Local Outlier Detection with Interpretation

Xuan Hong Dang[1], Barbora Micenková[1], Ira Assent[1], and Raymond T. Ng[2]

[1] Aarhus University, Denmark
{dang,barbora,ira}@cs.au.dk
[2] University of British Columbia, Canada
rng@cs.ubc.ca

Abstract. Outlier detection aims at searching for a small set of objects that are inconsistent or considerably deviating from other objects in a dataset. Existing research focuses on outlier identification while omitting the equally important problem of outlier interpretation. This paper presents a novel method named LODI to address both problems at the same time. In LODI, we develop an approach that explores the quadratic entropy to adaptively select a set of neighboring instances, and a learning method to seek an optimal subspace in which an outlier is maximally separated from its neighbors. We show that this learning task can be solved via the matrix eigen-decomposition and its solution contains essential information to reveal features that are most important to interpret the exceptional properties of outliers. We demonstrate the appealing performance of LODI via a number of synthetic and real world datasets and compare its outlier detection rates against state-of-the-art algorithms.

1 Introduction

Data mining aims at searching for novel and actionable knowledge from data. Mining techniques can generally be divided into four main categories: clustering, classification, frequent pattern mining and anomalies detection. Unlike the first three main tasks whose objective is to find patterns that characterize for majority data, the fourth one aims at finding patterns that only represent the minority data. Such kind of patterns usually do not fit well to the mechanisms that have generated the data and are often referred to as outliers, anomalies or surprising patterns. Mining that sort of rare patterns therefore poses novel issues and challenges. Yet, they are of interest and particularly important in a number of real world applications ranging from bioinformatics [28], direct marketing [18], to various types of fraud detection [4].

Outlying patterns may be divided into two types: global and local outliers. A global outlier is an object which has a significantly large distance to its k-th nearest neighbor (usually greater than a global threshold) whereas a local outlier has a distance to its k-th neighbor that is large *relatively to* the average distance of its neighbors to their own k-th nearest neighbors [6]. Although it is also possible to create a ranking of global outliers (and select the top outliers), it is noted in [6,3] that the notion of local outliers remains more general than that of

H. Blockeel et al. (Eds.): ECML PKDD 2013, Part III, LNAI 8190, pp. 304–320, 2013.

global outliers and, usually, a global outlier is also a local one but not vice versa, making the methods to discover local outliers typically more computationally expensive. In this study, our objective is to focus on mining and interpreting *local* outliers.

Although there is a large number of techniques for discovering global and local anomalous patterns [29,26], most attempts focus solely on the aspect of outlier *identification*, ignoring the equally important problem of outlier *interpretation*. For many application domains, especially those with data described by a large number of features, the description/intepretation of outliers is essential. As such, an outlier should be explained clearly and compactly, like a subset of features, that shows its exceptionality. This knowledge obviously assists the user to evaluate the validity of the uncovered outliers. More importantly, it offers him/her a facility to gain insights into why an outlier is exceptionally different from other regular objects. To our best knowledge, the study developed in [13] is the only attempt that directly addresses this issue, yet for global outliers but not for the more challenging patterns of local outliers (shortly reviewed in Section 2).

In this work, we introduce a novel approach that achieves both objectives of local outlier detection and interpretation at the same time. We propose a technique relying on the information theoretic measure of entropy to select an appropriate set of neighboring objects of an outlier candidate. Unlike most existing methods which often select the k closest objects as neighbors, our proposed technique goes further by requiring strong interconnections (or high entropy) amongst all neighboring members. This helps to remove irrelevant objects that can be nearby outliers or the objects coming from other distributions, and thus ensures all remaining objects to be truly normal inliers generated by the same distribution (illustrated via examples later). This characteristic is crucial since the statistical properties of the neighborhood play an essential role in our explanation of the outlierness. We then develop a method, whose solution firmly relies on the matrix eigen-decomposition, to learn an optimal one-dimensional subspace in which an outlier is most distinguishable from its neighboring set. The basic idea behind this approach is to consider the local outlier detection problem as a binary classification and thus ensure that a single dimension is sufficient to discriminate an outlier from its vicinity. The induced dimension is in essence a linear combination of the original features and thus contains all intrinsic information to reveal which original features are the most important to explain outliers. A visualization associated with the outlier interpretation is provided for intuitive understanding. Our explanation form not only shows the relevant features but also ranks objects according to their outlierness.

