International Journal of Data Science and Analytics (2025) 20:5171–5196
https://doi.org/10.1007/s41060-025-00784-1

REVIEW

Anomaly detection using data depth: multivariate case

Pavlo Mozharovskyi[1] · **Romain Valla**[1]

Received: 30 July 2024 / Accepted: 8 April 2025 / Published online: 28 May 2025

Abstract
Anomaly detection is a branch of data analysis and machine learning which aims at identifying observations that exhibit abnormal behavior. Be it measurement errors, disease development, severe weather, production quality default(s) (items) or failed equipment, financial frauds or crisis events, their on-time identification, isolation, and explanation constitute an important task in almost any branch of science and industry. By providing a robust ordering, data depth—statistical function that measures belongingness of any point of the space to a data set—becomes a particularly useful tool for detection of anomalies. Already known for its theoretical properties, data depth has undergone substantial computational developments in the last decade and particularly recent years, which has made it applicable for contemporary-sized problems of data analysis and machine learning. In this article, data depth is studied as an efficient anomaly detection tool, assigning abnormality labels to observations with lower depth values, in a multivariate setting. Practical questions of necessity and reasonability of invariances and shape of the depth function, its robustness and computational complexity, choice of the threshold are discussed. Illustrations include use cases that underline advantageous behavior of data depth in various settings.

Keywords Data depth · Anomaly detection · Robustness · Affine invariance · Computational statistics · Projection depth · Halfspace depth · Visualization · Data analysis

1 Motivation

Being applicable in a large variety of domains, anomaly detection increasingly gains popularity among researchers and practitioners. Having been in use since decades, it constitutes a contemporary domain of rapid development to meet growing demand in various areas such as industry, economy, social sciences, etc. With large amounts of data recorded in modern applications and constantly present probability of abnormal events, these cannot be identified by operator's hand anymore: Automatic procedures are necessary.

It is not the goal of the current article to provide a complete overview of anomaly detection methods, the reader is referred to [8]; see also following Sects. 1.1 and 1.2 for intuition. Here, a narrower question is in scope: Why and how to employ *data depth* for anomaly detection?

✉ Romain Valla
romain.valla@telecom-paris.fr

Pavlo Mozharovskyi
pavlo.mozharovskyi@telecom-paris.fr

[1] LTCI, Telecom Paris, Institut Polytechnique de Paris, Palaiseau, France

1.1 Difference from outlier detection

With two terms "outlier" and "anomaly" being used by two communities with small overlap, a discussion on their similarity is important.

From statistical point of view, both outlier and anomaly detection focus on identifying atypical observations. Nevertheless, there is a substantial difference in application of methods from these the two groups. First of all, while the term "outlier detection" is traditionally used by statisticians, "anomaly detection" has been adopted by the machine learning community. As a consequence, (more theoretically oriented) statisticians "did not need" and often were unaware of (some of the) anomaly detection methods developed by the machine learning community, while—when searching for practical solutions in applications—machine learners did not find outlier detection methods sufficiently flexible (w.r.t. the data space and shape of the distribution) and scalable (with number of observations and variables). Furthermore, rigorous statistical analysis and inference tools, being often in the center of attention for statisticians, often do not exist for anomaly detection methods, with latter taking frequently form of heuristics.

Indeed, perhaps in the best way the difference between "outlier" and "anomaly" can be described in application. Given a data set at hand, the task of identification of outliers consists in searching for observations not resembling the majority of the data set. "Anomaly detection" approach is more operational and follows rather the philosophy of machine learning. That is, given a training data set, which itself can contain anomalies or not, the task is to construct a rule (training phase) which can assign (on the detection phase) each observation of the space (including the observations of the training set) either to the category of anomalies or normal observations.

This work-flow imposes certain requirements on anomaly detection methodology, e.g., regarding the data set used to learn the anomaly detection rule. Should the rule simply save the entire training data set (this would be the case when directly applying data depth), only part of it, or not at all; should the rule be updated, and how often? Continuing the example with data depth, on the learning phase (again in direct application), training data set should be simply saved in the memory and no computations are to be done. When checking abnormality of a new observation, its data depth shall be computed w.r.t. the (saved) training data set based on which the decision about the observation's abnormality shall be made. To keep the rule scalable (and fitting in limited machine memory), only its subset can be stored instead. In the case of Mahalanobis depth, only parameters (center vector and scatter matrix) need to be saved, and no data at all.

It is important to keep attention on this operational aspect when underlining suitability of data depth for anomaly detection in industrial context in the following Sect. 1.2 and focus on this aspect later in Sect. 5.

1.2 Industrial context

Regard the following example simulating industrial data. Think of a (potential) production line that manufactures certain items. On several stages of the production process, measurements are taken on each of the items to ensure the quality of the produced pieces. These measurements can be numerous if the line is well automatized, or rare if this is not the case. If—for each item—these measurements can be assembled in a vector (of length d), then the item can be represented as a multivariate observation $\boldsymbol{x}$ in an Euclidean space ($\boldsymbol{x} \in \mathbb{R}^d$), and the entire manufacturing process as a data set in the same space ($\{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \subset \mathbb{R}^d$).

Regard Fig. 1, left. For visualization purposes, let us restrict to two measurements, whence each produced item is represented by an observation with two variables (=measurements). To construct an anomaly detection rule, a subset of production data is taken as a training set, which can itself contain anomalies or not; this corresponds to the 500 black pixels, let us denote them $\boldsymbol{X}_{tr} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_{500}\} \subset \mathbb{R}^2$. 8 new observations are now to be labeled either as normal observation or as anomalies, namely four green dots (corresponding to normal items), three red pluses and cross (anomalies). While in this bivariate visual case with $d = 2$ it is trivially resolved by a simple visual inspection, the task becomes much more complicated once d increases.

The simplest, though still frequently applied approach, is to define validation band for each measurement, i.e., upper and lower bound for each variable: This rule is depicted by black dashed lines parallel to variables' axes and—if well calibrated—allows to identify three out of four anomalies (red pluses) and is computationally extremely fast (computation, as well as following item's production, can even stop after crossing any of the bounds):

$$g_{\text{box}}(\boldsymbol{x}|\boldsymbol{X}_{tr}) = \begin{cases} \text{anomaly(=1)}, & \text{if } \boldsymbol{x} \notin \bigcap_{j=1,...,d} (\underline{H}_{j,l_j} \cap \overline{H}_{j,h_j}), \\ \text{normal(=0)}, & \text{otherwise.} \end{cases} \tag{1}$$

with $l_1, h_1, ..., l_d, h_d$ being lower and upper validation bounds (calibrated using $\boldsymbol{X}_{tr}$) for each axis and $\overline{H}_{j,a} = \{\boldsymbol{y} \in \mathbb{R}^d \mid \boldsymbol{y}^\top \boldsymbol{e}_j \le a\}$, $\underline{H}_{j,b} = \{\boldsymbol{y} \in \mathbb{R}^d \mid \boldsymbol{y}^\top \boldsymbol{e}_j \ge b\}$ where $\boldsymbol{e}_j$ is the orthant of the jth axis. The fourth anomaly (red cross) remains invisible for rule (1).

Obviously, this fourth anomaly can be identified using rule based on Mahalanobis depth D^{Mah}, defined later by (7)

$$g_{\text{Mah}}(\boldsymbol{x}|\boldsymbol{X}_{tr}) = \begin{cases} \text{anomaly}, & \text{if } D^{\text{Mah}}(\boldsymbol{x}|\boldsymbol{X}_{tr}) < t_{\text{Mah},\boldsymbol{X}_{tr}}, \\ \text{normal}, & \text{otherwise.} \end{cases} \tag{2}$$

where $t_{\text{Mah},\boldsymbol{X}_{tr}}$ is chosen based on $\boldsymbol{X}_{tr}(= 0.075)$ in a way that the Mahalanobis depth contour is largest not to exceed the variable-wise validation bounds. While rule (2) easily identifies all present anomalies, two aspects shall be taken into account: (i) The training data do not contain anomalies itself and (ii) is large (especially when compared to d).

In the beginning of the production process—the phase where diagnostic is particularly important—not many observations are available, but anomalies should still be identified among them; similar situation occurs when produced items are time/resources consuming and are not produced very often. To simulate this situation, regard Fig. 1, right. Here, the training data set contains 25 observations: 19 being generated from Gaussian distribution (gray dots), 4 former normal observations (green dots), and the same four anomalies (red pluses and cross). Rule (2) (with the same threshold $t_{\text{Mah},\boldsymbol{X}_{tr}}$) as before provides misleading ellipse (solid black line) that classifies all anomalies as normal observations. When employing a rule based on projection depth defined later by (12) instead (denoted in blue dashed line), i.e.,

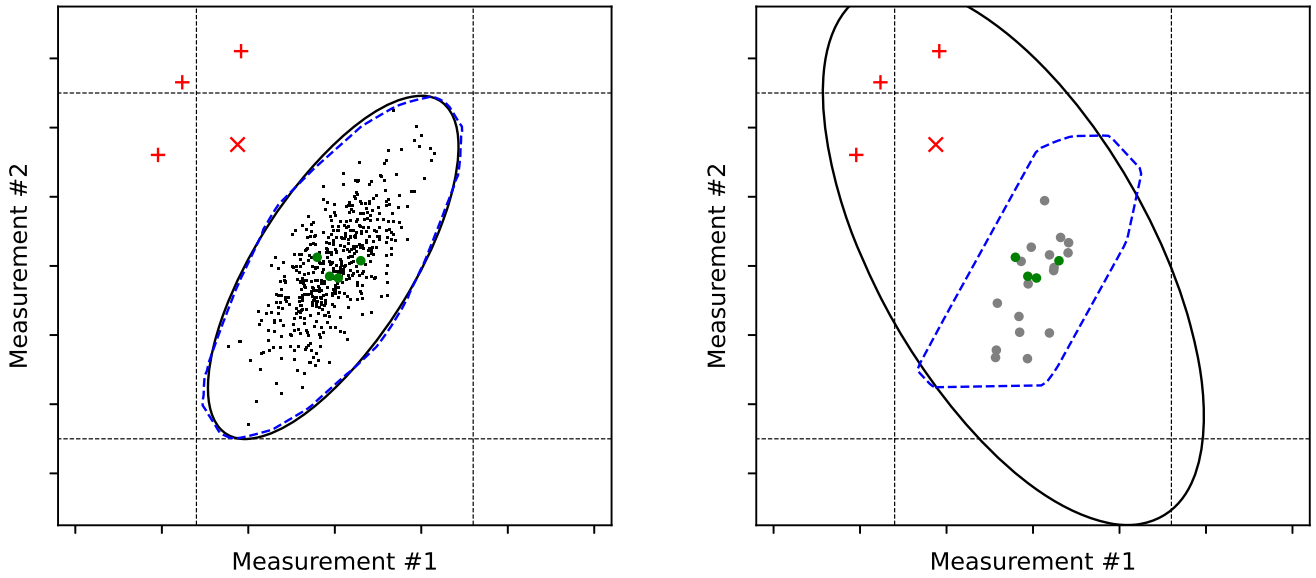


Fig. 1 Four normal observations (green dots) and four anomalies (red pluses and cross); contours of Mahalanobis (black solid) and projection (blue dashed) depths. Left: A sample of 500 bivariate Gaussian observations (black pixels). Right: 17 bivariate Gaussian observations (gray dots)

$$g_{\text{prj}}(\boldsymbol{x}|\boldsymbol{X}_{tr}) = \begin{cases} \text{anomaly}, & \text{if } D^{\text{prj}}(\boldsymbol{x}|\boldsymbol{X}_{tr}) < t_{\text{prj},\boldsymbol{X}_{tr}}, \\ \text{normal}, & \text{otherwise.} \end{cases} \tag{3}$$

all four anomalies are identified. Furthermore, when all 500 observations become available (e.g., when the production line works for longer period of time), rule from (3) (with the same threshold $t_{\text{prj},\boldsymbol{X}_{tr}} (= 0.1575)$) almost coincides with the Mahalanobis depth rule (2); see Fig. 1 (left) again. We refer the reader to Sect. 4.2 for a discussion on the choice of the threshold.

1.3 Outline of conceptual challenges

In the rest of the article, after a brief introduction of data depth (Sects. 2 and 3), challenges connected with its application to anomaly detection are discussed. These can be roughly split in three parts:

- Which depth notion to choose in a case at hand (Sect. 4)?
- Why to use data depth, i.e., why and in which cases is it advantageous over existing methods (Sect. 5)?
- How to deal with computational issues when employing data depth (Sect. 6)?

We gather remarks and a disclaimer in Sect. 8.

While being destined for practitioners, this article is entirely based on *simulated data*. This is mainly due to four reasons: First, in general enterprises are not willing to share data because of its confidentiality, often for competition reasons. Fortunately, this tendency starts to decrease, which can be witnessed by numerous data challenges, because—depending on the industrial sector—(a) data are getting quickly outdated and much more important is to (quickly) find clues how to treat it or/and (b) enterprise does not have enough internal expertise and searches for external ideas, releasing at least part of their data. Second, industrial cases are normally a result of continuous work (or collaboration) being augmented and labeled over time, often based on several data sets and using *a priori* knowledge of domain experts—a complex situation not necessarily presentable as a simple example. Third, the purpose of data illustrations of this article is to pin cases in which employing data depth-based methodology is advantageous, and illustrating it to better degree is more gainful with synthetic data. Fourth, for verification and comparison purposes, a feedback is needed. In any case, there is only very little probability that applicant would encounter exactly the same real data situation (repeated) in practice. (Though all the examples presented in the article are based on simulated data, we shall continue calling them *observations* in what follows.)

2 What is data depth?

Data depth is a statistical function that, given a data set $\boldsymbol{X} \subset \mathbb{R}^d$, assigns to each element of the space (where it is defined) a value (usually) between 0 and 1, which characterizes how deep this element is in the data set:

$$D : \mathbb{R}^d \times \mathbb{R}^{n\times d} \to [0, 1], \ (\boldsymbol{x}, \boldsymbol{X}) \mapsto D(\boldsymbol{x}|\boldsymbol{X}). \tag{4}$$

This element can be an observation that belongs to the data set, or any other arbitrary element of the space, e.g., future

observation. Being a function of data, data depth inherits statistical properties of the data set and thus describes it in one or another manner and can serve many purposes:

- it provides natural data-induced ordering on the space, a property not easily extendable beyond univariate data and which is widely used for statistical inference such as classification [30, 35, 40] or testing [14];
- its maximizer(s) is (are) a generalization of median (i.e., robust center) to higher dimensions;
- depth contours (constituted of space elements possessing the same depth level) describe data with respect to their location, scatter, and shape, a property that gave rise to the notions of bagplot [62] and curve boxplot [36, 50] being generalizations of the univariate boxplot (see Sect. 3.2 below for a discussion on depth contours);
- observations with very low depth values are natural candidates for anomalies—a property in the main focus of this article.

Let us focus again on the multivariate case, i.e., when the depth is defined in the d-variate Euclidean space $\mathbb{R}^d$. Since already in $\mathbb{R}^d$ infinite variety of possible functions fit the definition of data depth from above (including trivial ones, e.g., constant function), requirements (also called postulates) have been put on a depth function to be a proper one. There are two most know sets of such postulates. With the first one formulated by [37] (for simplicial depth) and generalized to further depths by [77], we here cite a later one by [21] [20, see also, forearlierversion], because it does not include statistical component (namely behavior for symmetric distribution) and is thus more practical. A depth from (4) should satisfy following postulates:

- *Affine invariance*: For any $\boldsymbol{b} \in \mathbb{R}^d$ and any non-singular $d \times d$ matrix $\boldsymbol{A}$ it holds:

$$D(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}|\boldsymbol{A}\boldsymbol{X} + \boldsymbol{b}) = D(\boldsymbol{x}|\boldsymbol{X}), \tag{5}$$

  where (in slight abuse of notation) $\boldsymbol{A}\boldsymbol{X} + \boldsymbol{b}$ is a shortcut for pre-multiplication with matrix $\boldsymbol{A}$ and adding vector $\boldsymbol{b}$ to each element of $\boldsymbol{X}$.
  This relatively strong but beneficial (as we shall see in Sect. 4) postulate can be weakened to orthogonal invariance only, i.e., with $\boldsymbol{A}$ being orthogonal matrix.
- *Vanishing at infinity*:

$$\lim_{\|\boldsymbol{x}\| \to \infty} D(\boldsymbol{x}|\boldsymbol{X}) = 0.$$

- *Monotone relative to deepest point*: For any $\boldsymbol{x}^*$ having maximal depth, i.e., such that $D(\boldsymbol{x}^*|\boldsymbol{X}) = \max_{\boldsymbol{x} \in \mathbb{R}^d} D(\boldsymbol{x}|\boldsymbol{X})$, and for any $\gamma \in [0, 1]$ it holds:

$$D(\boldsymbol{x}|\boldsymbol{X}) \leq D(\boldsymbol{x}^* + \gamma(\boldsymbol{x} - \boldsymbol{x}^*)|\boldsymbol{X}). \tag{6}$$

  This property ensures star-similar shape of the upper-level sets of the depth function. If necessary, it can be strengthened to quasi-concavity of the depth, which would yield convex upper-level regions.
- *Upper semicontinuity*: The upper-level sets (called also depth regions) defined as $D_\alpha(\boldsymbol{X}) := \{\boldsymbol{x} \in \mathbb{R}^d : D(\boldsymbol{x}|\boldsymbol{X}) \geq \alpha\}$ are closed, to make the depth function upper-semicontinuous.

