methods should have a unique advantage in anomaly detection problems, where only the group of normal data is sufficiently available. However, the existing C-NPFC methods originating from Troutt et al. (1996) are mainly designed to solve a classical binary classification problem. Therefore, this unique advantage of relying on only a single group of data to achieve correct classification has not received any attention in the literature related to C-NPFC methods.

To the best of our knowledge, the C-NPFC methods have not been applied to solve the anomaly detection problem. Moreover, the C-NPFC methods are essentially frontier analysis methods, and anomaly detection is one of the main tasks in supervised machine learning. Machine learning and frontier analysis are two relatively disconnected fields. In the literature, a research trend of applying well-known machine learning techniques to frontier analysis seems to emerge, e.g., Aparicio et al. (2021); Valero-Carreras et al. (2021); Zhu et al. (2021); Esteve et al. (2023). However, research applying frontier analysis methods to machine learning seems not to be developing in parallel. Therefore, the adaptation of C-NPFC methods to the anomaly detection problem can help bringing frontier analysis methods into the field of machine learning, thus creating some connection between two otherwise more or less unrelated fields.

In addition, there are in our opinion three key shortcomings of the existing C-NPFC methods which may constrain their classification capacity.

First, the existing C-NPFC methods are limited to construct a C nonparametric frontier. If the boundary of the class happens to be C, then a C nonparametric frontier offers a reasonable estimate. But, with our ignorance as to the real shape of the boundary, the convexity assumption can be overly optimistic. Pendharkar et al. (1999, p. 231) mention this as a potential harm to the capacity of the DEA frontier-based classification method while comparing it to the neural networks which are not constrained by convexity.

Second, the existing C-NPFC methods are limited to situations in which all characteristic variables have the property called conditional monotonicity. That is, acceptability of a case to a class increases with the increase or decrease in all characteristic variables. Thus, a radial DEA model without outputs or without inputs are adopted in the literature (Lovell and Pastor 1999).

Third, the existing C-NPFC methods are required to be constructed from all training observations. In other words, all training observations are presumed to be important in characterizing a group, yet some of them may be of very limited importance. Even worse, some of these training observations may be noises whose presence will overfit a misleading nonparametric frontier.

In conjunction, these restrictions are severe. A generic data set need not be separable by a C boundary, and it can simultaneously possess monotonically increasing and decreasing characteristics. Moreover, the data set may contain noisy or less important observations.

In this contribution, a general NPFC method is proposed to solve the anomaly detection problems and it can well compensate the above shortcomings. First, the convexity assumption is interpreted as reflecting a substitution relation between the characteristic variables. This relation does not always hold in practice. Therefore, we propose to relax the convexity assumption and construct a nonconvex-NPFC (NC-NPFC) method. This NC-NPFC method is based on the Free Disposal Hull (FDH) model, initially proposed by Deprins et al. (1984). Solving the FDH model results in a monotonous and staircase shaped nonparametric frontier enveloping the observed data. This NC nonparametric frontier is more conservative than the C nonparametric frontier. Second, the assumption of free disposability is interpreted as reflecting the monotonic relation between the characteristic variables and the membership. Therefore, both monotonically increasing and decreasing characteristic variables can be incorporated into the model simultaneously. Third, by allowing certain training observations to be misclassified, C and NC soft nonparametric frontiers are constructed with a super-efficiency model, initially proposed by Andersen and Petersen (1993). These soft nonparametric frontiers are believed to be less influenced by noise and less important observations. With these modifications, a generalized NPFC method is constructed: it can portray both monotonically increasing and monotonically decreasing characteristic variables, and it can generate a C hard frontier, a C soft frontier, an NC hard frontier, or an NC soft frontier.

To meet the above objectives, this contribution is structured as follows. Section 2 introduces the models and procedures used to construct the generalized NPFC method. In Sect. 3, both simulation studies and a real-life data set are used to show the classification performance of the NPFC methods relative to that of the OC-SVM and SVDD methods. Finally, Sect. 4 is concluded with a summary of the contributions and a discussion of potential future research topics.

2 Nonparametric frontier-based classification methods

2.1 Problem description

In anomaly detection problems, there is usually only a sufficient number of normal observations. The number of anomalous observations is very limited and therefore being insufficient for training a classifier. Thus, the training set consists of normal observations only. Let $G = \{Z_1, \dots, Z_n\}$ be the set of training observations. For a subset A of G, $i_G(A) = \{j \in \{1, \dots, n\} \mid Z_j \in A\}$ refers to the set of indexes of the elements of A in G.

Each training observation $Z_j \in G$ is characterized by a number of characteristic variables. These characteristic variables can be exclusively differentiated into two monotonic types, namely the monotonically decreasing characteristic variables denoted by $X = \{x_1, \dots, x_m\}$ and the monotonically increasing characteristic variables denoted by $Y = \{y_1, \dots, y_s\}$. The former is also termed as input-like characteristic variables, and the latter is termed as output-like characteristic variables. Generally, the observation is represented by $Z_j = (X_j, Y_j) \in \mathbb{R}^m \times \mathbb{R}^s$.

The monotonicity relations between the characteristic variables and the group membership are prior-knowledge of a classification problem. Consider the example of credit card default. All other characteristics being the same, cardholders with higher annual income are less likely to default compared to cardholders with lower income. That is, the probability of default should not decrease in the presence of better characteristics while the rest remains the same. Specifically, a characteristic variable is defined as being output-like if the probability of being normal increases (decreases) with the increase (decrease) of its value, e.g., the annual income in the example of credit card default. A characteristic variable is defined as being input-like if the probability of being normal increases (decreases) with the decrease (increase) of its value.

Given the training set G, an acceptance possibility set (APS) is constructed from the training observations and the imposition of some weak maintained axioms. It is a data-based description of the normal group. Any data point within this APS is perceived as normal and anomalous otherwise. Then, the boundary of the APS, termed as a nonparametric frontier, is used for anomaly detection. It consists part of the normal training observations. A test data point that has the same characteristic variables as the training observations is classified as normal if it lies within the nonparametric frontier and anomalous otherwise.

2.2 Convex and nonconvex acceptance possibility set

In this subsection, the normal observations from the training set G are used to describe the domain containing all possible normal data points. It describes all possible combinations of characteristic values for which the corresponding evaluated data point can be classified as normal.

In production analysis, a production possibility set (PPS) is used to describe the attainable set in production. For all the combinations of the inputs and the output within the PPS, these are attainable (producible) under a certain given technology. Instead of discussing the producibility under the PPS, the attainable set in classification describes the attainability in accepting an observation as normal. Hence, we define an APS to describe the attainable set of the normal group based on the training set G.

If a data point has the same characteristic values as a normal observation from the training set G, then it is in the APS. Based on the monotonicity relations, any data point with less X and more Y than an observation $Z_j \in G$ is perceived as having higher probability of being normal and thereby should be in the APS of the normal group. A free disposal set denoted by T_j is introduced to describe the situation under the monotonicity constraint. For every observation $Z_j \in G$, $T_j = \{(X, Y) \in \mathbb{R}^m \times \mathbb{R}^s \mid X \leq X_j \text{ and } Y \geq Y_j\}$.

The union of all the free disposal sets of the observations from G constitutes a NC APS denoted by T_{NC}. Specifically, T_{NC} depicts the normal group based on the n training observations as follows:

$$\begin{aligned} T_{NC} &= \bigcup_{j=1}^{n} T_j \\ &= \left\{ (X, Y) \in \mathbb{R}^m \times \mathbb{R}^s \mid \sum_{j=1}^{n} \lambda_j X_j \geq X, \sum_{j=1}^{n} \lambda_j Y_j \leq Y, \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \in \{0, 1\} \right\}. \end{aligned} \tag{1}$$

Figure 1 sketches a typical figure displaying a NC APS. All gray crosses are observations known from the normal group and constitute the training set G. These observations are characterized by two characteristic variables, namely $X \in \mathbb{R}^1$ and $Y \in \mathbb{R}^1$. For the monotonically decreasing characteristic variable X, the smaller its value the higher is the probability of belonging to the normal group. While for the monotonically increasing characteristic variable Y, the larger its value the higher is the probability of belonging to the normal group.