2 Related Work

Studies in outlier detection can generally be divided into two categories stemming from: (i) statistics and (ii) data mining. In the statistical approach, most methods assume that the observed data are governed by some statistical process to which a standard probability distribution (e.g., Binomial, Gaussian, Poisson

etc.) with appropriate parameters can be fitted to. An object is identified as an outlier based on how unlikely it could have been generated by that distribution [2]. Data mining techniques, on the other hand, attempt to avoid model assumptions; relying on the concepts of distance and density, as stated earlier. For most distance-based methods [12,27], two parameters called distance d and data fraction p are required. Following that, an outlier has at least fraction p of all instances farther than d from it [12]. As both d and p are parameters defined over the entire data, methods based on distance can only find *global* outliers. Techniques relying on density, in contrast, attempt to seek *local* outliers, whose outlying degrees ("local outlier factor"—LOF) are defined w.r.t. their neighborhoods rather than the entire dataset [3,6]. There are several recent studies that attempt to find outliers in spaces with reduced dimensionality. Some of them consider every single dimension [10] or every combination of two dimensions [7] as the reduced dimensional subspaces, others [19,11] go further in refining the number of relevant subspaces. While the work in [19] makes assumptions that outliers can only exist in subspaces with non-uniform distributions, the method developed in [11] assumes that outliers only appear in subspaces showing high dependencies amongst their related dimensions. These studies, exploring either subspace projections [19,11] or subspace samplings [18,10,7], appear to be appropriate for the purpose of outlier interpretation. Nonetheless, as the outlier score of an object is aggregated from multiple spaces, it remains unclear which subspace should be selected to interpret its outlierness property. In addition, the number of explored subspaces for every object should be large in order to obtain good outlier ranking results. These techniques are hence closer to outlier ensembles [25] rather than outlier interpretation. The recent SOD method [14] pursues a slightly different approach in which it seeks an axis-parallel hyperplane (w.r.t. an object) as one spanned by the attributes with the highest data variances. The anomaly degree of the object is thus computed in the space orthogonal to this hyperplane. This technique also adopts an approach based on the shared neighbors between two objects to measure their similarity, which alleviates the almost equi-distance effect among all instances in a high dimensional space and thus can achieve better selection for neighboring sets. SOD was demonstrated to be effective in uncovering outliers that deviate from the most variance attributes yet it seems somewhat limited in searching outliers having extreme values in such directions. A similar approach is adopted in [16] where the subspace can be arbitrarily oriented (not only axis-parallel) and a form of outlier characterization based on vector directions have been proposed. ABOD [15] pursues a different approach where variance of angles among objects is taken into account to compute outlierness, making the method suitable for high dimensional data. In terms of outlier detection, we provide experimental comparisons with state-of-the-art algorithms in Section 4.

3 Our Approach

In this work, we consider $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ a dataset of N instances and each $\mathbf{x}_i \in \mathcal{X}$ is represented as a vector in a D-dimensional space. Each dimension

represents a feature f_1 to f_D. We aim for an algorithm that can rank the objects in $\mathcal{X}$ w.r.t. their outlier degrees with the most outlying objects on the top. Having been queried for M outliers in $\mathcal{X}$, the algorithm returns the top M outliers and for a threshold $\lambda \in (0, 1)$ (to be clear in Section 3.3), each outlier $\mathbf{x}_i$ is associated with a small set of features $\{f_1^{(\mathbf{x}_i)}, \ldots, f_d^{(\mathbf{x}_i)}\}, d \ll D$ explaining why the object is exceptional. The value of d may vary across different outliers. In addition, $f_1^{(\mathbf{x}_i)}, \ldots, f_d^{(\mathbf{x}_i)}$ are also weighted according to the degree to which they contribute to discriminate $\mathbf{x}_i$ as an outlier.