Even these postulates are of course not sufficient to restrict a function to a reasonable and even practically useful depth. That is why up to hundred definitions have been attempted, with only a dozen being accepted and used by the community. While below we define six depth functions used in the simulation studies of this article, a longer list of depth notions (neither pretending on completeness) can be provided such as **Convex hull peeling depth** [3, 22], **Majority depth** [66], **Zonoid depth** [33], $\mathbb{L}_p$ **depth** [77], **Spatial depth** [64], **Expected convex hull depth** [7], **Geometrical depth** [15], **Lens depth** [39] generalized in $\boldsymbol{\beta}$**-skeleton depth** [76].

As one shall see from this point on, data depth provides a universal generic methodology for anomaly detection since (almost) any depth notion can be plugged in the rule (3), depending on desired properties and existing (computational and data) limitations. This list of depths is of course not complete, if a complete list can be provided in an article in general with many new notions or modifications of existing ones constantly appearing. Below, we define the five depths employed in this article, while letting the reader to consult provided (and other) references for the rest.

To underline the practical nature of the article, we shall introduce the depths in their empirical context, i.e., as a deterministic function $D(\boldsymbol{x}|\boldsymbol{X})$ computing representativeness of any point $\boldsymbol{x} \in \mathbb{R}^d$ w.r.t. a data set $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$. Indeed, numerically any empirical distribution can be treated with ties not being as exception. Below, we shall also illustrate the techniques how to turn a data depth notion into asymmetric (on the example of projection depth) or affine-invariant (5) (on the example of simplicial volume depth) depth notion.

Mahalanobis depth is defined as a strictly decreasing transform of the (squared) Mahalanobis distance [47] to the mean:

$$D^{\text{Mah}}(\boldsymbol{x}|\boldsymbol{X}) = \frac{1}{1 + (\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})}. \tag{7}$$

Here, $\boldsymbol{\mu}$ stands for the mean vector and $\boldsymbol{\Sigma}^{-1}$ for the inverse of the covariance matrix of the distribution generating $\boldsymbol{X}$.

Taking moment estimates for both quantities, i.e., when $\mathbb{R}^d \ni \boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i$ and (unbiased) $\boldsymbol{\Sigma} = \frac{1}{n-1}\sum_{i=1}^n (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^\top$, as it is the case in Fig. 1, results in a fast (having time complexity $O(nd^2+d^3)$ if $n >> d$) and affine-invariant but not robust estimator that can be perturbed even by a single anomaly in $\boldsymbol{X}$ with sufficiently high amplitude. A more robust is the minimum covariance determinant estimator [43, 61, MCD;] which estimates $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ as average and empirical covariance of $\alpha \in (0.5, 1]$ portion of $\boldsymbol{X}$ that minimizes the determinant of $\boldsymbol{\Sigma}$, and can be approximated by a fast stochastic algorithm of [63]. Generally speaking, any reasonable quantities can be used instead which estimate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ as center and scatter of $\boldsymbol{X}$-generating law.

Such "robustification" may conflict with affine invariance though: Approximately computed MCD (or another) estimator will be only approximately affine-invariant and more generally, robust estimation of the covariance matrix, especially with growing dimension d, is naturally challenging and can be computationally involving; the reader is referred to some recent works [2, 10], rather to illustrate the complexity of the task. This issue obstructs on the whole construction of affine-invariant robust depths which achieve affine invariance explicitly involving covariance matrix. This is in particular the case with mentioned above spatial depth; with four out of five depths defined below being implicitly affine-invariant, this issue of depth's affine invariance will be again illustrated during definition of simplicial volume depth.

Mahalanobis depth can be referred to as a parametric depth, since it can be described by $d + \frac{d(d+1)}{2}$ parameters (and thus to perform anomaly detection they are sufficient to be computed and stored on the training phase), but its level contours (being quadratic functionals) take shape of concentric ellipsoids and are thus limited in summarizing data. The following four depths are defined based on data geometry and are nonparametric.

Halfspace depth, also called Tukey or location depth [12, 74], of $\boldsymbol{x}$ w.r.t. $\boldsymbol{X}$ is defined as the smallest fraction of $\boldsymbol{X}$ that can be contained in a closed halfspace together with $\boldsymbol{x}$. Representing halfspace by the vector orthogonal to its boundary hyperplane leads to the following definition:

$$D^{\text{hfsp}}(\boldsymbol{x}|\boldsymbol{X}) = \min_{\boldsymbol{u}\in\mathcal{S}^{d-1}} \frac{1}{n}\sum_{i=1}^n \mathbf{1}(\boldsymbol{x}_i^\top \boldsymbol{u} \le \boldsymbol{x}^\top \boldsymbol{u}), \tag{8}$$

where $\mathcal{S}^{d-1}$ denotes the unit sphere in dimension d (being $(d-1)$-dimensional surface) and $\mathbf{1}(A)$ stands for indicator function that equals 1 if event A is true and 0 otherwise. Clearly, since this is an estimator of data depth, the average of the indicator functions estimates empirical mass. Being probably one of the most studied in the literature depth notions, halfspace depth is nonparametric, affine-invariant (without involving estimation of the covariance matrix), and robust: Anomalies in $\boldsymbol{X}$ (almost) do not distract depth values of the normal data. These properties come with high computational cost though, with the most efficient implemented algorithm [17] to compute halfspace depth in any dimension d having time complexity $O(n^{d-1}\log n)$; see also `R`-package `ddalpha` [56, 57] and `Python` library `data depth` for implementation. Further, halfspace depth vanishes (and equals zero) immediately beyond the convex hull of the data.

Simplicial volume depth, also called Oja depth, is defined based on outlyingness measure suggested by [53]:

$$D^{\text{smpv}}(\boldsymbol{x}|\boldsymbol{X}) = \Big(1 + \frac{1}{\binom{n}{d}} \sum_{\substack{i_1 < \ldots < i_d \\ i_1,\ldots,i_d \subset \{1,\ldots,n\}}} \text{vol}_d\big(\text{conv}(\{\boldsymbol{x}, \boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_d}\})\big)\Big)^{-1}, \tag{9}$$

with $\text{conv}(A)$ standing for the convex hull of the set A (the smallest convex set containing A), and $\text{vol}_d(\cdot)$ being the d-dimensional volume that can be calculated as follows:

$$\text{vol}_d\big(\text{conv}(\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d+1}\})\big) = \frac{1}{d!}\big|\det\big((1, \boldsymbol{v}_1^\top)^\top, \ldots, (1, \boldsymbol{v}_{d+1}^\top)^\top\big)\big|, \tag{10}$$

where $\det(\cdot)$ denotes the determinant of a matrix. The average approximates here expectation of the volume of the set containing d points from $\boldsymbol{X}$ and $\boldsymbol{x}$. Simplicial volume depth is not affine-invariant in its traditional version and additionally is computationally involved: Its time complexity is $O(n^d d^3)$.

Simplicial volume depth can be transformed into affine-invariant by dividing the mentioned above average over the square root of the determinant of the covariance matrix $\boldsymbol{\Sigma}$:

$$D^{\text{smpv(ai)}}(\boldsymbol{x}|\boldsymbol{X}) = \Big(1 + \frac{1}{\binom{n}{d}} \sum_{\substack{i_1 < \ldots < i_d \\ i_1,\ldots,i_d \subset \{1,\ldots,n\}}} \frac{\text{vol}_d\big(\text{conv}(\boldsymbol{x}, \boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_d})\big)}{\sqrt{\det(\boldsymbol{\Sigma})}}\Big)^{-1}. \tag{11}$$

While this technique is sufficient for the simplicial volume depth, in general, affine invariance can be achieved by substituting all points by their whitened versions, $\boldsymbol{L}\boldsymbol{x}_i$ with $\boldsymbol{\Sigma}^{-1} = \boldsymbol{L}^\top \boldsymbol{L}$, as it can be the case for spatial [65, see also] or lens depth not addressed here in detail.

Projection depth [77] is defined as a strictly decreasing transform of the projected outlyingness [19, 69], which idea can be briefly formulated as follows: A point is outlying if it is outlying in projection on at least one direction. Naturally, by definition, projection depth is thus "searching" for this outlyingness-proving direction:

$$D^{\text{prj}}(\boldsymbol{x}|\boldsymbol{X}) = \Big(1 + \max_{\boldsymbol{u}\in\mathcal{S}^{d-1}} \frac{\big|\boldsymbol{x}^\top\boldsymbol{u} - \text{med}(\boldsymbol{X}^\top\boldsymbol{u})\big|}{\text{MAD}(\boldsymbol{X}^\top\boldsymbol{u})}\Big)^{-1}, \quad (12)$$

where (in slight abuse of notation) $\boldsymbol{X}^\top\boldsymbol{u}$ is a shortcut for projecting $\boldsymbol{X}$ on $\boldsymbol{u}$ resulting in $\boldsymbol{X}^\top\boldsymbol{u} = \{\boldsymbol{x}_1^\top\boldsymbol{u}, ..., \boldsymbol{x}_n^\top\boldsymbol{u}\}$ and med and MAD stand for (univariate) median and median absolute deviation from the median, respectively (for a univariate set Y, $\text{MAD}(Y) = \text{med}\big(\big|Y - \text{med}(Y)\big|\big)$). Any other robust estimators of univariate location and scale can be used instead, of course. Projection depth is affine-invariant (implicitly, in that resembling the halfspace depth), (highly) robust [78, asymptotic breakdown point of its median equals 0.5;], and (different to the halfspace depth) it is positive on the entire $\mathbb{R}^d$ once the depth is well defined. When describing the data, its contours retain certain degree of symmetry. To improve on this, **asymmetric projection depth** [21] has been proposed:

$$D^{\text{prj(as)}}(\boldsymbol{x}|\boldsymbol{X}) = \Big(1 + \max_{\boldsymbol{u}\in\mathcal{S}^{d-1}} \frac{\big(\boldsymbol{x}^\top\boldsymbol{u} - \text{med}(\boldsymbol{X}^\top\boldsymbol{u})\big)_+}{\text{MAD}_+(\boldsymbol{X}^\top\boldsymbol{u})}\Big)^{-1}, \quad (13)$$

where $(a)_+ = \max\{a, 0\}$ and MAD_+ is the median of the positive deviations from the median.

Simplicial depth is defined as the fraction of the—based on $d+1$ data points—simplices that contain $\boldsymbol{x}$ [37]:

$$D^{\text{smp}}(\boldsymbol{x}|\boldsymbol{X}) = \frac{1}{\binom{n}{d}} \sum_{\substack{i_1 < ... < i_{d+1} \\ i_1, ..., i_{d+1} \subset \{1, ..., n\}}} \mathbf{1}\big(\boldsymbol{x} \in \text{conv}(\{\boldsymbol{x}_{i_1}, ..., \boldsymbol{x}_{i_{d+1}}\})\big). \quad (14)$$

It is (implicitly) affine-invariant and robust, but approximating the average belongingness to a simplex involves high computational time complexity $O(n^{d+1}d^3)$. Different to the five depth notions, previously mentioned in this section, simplicial depth has non-convex contours; the same as the halfspace depth, it vanishes immediately beyond the convex hull of the data set $\boldsymbol{X}$.

3 Algorithmic aspects

This section provides information about computational aspects of data depths, which are important for anomaly detection. Section 3.1 proposes a taxonomy of data depth based on their calculation properties. Section 3.2 refers to use of depth-trimmed regions for anomaly detection.

3.1 Computational taxonomy

Several taxonomies have been proposed to categorize existing data depth notions into groups, mostly by construction mechanism of depths; let us cite two of them. [77] divide multivariate data depth notions in four types: *A* (depths based on closeness to the sample), *B* (depths based on distance to the sample), *C* (depth based on outlyingness measure), and *D* (depths based on "tailedness" measured using a class of closed subsets). [49] categorizes depths in those based on distances (including $\mathbb{L}_p$, Mahalanobis, simplicial volume, and projection depths), weighted mean depths, and depths based on halfspaces and simplices.

[21] introduced a class of depths that satisfy the (weak) projection property: They can be computed as minimum of depths in univariate projections. Including zonoid, expected convex hull, geometrical, Mahalanobis, halfspace, projection depths, this class is very important since it allows to develop efficient algorithms based on computation of univariate depths only. Further, [15] [16, see also] define a subclass of these depths—depths defined via (convex) weighted mean trimmed regions that include zonoid, expected convex hull, and geometrical depths.

In the current article, we suggest a novel approach—the *computational taxonomy*. Having undergone substantial theoretical developments during last 30 years data depth has options to offer, and in many situations a depth notion (or its modification) that satisfies expected theoretical properties (important for application of interest and thus potential data generating distribution) can be found or constructed by modification of an existing notion. Today, in the era of big data, computational properties gain even more importance, and the choice of depth can be restricted by computational time and/or resources.

In what follows, we address exact computation (i.e., computing depth of observation $\boldsymbol{x}$ with respect to data set $\boldsymbol{X}$) of all above-mentioned depths in view of three axes: computational time complexity, affine invariance, and robustness; see Table 1. While only for several depths the complexity is either proved or obvious, the below summary is relying on complexities of developed algorithms (and their implementations) based on the preceding decades of research.

Several remarks are in order here:

- Exponential complexity for halfspace depth has been shown by [31], this of convex hull peeling depth is mentioned in its computing algorithm [4], while projection depth is believed to be so as well [45]. Complexities of majority, $\mathbb{L}_p$, spatial, lens, Mahalanobis, simplicial volume, and simplicial depths follow from definitions being combinatorial sums of certain order.

Table 1 Computational taxonomy of data depth along three axes: computational time complexity, affine invariance, and robustness; depth notions in *italics* are robust

		Exponential time	Polynomial time
Affine-	invariant	*convex hull peeling* *majority* expected convex hull geometrical *halfspace* *projection* *simplicial*	zonoid Mahalanobis
Not affine-	invariant	simplicial volume	$\mathbb{L}_2$ *spatial* *lens*

- Zonoid depth can be formulated as a linear programming task [13], which can be solved in polynomial time using the interior point method [32].
- Properly speaking, robustness properties of the $\mathbb{L}_p$-depth depend on the chosen norm degree p; we mention only better studied $\mathbb{L}_2$-depth here.
- No exact algorithms have been designed for computing expected convex hull and geometrical depths, which can be trivially done when optimizing these depths by searching through contours (exploiting the mentioned above monotonicity property (6)). Algorithm for computing such contours possesses exponential time complexity [6].
- The non-existent entry in Table 1 is a depth being simultaneously affine-invariant, robust, and having polynomial time complexity. Making spatial or lens depth affine-invariant is possible by using covariance matrix, as it is the case above for affine-invariant simplicial volume depth $D^{\text{smpv(ai)}}$ (11). On the other hand, no polynomial time algorithm exists for computing such a matrix exactly in a robust way without further assumptions.

3.2 A word about depth contours

Being a function defined on $\mathbb{R}^d$, data depth gives rise to depth-trimmed central regions, mentioned already above (when listing depth postulates) and defined—for the data set $\boldsymbol{X}$ and depth level $\alpha \in [0, 1]$—as:

$$D_\alpha(\boldsymbol{X}) := \{\boldsymbol{x} \in \mathbb{R}^d : D(\boldsymbol{x}|\boldsymbol{X}) \geq \alpha\}.$$

These regions describe data with respect to their location, scatter, and shape and can be used for insightful visualization in dimensions 2 and 3; see, e.g., [6, 41, 42] to name but a few. Further, they can be used to define anomaly detection rule, for a properly chosen depth level $\alpha(t_{\boldsymbol{X}_{tr}})$:

$$g_{\text{reg}}(\boldsymbol{x}|\boldsymbol{X}_{tr}) = \begin{cases} \text{anomaly}, & \text{if } \boldsymbol{x} \notin D_{\alpha(t_{\boldsymbol{X}_{tr}})}(\boldsymbol{X}_{tr}), \\ \text{normal}, & \text{otherwise.} \end{cases} \tag{15}$$

Seeming easy to write and implement, rule (15) conceals substantial computational difficulties. These lie in calculating the depth contours themselves and can be expressed as follows. Mahalanobis depth contours are simple ellipsoids and do not reflect properly the shape of the data. Algorithms computing contours of depth notions attractive from anomaly detection point of view have time complexity growing exponentially with space dimension and for a number of depths (e.g., lens or simplicial depth) are unknown to the literature. For certain depths (e.g., halfspace or zonoid depth) only necessary part of the contour—e.g., where (majority of) anomalies are expected—can be computed, but such an approach is difficult to justify in practice, while it still becomes intractable with growing space dimension. Indeed, taking into account the currently developed algorithmic basis, it is much more practical to first compute depth of an observation and then check whether it belongs to depth region (of normal data) by comparing with a threshold.

A possible solution to the above-mentioned issue could be development of approximating algorithms for depth contours relying on a small number of (possibly simple) surfaces elements.