In Fig. 1, the observations from G are known to be normal. Thus, they are apparently in the APS. Then, we take the observation Z_6 as an example to explain the free disposal set. The free disposal set is built based on the monotonic relation of the

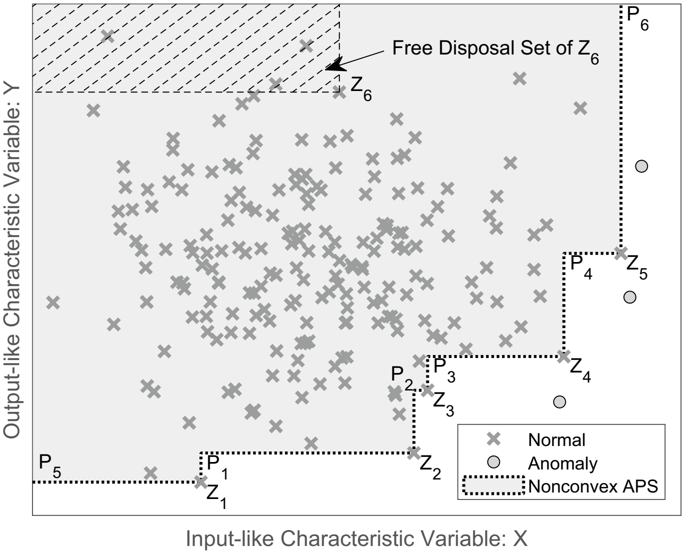


Fig. 1 Nonconvex APS of the normal group G

characteristic variables. If a data point has either a smaller X or a larger Y than Z_6, then it is more likely belonging to the normal group than Z_6 does. Since Z_6 belongs to the normal group, thus, a data point which has either a smaller X or a larger Y than Z_6 can be accepted as normal. This draws the dashed hatched area, which is located above and to the left of Z_6. This dashed hatched area represents the free disposal set of Z_6, namely T_6. If a new data point is located within this dashed hatched area, then it is regarded as normal just like the observation Z_6. For other observations from the training set G, their free disposal sets are derived in the same way. The union of all these free disposal sets constitutes the NC APS of the normal group. In Fig. 1, this is the shaded area restricted to the second quadrant located above and to the left of the dotted polyline $P_5Z_1P_1Z_2P_2Z_3P_3Z_4P_4Z_5P_6$.

In addition to the monotonicity assumption, the convexity assumption is commonly adopted in the literature. Mathematically, the axiom on convexity implies that for any two observations from one set, the linear combination of these two observations belong to the same set. In classification, this convexity axiom could explain a substitution relation between two characteristic variables. For example, both Z_2 and Z_4 in Fig. 1 are normal training observations. With the convexity assumption, their linear combinations, which locates on the line between Z_2 and Z_4, are also regarded as belonging to normal.

The C APS, denoted by T_C, is the convex hull of the NC APS. It depicts the normal group based on the n training observations as follows:

$$T_C = \left\{ (X, Y) \in \mathbb{R}^m \times \mathbb{R}^s \mid \sum_{j=1}^{n} \lambda_j X_j \geq X, \sum_{j=1}^{n} \lambda_j Y_j \leq Y, \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \geq 0 \right\}. \quad (2)$$

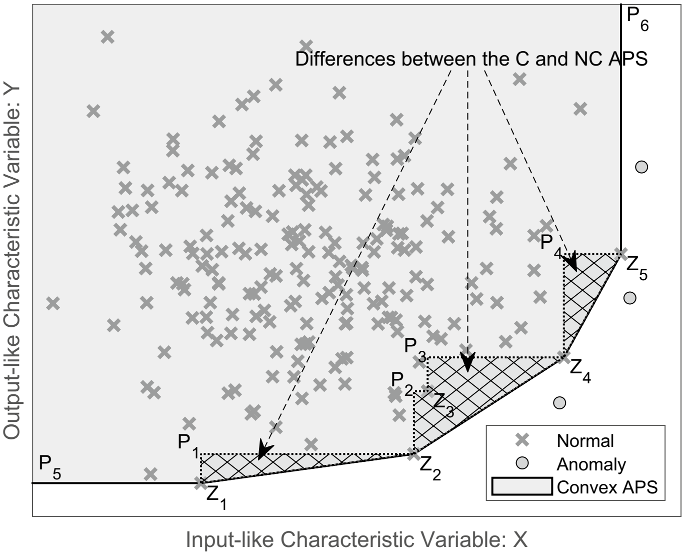


Fig. 2 Convex APS of the normal group G

Figure 2 shows a figure of a C APS. The same training observations as in Fig. 1 are used to illustrate the construction of a C APS. This C APS is built based on the NC APS with an additional assumption on convexity. Under the convexity assumption, the data points derived from the linear combination of observations must also be regarded as belonging to the normal group. For example, the data points on the line Z_2Z_4 should be classified into the normal group due to the convexity assumption. Then, the data points in the polygon $Z_2P_2P_3Z_4Z_2$ marked by grid lines can also be classified into the normal group with the monotonicity assumption. Overall, three polygons marked by grid lines are added, namely $Z_1P_1Z_2Z_1$, $Z_2P_2P_3Z_4Z_2$ and $Z_4P_4Z_5Z_4$. Thus, the C APS of the normal group is the shaded area restricted to the second quadrant located above and to the left of the solid polyline $P_5Z_1Z_2Z_4Z_5P_6$.

For the NC case, the APS consists of data points which are located within the free disposal area of certain training observation from G. While for the C case, except for the above situation, if a data point is located within the free disposal area of a convex combination of two training observations from G, it also constitutes the APS and therefore belongs to the normal group. Obviously, $T_{NC} \subseteq T_C$: a NC monotonic hull is a subset of a C monotonic hull. Put differently, the NC APS provides a tighter envelopment of the training observations than the C APS does.

To simplify the expressions, we use the following notation to stand for the APS of the normal group under the NC and C cases:

$$T_{\Lambda} = \left\{ (X, Y) \in \mathbb{R}^m \times \mathbb{R}^s \mid \sum_{j=1}^{n} \lambda_j X_j \geq X, \sum_{j=1}^{n} \lambda_j Y_j \leq Y, \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \in \Lambda \right\}. \quad (3)$$

where

$$\text{(i) } \Lambda \equiv \Lambda^{\mathrm{C}} = \left\{ \lambda_j \geq 0 \right\}, \quad \text{or (ii) } \Lambda \equiv \Lambda^{\mathrm{NC}} = \left\{ \lambda_j \in \{0, 1\} \right\}.$$

2.3 Convex and nonconvex hard nonparametric frontiers

Instead of using all the training observations, the APS of the normal group can be simply described by a number of training observations located on the nonparametric frontier. These training observations have the least preferable characteristic values and are located on the worst-practice frontier. Any data point with better characteristics than these training observations is assigned to be normal. On the contrary, any data point with worse characteristics than these training observations is assigned to be abnormal. In the following, the hard nonparametric frontiers under the C and NC settings are introduced correspondingly. These nonparametric frontiers are hard in the sense that all the normal training observations are used to construct a nonparametric frontier. In other words, all the normal training observations are required to be correctly classified during the training process.

Before constructing the hard nonparametric frontiers, we introduce the directional distance function (DDF) measure to gauge the relative distance of a data point Z to the frontier. Following Chambers et al. (1998), T_{Λ} is represented by the DDF measure ($D_{\Lambda,g}(Z)$) defined below:

$$D_{\Lambda,g}(Z) = \sup\{\delta \in \mathbb{R} \mid Z + \delta g \in T_{\Lambda}\}. \quad (4)$$

where $g = (g_X, g_Y) \in \mathbb{R}^m \times \mathbb{R}^s$ represents the projection direction. To be meaningful, $g_{x_i} > 0$ for all $i \in \{1, \dots, m\}$ and $g_{y_r} < 0$ for all $r \in \{1, \dots, s\}$. In this way, the characteristic variables X are non-decreasing and the characteristic variables Y are non-increasing while increasing the value of δ, which is the favorable behavior. Note that δ is a free decision variable that can take positive, zero or negative values.

All the C nonparametric frontier-based classification methods in the literature adopt either an input-oriented or an output-oriented radial efficiency measure. However, the adoption of these radial efficiency measures may lead to infeasibilities for the observations located outside the APS, when there are both input-like and output-like characteristic variables. While the DDF measure in expression (4) is well-defined for all possible data points, and for different monotonic types of characteristic variables.

The value of $D_{\Lambda,g}(Z)$ serves as an indicator that positions an observation relative to the hard frontier of the APS (T_{Λ}). A non-negative $D_{\Lambda,g}(Z)$ means that Z belongs to T_Λ. Specifically, an observation with $D_{\Lambda,g}(Z)$ equal to 0 means this observation is located on the hard frontier. If an observation Z is located outside T_Λ, then $D_{\Lambda,g}(Z)$ becomes negative and this observation is projected onto the hard frontier in the direction opposite to g.

Note that different choices of the direction vectors g lead to various distance values denoted by $D_{\Lambda,g}(Z)$. However, this choice does not change the sign of $D_{\Lambda,g}(Z)$. In the following, the direction vector is applied with $g = (|X_0|, -|Y_0|)$ for the observation $Z = (X_0, Y_0)$. This invests the DDF measure with a proportional interpretation, see Briec (1997); Kerstens and Van de Woestyne (2011). Such a percentage interpretation is not indispensable to assign a membership, but it remains convenient.