3.1 Neighboring Set Selection

Compared to global anomalous patterns, mining local outliers is generally harder and more challenging since it has to further deal with the problem of locally different densities in the data distribution. An outlier is considered anomalous if its density value is significantly different from the average density computed from the neighboring objects. The anomalous property of an outlier is thus decided by the local density distribution rather than the global knowledge derived from the entire distribution. For most existing studies [3,14], the set of k nearest neighboring objects (kNNs) is used. Nonetheless, this approach has not been thoroughly investigated and may be misleading for outlier explanation. The difficulty comes from the fact that identifying a proper value of k is not only a non-trivial task [22,3] but such a set of k closest neighbors might also contain nearby outliers or inliers from several distributions, which both strongly affect the statistical properties of the neighboring set. To give an illustration, we borrow a very popular data set from subspace clustering [23,17] which includes four clusters in a 3-dimensional space with 20 outliers randomly added as shown in Figure 1(a). Each cluster is only visible in 2-dimensional subspace [17] and each outlier is considered anomalous w.r.t. its closest cluster. Now taking the outlier $\mathbf{o}_1$ as an example, regardless of how small k is selected, other nearby outliers such as $\mathbf{o}_2, \mathbf{o}_3$ or $\mathbf{o}_4$ are included in its neighbors since they are amongst the closest objects (see Figure 1(a)). On the other hand, increasing k to include more inliers from the upper distribution can alleviate the effect of these outliers on the $\mathbf{o}_1$'s anomalous property. Unfortunately, such a large setting also comprises instances from the lower right distribution as shown in Figure 1(b). To cope with these issues, our objective is to ensure that all $\mathbf{o}_1$'s neighbors are truly inliers coming from a single closest distribution and thus $\mathbf{o}_1$ can be considered as its local outlier. Our proposed approach to handle this issue stems from the well-studied concept of entropy in information theory. The technique is adaptive by not fixing the number of neighboring inliers k. Instead, we only use k as a lower bound to ensure that the number of final nearby inliers is no less than k.

In information theory, entropy is used to measure the uncertainty (or disorder) of a stochastic event. Following the definition by Shannon, the entropy of that event is defined by $H(X) = -\int p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x}$, of which X is the stochastic event or more specifically, a continuous random variable, and $p(\mathbf{x})$ is its corresponding probability distribution. If the entropy of X is large, its purity is

low, or equivalently, X's uncertainty is high. Therefore, it is natural to exploit entropy for our task of selecting neighboring inliers. Intuitively, for the entropy computed with respect to this set, we would expect its value to be small in order to infer that objects within the set are all similar (i.e., high purity) and thus there is a high possibility that they are being generated from the same statistical mechanism or distribution. Nonetheless, computing entropy in Shannon's definition is not an easy task since it requires $p(\mathbf{x})$ to be known. We thus utilize a more general form, the Renyi entropy [24], which enables a straightforward computation. Mathematically, given α as an order, Renyi entropy is defined as:

$$H_{R_\alpha}(X) = \frac{1}{1-\alpha} \log \int p(\mathbf{x})^\alpha d\mathbf{x}, \text{ for } \alpha > 0,\ \alpha \neq 1. \tag{1}$$

in which Shannon entropy is a special case when α is approaching 1 (i.e., $\lim_{\alpha \to 1} H_{R_\alpha}(X) = H(X)$ [24]). However, in order to ensure the practical computation and impose no assumption regarding the probability distribution $p(\mathbf{x})$, we select $\alpha = 2$, yielding the quadratic form of entropy, and use the non-parametric Parzen window technique to estimate $p(\mathbf{x})$. More specifically, let us denote $R(\mathbf{o}) = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_s\}$ as the initial set of nearest neighboring instances closest to an outlier candidate $\mathbf{o}$. Following the Parzen window technique, we approximate $p(\mathbf{x})$ w.r.t. this set via the sum of kernels placed at each $\{\mathbf{x}_i\}_{i=1}^s$ and it follows that:

$$p(\mathbf{x}) = s^{-1} \sum_i G(\mathbf{x} - \mathbf{x}_i, \sigma^2) \tag{2}$$

where $G(\mathbf{x} - \mathbf{x}_i, \sigma^2) = (2\pi\sigma)^{-D/2} \exp\left\{-\frac{||\mathbf{x}-\mathbf{x}_i||^2}{2\sigma^2}\right\}$ is the Gaussian in the D-dimensional space used as the kernel function. In combination with setting $\alpha = 2$, this leads to a direct computation of the local quadratic Renyi entropy as follows:

$$\begin{aligned} QE(R(\mathbf{o})) &= -\ln \int \left(\frac{1}{s}\sum_{i=1}^{s} G(\mathbf{x} - \mathbf{x}_i, \sigma^2)\right)\left(\frac{1}{s}\sum_{j=1}^{s} G(\mathbf{x} - \mathbf{x}_j, \sigma^2)\right) \\ &= -\ln \frac{1}{s^2} \sum_i^s \sum_j^s G(\mathbf{x}_i - \mathbf{x}_j, 2\sigma^2) \end{aligned} \tag{3}$$

Notice that, unlike Shannon entropy, the above computation removes burden of the computation of the numerical integration due to the advantages of the quadratic form and the convolution property of two Gaussian functions. Essentially, the sum within the logarithm operation can be interpreted as the local information potential. Each term in the summation satisfies the positivity and increases as the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ decreases, very much analogous to the potential energy between two physical particles. As such, our objective of minimizing the entropy is equivalent to maximizing the information potential within the neighboring set. The higher the information potential of the set is, the more similar the elements within the set are.