4 Suitability for anomaly detection

With the main task of this section being illustration of the very mechanism of application of data depth to the anomaly detection task, we focus on the task of identifying anomalies in the training data. This allows to keep the exposition simple preserving the main difficulty of unsupervised anomaly detection, i.e., to construct a rule not perturbed by the anomalies in the training data. It is from Sect. 5 that we follow the machine learning framework and consider two separate data sets for training and testing.

To perform anomaly detection in practice, two natural questions arise first: (i) Which depth to choose how and (ii) how (exactly) to proceed?

4.1 Choice of the depth function

[48] discuss that when applying data depth, choice of a suitable depth notion is crucial and usually consists in finding a compromise between its statistical and computational properties. To provide insights on this, let us take a look at the following (somewhat general) simulated example.

Given a training data set $\boldsymbol{X}_{tr}$ of 100 observations, where 90 of them are generated from bivariate normal distribution $\mathcal{N}\left((1, 1)^\top, \left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix}\right)\right)$ with the remaining 10 stemming also

from bivariate normal distribution but with 36 times smaller covariance and located at the previous mean shifted 2.5 in direction of the second principal component of normal data: $\mathcal{N}\Big((3.181, -0.222)^\top, \left(\begin{smallmatrix}1 & 1\\ 1 & 2\end{smallmatrix}\right)/36\Big)$. This example constitutes a rather favorable case, e.g., the 10 anomalies can be identified by visual inspection (which is of course impossible in higher dimensions).

First, consider projection depth (12) for anomaly detection. For the training sample mentioned right above, Fig. 2 (left) plots depth values $D^{\text{prj}}(\boldsymbol{x}|\boldsymbol{X}_{tr})$ of the entire Euclidean space (i.e., for all $\boldsymbol{x} \in \mathbb{R}^2$, see the depth scale on right side of the figure), as well as three depth contours for minimal depth of normal observations, and maximal and minimal depth of anomalies. As it is seen visually, based on the provided training set of 100 observations (including 10 anomalies), rule (3) allows to detect anomalies when choosing a reasonable threshold $t_{\text{prj},\boldsymbol{X}_{tr}}$ (between 0.151 and 0.194 here).

Second, consider employing halfspace depth (8) in the same way. Plot of Fig. 2 (right), which depicts both depth values and contours according to the same logic, shows that correct detection of anomalies is rather impossible, which is in part due to immediate vanishing of the halfspace depth beyond the convex hull of the data. A number of works attempted to improve on this last issue, e.g., [23, 51, 60] to name a few, with halfspace mass [11] being a computationally tractable alternative lacking affine invariance property though.

For an additional visualization, Fig. 3 (left) plots ordered projection and halfspace depth values for all 100 observations of the training sample.

Let us consider another couple of widely known depths, namely simplicial volume (9) and simplicial (14) depths. For the same training set, similar visualization is depicted in Fig. 4. As before, by construction and due to the immediate vanishing property, simplicial depth fails to detect the group of 10 anomalies, a task being coped with by simplicial volume depth.

While this example is very typical for anomaly detection, other situations appear. For instance, regard an example where additional (single) anomalies mask the clustered ones. For this, let us add to the previous training sample 25 observations distributed in a similar manner as normal training data but having Mahalanobis distance (calculated using normal population mean vector and covariance matrix) between minimal and maximal Mahalanobis distance of the 10 clustered anomalies. As we can see from Fig. 5, as before projection depth copes with the task correctly retrieving all 35 anomalies, but this time the halfspace depth also detects majority of them ("thanks" to masking effect); see additionally Fig. 3 (right). Another couple of depths—simplicial volume and simplicial—behave similarly with visualization omitted here for space purposes.

While projection depth is very suitable for the examples from above, its upper-level sets retain certain degree of symmetry, which can be inadequate for, e.g., skewed data. To get more insights for this latter case, we shall contrast projection depth with its asymmetric version (13), in the following setting. First, we generate 100 (normal) observations from a skewed bivariate distribution with independent marginals being skewed normal with parameter 9 [1, abscissa, according to] and centered normal with standard deviation 1/4 (ordinate). Then, we add 15 anomalies from the same distribution, with probability density $\in [0.01, 0.05]$ and positive values for abscissa. The corresponding plots for both depth notions are indicated in Fig. 6, top. One observes that while in this particular case projection depth still copes with the task, asymmetric projection depth allows to immediately identify the group of anomalies, see Fig. 6, bottom.

4.2 Choice of the threshold

Together with Sect. 1.2, this section in a natural way attracts attention to the following practical question: Which value should one choose for the threshold that cuts off anomalies (as observations with small values of depth)?

In view of generality and complexity of the task of unsupervised anomaly detection, it would be apparently too hopeful to expect a universal answer. One practical way is to use the so-called elbow rule, i.e., to place threshold where the slope of the (locally) straight line following plotted—in decreasing order—depth values suddenly decreases. This works if plotted in decreasing (or increasing) order depth values can be well interpolated by two connected straight lines such as in left Fig. 7 (the threshold is then set where the two lines connect), but is much more ambiguous otherwise. Another practical way is to place the threshold in the "gap" between ordered depth values, as this can be observed in Figs. 3 (both left and right) for projection depth and Fig. 6 (right) for asymmetric projection depth or Fig. 7 (right) for an explicit example; as before application of this rule becomes ambiguous if such a gap on the depth graph does not exist.

Furthermore, in practice the anomaly-detecting threshold is rarely fixed (forever) in advance and is periodically altered, either due to distribution change or (enterprise) policy evolution. Guided by such a policy, one naturally attempts to (always) detect all anomalies, which can optimistically be the case, but more often one has to admit that part of anomalies will remain unnoticed. In this latter case it is important to deliver a reasonable score reflecting a degree of abnormality—advocated as data depth in this article. Such a score shall further allow for adjusting the threshold depending on the application-related trade-off, e.g., between undetected anomalies (false negatives) and normal data marked as anomalies (false positives); there are of course other formulations of this trade-off using different quantities.

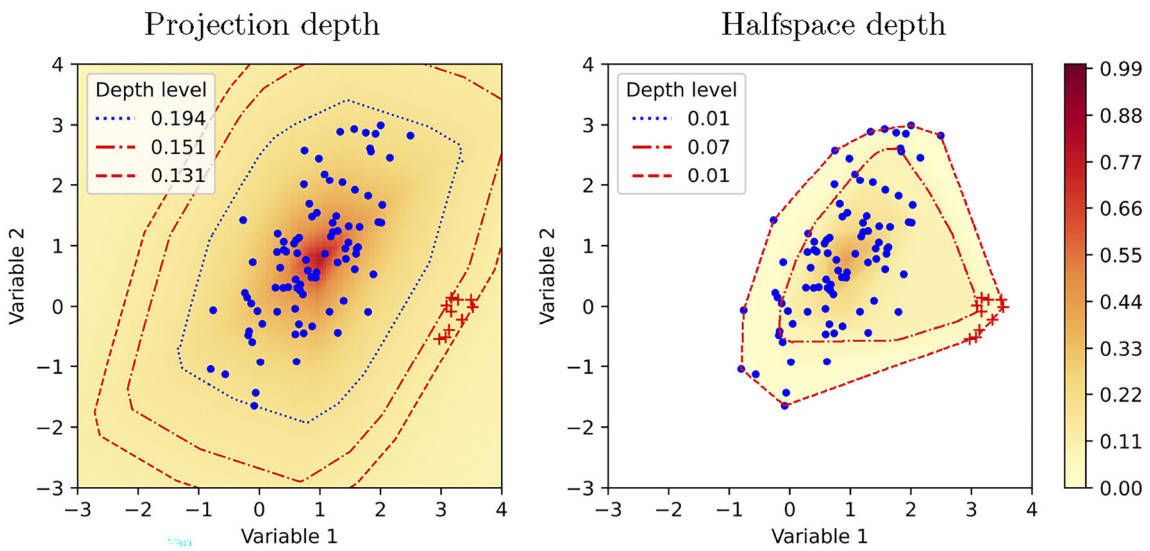


Fig. 2 90 observations stemming from bivariate normal distribution (blue dots) contaminated with 10 observations (red pluses). Depth contours at three levels: minimal depth of normal observations (blue dotted line), maximal depth of 10 anomalies contaminating the training sample (red dash-dotted line), minimal depth of 10 anomalies contaminating the training sample (red dashed line). Left: depth values (in color) for projection depth. Right: depth values (in color) for the halfspace depth (with white corresponding to zero). Color scale for both plots is depicted on the right side

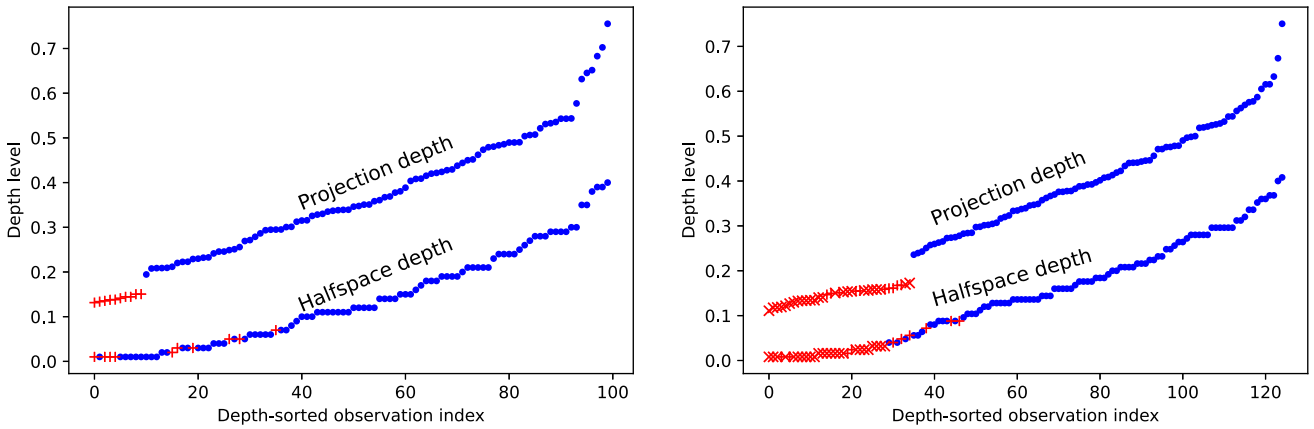


Fig. 3 Ordered depth values for projection and halfspace depth, with normal data corresponding to blue points and anomalies depicted with red pluses and crosses. Left: For 100 observations of the training data from first example in Sect. 4. Right: For 125 observations of the training data from second example in Sect. 4

It is noteworthy that getting a deeper insight in the nature of anomalies, e.g., by explaining those (presumably) detected in the training sample (see Sect. 5.3 below for an example) is beneficial for threshold definition as well.

For the reasons described above, the approach used in benchmark studies of Sects. 5 and 6 shall be based on evaluation of the obtained ordering; see also (16).

In case of the data arriving sequentially (e.g., in time), we suggest the following general anomaly detection scheme, based on visual inspection for threshold selection: Let $\boldsymbol{x}(t_i)$ for $i = 1, 2, \ldots$ be the sequence of observations appearing at time moments t_i, with $\boldsymbol{x}(t_i) \in \mathbb{R}^d$ for all i. Denote $\boldsymbol{X}_{tr} = \{\boldsymbol{x}(t_{tr_1}), \ldots, \boldsymbol{x}(t_{tr_T})\}$, $\boldsymbol{X}_{te} = \{\boldsymbol{x}(t_{te_1}), \ldots, \boldsymbol{x}(t_{te_T})\}$, and $\boldsymbol{X}_{in} = \{\boldsymbol{x}(t_{in_1}), \ldots, \boldsymbol{x}(t_{in_T})\}$ train, test, and inference sets, respectively.

1. Choose the training set window, i.e., determine tr_1 and tr_T.
2. Choose the testing set window, i.e., determine ts_1 and ts_T. (In the unsupervised framework, i.e., when one does not have *a priori* information about presence of anomalies in the training set, it is not uncommon to choose $te_1 = tr_1$ and $te_T = tr_T$.)
3. By plotting depth $D(\boldsymbol{x}|\boldsymbol{X}_{tr})$ of each $\boldsymbol{x} \in \boldsymbol{X}_{ts}$ in increasing order, choose a threshold $t_{D,\boldsymbol{X}_{te}}$ following logic similar to this of Fig. 7.

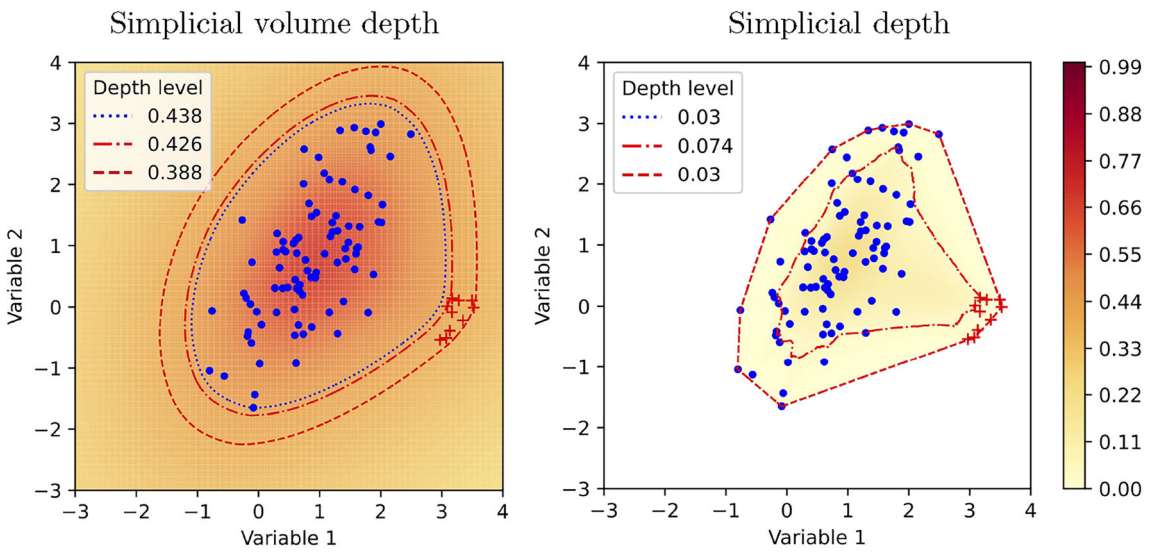


Fig. 4 90 observations stemming from bivariate normal distribution (blue dots) contaminated with 10 observations (red pluses). Depth contours at three levels: minimal depth of normal observations (blue dotted line), maximal depth of 10 anomalies contaminating the training sample (red dash-dotted line), minimal depth of 10 anomalies contaminating the training sample (red dashed line). Left: depth values (in color) for simplicial volume depth. Right: depth values (in color) for simplicial depth (with white corresponding to zero). Color scale for both plots is depicted on the right side

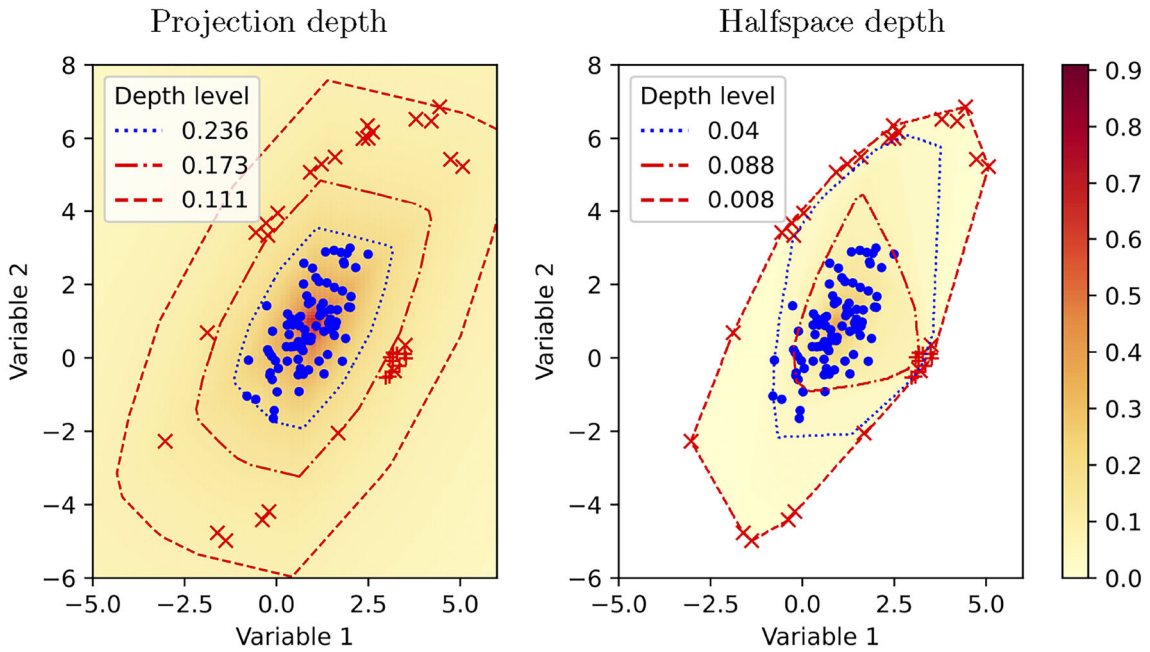


Fig. 5 90 observations stemming from bivariate normal distribution (blue dots) contaminated with 10 clustered (red pluses) and 25 masking (red crosses) anomalies. Depth contours at three levels: minimal depth of normal observations (blue dotted line), maximal depth of all 35 anomalies contaminating the training sample (red dash-dotted line), minimal depth of all 35 anomalies contaminating the training sample (red dashed line). Left: depth values (in color) for simplicial volume depth. Right: depth values (in color) for simplicial depth (with white corresponding to zero). Color scale for both plots is depicted on the right side