Based on the APS of the normal group defined by expression (3), the proportional DDF measure is then computed accordingly. With respect to T_Λ, the DDF of a data point $Z_0 = (X_0, Y_0)$ is obtained by solving model (5):

$$\begin{aligned}
&\max_{\lambda_j,\overline{\delta}_\Lambda} \ \overline{\delta}_\Lambda \\
&s.t. \quad \sum_{j=1}^{n} \lambda_j x_{i,j} \geq x_{i,0} + \overline{\delta}_\Lambda |x_{i,0}| \qquad \forall i \in \{1,\dots,m\} \\
&\qquad \sum_{j=1}^{n} \lambda_j y_{r,j} \leq y_{r,0} - \overline{\delta}_\Lambda |y_{r,0}| \qquad \forall r \in \{1,\dots,s\} \\
&\qquad \sum_{j=1}^{n} \lambda_j = 1 \\
&\qquad \lambda_j \in \Lambda \qquad \forall j \in \{1,\dots,n\}
\end{aligned} \tag{5}$$

where

$$\text{(i) } \Lambda \equiv \Lambda^{\mathrm{C}} = \left\{\lambda_j \geq 0\right\}, \ \text{ or (ii) } \Lambda \equiv \Lambda^{\mathrm{NC}} = \left\{\lambda_j \in \{0,1\}\right\}.$$

In the C case, model (5) is a linear programming (LP) problem, while it involves solving a binary mixed integer program (BMIP) for the NC case. To remedy the computational issue in the NC case, a fast implicit enumeration-based method is proposed by Cherchye et al. (2001) requiring only to compute minima and maxima of lists of ratios. Instead of solving a BMIP model, the following exact solution is obtained for model (5) under the NC case:

$$\overline{\delta}^*_{\Lambda^{\mathrm{NC}}} = \max_{j=1,\dots,n}\left(\min_{i=1,\dots,m}\left(\frac{x_{i,0} - x_{i,j}}{|x_{i,0}|}\right), \min_{r=1,\dots,s}\left(\frac{y_{r,j} - y_{r,0}}{|y_{r,0}|}\right)\right). \tag{6}$$

The assumption on convexity differentiates the NC APS ($T_{\Lambda^{\mathrm{NC}}}$) from the C APS ($T_{\Lambda^{\mathrm{C}}}$). However, this does not change the definition of the DDF measure, only the value of the DDF measure may change. Thus, $\overline{\delta}_{\Lambda^{\mathrm{NC}}} \leq \overline{\delta}_{\Lambda^{\mathrm{C}}}$, since $T_{\Lambda^{\mathrm{NC}}} \subseteq T_{\Lambda^{\mathrm{C}}}$.

By solving model (5) for all observations from G, a frontier set denoted by F_Λ is generated. F_Λ consists of the observations from G that have $\overline{\delta}^*_\Lambda = 0$. Normally, the set F_Λ under the NC case is different from that under the C case. All frontier observations in $F_{\Lambda^{\mathrm{C}}}$ can also be found in $F_{\Lambda^{\mathrm{NC}}}$. However, not all frontier observations in $F_{\Lambda^{\mathrm{NC}}}$ belong to $F_{\Lambda^{\mathrm{C}}}$, since some frontier observations under the NC case might be dominated by convex combinations of certain observations. Therefore, $F_{\Lambda^{\mathrm{C}}} \subseteqq F_{\Lambda^{\mathrm{NC}}}$.

We consider Figs. 1 and 2 in Sect. 2.2 as an example to illustrate the results of model (5). Under the NC setting, model (5) is executed for all training observations. Only observations Z_1, Z_2, Z_3, Z_4 and Z_5 have $\overline{\delta}^*_{\Lambda^{\mathrm{NC}}} = 0$, while the other observations have $\overline{\delta}^*_{\Lambda^{\mathrm{NC}}} > 0$. Thus, the NC hard frontier is represented by the frontier set $F_{\Lambda^{\mathrm{NC}}} = \{Z_1, Z_2, Z_3, Z_4, Z_5\}$. The NC hard frontier is the dotted polyline $P_5Z_1P_1Z_2P_2Z_3P_3Z_4P_4Z_5P_6$. Similarly, model (5) is executed for all observations under the C setting. Observations Z_1, Z_2, Z_4 and Z_5 still have $\overline{\delta}^*_{\Lambda^{\mathrm{C}}} = 0$, but observation Z_3 has $\overline{\delta}^*_{\Lambda^{\mathrm{C}}} > 0$ as all the other observations do. Thus, $F_{\Lambda^{\mathrm{C}}} = \{Z_1, Z_2, Z_4, Z_5\}$. The C hard frontier is the solid polyline $P_5Z_1Z_2Z_4Z_5P_6$.

2.4 Convex and nonconvex soft nonparametric frontiers

In order to limit the impact of potentially noisy and less important observations on constructing the nonparametric frontiers, C and NC soft nonparametric frontiers are introduced in this section. These C and NC soft model variations have been introduced by Andersen and Petersen (1993) and Kerstens et al. (2022), respectively.

The C and NC nonparametric frontiers are soft in the sense that some of the normal training observations are allowed to be misclassified during training. Allowing some normal training observations to be misclassified means that these misclassified normal training observations are excluded from the construction of a nonparametric frontier. The potentially noisy and less important training observations are the ones that should be excluded from the construction of the nonparametric frontier.

According to the monotonicity relation, the frontier observations have the least preferable characteristic values. If these frontier observations are distant from the rest of the training observations, they may be noisy or of low importance. Thus, in this contribution, training observations that are close to the hard nonparametric frontier but distant from the other training observations are identified as training observations that should be misclassified.

The distance of a training observation to the other training observations can be measured by a super-efficiency model. Specifically, model (7) is used to calculate the distance of a training observation $Z_0 = (X_0, Y_0)$ relative to the rest of the training observations. In this model, S_Λ represents some subset of the set of training observations G: its precise meaning is defined in Algorithm 1.

The optimal value $\delta^*_{\mathrm{super},\Lambda}$ measures the proportional distance of the training observation $Z_0 = (X_0, Y_0)$ to the rest of training observations denoted by $S_\Lambda \setminus \{Z_0\}$. If $\delta^*_{\mathrm{super},\Lambda}$ is non-negative, then the training observation $Z_0 = (X_0, Y_0)$ is located within the APS generated from $S_\Lambda \setminus \{Z_0\}$; otherwise, it is located outside the corresponding APS.

In order to detect the noisy and less important training observations that should be misclassified, a negative cut-off super-efficiency denoted by c_Λ is introduced. Specifically, if a training observation $Z_0 = (X_0, Y_0)$ satisfies $\delta^*_{\mathrm{super},\Lambda} \leq c_\Lambda$, then this training observation is identified as the one that should be misclassified.

$$
\begin{aligned}
\max_{\lambda_j,\delta_{\text{super},\Lambda}} \quad & \delta_{\text{super},\Lambda} \\
s.t. \quad & \sum_{j\in i_G(S_\Lambda\setminus\{Z_0\})} \lambda_j x_{i,j} \geq x_{i,0} + \delta_{\text{super},\Lambda}|x_{i,0}| && \forall i \in \{1,\ldots,m\} \\
& \sum_{j\in i_G(S_\Lambda\setminus\{Z_0\})} \lambda_j y_{r,j} \leq y_{r,0} - \delta_{\text{super},\Lambda}|y_{r,0}| && \forall r \in \{1,\ldots,s\} \\
& \sum_{j\in i_G(S_\Lambda\setminus\{Z_0\})} \lambda_j = 1 \\
& \lambda_j \in \Lambda && \forall j \in i_G(S_\Lambda \setminus \{Z_0\})
\end{aligned}
\tag{7}
$$

where

$$
\text{(i) } \Lambda \quad \equiv \quad \Lambda^{\text{C}} = \left\{\lambda_j \geq 0\right\}, \ \text{ or (ii) } \Lambda \equiv \Lambda^{\text{NC}} = \left\{\lambda_j \in \{0,1\}\right\}.
$$

In this contribution, we suggest one way of deciding the value of c_Λ. First, model (7) is calculated for every training observation in G. The $\delta^*_{\text{super},\Lambda}$ calculated for the frontier observations in G should be non-positive. Then, these non-positive $\delta^*_{\text{super},\Lambda}$ are ordered from largest to smallest and are denoted by $\delta^*_{\text{super},\Lambda}(Z_1)$, $\delta^*_{\text{super},\Lambda}(Z_2)$,.... The difference between two adjacent $\delta^*_{\text{super},\Lambda}$ should be small. If there is a considerable jump observed in the difference, i.e., if there is a significant large difference between $\delta^*_{\text{super},\Lambda}(Z_j)$ and $\delta^*_{\text{super},\Lambda}(Z_{j+1})$, then c_Λ is taken to be the decile between $\delta^*_{\text{super},\Lambda}(Z_j)$ and $\delta^*_{\text{super},\Lambda}(Z_{j+1})$, and closest to $\delta^*_{\text{super},\Lambda}(Z_j)$. An illustrative example for deciding c_Λ is given in Appendix. Note that our suggested method for determining the value of c_Λ is a feasible approach, but it is not necessarily the best one. Further research on this issue is certainly valuable: we return to this in Sect. 4.