Having the way to capture the local quadratic entropy, an appropriate set of nearest neighbors can be selected adaptively as follows. We begin by setting

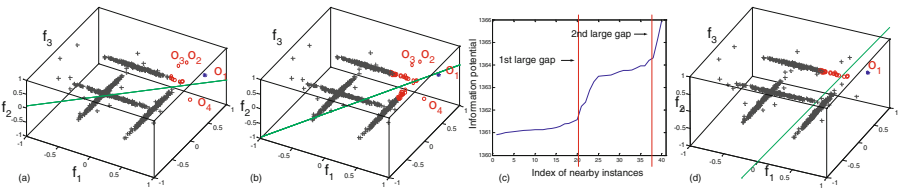


Fig. 1. Neighbors selection: object under consideration is $\mathbf{o}_1$ and circle points are its nearest neighbors (figures are best visualized in colors).

the number of initial nearest neighbors to s (in our experiments, a setting of $s = 2k$ often gives good results), and aim to find an optimal subset of no less than k instances with maximum local information potential. Obviously, a naive way to find such an optimal set may require computing all $\sum_{i=k}^{s} \binom{s}{i}$ possible combinations, which is prohibitively expensive. We thus make use of an heuristic approach to select such a subset. Specifically, removing an object from the neighboring set will lead to a decrement in the total information potential. Those instances resulting in the most decrement are important ones whereas those causing least decrement tend to be irrelevant for the neighboring set. With the latter objects, their potential energy is minor as they loosely interact with the rest of neighboring objects and thus excluding them makes the neighboring set less uncertain or more pure. These objects in fact can be either other outliers or part of nearby distributions. Our method thus ranks the total information potential left in the increasing order and removes objects behind the first significant gap as long as the number of remaining instances is no less than k. A significant gap is defined to have a value larger than the average gap.

For illustration, we plot in Figure 1(c) the total information potential left (ordered increasingly) after excluding each of nearest neighboring objects represented in Figure 1(b). One may observe that there are two remarkably large gaps in the plot (noted by the red vertical lines in Figure 1(c)), which indeed reflect the nature of local distribution surrounding outlier $\mathbf{o}_1$. In particular, the first large gap signifies the information decrement in removing instances from the lower right distribution whereas the second one corresponds to the removal of nearby outliers. By excluding these irrelevant objects from the set of $\mathbf{o}_1$'s neighboring instances, the remaining ones are true inliers coming from the same and closest distribution shown as blue points in Figure 1(d).

3.2 Anomaly Degree Computation

Given a way to compute the neighboring (or "reference") set above, we develop a method to calculate the anomaly degree for each object in the dataset $\mathcal{X}$. Essentially, directly computing that measure in the original multidimensional data space is often less reliable since many features may not be relevant for the task of identifying local outliers. We thus exploit an approach of a *local* dimensionality reduction. For the remaining discussion, let us denote $\mathbf{o}$ as an

outlier candidate under consideration, $R(\mathbf{o})$ as its neighboring inliers found by the entropy-based technique presented in the previous section and $\mathbf{R}$ as the matrix form of $R(\mathbf{o})$. Each neighboring inlier $\mathbf{x}_i \in R(\mathbf{o})$ corresponds to a column in $\mathbf{R}$ and together with $\mathbf{o}$, they are all vectors in the $\mathbb{R}^D$ space.

Essentially, we view the local outlier detection as a binary classification problem in the sense that the outlier candidate $\mathbf{o}$ should be distinguished from its neighbors $R(\mathbf{o})$. By dimensionality reduction, this objective is equivalent to the objective of learning an optimal subspace such that $\mathbf{o}$ is maximally separated from every object in $R(\mathbf{o})$. More specifically, $\mathbf{o}$ needs to be strongly deviating from $R(\mathbf{o})$ while at the same time $R(\mathbf{o})$ shows high density or low variance in that induced subspace. Following this approach, we denote the optimal 1-dimensional subspace as $\mathbf{w}$ and in order to achieve our goal, data variance is obviously an important statistical measure to explore. Toward this goal, we define the first variance of all neighboring objects projected onto $\mathbf{w}$ as follows:

$$Var(R(\mathbf{o})) = \mathbf{w}^T \left(\mathbf{R} - \mathbf{R}\mathbf{e}\mathbf{e}^T/N_{\mathbf{o}}\right) \left(\mathbf{R} - \mathbf{R}\mathbf{e}\mathbf{e}^T/N_{\mathbf{o}}\right)^T \mathbf{w} = \mathbf{w}^T \mathbf{A}\mathbf{A}^T \mathbf{w} \quad (4)$$

where $\mathbf{A} = \left(\mathbf{R} - \mathbf{R}\mathbf{e}\mathbf{e}^T/N_{\mathbf{o}}\right)$, $N_{\mathbf{o}}$ is the number of neighboring instances in $R(\mathbf{o})$ and $\mathbf{e}$ is the vector with all entries equal to 1.