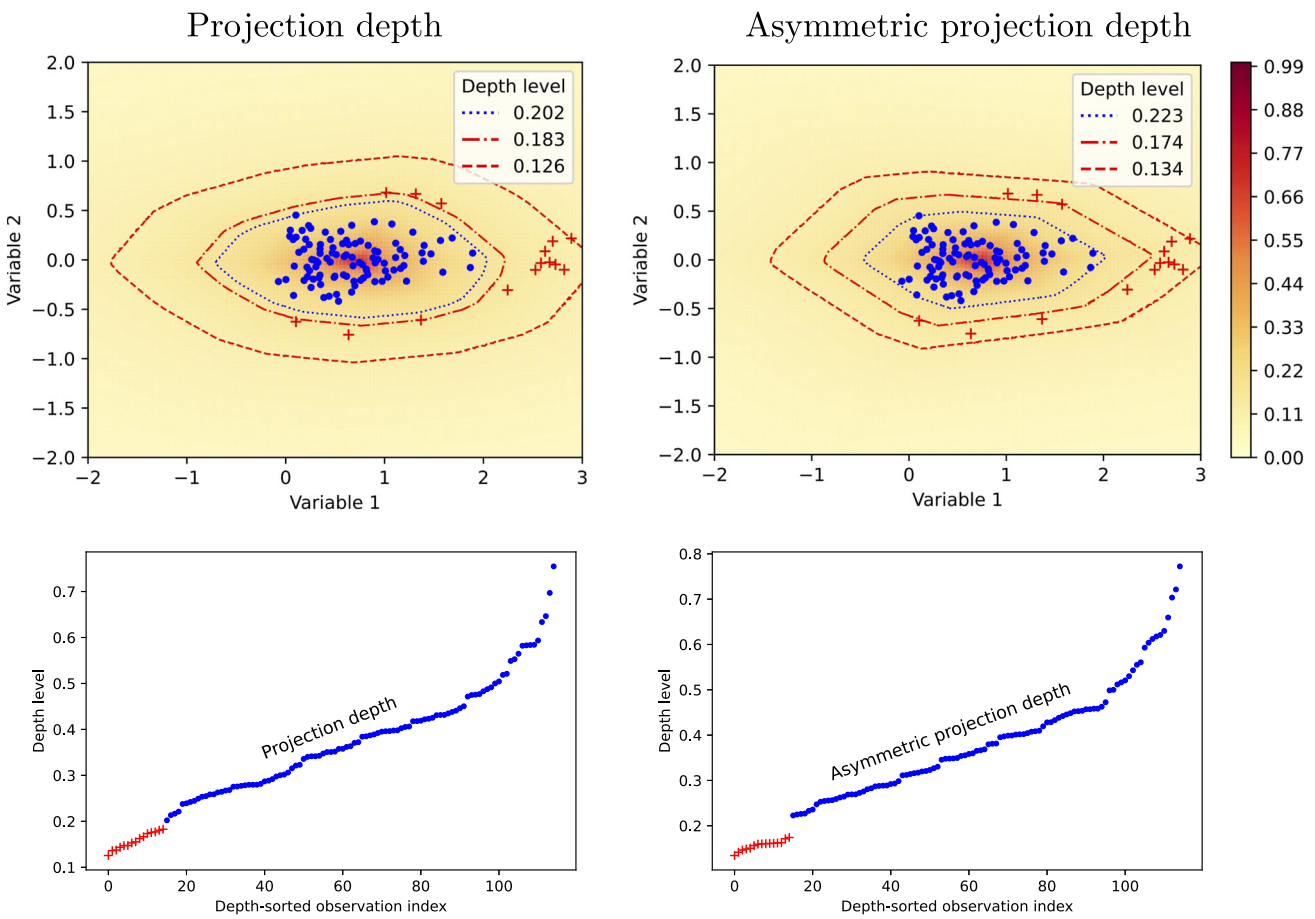


Fig. 6 100 observations stemming from skewed bivariate distribution (blue dots) contaminated with 15 anomalies (red pluses). Depth contours at three levels: minimal depth of normal observations (blue dotted line), maximal depth of 15 anomalies (red dash-dotted line), minimal depth of 15 anomalies (red dashed line). Top: depth values (in color) for projection depth (left) and asymmetric projection depth (right); color scale for both plots is depicted on the right side. Bottom: Ordered depth values for projection (left) and asymmetric projection (right) depth

4. For each $\boldsymbol{x} \in \boldsymbol{X}_{ti}$, assign a label according to the following rule:

$$g(\boldsymbol{x}) = \begin{cases} \text{anomaly}, & \text{if } D(\boldsymbol{x}|\boldsymbol{X}_{tr}) < t_{D,\boldsymbol{X}_{te}}, \\ \text{normal}, & \text{otherwise.} \end{cases}$$

5. Continue Step 4 for newly arriving observations as long as necessary (e.g., stable anomaly detection rate over time). Once the anomaly landscape is suspected to have changed, restart from Step 1.

5 Advantages

As announced in the introduction, the main goal of this section is to illustrate advantages of the data depth when performing anomaly detection. In order to do this, its subsections shall include comparisons with the most used methods for anomaly detection existing in the literature. It is important to stress that our goal here is not to state that data depth performs better—in frame of anomaly detection—than the mentioned methods neither to provide a comprehensive comparison, but rather to identify potential cases where its deployment can be advantageous.

In Sect. 5.1 we start with illustrating robustness of the depth-based methodology comparing with widely used anomaly detection neural network—auto-encoder. Further, in Sect. 5.2 depth is contrasted with three major methods for anomaly detection: isolation forest, local outlier factor, and one-class support vector machine. Finally, Sect. 5.3 shall highlight explainability capacities of the depth-based approach—a highly demanded feature today lacking for neural networks.

Anomaly detection rule (3) based on projection depth (12) shall be used throughout this section.

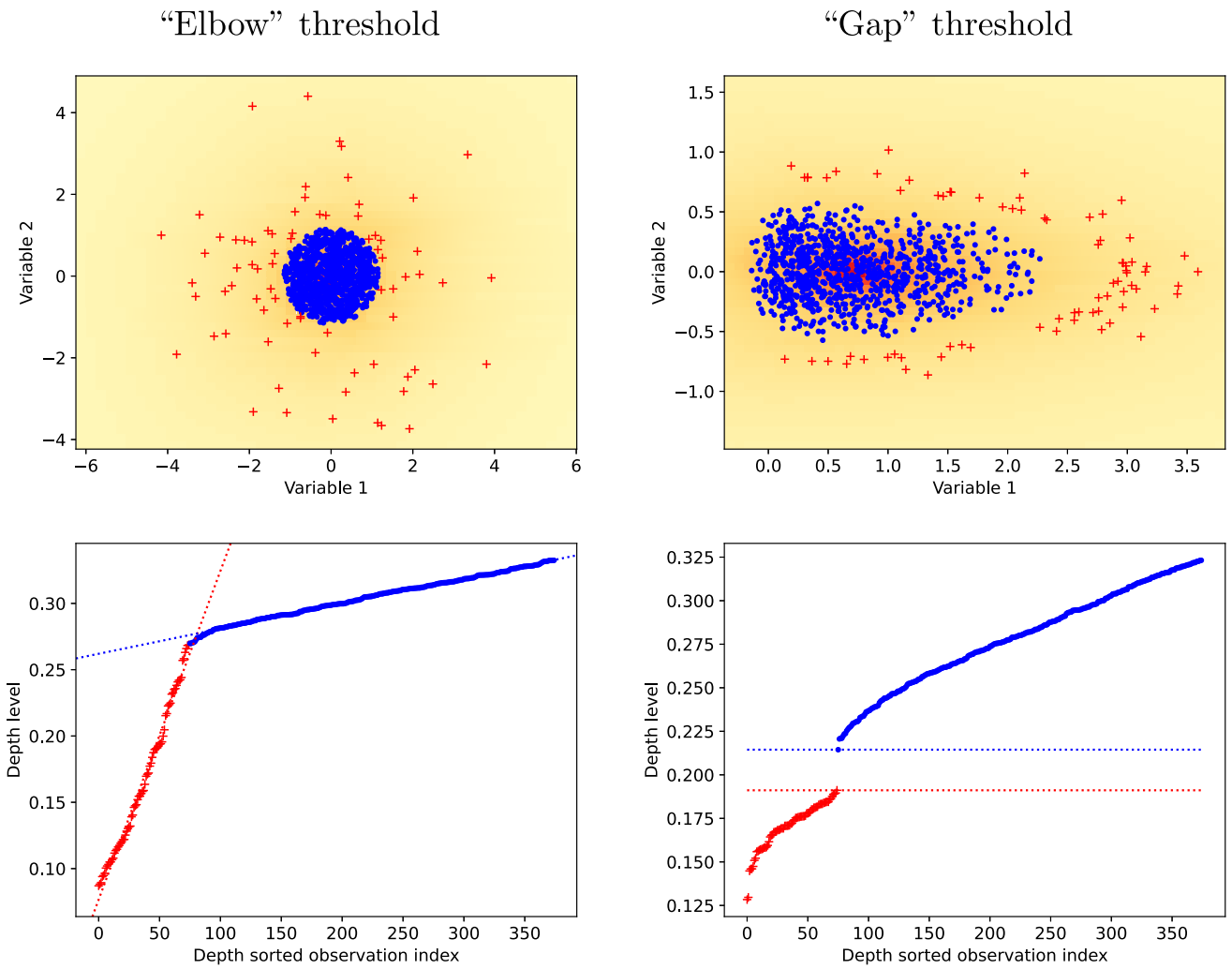


Fig. 7 Simulated data set for an elbow-shaped threshold (left) and for a gap-shaped threshold (right). To ease visualization, depth level is zoomed on the lower 30% values

5.1 Robustness

Autoencoder is a widely used contemporary method for anomaly detection, applied to various types of data. The idea is to use bottleneck when joining sequentially neuronal encoder and decoder, and detect anomalies by high reconstruction loss. Being a neural network, it inherits all the advantages of scalability and flexibility from artificial-neuron-based architectures, demanding meticulous tuning and (especially for complex data sets) computational resources though. Complexity of the autoencoder model (which can be quantified by its exact architecture as well as by the number of parameters to estimate, with the last number sometimes reaching millions) impedes human-understandable interpretability of its output. Elaborating on this explainability issue later in Sect. 5.3, let us discuss autoencoder's robustness contrasting it with data depth.

Deep support vector data description (deep SVDD) acts similarly to the spherical one-class support vector machine [67], but (like autoencoder) improves the anomaly detection performance using neural-network-based representation instead of RKHS [73]. With the method being widely used in practice and inheriting advantages from both statistical and machine learning sides, we include it in the comparison.

In the unsupervised setting, when no feedback is given about abnormality of the observations of the training set, autoencoder detects anomalies assuming that majority of the data are normal and deviations of anomalies are not critically large. This is due to the fact that, when being trained using stochastic gradient descent algorithm, derivative of the loss function is required. This smoothness condition impedes autoencoder from being robust, since larger losses—in particular with substantial portions of anomalies—"distract" the model (on an average) during training. This same assumption

of majority of data being normal also holds for Deep SVDD and in the presence of higher fractions of anomalies in the training data can lead to a poor embedding of normal data in the neural network part and consequently may result in a bad hypersphere minimization.

A robust depth function (e.g., projection or halfspace depth), on the other hand, can include elements of indicator nature (corresponding to the so-called 0-1-loss) in the definition, which are not influenced by the amplitude of anomalies. Such indicator-containing functions do not possess derivative everywhere (and even have steps/breaks), which makes them unoptimizable for neural network architectures. Such an approach rather shifts the problem from the definition part (e.g., surrogate loss) to the numerical/computational stage; solutions to this problem have been recently proposed for a number of data depth functions in larger dimensions [18, see, e.g.,]. (In the present article, computation of data depth covers spaces up to $\mathbb{R}^{50}$.) We illustrate this difference on a short simulation study right below.

Consider a training data set $\boldsymbol{X}_{tr} \subset \mathbb{R}^d$ consisting of $n = 700$ observations simulated in the following way. First, subset of $\lfloor n(1-\varepsilon) \rfloor$ (normal) points $\boldsymbol{Y}_{tr} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_{\lfloor n(1-\varepsilon) \rfloor}\}$ is drawn from the normal distribution in $\mathbb{R}^d$ ($\mathcal{N}(\boldsymbol{i}_d, \boldsymbol{I}_{d\times d})$, where $\boldsymbol{i}_d = (1, 1, ..., 1)^\top$ is the d-vector of 1s and $\boldsymbol{I}_{d\times d}$ is the $d \times d$ matrix of 0s with 1s on the main diagonal only). Then, second subset $\boldsymbol{Z}_{tr}$ of $\lceil n\varepsilon \rceil$ (abnormal) points is drawn from the conditional Cauchy distribution $Z \mid \|Z\| > 1.5 \max(\|\boldsymbol{y}_1\|, \|\boldsymbol{y}_2\|, ..., \|\boldsymbol{y}_{\lfloor n(1-\varepsilon) \rfloor}\|)$, where Z is the d-variate random vector stemming from elliptical Cauchy distribution. (Elliptical distribution is a generalization of the multivariate normal distribution with density contours being ellipsoids to further univariate laws then the Gaussian; the reader is referred to, e.g., [24] for more details.) Third, $\boldsymbol{X}_{tr} = \boldsymbol{Y}_{tr} \cup \boldsymbol{Z}_{tr}$, followed by (numerical) random shuffling of the elements of $\boldsymbol{X}_{tr}$. Likewise, the same procedure is used to generate testing data $\boldsymbol{X}_{te}$, with $n = 300$.

The depth-based anomaly scoring rule of type (3) is then applied to each observation of the testing data, based on projection depth. The smallest threshold is chosen to correctly detect all anomalies in $\boldsymbol{X}_{te}$: $t_{\text{prj},\boldsymbol{X}_{te}} = \max_{z\in\boldsymbol{Z}_{te}} D^{\text{prj}}(z|\boldsymbol{X}_{tr}) + \epsilon$ for some infinitesimal positive ϵ. It is shown this threshold is built on new observations $\boldsymbol{X}_{te}$ but using $\boldsymbol{X}_{tr}$ as a reference to emulate the standard machine learning framework (training followed by testing). The following quantity is then used for comparison:

$$p(\boldsymbol{X}_{te}) = \frac{\sum_{z\in\boldsymbol{Z}_{te}} g_{\text{prj}(z|\boldsymbol{X}_{tr})}}{\sum_{z\in\boldsymbol{X}_{te}} g_{\text{prj}(z|\boldsymbol{X}_{tr})}}, \tag{16}$$

which reflects the (largest) portion of the anomalies in the—ordered by depth—part of the testing sample identified as anomalies such that all anomalies are correctly detected.

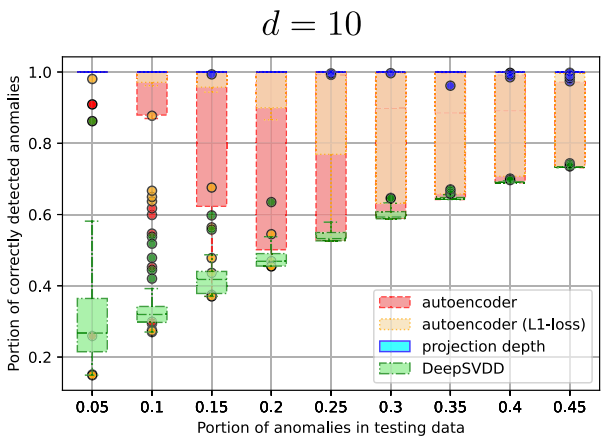
With autoencoder, the anomaly scoring rule is based on (quadratic) reconstruction error: Observations with higher (i.e., beyond a threshold) reconstruction error are identified as anomalies. As well as in (16), the threshold is chosen to be (slightly smaller than) the highest value to recognize correctly all anomalies. The autoencoder is trained on $\boldsymbol{X}_{tr}$ with both quadratic and L_1 losses, but only quadratic loss is used to detect anomalies. Boxplots of p from (16) over 50 independent draws of $\boldsymbol{X}_{te}$ for both autoencoders, as well as for the projection depth, are indicated in Fig. 8, where $\varepsilon = 0.05, 0.1, ..., 0.45$ and $d = 10, 20$ are tried. (Projection depth was approximated using Nelder-Mead algorithm as in [18] with 100 directions for $d = 10$ and 200 directions for $d = 20$. The autoencoder contains three hidden layers each having $5-2-5$ (for $d = 10$) neurons ($20-10-5-10-20$ for $d = 20$, respectively) and is trained 100 epochs with 10 observations per minibatch, using stochastic gradient descent algorithm with learning rate 0.005. Python libraries `data depth` and `PyTorch` were used, respectively.)