Algorithm 1 Training a nonparametric frontier

```
Given:
    Training data set: $G$, Input-like characteristics variables: $X$
    Assumption on Convexity, Output-like characteristics variables: $Y$
    Cut-off super-efficiency: $c_\Lambda$
Training:
    Let $S_\Lambda = G$, $Mis_\Lambda = \emptyset$, $l = 1$
    while $l \neq 0$
      $l = 0$, $F_\Lambda = \emptyset$
      for every $Z_j \in S_\Lambda$
        Calculate $\delta^*_{\text{super},\Lambda}$ using model (7)
        if $\delta^*_{\text{super},\Lambda} \leq c_\Lambda$, then
                $Mis_\Lambda = Mis_\Lambda \cup \{Z_j\}$, $l = 1$
        end if
        if $c_\Lambda < \delta^*_{\text{super},\Lambda} \leq 0$, then
                $F_\Lambda = F_\Lambda \cup \{Z_j\}$
        end if
      end for
      $S_\Lambda = G \setminus Mis_\Lambda$
    end while
    Export $F_\Lambda$ for characterizing the nonparametric frontier
```

Algorithm 1 is designed to train a nonparametric frontier with a given c_Λ. First, let $S_\Lambda = G, Mis_\Lambda = \emptyset$. Then, the loop starts by calculating the optimal $\delta^*_{\text{super},\Lambda}$ for every training observation in S_Λ. If a training observation Z_j satisfies $\delta^*_{\text{super},\Lambda}(Z_j) \leq c_\Lambda$, it is identified as a misclassified training observation and will be collected to Mis_Λ. As long as there are misclassified training observations identified, the loop continues with $S_\Lambda = G \setminus Mis_\Lambda$; otherwise, the loop ends. After executing Algorithm 1, all the training observations in S_Λ should have $\delta^*_{\text{super},\Lambda} > c_\Lambda$. Those training observations in S_Λ that satisfy $0 \geq \delta^*_{\text{super},\Lambda} > c_\Lambda$ are recognized as the frontier observations and will be collected to the frontier set F_Λ.

Note that Algorithm 1 is quite general, since it can train both soft and hard nonparametric frontiers, as well as C and NC nonparametric frontiers. Specifically, the assumption on convexity differentiates the nonparametric frontier as a C or a NC one. A hard nonparametric frontier can be trained by using a relatively small value of c_Λ, e.g., $c_\Lambda = -100$. A soft nonparametric frontier is trained by deciding a suitably large value of c_Λ based on the suggested way.

2.5 Nonparametric frontier-based classification rules

Regardless of whether the NPFC method uses a hard or soft nonparametric frontier, their classification rules are the same. Therefore, this subsection does not differentiate between NPFC methods that use hard or soft frontiers when presenting the classification rules.

The membership of a new data point is decided by the relative location with respect to the hard or soft nonparametric frontier characterized by the frontier set F_Λ. Specifically, model (8) is used to calculate the distance of the new data point $Z_0 = (X_0, Y_0)$ relative to the nonparametric frontier.

$$\begin{aligned}
\max_{\lambda_j, \delta_\Lambda} \quad & \delta_\Lambda \\
s.t. \quad & \sum_{j \in i_G(F_\Lambda)} \lambda_j x_{i,j} \geq x_{i,0} + \delta_\Lambda |x_{i,0}| && \forall i \in \{1, \dots, m\} \\
& \sum_{j \in i_G(F_\Lambda)} \lambda_j y_{r,j} \leq y_{r,0} - \delta_\Lambda |y_{r,0}| && \forall r \in \{1, \dots, s\} \\
& \sum_{j \in i_G(F_\Lambda)} \lambda_j = 1 \\
& \lambda_j \in \Lambda && \forall j \in i_G(F_\Lambda)
\end{aligned} \tag{8}$$

where

$$\text{(i) } \Lambda \equiv \Lambda^{\text{C}} = \left\{\lambda_j \geq 0\right\}, \ \text{ or (ii) } \Lambda \equiv \Lambda^{\text{NC}} = \left\{\lambda_j \in \{0, 1\}\right\}.$$

The decision variable δ_Λ in model (8) is a free variable. If $\delta^*_\Lambda \geq 0$, then it indicates that there exists a projection point that dominates the new data point $Z_0 = (X_0, Y_0)$. This projection point is generated from the left-hand side of the inequality constraints in model (8) and is represented by $Z_b = (\sum_{j\in i_G(F_\Lambda)} \lambda_j^* X_j, \sum_{j\in i_G(F_\Lambda)} \lambda_j^* Y_j)$. It is either an observation from the NC frontier set $F_{\Lambda^{\mathrm{NC}}}$ or a convex combination of the observations from $F_{\Lambda^{\mathrm{C}}}$. In the case where $\delta_\Lambda^* \geq 0$, the following inequalities hold: $\sum_{j\in i_G(F_\Lambda)} \lambda_j^* X_j \geq X_0$ and $\sum_{j\in i_G(F_\Lambda)} \lambda_j^* Y_j \leq Y_0$. Comparing to the projection point Z_b which is normal, the new data point $Z_0 = (X_0, Y_0)$ has less X and more Y. Therefore, it should be assigned to the normal group.

By contrast, if $\delta_\Lambda^* < 0$, then the new data point Z_0 dominates the projection point Z_b. That is, $\sum_{j\in i_G(S_\Lambda)} \lambda_j^* X_j < X_0$ and $\sum_{j\in i_G(S_\Lambda)} \lambda_j^* Y_j > Y_0$. The projection point Z_b is on the boundary of the APS. Comparing to the projection point Z_b, the new data point Z_0 has more X and less Y, and therefore, it is situated outside the APS. Hence, the new data point Z_0 is assigned as an anomaly if there is no further information.

For a new data point Z_0 whose group membership is unknown, model (8) is executed and the optimal δ_Λ^* is calculated. Then, the group membership of Z_0 is decided based on the following Rule (9):

$$\begin{array}{c}\text{If } \delta_\Lambda^* \geq 0, \text{then } Z_0 \text{ belongs to the normal group;}\\ \text{Otherwise, } Z_0 \text{ belongs to the group of anomalies.}\end{array} \tag{9}$$

3 Experimental analysis

3.1 Experimental setup

The classification performance of the classifiers is characterized by 6 measures, namely, accuracy, precision, recall, specificity, F_1 score and G-mean. These performance measures are listed in Eqs. (10) to (15):

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{11}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{12}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{13}$$

$$F_1 \text{ score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

$$\text{G-mean} = \sqrt{\text{Precision} \cdot \text{Recall}} \tag{15}$$

whereby *TN*, *TP*, *FN*, and *FP* represent true negative, true positive, false negative and false positive cases, respectively. An anomaly is depicted as positive, while the normal case is depicted as negative.

Accuracy, which captures the percentage of correctly predicted samples out of all test samples, is the most commonly used overall performance measure. However, it fails to comprehensively reflect the overall performance when facing an unbalanced data set (Akbani et al. 2004). Instead of using accuracy, a common choice in the literature is to calculate the harmonic and geometric means of the recall and the precision, namely the F_1 score and the G-mean. These measures can alleviate the problem that the accuracy measure has with respect to the unbalanced data set (He and Garcia 2009). In addition to the F_1 score and G-mean, we also list the precision, recall and specificity to give the readers a better understanding of the performance of different groups. Specifically, precision depicts the percentage of true positive (abnormal) samples out of all predicted positive samples. Recall represents the percentage of correctly predicted positive (abnormal) samples out of all true positive samples. Specificity indicates the percentage of correctly predicted negative (normal) samples out of all true negative samples.

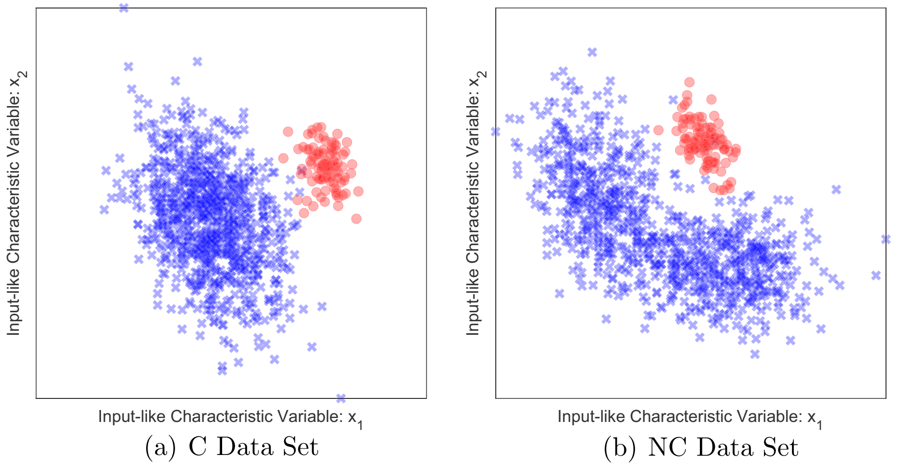
Four NPFC methods are evaluated in the experimental analysis, namely, the C-NPFC method with a hard frontier, the C-NPFC method with a soft frontier, the NC-NPFC method with a hard frontier and the NC-NPFC method with a soft frontier. The performance of the proposed NPFC methods is compared with two existing methods, namely, the OC-SVM and SVDD methods. It is important to remark that this comparison is intended to show that the proposed NPFC methods can be a good candidate for the anomaly detection problem, rather than always beat the best OC-SVM and SVDD methods. Thus, the OC-SVM method with a Gaussian kernel and the SVDD method with a polynomial kernel are chosen to be compared with. The OC-SVM method with a Gaussian kernel is implemented using the MATLAB function “fitcsvm”. The SVDD method with a polynomial kernel is implemented using the MATLAB code available from Qiu (2022).