From Fig. 8, one can clearly observe that while depth-based anomaly detection perfectly copes with the task, the autoencoder fails in most of the cases if portion of anomalies in the training set exceeds 10% and with improving but becoming more concentrated error if this portion grows. Further, higher non-stability of training of autoencoder is observed, while introducing (robust) L_1 loss does not improve the output much. It is important to notice that the autoencoder's architecture and training schedule/loss used here are typical and chosen for illustration purposes; when robustness is a potential issue more sophisticated architectures as well as losses and training schedules could improve the results.

To strengthen the outcome of our analysis, a similar simulation is carried out, this time using real data sets. The aim is to obtain normal observations with a real distribution (by removing real anomalies whose distribution is unknown) and to add anomalies in a controlled setting (i.e., with a known distribution) to validate the results obtained previously. Both chosen data sets ("PageBlocks" with $d = 10$ and "cardio" with $d = 21$) have dimensions comparable to first experiment. One can see from Fig. 9 that the performance is of the same order of magnitude as that of the depth-based method, achieving excellent detection, while others barely reach above 70% of anomalies correctly assigned for any fraction of contamination.

5.2 Extrapolation

While fundamental results of the empirical risk minimization theory request the distributions of the training and the test data to be identical, anomaly detection preserves additional difficulties since operational abnormal patterns' behavior can differ (substantially) from anomalies present in the train-



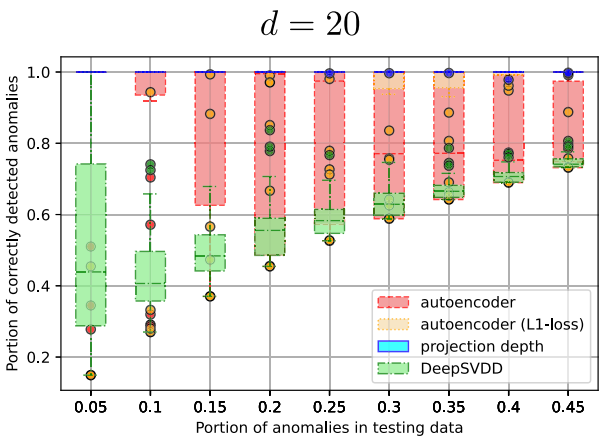
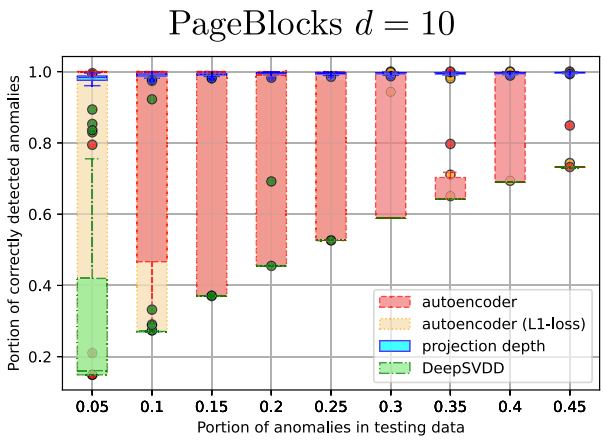


Fig. 8 Boxplots of portion of correctly detected anomalies of the testing data set when setting the threshold to detect all anomalies, over 50 random draws. Normal data stem from the multivariate Gaussian distribution in $\mathbb{R}^d$ ($d = 10$ on the left and $d = 20$ on the right), while anomalies are drawn from elliptical Cauchy distribution and having distance > 1.5 of the most distant normal observation from the origin. The three methods are: autoencoder with quadratic loss (red, dashed), autoencoder with L_1 loss (orange, dotted), Deep SVDD (green, dashed-dotted), and the projection depth (blue, solid)



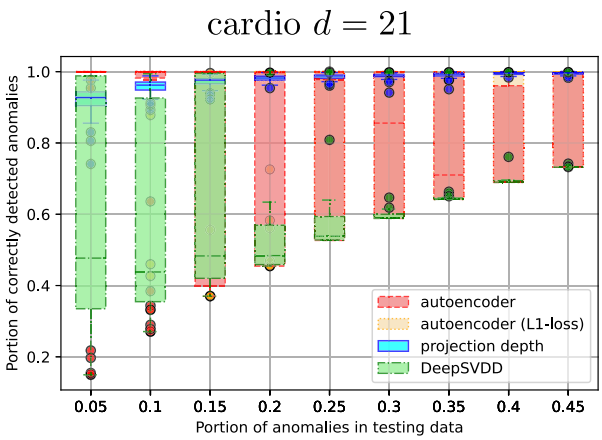


Fig. 9 Boxplots of portion of correctly detected anomalies of the testing data set when setting the threshold to detect all anomalies, over 50 random draws. Normal data stem from two real data sets from ODDS library [59] where anomalies were removed to add anomalies from known distribution (on the left PageBlocks with $d = 10$ and on the right cardio with $d = 21$). Anomalies are drawn from elliptical Cauchy distribution and having distance > 1.5 of the most distant normal observation from the origin. The three methods are: autoencoder with quadratic loss (red, dashed), autoencoder with L_1 loss (orange, dotted), Deep SVDD (green, dashed-dotted), and the projection depth (blue, solid)

ing data. To tackle this issue, employed anomaly detection method at hand should be able to extrapolate the knowledge about normal data to "protect" it from (preferably) all possible occurrences of future anomalies.

In this subsection we shall pay more attention to the operational nature of anomaly detection, i.e., the presence of decision function (or a rule), training data set, and new data (often called test set in study setting). For illustration purposes, we contrast data depth with *local outlier factor* [5, LOF], *one-class support vector machine* [67, OC-SVM], *isolation forest* [44, IF], and *autoencoders* [72, AE] on the same training and test data sets, which we construct as follows.

100 observations $\in \mathbb{R}^2$ constitute (viewable) *training data set*. 90 of them—normal data—follow uncorrelated normal distribution with mean $(1/2, 1/2)^\top$ and equal standard deviations $1/4$ for both variables. 10 anomalies are drawn from uncorrelated normal distribution with mean $(-3/4, 1/2)^\top$ and both standard deviations equal $1/10$. For the *testing set* (consisting of 300 observations), 250 observations are generated from the distribution of normal data, 25 from the distribution of anomalies, and 25 from uncorrelated normal distribution with mean $(7/4, 1/2)^\top$ and both standard deviations equal to $1/10$.

The parameters of the anomaly detection methods were chosen as follows: For LOF, number of neighbors is set to

25; OC-SVM is used with Gaussian kernel with parameter $\gamma = 0.1$ and regularization constant $\nu = 0.1$ (which after has illustrated best results); IF is constituted of 500 isolation trees; AE is built on $2-1-2$ layers' structure with L_1 loss using a learning rate of 0.005 during maximum 100 epochs and a batch size of 10 (that has illustrated best results); depth-based rule (3) is used with projection depth approximated with the spherical Nelder-Mead algorithm using 500 directions. (After multiple tries, values delivering best results on these data have been chosen; eventually the methods exhibit their regular behavior.) `Python` libraries `scikit-learn` [58], `PyTorch` [55], and `data depth` were used for implementation.

Figure 10 illustrates anomaly detection of both training and test sets using LOF. In general, LOF copes with the task to certain degree, where only very little normal observations of the testing set (which are further from the center) have same score as anomalies. Moreover, better choice of number of nearest neighbors could even improve the results. The only minor disadvantage is that anomalies on the right side of the normal data (i.e., where there were no anomalies in the training set) have different score from those on the left side, which can create a false impression that they are more abnormal, but indeed normal data were contaminated. As we shall see in Appendix A, this behavior is as well preserved with growing dimension, while the accuracy of a neighbor-based method is expected to deteriorate. Further, the number of nearest neighbors should be chosen, which can be a difficult task in the non-supervised context, and may not always provide satisfactory results if the data's density varies.

Figure 11 (left) indicates the anomaly score of OC-SVM. Being initially designed for estimation of support, OC-SVM fails to detect anomalies (even though assigning them rather low score, proper for non-central data) where they were already present in the training set, and detects only newly introduced anomalies. Better tuning of the kernel bandwidth does not seem to substantially change the plot. It is noteworthy that for a training set without (or with only several) anomalies OC-SVM would perform better, while also coping with the curse of dimension.

Anomaly score of IF is depicted in Fig. 11 (right). It does not detect any of the anomalies, with the reason being their location on the level of the center of normal data in one of the coordinates (ordinate). This happens while IF treats the dimensions one-by-one and becomes less important with growing dimension since the volume of such areas decreases. [28] suggest a modification of IF (called extended isolation forest), which using random directions suppresses this effect. It is of course important to mention that scores of anomalies are nevertheless low, similar to those of non-central normal data.

Figure 12 (right) exhibits anomaly score for autoencoder with satisfying detection for both training and testing data without any distinction between left and right clusters. Mapping of the score on the left displays a correct characterization of the normal data (dark green) with no asymmetry due to training data (red pluses). A finer tuning of the autoencoder parameters could refine the location of normal data (visible in Fig. 12 (left) as the darkest color region shifted up right from the data cloud) and would probably reduce the already small number of misclassification.

In this case, data depth (see Fig. 13) well copes with the task and detects anomalies in both training and testing set, including the group of newly introduced anomalies on the right of the normal data. Due to its robustness, the depth of both groups of anomalies is very similar reflecting their equal degree of abnormality. Similar experiments with $n = 1000$ points in dimensions 10 and 20 are available in Appendix A.

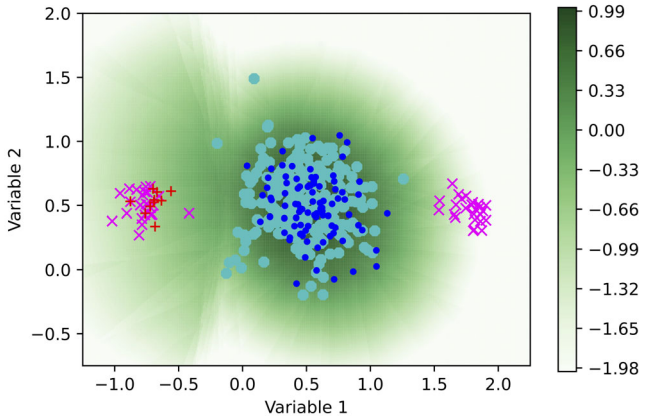
5.3 Explanation of anomalies

Not only detecting anomalies, but also providing explanation about their nature is a highly demanded contemporary topic treatable by data depth. Let us take a deeper look at the first example of Sect. 4; when using projection depth, visualizations are provided in Figs. 2 (left) and 3 (left). In this subsection, only the training data $\boldsymbol{X}_{tr}$ will be used with the goal being to explain anomalies, while the same idea can be readily applied to test data $\boldsymbol{X}_{te}$. When applying rule (3), choosing a proper threshold $t_{\mathrm{prj},\boldsymbol{X}_{tr}}$ (e.g., 0.175) does not seem to be a difficult task in this particular case: Even when imagining that abnormal observations do not differ in marker, a characteristic jump in (low) depth values provides a good indication.

Further, again sticking to projection depth, explanation of anomalies can be obtained when studying them revealing directions, *e.g.*, for the observation with the smallest depth (≈ 0.131), direction minimizing projection depth has coordinates $(0.863, -0.505)^\top$, which is very close to the second principal vector (of normal data) which equals $(0.872, -0.489)^\top$ as we have generated the anomalies. These vector's constituents may indicate which variables and to which degree are "responsible" for the abnormality (of this most abnormal) observation.

Let us now take a look at the two following visualizations of these directions for the entire data set. First, for each $\boldsymbol{x}_i \in \boldsymbol{X}_{tr}$, let $\mathcal{S}^{d-1} \ni \boldsymbol{u}^*_{\boldsymbol{x}_i} = \mathrm{argmin}_{\boldsymbol{u}\in\mathcal{S}^{d-1}} D^{\mathrm{prj}}(\boldsymbol{x}_i|\boldsymbol{X}_{tr})$ be the direction that minimizes (12) (we will call it *optimal direction* for $\boldsymbol{x}_i$). Further, consider the sequence of observations' projections on this optimal—for observation $\boldsymbol{x}_i$—direction:

$$\left(\boldsymbol{u}^{*\top}_{\boldsymbol{x}_i}\boldsymbol{x}_{1_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}} - m_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}, \boldsymbol{u}^{*\top}_{\boldsymbol{x}_i}\boldsymbol{x}_{2_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}} - m_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}, \ldots, \boldsymbol{u}^{*\top}_{\boldsymbol{x}_i}\boldsymbol{x}_{n_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}} - m_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}\right),$$



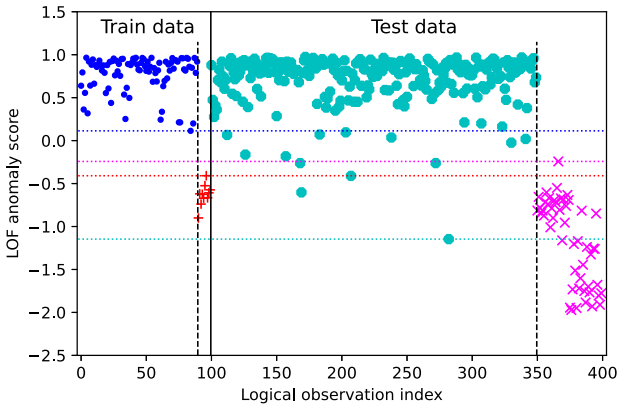
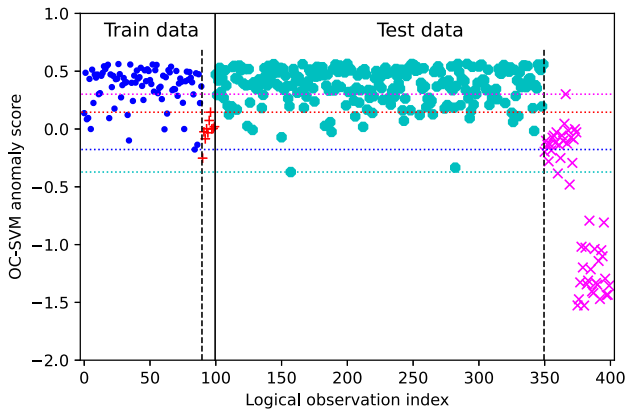


Fig. 10 Anomaly detection using LOF, for the data from Sect. 5.2. Left: Plot of training (blue dots for normal observations and red pluses for anomalies) and testing (bigger cyan dots for normal observations and magenta crosses for anomalies) sets and anomaly score for the entire space $\mathbb{R}^2$, with scale. Right: Using same markers, for both training and testing sets, anomaly score (on ordinate) by LOF. Separated by vertical lines from left to right are normal observations (indices on abscissa 1-90) and anomalies (91-100) of the training set, normal observations (101-350) and anomalies (351-400) of the test set



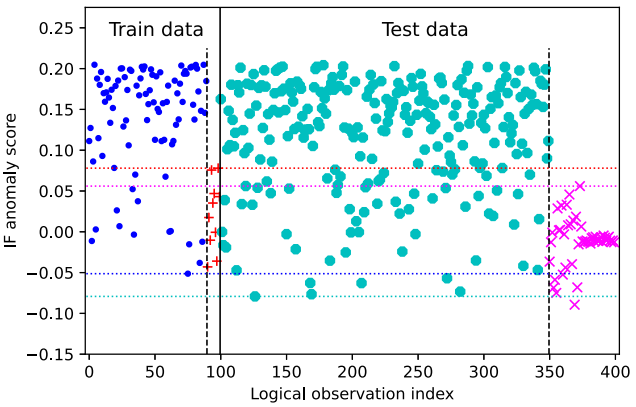


Fig. 11 For both training and testing sets, anomaly score (on ordinate) by OC-SVM (left) and IF (right). Separated by vertical lines from left to right are normal observations (indices on abscissa 1-90, blue dots) and anomalies (91-100, red pluses) of the training set, normal observations (101-350, bigger cyan dots) and anomalies (351-400, magenta crosses) of the test set

where $m_{\boldsymbol{u}^*_{\boldsymbol{x}_i}} = \min_{j\in\{1,\ldots,n\}} {\boldsymbol{u}^*_{\boldsymbol{x}_i}}^\top \boldsymbol{x}_{j_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}}$ and ${\boldsymbol{u}^*_{\boldsymbol{x}_i}}^\top \boldsymbol{x}_{1_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}} \leq {\boldsymbol{u}^*_{\boldsymbol{x}_i}}^\top \boldsymbol{x}_{2_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}} \leq \ldots \leq {\boldsymbol{u}^*_{\boldsymbol{x}_i}}^\top \boldsymbol{x}_{n_{\boldsymbol{u}^*_{\boldsymbol{x}_i}}}$. For each observation of the same data set, sorted in depth increasing order (on the ordinate), these sequences are plotted in Fig. 14 (left), with the projection of observation itself $\boldsymbol{x}_i$ (for $i = 1, \ldots, n$) in bold on each optimal direction.