3.2 Simulation studies

In this subsection, two simulation studies, one based on C data sets and the other on NC data sets, are designed to evaluate the proposed NPFC methods. A data set is described as C if its boundary has an overall convex shape. Correspondingly, if the boundary of the data set as a whole exhibits a distinctly NC shape, it is described as NC.

A graphical representation of the C and NC data sets is depicted in Fig. 3. In both sub-figures, the blue crosses represent the normal observations, while the red dots represent the abnormal observations. Apparently, the boundary of the normal observations in Fig. 3a is C, while the boundary of the normal observations in Fig. 3b is NC.

In the first simulation study, the normal data sets are generated from a bivariate Normal density distribution $\mathcal{N}_1(\mu_1, \Sigma_1)$ with the following parameters: $\mu_1 = (0, 5)$



(a) C Data Set (b) NC Data Set

Fig. 3 Illustration of the simulated data sets

and $\Sigma_1 = \begin{pmatrix} 10 & -5 \\ -5 & 20 \end{pmatrix}$. The abnormal data sets are also generated from a bivariate Normal density distribution $\mathcal{N}_0(\mu_0, \Sigma_0)$, but with different parameters: $\mu_0 = (10, 10)$ and $\Sigma_0 = \begin{pmatrix} 2 & -1 \\ -1 & 3 \end{pmatrix}$. Each simulation generates 700 normal observations for training the classifier. Then, 100 normal observations and 100 abnormal observations are generated to form a test sample. This simulation with a C data set is repeated 100 times, and the average performance is reported for different performance measures.

While implementing the NPFC methods with a soft frontier, different values of c_Λ are used for the C and NC cases. Specifically, we use $c_{\Lambda^{\mathrm{C}}} = -0.1$ and $c_{\Lambda^{\mathrm{NC}}} = -0.2$. Note that we do not intend to suggest that these values of c_Λ are the optimal choices. A careful choice of c_Λ is important for identifying noisy and less important training observations as accurately as possible (see Appendix for a discussion on different choices of c_Λ).

The performance results are summarized in Table 1. The first column displays the six performance measures used in this contribution. Columns 2–7 correspond to the performance results under the SVDD, OC-SVM, C-NPFC method with a hard frontier, C-NPFC method with a soft frontier, NC-NPFC method with a hard frontier and NC-NPFC method with a soft frontier, respectively. Horizontally, each row reports the average classification performance of the various methods under the corresponding measures (10)–(15). In each row, the best result among the six methods is highlighted in bold.

Several observations can be made from the performance results reported in Table 1.

First, benchmarking the classification performance of the two existing methods, the performance results of the proposed four NPFC methods can be found to be rather competitive. Specifically, in this simulation, the proposed

C-NPFC methods have better performance results than the two existing methods in almost all measures. The only exception is that the recall value of the OC-SVM method is slightly higher by 0.3%.

Second, the adoption of a soft nonparametric frontier improves the overall performance for both the C- and NC-NPFC methods. Specifically, in this simulation, the improvement in overall performance brought by the soft frontiers to the NC-NPFC method occurs mainly in the third decimal place, while the improvement to the C-NPFC method is more pronounced and occurs in the second decimal place.

Third, the comparison between the C- and NC-NPFC methods demonstrates the relative strengths of each. Compared to the C-NPFC methods, the NC-NPFC methods provide a more conservative frontier enveloping the normal training observations. As a result, the NC-NPFC methods always have a higher recall value, indicating their better performance in correctly detecting abnormal observations. The C-NPFC methods, on the other hand, always have a higher specificity value, implying their good performance in correctly predicting normal observations.

Fourth, in terms of overall performance, it is expected that the C-NPFC methods should perform better when dealing with the C data set. However, the results show that the hard NC-NPFC method outperforms the hard C-NPFC method by about 0.52%. The above expectation is only validated when a soft frontier is used, i.e., the overall performance of the C-NPFC method is then about 0.39% better than that of the NC-NPFC method.

In the following, another simulation setting is introduced for generating NC data sets. In this setting, the normal data sets are characterized by two bivariate Normal density distributions $\mathcal{N}_1(\mu_1, \Sigma_1)$ and $\mathcal{N}_2(\mu_2, \Sigma_2)$, with the following parameters: $\mu_1 = (0, 5),\ \Sigma_1 = \begin{pmatrix} 10 & -5 \\ -5 & 20 \end{pmatrix},\ \mu_2 = (12, -2),\ \Sigma_2 = \begin{pmatrix} 20 & -1 \\ -1 & 8 \end{pmatrix}$. The abnormal data sets are generated from the same bivariate Normal density distribution $\mathcal{N}_0(\mu_0, \Sigma_0)$ as in the first study. In each simulation, the training sample consists of two parts of data: one is a sample of 350 normal data generated from $\mathcal{N}_1(\mu_1, \Sigma_1)$, the other is a sample of 350 normal data generated from $\mathcal{N}_2(\mu_2, \Sigma_2)$. The test sample consists of three parts of data: the first is a sample of 50 normal data generated from $\mathcal{N}_1(\mu_1, \Sigma_1)$, the second is a sample of 50 normal data generated from $\mathcal{N}_2(\mu_2, \Sigma_2)$, and the third is a sample of 100 normal data generated from $\mathcal{N}_0(\mu_0, \Sigma_0)$. This simulation is also repeated 100 times, and the average performance is reported for different measures. While implementing the NPFC methods with a soft frontier, we use $c_{\Lambda^{\mathrm{C}}} = -0.1$ and $c_{\Lambda^{\mathrm{NC}}} = -0.2$.

The performance results are summarized in Table 2, which is structured in the same way as that of Table 1. Compared to the performance results under the C data set, there are still some observations worth mentioning in Table 2.

First, the performance of the NC-NPFC methods is still quite competitive in comparison with the two existing methods, and even outperform them on all performance measures in this simulation. However, this is not the case for the C-NPFC methods. The overall performance of the C-NPFC methods is still

Table 1 Performance results under the C data sets

	Existing methods		C-NPFC		NC-NPFC	
	SVDD	OC-SVM	Hard	Soft	Hard	Soft
Accuracy	0.8567	0.9731	0.9835	**0.9929**	0.9882	0.9890
Precision	0.8863	0.9501	**0.9948**	0.9899	0.9865	0.9818
Recall	0.9073	**0.9990**	0.9720	0.9960	0.9901	0.9966
Specificity	0.8061	0.9471	**0.9949**	0.9897	0.9863	0.9813
F_1 Score	0.8724	0.9738	0.9824	**0.9929**	0.9880	0.9890
G-mean	0.8842	0.9742	0.9829	**0.9929**	0.9882	0.9891

competitive with that of the SVDD method, but much inferior to that of the OC-SVM method.

Second, the improvement in overall performance brought by the soft frontiers still exists and is much more pronounced in terms of the increase. Specifically, in this simulation, the NC-NPFC approach shows an improvement of about 0.57–0.67%. For the C-NPFC methods, this improvement can be up to 18.13%.

Third, the relative strengths of the C- and NC-NPFC methods can still be observed in Table 2. Specifically, the C-NPFC methods still have a higher specificity value, and the NC-NPFC methods have a higher recall value.

Fourth, the C-NPFC methods perform rather poor when dealing with the NC data set. Specifically, the C-NPFC methods identify most of the normal observations but over-optimistically predict some of the abnormal observations as normal. As a result, it fails miserably in detecting abnormal observations, with observed recall values around or well below 50%. While dealing with the same NC data sets, the NC-NPFC methods have a balanced and good performance in predicting both normal and abnormal observations. Thus, their overall performance is the best among all the listed methods.

To sum up, the simulation results show that the proposed NPFC methods are competitive in solving the anomaly detection problem, using the two existing methods as benchmarks. For both the C- and NC-NPFC methods, the adoption of a soft frontier can improve the overall performance. The C-NPFC method with a soft frontier may have the best overall performance if the boundary shape of the data set is C as in the first simulation study, but the performance of the NC-NPFC method is also competitive. However, if the boundary shape of the data set is NC as in the second simulation study, then the overall performance of the NC-NPFC methods will be much better than that of the C-NPFC methods.

3.3 Experiments on a real-life data set

The proposed NPFC methods are applied to a real-life data set, see Cox et al. (1982).[1] This data set arose in a study that aims at identifying carriers of a rare

[1] The data set is collected from the Statlib data archive at: `http://lib.stat.cmu.edu/datasets/`

genetic disorder. Because the disease is rare, the number of carriers whose data are available is relatively smaller comparing to the number of normal samples. Specifically, this biomedical data set contains 194 observations after excluding 15 observations which have missing values. Among them, 127 observations are normal samples and 67 observations are disease carriers which are deemed as anomalies. Each observation is characterized by five characteristic variables, namely, age and 4 blood measurements.