This first plot confirms correctness of calculation of the (projection) depth as we see the projections of points $\boldsymbol{x}_i$ (red points) are more outside for lower depth values. Further, we observe apartness of the 10 anomalies: Depth-sorted indices of their optimal directions are 1–10 on ordinate since they possess lowest depth.

Second plot (Fig. 14, right) depicts heat map of scalar products between all pairs of optimal directions, with their indices again sorted increasing due to depth values. One observes very high values for optimal directions of the ten anomalies, which indicates that all these ten optimal directions are very close to each other: Thus, the ten anomalies lie in the same direction from normal data, and with high probability in a cluster (which is the case, since they have close depth values).

It is important to note that Fig. 14 can serve as a visualization-explanation tool for space of any dimension $\mathbb{R}^d$.

Another setting is explored in Fig. 16 in higher dimension and with a contamination represented (only for visualization) in Fig. 15. The same setting found in Sect. 4 is used, but we increased dimensions to 20 and added a second anomaly cluster (yellow "×") to the first (red "+"). They are placed in the two last principal components obtained from normal data.

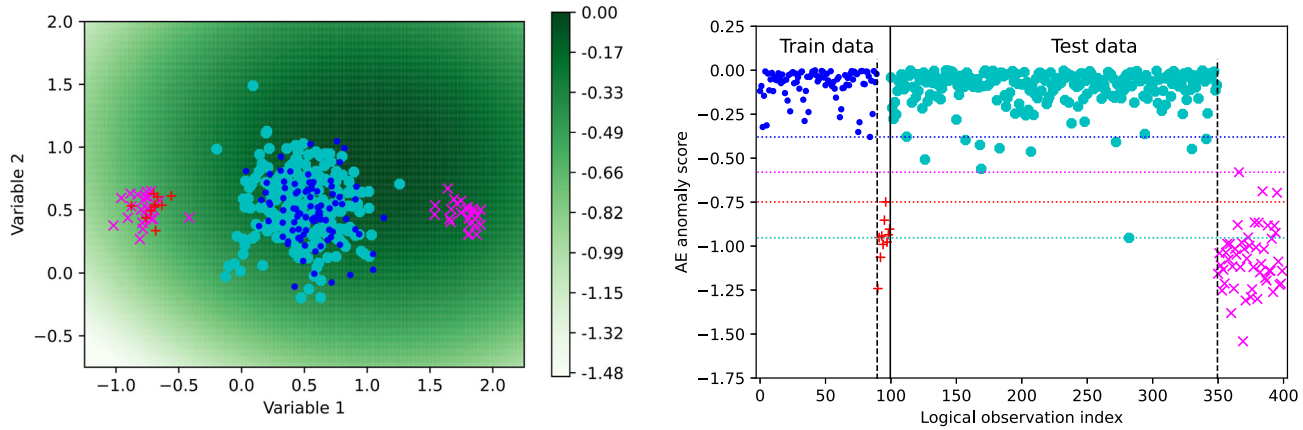


Fig. 12 Anomaly detection using autoencoder, for the data from Sect. 5.2. Left: Plot of training (blue dots for normal observations and red pluses for anomalies) and testing (bigger cyan dots for normal observations and magenta crosses for anomalies) sets and anomaly score for the entire space $\mathbb{R}^2$, with scale. Right: Using same markers, for both training and testing sets, anomaly score (on ordinate) by autoencoder. Separated by vertical lines from left to right are normal observations (indices on abscissa 1-90) and anomalies (91-100) of the training set, normal observations (101-350) and anomalies (351-400) of the test set

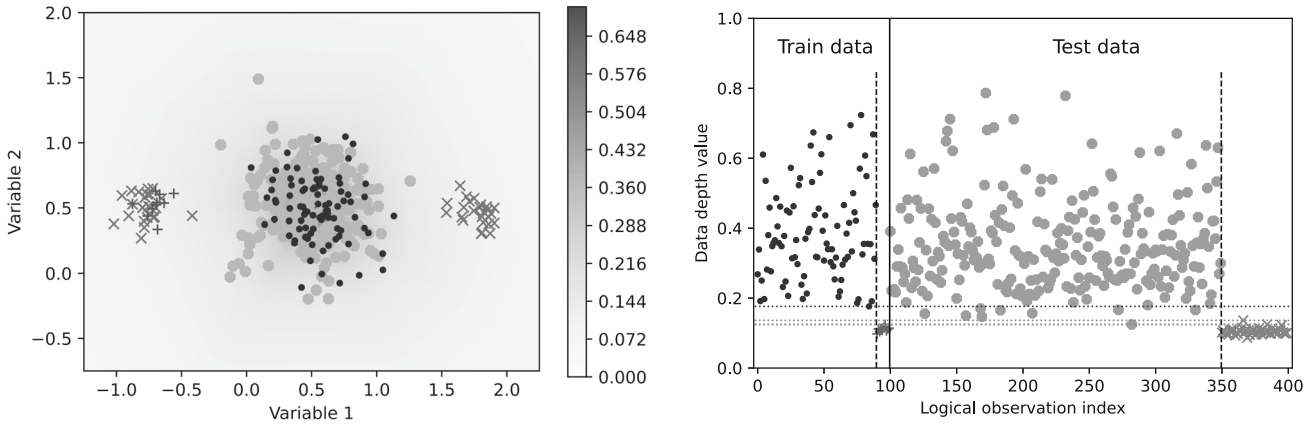


Fig. 13 Anomaly detection using data depth, for the data from Sect. 5.2. Left: Plot of training (blue dots for normal observations and red pluses for anomalies) and testing (bigger cyan dots for normal observations and magenta crosses for anomalies) sets and anomaly score for the entire space $\mathbb{R}^2$, with scale. Right: Using same markers, for both training and testing sets, anomaly score (on ordinate) by data depth. Separated by vertical lines from left to right are normal observations (indices on abscissa 1-90) and anomalies (91-100) of the training set, normal observations (101-350) and anomalies (351-400) of the test set

Finally we placed some isolated anomalies (magenta "▲") sampled from elliptical Cauchy distribution and placed at a distance between normal points and both anomaly clusters.

It is shown on left Fig. 16, the distinction between the three sources of abnormal points (red and yellow clusters and magenta isolated ones) on different level of abnormality (measured by depth in ascending order). But now, the additional plot on right Fig. 16 shows only the two clusters with optimal directions near each other. This actually gives us additional information about nature of these anomalies.

The reader is additionally referred to a recent work on explanation of anomalies using the optimal direction [75].

6 Computational tractability

In this section, we shall explore computational properties of data depths from two points of view: numerical and statistical. As we have seen in Sect. 3.1, data depth notions possessing most attractive (for anomaly detection purposes) properties (e.g., affine invariance, robustness) demand computational time exponentially increasing with space dimension for exact calculation. This impedes their application in dimensions of order 50 or 10 and, depending on data set size, even 5. The proposed solution is using approximate computation to reasonably sacrifice precision for dramatic decrease in computational time. The goal here is to develop more intu-

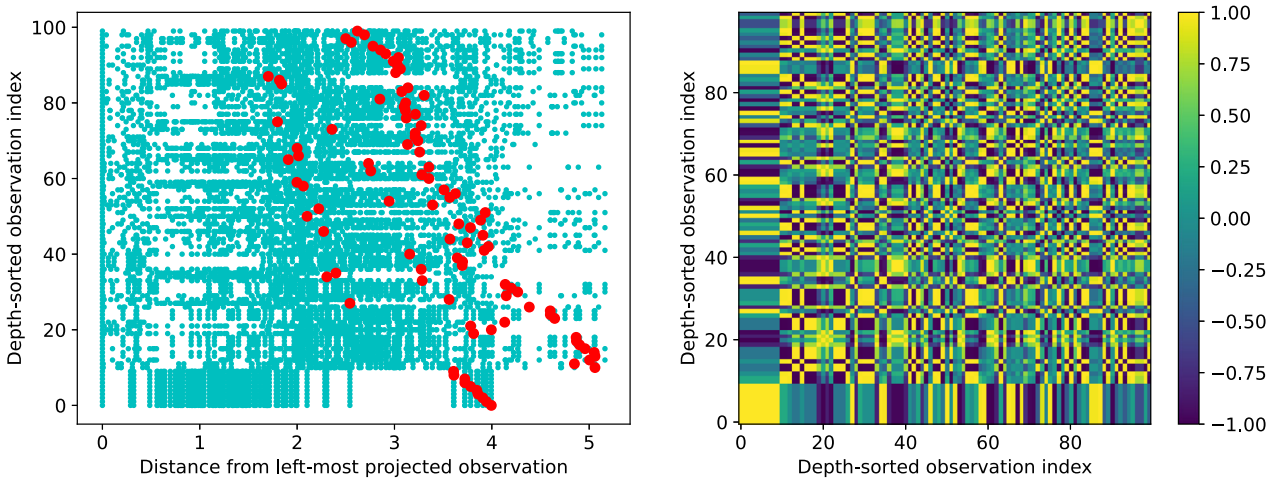


Fig. 14 Left: Projections of data set's observations (cyan dots) from first example of Sect. 4 on directions minimizing (projection) depth (optimal directions) for each of the observations of the data set; value of left-most projection is subtracted from each projection. Projections of the observations for which the depth was computed are red points. The projections are ordered by increasing depth value of the observations for which the depth was computed. Right: Scalar products between optimal directions (for projection depth) for each observation (ordered by their depth values) of the data set from first example of Sect. 4

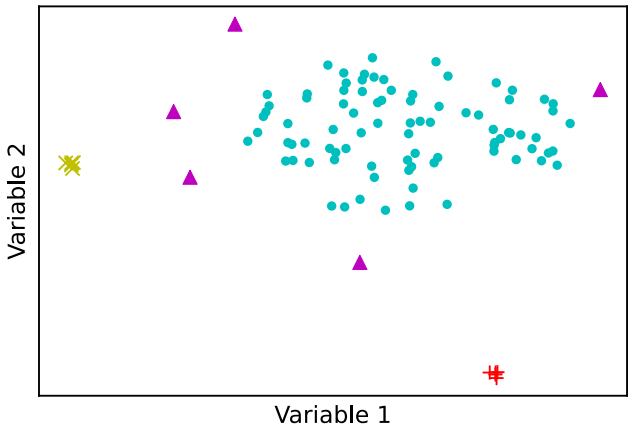


Fig. 15 2−dimensions visualization of the setting used in Fig. 16

ition about the trade-off between computational time and precision of anomaly detection. As we shall see, even if data depth is computed only approximately, this can be sufficient to identify maximum (and up to all, depending on setting) of abnormal observations.

The following distributional setting shall be used in the rest of this section: data set $\boldsymbol{X}_{tr}$ consisting of $n = 1000$ observations and containing 5% of anomalies. Normal data are generated from multivariate Gaussian distribution centered in the origin with Toeplitz covariance matrix. Anomalies are drawn from multivariate Gaussian distribution with covariance matrix $\boldsymbol{I}_{d\times d}/10$ and located 1.25 Mahalanobis distances from the origin in direction of the smallest principal vector of normal data. This is a relatively difficult setting requiring depth computation precision and not allowing to identify all anomalies for higher dimensions.

First, based on [18], let us explore the numerical aspect of approximation, i.e., how precision of anomaly detection depends on time (represented by the number of used directions). For $d = 10, 50$, we measure (16) over 50 repetitions and visualize it in boxplots of Fig. 17, approximating projection depth (12) using *random search* (RS), *refined random search* (RRS), and *Nelder-Mead* (NM) algorithms, with the last two containing elements of optimization. Thus, this experiment shall also illustrate advantage of optimization—involving approximation over purely random one, with the last one suffering from curse of dimension as it has been shown by [52]. As we can see from Fig. 17 (left), depth-based anomaly detection rule copes with the task for $d = 10$ perfectly even with small number of directions when using RRS approximation (e.g., with 200 directions depth calculation for all 1000 points of one data set took less than 20 seconds on a single core of Apple M1 Max chip). RS algorithm improves with growing number of directions (for 5000 directions the same calculation took up to 500 seconds).

When increasing dimension to $d = 50$, the task of anomaly detection becomes difficult to tackle since due to growing distances the cluster of abnormal observations becomes unrecognizable. On the other hand this allows to benchmark the two algorithms along the entire abscissa, see Fig. 17 (right). It is noteworthy that such situations are not rare in practice and still beneficial: E.g., when searching for inconsistencies in enterprise's transactions (the operation to be done by hand on up to millions records) narrowing down

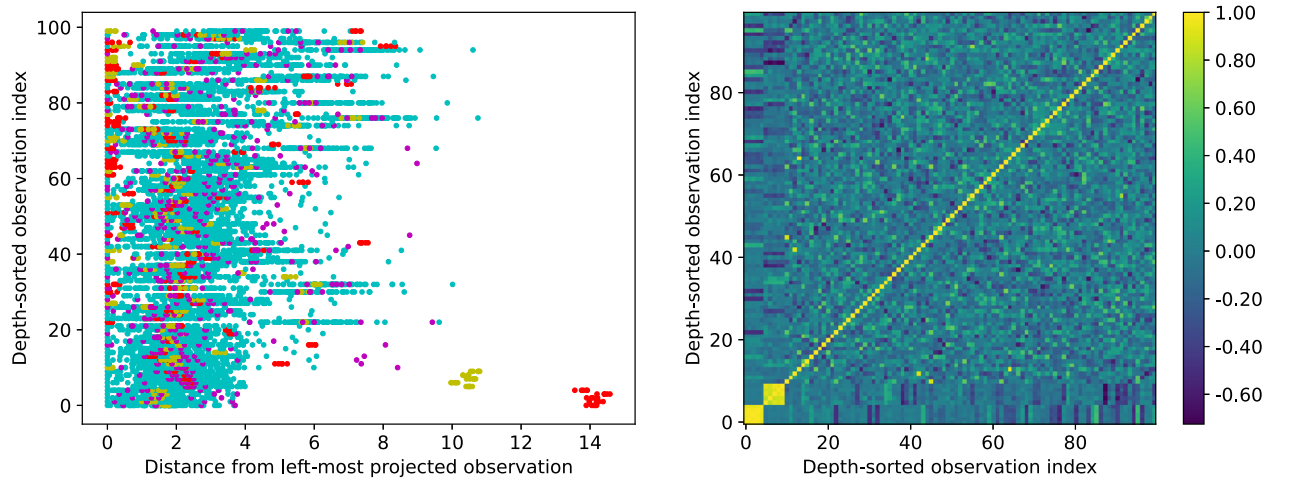


Fig. 16 Example with $d = 20$, 2 clusters, and isolated anomalies

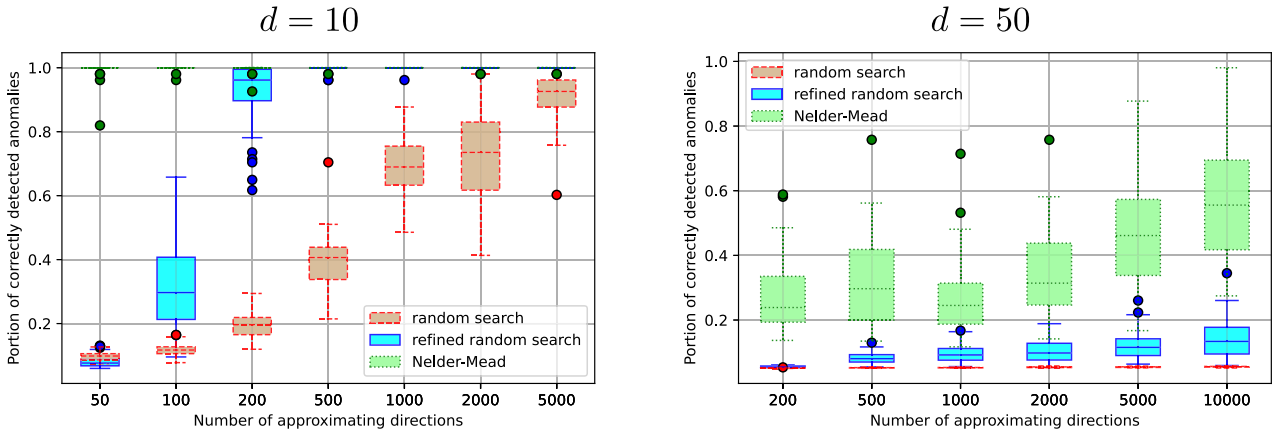


Fig. 17 Boxplots for measure (16) over 50 repetitions for the distribution from Sect. 6 (for $n = 1000$ observations containing 5% of anomalies) for random search, refined random search, and Nelder-Mead approximation algorithms from [18]. Left: in $\mathbb{R}^{10}$. Right: in $\mathbb{R}^{50}$.