Since the data set is small, a repeated k-fold cross-validation (CV) is used to evaluate the performance of various classifiers (Kim 2009). Moreover, we stick to the general situation of anomaly detection where only one group of data is available. That is, only the normal observations are used for the training process, while all the abnormal observations are excluded from the training process and are waited to be classified in the test process. The specific process of dividing the data into training and test data is as follows. The normal data set is randomly partitioned into k disjoint folds of approximately the same size. A fold of normal observations is set aside for testing and the remaining folds are used as training data to train a classifier. The trained classifier is then tested against the previously set aside fold of normal observations as well as the abnormal observations. The above process is repeated 100 times, and the average performance is reported for different performance measures. With respect to the choice of k, there is no formal rule for deciding its optimal value. Thus, both the commonly used k values of 5 and 10 are adopted in this contribution, as these two values are believed to give test error estimates that suffer neither from extremely high bias nor very high variance (Marcot and Hanea 2021).

In implementing the NPFC methods, the monotonicity relations of the characteristic variables are derived from the expert knowledge released in Cox et al. (1982). Since all normal observations are younger than 40, the expert knowledge should be descriptive of carriers. Specifically, the expert knowledge suggests that high measurements are more likely to correspond to carriers. In addition, young carriers tend to have even higher measurements compared to old carriers. That is, the probability of being normal decreases with the increase in the measurements, and therefore, the blood measures should be input-like characteristic variables. Given the same measurements, the probability of being normal increases with age, so age should be an output-like characteristic variable.

As for the assumption on convexity, it is decided by the potential substitution relation between the characteristic variables. However, in this biomedical data set, there is no prior information on this substitution relation. Thus, both C- and NC-NPFC methods are adopted in this analysis. For the cut-off super-efficiencies, we use $c_{\Lambda^{\mathrm{C}}} = -0.3$ and $c_{\Lambda^{\mathrm{NC}}} = -0.4$.

The classification performance results under various k-fold CVs are presented in Table 3. Table 3 is structured in a similar way as that of Table 1. Horizontally, each block reports the performance results under a specific k-fold CV. Within each block, the results of the performance measures (10)-(15) are arranged in the same way as that in Table 1. In each row, the best performance is highlighted in bold. Note that the test data set is unbalanced for both 5-fold and 10-fold CVs. Specifically, under 5-fold CV, the number of normal and abnormal test observations are 25 and 67, respectively, while under 10-fold CV, they are 13 and 67, respectively. For the

Table 2 Performance results under the NC data sets

	Existing methods		C-NPFC		NC-NPFC	
	SVDD	OC-SVM	Hard	Soft	Hard	Soft
Accuracy	0.5701	0.9670	0.6625	0.7540	0.9767	**0.9824**
Precision	0.6050	0.9492	0.9731	0.9807	**0.9827**	0.9735
Recall	0.3638	0.9873	0.3304	0.5175	0.9707	**0.9922**
Specificity	0.7763	0.9466	0.9946	0.9905	**0.9826**	0.9726
F_1 Score	0.3541	0.9677	0.4657	0.6470	0.9758	**0.9825**
G-mean	0.4028	0.9680	0.5425	0.6919	0.9763	**0.9827**

unbalanced data set, using accuracy as an overall performance measure is perceived as virtually useless (Akbani et al. 2004). Thus, the results on accuracy are displayed but are not analyzed.

The experiments under the 5-fold and 10-fold CVs yield the same observations. Therefore, in the following discussion, we do not make any special distinction between them.

First, a comparison is made between the four NPFC methods and the two existing methods. It is observed that the precision value of the two existing methods is quite high, but their recall value is relatively lower. This implies that the two existing methods are quite conservative in identifying observations as abnormal, consequently leading to a relatively small proportion of abnormal observations being identified. Comparatively, the recall value of the NPFC methods is always higher, although the precision value may be lower. This suggests that the NPFC methods perform well in identifying abnormal observations, but entails some risk of misclassifying normal observations as abnormal. Each of the listed methods has its own relative advantages, either in correctly predicting normal or in correctly predicting abnormal observations. However, in this experiment, the overall performance of the four NPFC methods is better than the two existing methods, with a range of 2.11–28.88%.

Second, a comparison in made between the NPFC methods with a hard and the NPFC methods with a soft frontier. In this experiment, the NPFC methods with a soft frontier slightly outperform the ones with a hard frontier in terms of the overall performance, with a range of 0.03–1.62%.

Third, a comparison in made between the C- and NC-NPFC methods to reveal their relative advantages. The C-NPFC methods have better performance in correctly predicting normal observations (reflected by a higher specificity value), while the NC-NPFC methods have better performance in correctly identifying abnormal observations (reflected by a higher recall value).

Fourth, in terms of the overall performance, the NC-NPFC methods outperform the C-NPFC methods with a range of 2.31–6.13%.

The performance results in Table 3 are visualized in Fig. 4. Figures 4a, 4b report the performance results under the 5-fold and 10-fold CVs, respectively. In every sub-figure, the green dashed line marked with upward-pointing triangles reports the recall values; the black dotted line marked with downward-pointing triangles reports

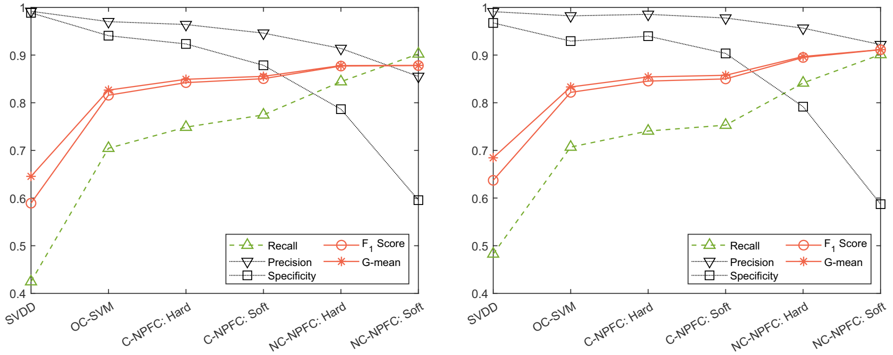
Table 3 Performance Results under Various k-fold CVs

		Existing Methods		C-NPFC		NC-NPFC	
		SVDD	OC-SVM	Hard	Soft	Hard	Soft
k=5	Accuracy	0.5798	0.7696	0.7968	0.8032	**0.8282**	0.8179
	Precision	**0.9919**	0.9702	0.9641	0.9461	0.9139	0.8553
	Recall	0.4246	0.7045	0.7487	0.7745	0.8442	**0.9024**
	Specificity	**0.9886**	0.9407	0.9234	0.8786	0.7865	0.5956
	F_1 Score	0.5892	0.8158	0.8422	0.8505	0.8770	**0.8780**
	G-mean	0.6455	0.8265	0.8492	0.8554	0.8780	**0.8784**
k=10	Accuracy	0.5592	0.7417	0.7719	0.7766	0.8336	**0.8518**
	Precision	**0.9911**	0.9823	0.9855	0.9778	0.9565	0.9220
	Recall	0.4827	0.7070	0.7407	0.7530	0.8415	**0.9012**
	Specificity	**0.9674**	0.9294	0.9397	0.9035	0.7917	0.5872
	F_1 Score	0.6372	0.8218	0.8454	0.8500	0.8950	**0.9113**
	G-mean	0.6845	0.8331	0.8542	0.8576	0.8970	**0.9114**

the precision values; the black dotted line marked with squares reports the specificity values; the red solid line marked with circles reports the F_1 scores; the red solid line marked with asterisks reports the G-mean values. Six classification methods listed in Table 3 are displayed accordingly on the horizontal axis.

The four observations from Table 3 can still be drawn from Fig. 4. Moreover, it is easier to analyze the general trend with Fig. 4. From the left to the right, both Figs. 4a, b show a progressively upward trend in F_1 score, G-mean and recall, while there is a continuous downward trend in precision and specificity. That is, the performance of correctly detecting abnormal observations is improving, while the performance of correctly predicting normal observations is deteriorating. The positive effect of the former is stronger than the negative effect of the latter, so the overall performance represented by F_1 score and G-mean shows some improvement. In addition, the smaller the difference between the recall value and the specificity value, the more balanced a model's performance in predicting normal and abnormal observations. In this sense, the hard NC-NPFC method is the one with the most balanced performance because it has the smallest distance between the recall and specificity lines.

In general, the proposed NPFC methods show a competitive classification performance, and even outperform the listed OC-SVM and SVDD methods in terms of the overall performance. Moreover, they show unique advantages in correctly detecting abnormal samples, especially the NC-NPFC methods. All these support that the proposed NPFC methods, especially the NC-NPFC methods, can be well applied to the anomaly detection problem.



(a) k=5 (b) k=10

Fig. 4 Performance results for various k-fold CVs

4 Conclusions

Although anomaly detection is a popular research problem, no consensus has been reached on the best classification method. This contribution proposes for the first time that the NPFC method can be used for anomaly detection. In the NPFC method, the nonparametric frontier is generated from the group of normal training data and consists of only a few frontier training observations. Moreover, the shape of the nonparametric frontier is determined by the training observations and the imposition of some weak maintained axioms, rather than being predetermined as a hyperplane or a hypersphere. A test data point only has to be compared with this nonparametric frontier for deciding its membership.

In addition to bringing the NPFC method to anomaly detection, this contribution also makes three innovations at the methodological level. First, the convexity assumption is explained as a substitution relation between the characteristic variables; thus, it can be reasonably relaxed. Having the convexity assumption relaxed, a NC-NPFC method is constructed for anomaly detection and it ends up with a more conservative nonparametric frontier describing the observed normal group. Second, the other assumption of free disposability is explained to reflect the monotonic relation between the characteristic variables and the membership. Therefore, characteristic variables with both the monotonically increasing and the monotonically decreasing relations can be included in the model without data transformation. Third, by allowing certain training observations to be misclassified, the hard nonparametric frontier is extended to a soft nonparametric frontier. Compared to the hard frontier, the soft nonparametric frontier is more conservative in describing the observed normal group. We leave the determination of the value c_Λ as an avenue of future research. To sum up, assigning reasonable interpretations to the assumptions justifies the NPFC methods and also contributes to the construction of a more generalized NPFC method.

The simulation studies and the experiment analysis on a biomedical data set both reveal that the proposed NPFC methods have competitive overall performance and have consistent advantages in detecting abnormal samples. This advantage in correctly detecting abnormal samples is consistent with the goal of anomaly detection. Moreover, the overall performance can be further improved by using a soft nonparametric frontier. This improvement exists for both the C- and NC-NPFC methods. Last but not least, if there is no prior information on the substitution relation among characteristic variables, then the NC-NPFC methods should be favored so as to correctly detect more abnormal samples.

We end with developing some perspectives for potential future research. First, it is worthwhile to further connect axioms inherited from production theory (e.g., the axiom of returns-to-scale) with explainable knowledge in anomaly detection such that these axioms can be reasonably exploited or even relaxed. Second, although the monotonicity relation contributes to explainable classification results, it is not always known to the decision maker. For situations where the monotonicity relation is not prior known, one may wonder whether it is possible to weaken the currently maintained axiom of disposability. A recent attempt in the production theory to do so is developed in Briec et al. (2016) and empirically implemented in Briec et al. (2018). Third, one may equally wonder to which extent the same ideas can be transposed in the limited literature employing double separating frontiers in a classification setting, e.g, Sueyoshi (2006); Chang and Kuo (2008); Wu et al. (2011). Finally, while we have in this contribution compared the NPFC methods to the OC-SVM and SVDD methods, it could be interesting to compare the best of the NPFC methods to some of the best performing state of the art classification methods in anomaly detection to check their relative classification and prediction accuracies.

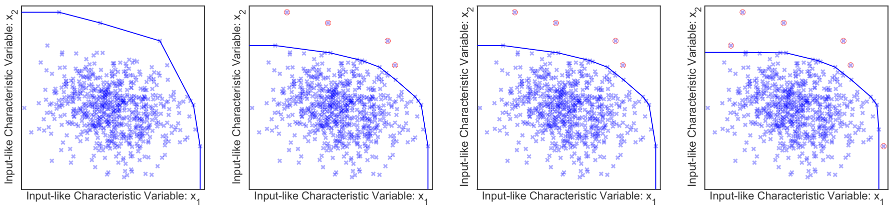
Appendix: Discussions on the choices of cut-off super-efficiency

In this contribution, a negative cut-off super-efficiency denoted by c_Λ is introduced to decide the noisy and less important training observations. A larger value of c_Λ means that more training observations will be identified as noisy and of low importance and thus, will be excluded from the construction of a soft nonparametric frontier.

For both the C and NC cases in Sect. 3.2, the simulation is executed for 100 times. Here, an example for the C case is extracted to show different C and NC soft nonparametric frontiers constructed from different choices of c_Λ.

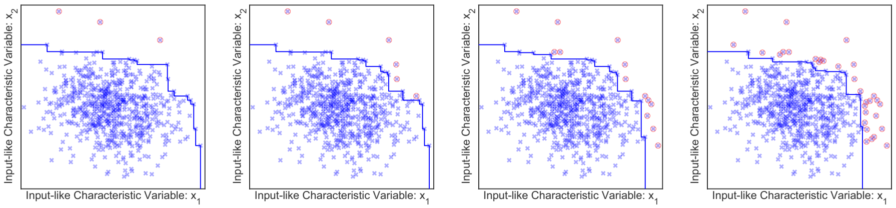
The resulted frontiers under different choices of c_Λ are displayed in Figs. 5 and 6. In every sub-figure, the normal training observations are represented by blue crosses. The training observations that are identified as noisy and of low importance are further marked by red circles. These training observations are excluded while constructing the corresponding soft nonparametric frontier. The soft nonparametric frontier is represented by the blue solid lines.

Similar observations can be derived from Figs. 5 and 6. With the increase in c_Λ, more training observations are identified as noisy and of low importance.



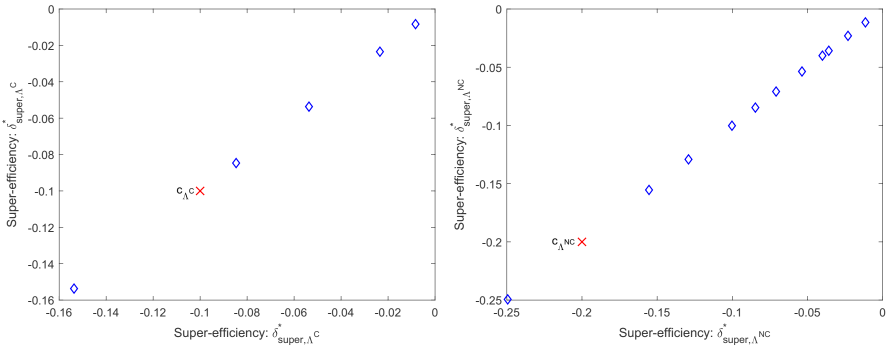
(a) $c_{\Lambda^{\mathrm{C}}}$=-0.2 (b) $c_{\Lambda^{\mathrm{C}}}$=-0.1 (c) $c_{\Lambda^{\mathrm{C}}}$=-0.075 (d) $c_{\Lambda^{\mathrm{C}}}$=-0.05

Fig. 5 C soft nonparametric frontiers with different choices of c_Λ



(a) $c_{\Lambda^{\mathrm{NC}}}$=-0.2 (b) $c_{\Lambda^{\mathrm{NC}}}$=-0.1 (c) $c_{\Lambda^{\mathrm{NC}}}$=-0.075 (d) $c_{\Lambda^{\mathrm{NC}}}$=-0.05

Fig. 6 NC soft nonparametric frontiers with different choices of c_Λ



(a) C case (b) NC case

Fig. 7 Illustrative diagram of deciding a cut-off super-efficiency

Accordingly, the soft nonparametric frontier becomes more conservative. In comparison with the C soft nonparametric frontier, the NC soft nonparametric frontier is more conservative, since more training observations are excluded.

A proper choice of c_Λ is important for identifying as accurately as possible noisy and less important training observations. In the following, the suggested way of deciding the value of c_Λ is explained with the above simulation example.

By solving model (7) under the C case for 700 normal training observations, 5 of them are identified as frontier observations. Their values of $\delta^*_{\mathrm{super},\Lambda^{\mathrm{C}}}$ ordered from largest to smallest are represented by blue diamonds in Fig. 7a. It is observed that 4 out of 5 frontier observations have a $\delta^*_{\mathrm{super},\Lambda^{\mathrm{C}}}$ larger than −0.1. Only one frontier observations has $\delta^*_{\mathrm{super},\Lambda^{\mathrm{C}}} = -0.1537$. Thus, $c_{\Lambda^{\mathrm{C}}} = -0.1$ is chosen.

Similarly, by solving model (7) under the NC case for the same normal training observations, 11 of them are identified as frontier observations. Their values of $\delta^*_{\mathrm{super},\Lambda^{\mathrm{NC}}}$ ordered from largest to smallest are as shown in Fig. 7b. It is observed that 10 out of 11 frontier observations have the value of $\delta^*_{\mathrm{super},\Lambda^{\mathrm{NC}}}$ larger than −0.2. Only one frontier observations has $\delta^*_{\mathrm{super},\Lambda^{\mathrm{NC}}} = -0.2494$. Thus, $c_{\Lambda^{\mathrm{NC}}} = -0.2$ is chosen.

One suggested way of deciding the value of c_{Λ} is illustrated. Note that we do not intend to suggest that this is the optimal way. It is worthwhile for future researches to explore the other methods of deciding a proper c_{Λ}.