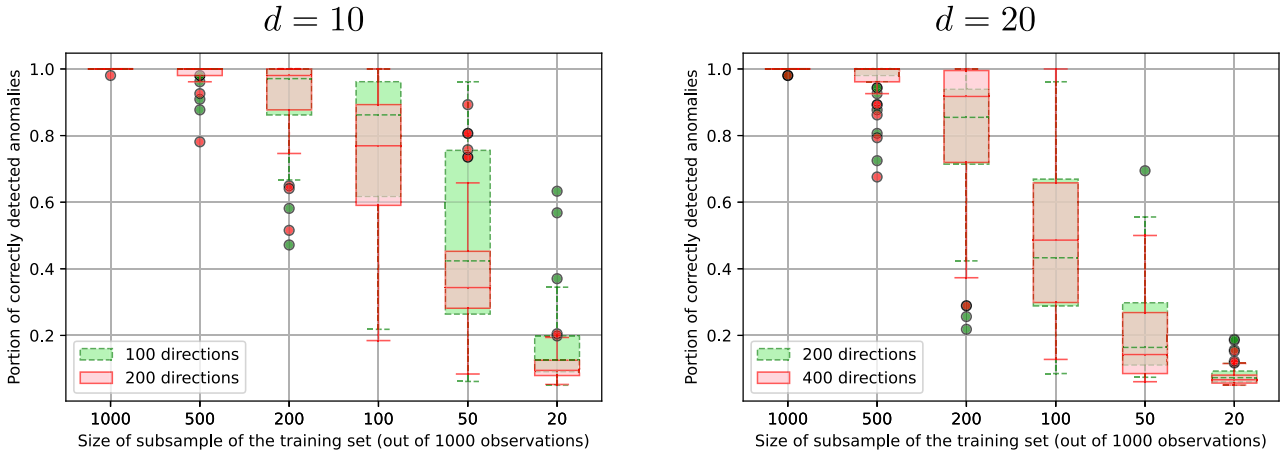


Fig. 18 Boxplots for measure (16) over 50 repetitions for the distribution from Sect. 6 (for $n = 1000$ observations containing 5% of anomalies) for Nelder-Mead approximation algorithm from [18]. For two numbers of approximating directions (linearly increasing with dimension), the sub-sample of $\boldsymbol{X}_{tr}$ with respect to which the depth of each observation $\in \boldsymbol{X}_{tr}$ is computed is gradually reduced. Left: in $\mathbb{R}^{10}$. Right: in $\mathbb{R}^{20}$.

Table 2 Performances on real data sets for AE (MSE and L1 losses), Deep SVDD, IF, LOF, OCSVM, PD, CBLOF, COPOD, HBOS, VAE, MO_GAAL, AnoGAN, and LUNAR. The metric used is precision (also called positive predictive value) between 0 and 100%. When size is -, only 10 000 observations were sampled

Name	Size	Dimension	% Anomaly	AE L1	AE MSE	DSVDD	IF	LOF	OCSVM	PD	CBLOF	COPOD	HBOS	VAE	MO_GAAL	AnoGAN	LUNAR
ALOI	-	27	3.04	10.1	11.0	9.2	5.5	**21.1**	9.2	6.4	12.8	5.5	9.2	10.1	8.3	12.8	**21.1**
annthyroid	7200	6	7.42	43.8	39.8	62.5	60.2	58.0	23.3	**77.8**	44.3	52.3	42.0	50.6	18.8	37.5	50.0
breastw	683	9	34.99	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
cardio	1831	21	9.61	**100.0**	**100.0**	24.1	**100.0**	43.1	65.5	93.1	69.0	93.1	74.1	**100.0**	44.8	65.5	31.0
Cardiotocography	2114	21	22.04	92.9	**93.5**	69.5	86.4	78.6	73.4	86.4	80.5	84.4	61.0	84.4	74.0	82.5	66.2
celeba	-	39	2.24	**33.8**	**33.8**	10.0	23.8	3.8	5.0	23.8	21.2	28.8	30.0	32.5	12.5	0.0	8.8
cover	-	10	0.96	28.6	35.7	14.3	25.0	**78.6**	0.0	10.7	0.0	17.9	10.7	32.1	3.6	0.0	21.4
donors	-	10	5.93	**54.7**	15.3	22.1	37.9	5.8	4.2	41.1	43.7	48.9	14.7	25.3	25.8	0.5	52.6
fault	1941	27	34.67	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
fraud	-	29	0.17	66.7	66.7	**100.0**	66.7	**100.0**	50.0	66.7	**100.0**	83.3	66.7	66.7	0.0	**100.0**	83.3
glass	214	7	4.21	33.3	33.3	0.0	33.3	33.3	0.0	33.3	33.3	33.3	33.3	0.0	0.0	0.0	**66.7**
Hepatitis	80	19	16.25	**75.0**	**75.0**	50.0	50.0	50.0	50.0	25.0	25.0	**75.0**	**75.0**	**75.0**	**75.0**	50.0	50.0
http	-	3	0.39	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	0.0	92.9	**100.0**	**100.0**	**100.0**	0.0
Ionosphere	351	32	35.90	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
landsat	6435	36	20.71	44.3	44.5	77.0	57.5	64.1	55.7	65.0	74.5	52.0	73.2	68.6	62.3	**80.2**	67.0
letter	1600	32	6.25	24.2	33.3	39.4	27.3	57.6	**78.8**	33.3	57.6	30.3	21.2	39.4	12.1	15.2	75.8
Lymphography	148	18	4.05	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	50.0	**100.0**	**100.0**	**100.0**	**100.0**	0.0	**100.0**	**100.0**
magic.gamma	-	10	35.16	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	
mammography	-	6	2.32	55.0	51.2	43.8	50.0	47.5	31.2	38.8	36.2	**65.0**	33.8	53.8	40.0	50.0	46.2

Table 2 continued

Name	Size	Dimension	% Anomaly	AE L1	AE MSE	DSVDD	IF	LOF	OCSVM	PD	CBLOF	COPOD	HBOS	VAE	MO_GAAL	AnoGAN	LUNAR
optdigits	5216	64	2.88	0.0	0.0	0.0	12.0	6.0	8.0	2.0	4.0	8.0	**44.0**	0.0	10.0	10.0	2.0
PageBlocks	5393	10	9.46	86.3	88.7	36.3	85.1	73.2	28.0	75.0	33.9	85.1	85.7	**89.9**	26.2	32.1	62.5
pendigits	6870	16	2.27	57.7	34.6	11.5	53.8	7.7	32.7	15.4	**67.3**	40.4	51.9	55.8	0.0	23.1	11.5
Pima	768	8	34.90	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
satellite	6435	36	31.64	95.1	93.9	97.9	97.5	93.2	98.1	98.2	98.5	**99.4**	99.1	97.5	96.1	94.5	98.8
satimage-2	5803	36	1.22	91.3	95.7	4.3	95.7	8.7	0.0	**100.0**	95.7	87.0	87.0	91.3	0.0	87.0	30.4
shuttle	-	9	7.15	98.0	98.4	35.1	99.2	44.4	45.6	**100.0**	98.4	99.2	96.4	98.4	24.2	99.2	48.0
skin	-	3	20.75	95.4	89.0	76.6	**100.0**	60.3	41.7	**100.0**	84.8	69.3	93.8	90.7	56.7	76.4	95.4
smtp	-	3	0.03	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
SpamBase	4207	57	39.91	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
Stamps	340	9	9.12	**100.0**	90.0	50.0	**100.0**	50.0	40.0	80.0	50.0	**100.0**	**100.0**	**100.0**	20.0	80.0	80.0
thyroid	3772	6	2.47	80.6	67.7	0.0	**93.5**	77.4	51.6	**93.5**	61.3	67.7	77.4	90.3	3.2	22.6	64.5
vertebral	240	6	12.50	20.0	20.0	20.0	30.0	40.0	30.0	10.0	30.0	20.0	40.0	50.0	**90.0**	10.0	20.0
vowels	1456	12	3.43	29.4	29.4	23.5	47.1	82.4	47.1	52.9	64.7	17.6	23.5	35.3	0.0	17.6	**94.1**
Waveform	3443	21	2.90	12.1	12.1	15.2	18.2	**48.5**	39.4	12.1	42.4	12.1	9.1	12.1	3.0	6.1	24.2
WBC	223	9	4.48	**100.0**	**100.0**	33.3	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
WDBC	367	30	2.72	**100.0**	**100.0**	33.3	**100.0**	**100.0**	0.0	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	0.0	**100.0**	66.7
Wilt	4819	5	5.33	1.2	2.4	12.9	2.4	38.8	21.2	**54.1**	8.2	1.2	8.2	4.7	21.2	4.7	9.4
wine	129	13	7.75	**100.0**	**100.0**	33.3	**100.0**	**100.0**	**100.0**	**100.0**	66.7	**100.0**	66.7	66.7	0.0	**100.0**	0.0
WPBC	198	33	23.74	75.0	75.0	75.0	81.2	81.2	68.8	**87.5**	81.2	**87.5**	**87.5**	68.8	62.5	75.0	75.0
yeast	1484	8	34.16	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
Top Score	.	.	.	18	17	11	17	17	13	19	14	17	15	16	12	15	13
Top Unique	.	.	.	1	1	0	0	2	1	4	1	2	1	1	1	1	2

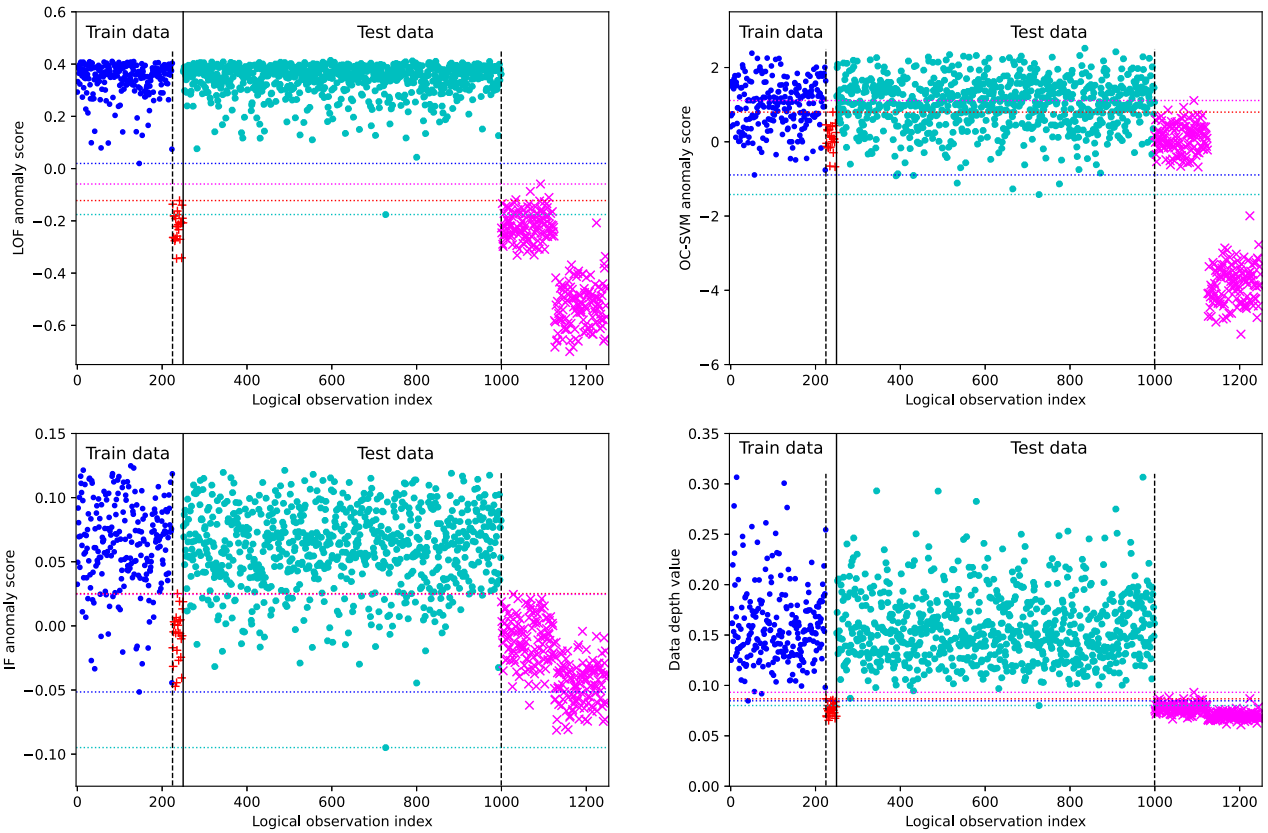


Fig. 19 Experiment with $d = 10$

search of few anomalies to hundreds transactions facilitates dramatically the audit work. The difference between approximation methods becomes even more visible, while purely random RS does not cope with the task, optimization NM starts to get interesting results considering the size of d.

Second, we study the statistical aspect and try to increase computation speed by sub-sampling. Here, we computed depth of each point of $\boldsymbol{X}_{tr}$ with respect to its random subset, and repeat the experiment 50 times, employing the Nelder-Mead algorithm as described in [18]. For two different numbers of directions in each case, for $d = 10, 20$, (16) is indicated in Fig. 18. Clearly, when reducing the size of the data set used the quality of anomaly detection decreases, while this does not happen immediately suggesting a possible compromise. Indeed, in the problems where the data set is too large, sub-sampling is not expected to substantially reduce the quality, whereas for a small data set computation should be fast enough even without sub-sampling.

While continuing the example with projection depth, as experimental evidence of [18] suggests, the results from the first experiment here are (accounting for robustness and anomalies' configuration) extendable to the multitude of depths satisfying the projection property [21]. The conclusions of the second experiment are even more general.

7 Real data sets

To shed more light on performance of the depth-based anomaly detection approach in a general context, we consider 40 real-world data sets [59, knownastheODDSlibrary] in a comparative study. We thus contrast data depth with autoencoder (using both L_1 and L_2 losses), deep SVDD, IF, OC-SVM, LOF as well as CBLOF [29], COPOD [46], HBOS [25], VAE [34], MO_GAAL [38], AnoGAN [68], and LUNAR [26]. For each data set, we use a stratified split (to preserve contamination rate) with proportion 66%/33% for train and test data, respectively. We assess the methods' performance using the true contamination rate $\boldsymbol{c} \in [0, 1]$ to compute the precision (or positive predictive rate) and indicate the portion of observations correctly labeled as abnormal among the proportion $\boldsymbol{c}$ of highest anomaly score (or lowest depending on the method). The results are indicated in Table 2.

With the goal of this article being not to illustrate superior performance of depth-based anomaly detection, but to indicate its place in machine learning, the results are satisfactory. Data depth score is best in 21 out of 40 data sets, ranging in dimension from 3 to 64 and having up to 10 000 observations.

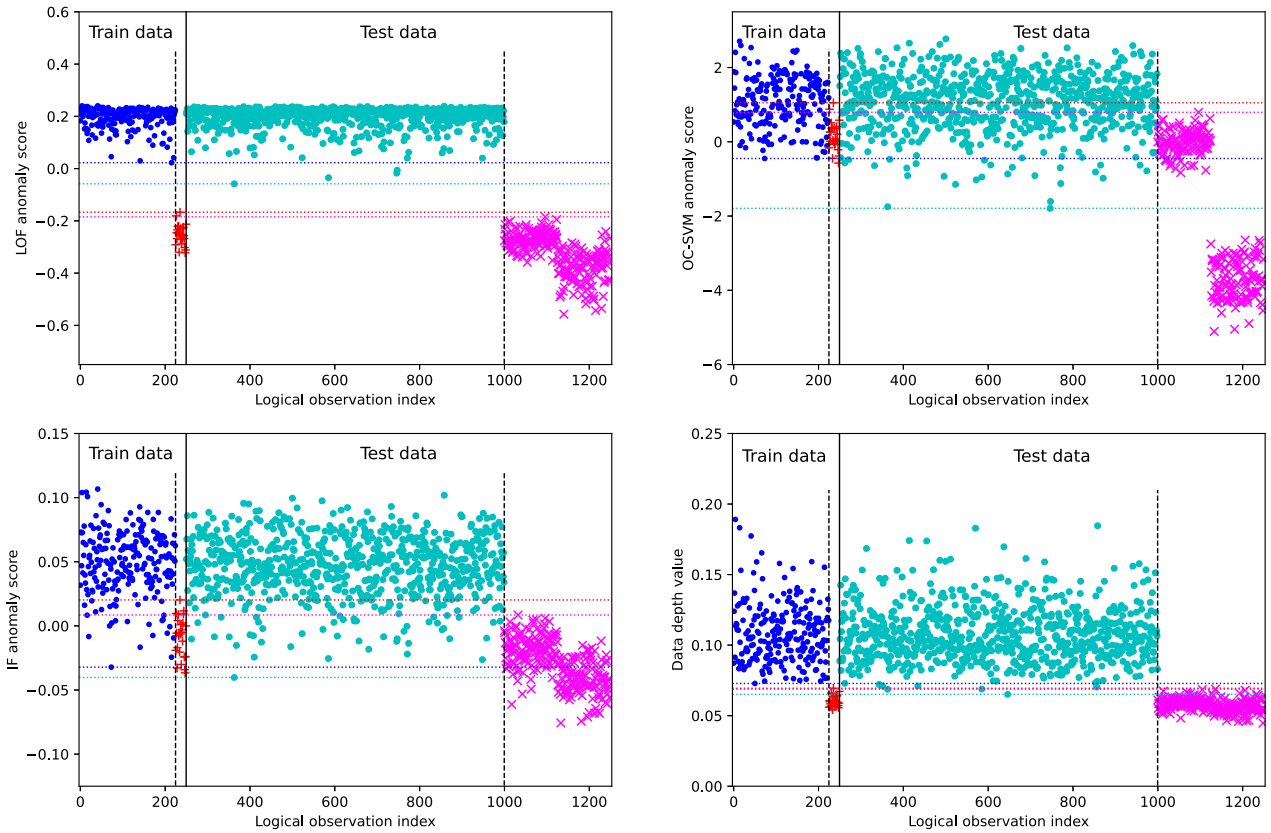


Fig. 20 Experiment with $d = 20$

Furthermore, for such data sets as "annthyroid," "Wilt," or "WPBC," it substantially outperforms the state of the art.