Acknowledgements We appreciate the constructive comments of the Editor-in-Chief, G. Voigt, and two anonymous referees, which helped to improve this contribution. The usual disclaimer applies. Q. Jin appreciates financial support of the National Natural Science Foundation of China (No. 72201127).

References

Ahmed M, Mahmood AN, Islam MR (2016) A survey of anomaly detection techniques in financial domain. Futur Gener Comput Syst 55:278–288

Akbani R, Kwek S, Japkowicz N (2004) Applying support vector machines to imbalanced datasets. In: Boulicaut J, Esposito F, Giannotti F, Pedreschi D (eds) Machine learning: ECML 2004. Springer, Berlin, pp 39–50

Al-Hashedi KG, Magalingam P (2021) Financial fraud detection applying data mining techniques: a comprehensive review from 2009 to 2019. Comput Sci Rev 40:100402

Alam S, Sonbhadra SK, Agarwal S, Nagabhushan P (2020) One-class support vector classifiers: a survey. Knowl-Based Syst 196:105754

Andersen P, Petersen NC (1993) A procedure for ranking efficient units in data envelopment analysis. Manage Sci 39(10):1261–1264

Aparicio J, Esteve M, Rodriguez-Sala JJ, Zofio JL (2021) The estimation of productive efficiency through machine learning techniques: efficiency analysis trees. In: Zhu J, Charles V (eds) Data-enabled analytics: DEA for big data. Springer, Cham, pp 51–92

Banker RD, Charnes A, Cooper WW (1984) Some models for estimating technical and scale inefficiencies in data envelopment analysis. Manage Sci 30(9):1078–1092

Bhattacharyya S, Jha S, Tharakunnel K, Westland JC (2011) Data mining for credit card fraud: a comparative study. Decis Support Syst 50(3):602–613

Briec W (1997) A graph-type extension of Farrell technical efficiency measure. J Prod Anal 8(1):95–110

Briec W, Kerstens K, Van de Woestyne I (2016) Congestion in production correspondences. J Econ 119(1):65–90

Briec W, Kerstens K, Van de Woestyne I (2018) Hypercongestion in production correspondences: an empirical exploration. Appl Econ 50(27):2938–2956

Chambers R, Chung Y, Färe R (1998) Profit, directional distance functions, and nerlovian efficiency. J Optim Theory Appl 98(2):351–364

Chang D, Kuo Y (2008) An Approach for the two-group discriminant analysis: an application of DEA. Math Comput Model 47(9–10):970–981

Cherchye L, Kuosmanen T, Post T (2001) FDH directional distance functions with an application to European commercial banks. J Prod Anal 15(3):201–215

Clifton L, Clifton DA, Zhang Y, Watkinson P, Tarassenko L, Yin H (2014) Probabilistic novelty detection with support vector machines. IEEE Trans Reliab 63(2):455–467

Cox LH, Johnson MM, Kafadar K (1982) Exposition of statistical graphics technology. In: Proceedings of the statistical computation section, American Statistical Association, Washington, D.C, pp 55–56

da Silva DB, Schmidt D, da Costa CA, da Rosa Righi R, Eskofier B (2021) Deepsigns: a predictive model based on deep learning for the early detection of patient health deterioration. Expert Syst Appl 165:113905

Deprins D, Simar L, Tulkens H (1984) Measuring labor efficiency in post offices. In: Marchand M, Pestieau P, Tulkens H (eds) The performance of public enterprises: concepts and measurements. North Holland, Amsterdam, pp 243–268

El-Yaniv R, Nisenson M (2006) Optimal single-class classification strategies. Adv Neural Inform Process Syst 19

Emrouznejad A, Banker RD, Neralic L (2019) Advances in data envelopment analysis: celebrating the 40th anniversary of DEA and the 100th anniversary of professor Abraham Charnes, Birthday. Eur J Op Res 278(2):365–367

Emrouznejad A, Yang G-L (2018) A survey and analysis of the first 40 years of scholarly literature in DEA: 1978–2016. Socioecon Plann Sci 61:4–8

Esteve M, Aparicio J, Rodriguez-Sala JJ, Zhu J (2023) Random forests and the measurement of super-efficiency in the context of free disposal hull. Eur J Oper Res 304(2):729–744

He H, Garcia EA (2009) Learning from imbalanced data. IEEE Trans Knowl Data Eng 21(9):1263–1284

Jiang T, Li Y, Xie W, Du Q (2020) Discriminative reconstruction constrained generative adversarial network for hyperspectral anomaly detection. IEEE Trans Geosci Remote Sens 58(7):4666–4679

Kaffash S, Azizi R, Huang Y, Zhu J (2020) A survey of data envelopment analysis applications in the insurance industry 1993–2018. Eur J Oper Res 284(3):801–813

Kerstens K, Sadeghi J, Toloo M, Van de Woestyne I (2022) Procedures for ranking technical and cost efficient units: with a focus on nonconvexity. Eur J Oper Res 300(1):269–281

Kerstens K, Van de Woestyne I (2011) Negative data in DEA: a simple proportional distance function approach. J Op Res Soc 62(7):1413–1419

Khan SS, Madden MG (2014) One-class classification: taxonomy of study and review of techniques. Knowl Eng Rev 29(3):345–374

Kim J-H (2009) Estimating classification error rate: repeated cross-validation, repeated hold-out and bootstrap. Comput Stat Data Anal 53(11):3735–3745

Leon CF, Palacios F (2009) Evaluation of rejected cases in an acceptance system with data envelopment analysis and goal programming. J Op Res Soc 60(10):1411–1420

Lovell K, Pastor JT (1999) Radial DEA models without inputs or without outputs. Eur J Oper Res 118(1):46–51

Marcot BG, Hanea AM (2021) What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis? Comput Stat 36(3):2009–2031

Markou M, Singh S (2003) Novelty detection: A review-part 1: statistical approaches. Signal Process 83(12):2481–2497

Markou M, Singh S (2003) Novelty detection: a review-part 2: neural network based approaches. Signal Process 83(12):2499–2521

Moya MM, Koch MW, Hostetler LD (1993) One-class classifier networks for target recognition applications. In: World congress on neural networks, vol 3. Lawrence Erlbaum Associates, Portland, pp 797–801

Noble WS (2006) What is a support vector machine? Nat Biotechnol 24(12):1565–1567

Pang G, Shen C, Cao L, Hengel AVD (2021) Deep learning for anomaly detection: a review. ACM Comput Surv 54(2):1–38

Park C, Huang JZ, Ding Y (2010) A computable plug-in estimator of minimum volume sets for novelty detection. Oper Res 58(5):1469–1480

Pendharkar P, Khosrowpour M, Rodger J (2000) Application of Bayesian network classifiers and data envelopment analysis for mining breast cancer patterns. J. Comput. Inform. Syst. 40(4):127–132

Pendharkar P, Rodger J, Yaverbaum G (1999) Association, statistical, mathematical and neural approaches for mining breast cancer patterns. Expert Syst Appl 17(3):223–232

Pendharkar PC (2002) A potential use of data envelopment analysis for the inverse classification problem. Omega 30(3):243–248

Qiu K (2022) Support Vector Data Description (SVDD Version 2.2), https://github.com/iqiukp/SVDD–MATLAB

Ruff L, Kauffmann JR, Vandermeulen RA, Montavon G, Samek W, Kloft M, Dietterich TG, Müller K-R (2021) A unifying review of deep and shallow anomaly detection. Proc IEEE 109(5):756–795

Schölkopf B, Williamson RC, Smola A, Shawe-Taylor J, Platt J (1999) Support vector method for novelty detection. In: Solla A, Müller K-R, Leen TK (eds) Advances in neural information processing systems, vol 12. MIT Press, Cambridge, Massachusetts, pp 582–588

Seiford L, Zhu J (1998) An acceptance system decision rule with data envelopment analysis. Comput Op Res 25(4):329–332

Sueyoshi T (2006) DEA-discriminant analysis: methodological comparison among eight discriminant analysis approaches. Eur J Oper Res 169(1):247–272

Tax DMJ, Duin RPW (1999) Support vector domain description. Pattern Recogn Lett 20(11–13):1191–1199

Tax DMJ, Juszczak P (2003) Kernel whitening for one-class classification. Int J Pattern Recognit Artif Intell 17(3):333–347

Troutt M, Rai A, Zhang A (1996) The potential use of DEA for credit applicant acceptance systems. Comput Op Res 23(4):405–408

Valero-Carreras D, Aparicio J, Guerrero NM (2021) Support vector frontiers: a new approach for estimating production functions through support vector machines. Omega 104:102490

Wu J, An Q, Liang L (2011) A modified super-efficiency DEA approach for solving multi-groups classification problems. Int J Comput Intell Syst 4(4):606–618

Yan H, Wei Q (2011) Data envelopment analysis classification machine. Inf Sci 181(22):5029–5041

Zhu N, Zhu C, Emrouznejad A (2021) A combined machine learning algorithms and DEA method for measuring and predicting the efficiency of chinese manufacturing listed companies. J Manage Sci Eng 6(4):435–448

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.