8 Outlook

Data depth has undergone substantial theoretical developments during a few recent decades and possesses attractive properties, such as nonparametricity, robustness, affine invariance, *etc*. After having been computationally reachless for long time, more recently efficient exact and approximate computation methods have been proposed.

When applying data depth for anomaly detection, several aspects should be taken into account: These were addressed in sections of the present article. Section 3 analyzes computational aspects of data depth. In Sect. 4, on an example of several depth notions, importance of robustness, affine invariance, non-vanishing beyond data's convex hull is underlined. In certain (not uncommon) settings highlighted in Sect. 5, data depth competitively compares with such widely used anomaly detection tools as autoencoder, local outlier factor, one-class support vector machine, and isolation forest, additionally providing explainability. Experiments of Sect. 6 illustrate that—with reasonably limited resources—anomaly detection can be performed in relatively high dimensions when properly choosing the degree of approximation and computed data depth with respect to a sub-sample only.

Even though the topic itself deserves a separate work, it is noteworthy that experimental evidence from the current article can be useful for anomaly detection in functional data. When employing (a selected notion of) functional data depth, anomaly detection rule (3) can be readily applied to functions (or time series), given the threshold is properly chosen. Furthermore, in case if a notion of integrated functional depth is used, for each argument value, multivariate depth is to be considered. Already from computational perspective, the ideas of Sect. 6 can be applied as well.

It is important to conclude with the following *disclaimer*: The presented in the current article examples were designed to illustrate advantages of depth-based anomaly detection and their immediate generalization can be limited.

It is further noteworthy to underline two—still limiting—aspects of the typical data depth approach to anomaly detection: scalability and multimodality:

First, while recently numerous advances took place regarding computing (in particular approximately) a number of depth notions (see, e.g., [18] for depths satisfying the projection property or [70, 71] for the halfspace depth), its

scalability to reach the state of the art of data compared to such contemporary tools as neural networks should still be improved. While data sets containing thousands of points in dimensions up to $d \leq 20$ can be fairly treated, computations for data with millions of observations are slow even in approximate mode. Likewise, when dimension $d \geq 50$, at least task-sufficient precision of depth computation should still be (im)proved.

Second, when handling data stemming from a probability distribution consisting of several multivariate modes, typical multivariate data depth might not necessarily reflect everywhere the geometry of the data. Monge-Kantorovich depth by [9, 27] as well as localized depth by [54] though possesses mild artifacts. Clustering before applying the typical multivariate data depth can be seen as an alternative.

The source codes (in `Python`) of all examples and experiments contained in the current article can be downloaded from the author's website.

Appendix A Appendix A

Appendix A.1 Experiment 5.2 with higher dimension

For both training and testing sets with $d = 10$ and $d = 20$, anomaly score (on ordinate) by LOF (top left), OCSVM (top right), IF (bottom left), and data depth (bottom right). Separated by vertical lines from left to right are normal observations (indices on abscissa 1-225, blue dots), and anomalies (226-250, red pluses) of the training set, normal observations (251-750, bigger cyan dots) and anomalies (751-1000, magenta crosses) of the test set.

Acknowledgements The authors are grateful for the insightful remarks of Prof. Florence d'Alché-Buc on their manuscript.

Funding Open access funding provided by Télécom Paris. The authors further gratefully acknowledge the support of the Young Researcher Grant of the French National Agency for Research (ANR JCJC 2021) in category Artificial Intelligence registered under the number ANR-21-CE23-0029-01 and the support of the CIFRE grant number 2021/1739.

References

1. Azzalini, A., Capitanio, A.: Statistical applications of the multivariate skew normal distribution. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **61**(3), 579–602 (1999)
2. Avella-Medina, M., Battey, H.S., Fan, J., Li, Q.: Robust estimation of high-dimensional covariance and precision matrices. Biometrika **105**(2), 271–284 (2018)
3. Barnett, V.: The ordering of multivariate data. J. R. Stat. Soc. Ser. A **139**(3), 318–344 (1976)
4. Barber, C.B., Dobkin, D.P., Huhdanpaa, H.: The quickhull algorithm for convex hulls. ACM Trans. Math. Softw. **22**(4), 469–483 (1996)
5. Breunig, M.M., Kriegel, H.-P., Ng, R.T., Sander, J.: Lof: Identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, vol. 29, pp. 93–104 (2000)
6. Bazovkin, P., Mosler, K.: An exact algorithm for weighted-mean trimmed regions in any dimension. J. Stat. Softw. **47**(13), 1–29 (2012)
7. Cascos, I.: The expected convex hull trimmed regions of a sample. Comput. Stat. **22**, 557–569 (2007)
8. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surv. **41**(3), 1–58 (2009)
9. Chernozhukov, V., Galichon, A., Hallin, M., Henry, M.: Monge-Kantorovich depth, quantiles, ranks and signs. Ann. Stat. **45**(1), 223–256 (2017)
10. Cai, T., Liu, W.: Adaptive thresholding for sparse covariance matrix estimation. J. Am. Stat. Assoc. **106**, 494 (2011)
11. Chen, B., Ting, K.M., Washio, T., Haffari, G.: Half-space mass: a maximally robust and efficient data depth method. Mach. Learn. **100**, 677–699 (2015)
12. Donoho, D.L., Gasko, M.: Breakdown properties of location estimates based on halfspace depth and projected outlyingness. Ann. Stat. **20**(4), 1803–1827 (1992)
13. Dyckerhoff, R., Koshevoy, G., Mosler, K.: Zonoid data depth: Theory and computation. In: Prat, A. (ed.) COMPSTAT '96 - Proceedings in Computational Statistics, pp. 235–240. Physica-Verlag, Heidelberg (1996)
14. Dyckerhoff, R., Ley, C., Paindaveine, D.: Depth-based runs test for bivariate central symmetry. Ann. Inst. Stat. Math. **67**, 917–941 (2015)
15. Dyckerhoff, R., Mosler, K.: Weighted-mean trimming of multivariate data. J. Multivar. Anal. **102**(3), 405–421 (2011)
16. Dyckerhoff, R., Mosler, K.: Weighted-mean regions of a probability distribution. Stat. Prob. Lett. **82**, 318–325 (2012)
17. Dyckerhoff, R., Mozharovskyi, P.: Exact computation of the halfspace depth. Comput. Stat. Data Anal. **98**, 19–30 (2016)
18. Dyckerhoff, R., Mozharovskyi, P., Nagy, S.: Approximate computation of projection depths. Comput. Stat. Data Anal. **157**, 107166 (2021)
19. Donoho, D.: Breakdown properties of multivariate location estimators. PhD thesis, Harvard University (1982)
20. Dyckerhoff, R.: Datentiefe: Begriff, berechnung, tests. Fakultät für Wirtschafts-und Sozialwissenschaften, Universität zu Köln, Mimeo (2002)
21. Dyckerhoff, R.: Data depths satisfying the projection property. Allgemeines Statistisches Archiv **88**(2), 163–190 (2004)
22. Eddy, W.F.: Graphics for the multivariate two-sample problem: comment. J. Am. Stat. Assoc. **76**(374), 287–289 (1981)
23. Einmahl, J.H.J., Li, J., Liu, R.Y.: Bridging centrality and extremity: refining empirical data depth using extreme value statistics. Ann. Stat. **43**(6), 2738–2765 (2015)

24. Fang, K.-T., Kotz, S., Ng, K.-W.: Symmetric Multivariate and Related Distributions (Monographs on Statistics and Applied Probability). Chapman and Hall, New York (1990)
25. Goldstein, M., Dengel, A.: Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. KI-2012: poster and demo track 1, 59–63 (2012)
26. Goodge, A., Hooi, B., Ng, S.-K., Ng, W.S.: Lunar: Unifying local outlier detection methods via graph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 6737–6745 (2022)
27. Hallin, M., Barrio, E., Cuesta-Albertos, J., Matrán, C.: Distribution and quantile functions, ranks and signs in dimension d: a measure transportation approach. Ann. Stat. **49**(2), 1139–1165 (2021)
28. Hariri, S., Kind, M.C., Brunner, R.J.: Extended isolation forest. IEEE Trans. Knowl. Data Eng. **33**(4), 1479–1489 (2021)
29. He, Z., Xu, X., Deng, S.: Discovering cluster-based local outliers. Patt. Recognit. Lett. **24**(9–10), 1641–1650 (2003)
30. Jörnsten, R.: Clustering and classification based on the l_1 data depth. J. Multivar. Anal. **90**(1), 67–89 (2004)
31. Johnson, D.S., Preparata, F.P.: The densest hemisphere problem. Theor. Comput. Sci. **6**, 93–107 (1978)
32. Karmarkar, N.: A new polynomial-time algorithm for linear programming. Combinatorica **4**(4), 373–395 (1984)
33. Koshevoy, G., Mosler, K.: Zonoid trimming for multivariate distributions. Ann. Stat. **25**(5), 1998–2017 (1997)
34. Kingma, D.P., Welling, M., et al.: Auto-encoding variational bayes. Banff, Canada (2013)
35. Li, J., Cuesta-Albertos, J.A., Liu, R.Y.: DD-classifier: nonparametric classification procedure based on DD-plot. J. Am. Stat. Assoc. **107**, 737–753 (2012)
36. Lafaye De Micheaux, P., Mozharovskyi, P., Vimond, M.: Depth for curve data and applications. J. Am. Stat. Assoc. **116**(536), 1881–1897 (2022)
37. Liu, R.Y.: On a notion of data depth based on random simplices. Ann. Stat. **18**(1), 405–414 (1990)
38. Liu, Y., Li, Z., Zhou, C., Jiang, Y., Sun, J., Wang, M., He, X.: Generative adversarial active learning for unsupervised outlier detection. IEEE Trans. Knowl. Data Eng. **32**(8), 1517–1528 (2019)
39. Liu, Z., Modarres, R.: Lens data depth and median. Journal of Nonparametric Statistics **23**(4), 1063–1074 (2011)
40. Lange, T., Mosler, K., Mozharovskyi, P.: Fast nonparametric classification based on data depth. Stat. Papers **55**(1), 49–69 (2014)
41. Liu, X., Mosler, K., Mozharovskyi, P.: Fast computation of Tukey trimmed regions and median in dimension $p>2$. J. Comput. Gr. Stat. **28**(3), 682–697 (2019)
42. Liu, R.Y., Parelius, J.M., Singh, K.: Multivariate analysis by data depth: descriptive statistics, graphics and inference. Ann. Stat. **27**(3), 783–858 (1999)
43. Lopuhaa, H.P., Rousseeuw, P.J.: Breakdown points of affine equivariant estimators of multivariate location and covariance matrices. Ann. Stat. **19**(1), 229–248 (1991)
44. Liu, F.T., Ting, K.M., Zhou, Z.-H.: Isolation forest. In: Proceedings of the Eighth IEEE International Conference on Data Mining, pp. 413–422 (2008). IEEE Computer Society
45. Liu, X., Zuo, Y.: Computing projection depth and its associated estimators. Stat. Comput. **24**(1), 51–63 (2014)
46. Li, Z., Zhao, Y., Botta, N., Ionescu, C., Hu, X.: Copod: copula-based outlier detection. In: 2020 IEEE International Conference on Data Mining (ICDM), pp. 1118–1123 (2020). IEEE
47. Mahalanobis, P.C.: On the generalized distance in statistics. Proc. Nat. Inst. Sci.(India) **2**(1), 49–55 (1936)
48. Mosler, K., Mozharovskyi, P.: Choosing among notions of multivariate depth statistics. Stat. Sci. **37**(3), 348–368 (2022)
49. Mosler, K.: Depth statistics. In: Becker, C., Fried, R., Kuhnt, S. (eds.) Robustness and Complex Data Structures: Festschrift in Honour of Ursula Gather, pp. 17–34. Springer, Berlin (2013)
50. Mirzargar, M., Whitaker, R.T., Kirby, R.M.: Curve boxplot: generalization of boxplot for ensembles of curves. IEEE Trans. V. Comput. Gr. **20**(12), 2654–2663 (2014)
51. Nagy, S., Dvořák, J.: Illumination depth. J. Comput. Gr. Stat. **30**(1), 78–90 (2021)
52. Nagy, S., Dyckerhoff, R., Mozharovskyi, P.: Uniform convergence rates for the approximated halfspace and projection depth. Electron. J. Stat. **14**(2), 3939–3975 (2020)
53. Oja, H.: Descriptive statistics for multivariate distributions. Stat. Probab. Lett. **1**(6), 327–332 (1983)
54. Paindaveine, D., Bever, G.V.: From depth to local depth: a focus on centrality. J. Am. Stat. Assoc. **108**(503), 1105–1119 (2013)
55. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: an imperative style, high-performance deep learning library. Adv. Neural Inf. Process. Syst. **32**, 8026 (2019)
56. Pokotylo, O., Mozharovskyi, P., Dyckerhoff, R.: Depth and depth-based classification with R-package ddalpha. J. Stat. Softw. **91**(5), 1–46 (2019)
57. Pokotylo, O., Mozharovskyi, P., Dyckerhoff, R., Nagy, S.: Ddalpha: Depth-Based Classification and Calculation of Data Depth. (2022). R package version 1.3.13. https://CRAN.R-project.org/package=ddalpha
58. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M.: Duchesnay: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
59. Rayana, S.: Outlier Detection DataSets (ODDS) Library (2016). https://odds.cs.stonybrook.edu
60. Ramsay, K., Durocher, S., Leblanc, A.: Integrated renk-weighted depth. J. Multivar. Anal. **173**, 51–69 (2019)
61. Rousseeuw, P.J., Leroy, A.M.: Robust Regression and Outlier Detection. John Wiley & Sons, New York (1987)
62. Rousseeuw, P.J., Ruts, I., Tukey, J.W.: The bagplot: a bivariate boxplot. Am. Stat. **53**(4), 382–387 (1999)
63. Rousseeuw, P.J., Van Driessen, K.: A fast algorithm for the minimum covariance determinant estimator. Technometrics **41**, 212–223 (1999)
64. Serfling, R.: A depth function and a scale curve based on spatial quantiles. In: Dodge, Y. (ed.) Statistical Data Analysis Based on the L_1-Norm and Related Methods. Statistics for Industry and Technology book series (SIT), Birkhäuser, Basel (2002)
65. Serfling, R.: Equivariance and invariance properties of multivariate quantile and related functions, and the role of standardisation. J. Nonparametric Stat. **22**(7), 915–936 (2010)
66. Singh, K.: A notion of majority depth. Techinical report, Rutgers University, Department of Statistics (1991)
67. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**(7), 1443–1471 (2001)
68. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Information Processing in Medical Imaging", pp. 146–157. Springer, (2017)
69. Stahel, W.A.: Robust estimation: Infinitesimal optimality and covariance matrix estimators. PhD thesis, Eidgenössische Technische Hochschule Zürich (1981)
70. She, Y., Tang, S., Liu, J.: On generalization and computation of tukey's depth: Part i. Journal of Data Science, Statistics, and Visualisation **2**(1) (2022)
71. She, Y., Tang, S., Liu, J.: On generalization and computation of tukey's depth: Part ii. Journal of Data Science, Statistics, and Visualisation **2**(2) (2022)
72. Sakurada, M., Yairi, T.: Anomaly detection using autoencoders with nonlinear dimensionality reduction. In: Proceedings of the

MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, pp. 4–11 (2014)
73. Tax, D.M.J., Duin, R.P.W.: Support vector data description. Mach. Learn. **54**, 45–66 (2004)
74. Tukey, J.W.: Mathematics and the picturing of data. In: James, R.D. (ed.) Proceedings of the International Congress of Mathematicians, vol. 2, pp. 523–531. Canadian Mathematical Congress, Vancouver (1975)
75. Valla, R., Mozharovskyi, P., d'Alché-Buc, F.: Anomaly component analysis. arXiv preprint arXiv:2312.16139 (2023)
76. Yang, M., Modarres, R.: β-skeleton depth functions and medians. Commun. Stat. - Theor. Methods **47**(20), 5127–5143 (2018)
77. Zuo, Y., Serfling, R.: General notions of statistical depth function. Ann. Stat. **28**(2), 461–482 (2000)
78. Zuo, Y.: Projection-based depth functions and associated medians. Ann. Stat. **31**(5), 1460–1490 (2003)