Another important statistic in our approach is the distance between $\mathbf{o}$ and every object in $R(\mathbf{o})$. This resembles an *average proximity* in a hierarchical clustering technique[9] where all pairwise data distances are taken into account. Compared to the two extremes of using minimum or maximum distance, this measure often shows better stability. We hence formulate their variance in the projected dimension $\mathbf{w}$ as the following quantity:

$$D_{(\mathbf{o},R(\mathbf{o}))} = \mathbf{w}^T \left(\sum (\mathbf{o} - \mathbf{x}_i)(\mathbf{o} - \mathbf{x}_i)^T\right) \mathbf{w} = \mathbf{w}^T \mathbf{B}\mathbf{B}^T \mathbf{w}, \quad (5)$$

where $\mathbf{x}_i \in R(\mathbf{o})$ and $\mathbf{B}$ is defined as the matrix whose each column corresponds to a vector $(\mathbf{o} - \mathbf{x}_i)$. Intuitively, in order to achieve the goal of optimally distinguishing $\mathbf{o}$ from its neighboring reference inliers, we want to learn a direction for $\mathbf{w}$ such that the variance of $R(\mathbf{o})$ projected onto it is minimized whereas the variance between $\mathbf{o}$ and $R(\mathbf{o})$ also projected on that direction is maximized. One possible way to do that is to form an objective function resembling Rayleigh's quotient which maximizes the ratio between $D_{(\mathbf{o},R(\mathbf{o}))}$ and $R(\mathbf{o})$ as follows:

$$\arg\max_{\mathbf{w}} J(\mathbf{w}) = \frac{D_{(\mathbf{o},R(\mathbf{o}))}}{Var(R(\mathbf{o}))} = \frac{\mathbf{w}^T \mathbf{B}\mathbf{B}^T \mathbf{w}}{\mathbf{w}^T \mathbf{A}\mathbf{A}^T \mathbf{w}}. \quad (6)$$

It is obvious that setting the derivative of $J(\mathbf{w})$ w.r.t. $\mathbf{w}$ equal to 0 results in $(\mathbf{w}^T \mathbf{B}\mathbf{B}^T \mathbf{w})\mathbf{A}\mathbf{A}^T \mathbf{w} = (\mathbf{w}^T \mathbf{A}\mathbf{A}^T \mathbf{w})\mathbf{B}\mathbf{B}^T \mathbf{w}$, which is in essence equivalent to solving the following generalized eigensystem:

$$J(\mathbf{w})\mathbf{A}\mathbf{A}^T \mathbf{w} = \mathbf{B}\mathbf{B}^T \mathbf{w}. \quad (7)$$

In dealing with this objective function, note that $\mathbf{A}\mathbf{A}^T$, though symmetric, may not be full rank as the number of neighbors can be smaller than the number

of features. This matrix is thus not directly invertible. Moreover, the size of $\mathbf{A}\mathbf{A}^T$ can be large and quadratically proportional to the feature number which makes its eigendecomposition computationally expensive. To alleviate this problem, we propose to approximate $\mathbf{A}$ via its singular value decomposition and consequently $\mathbf{w}$ can be computed using the pseudo inversion of $\mathbf{A}\mathbf{A}^T$.

Specifically, since $\mathbf{A}$ in general is a rectangular matrix, it can be decomposed into three matrices $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ of which $\mathbf{U}$ and $\mathbf{V}$ are matrices whose columns are $\mathbf{A}$'s left and right singular eigenvectors and $\mathbf{\Sigma}$ is the diagonal matrix of its singular values. In essence, as our objective is to compute matrix inversion, we remove singular values which are very close to 0 and approximate $\mathbf{A}$ by its set of leading singular values and vectors. More concretely, we estimate $\mathbf{A} = \sum_\ell \mathbf{u}_\ell \sigma_\ell \mathbf{v}_\ell^T$ such that the sum over keeping singular values σ_ℓ's explains for 95% (as demonstrated in our experimental studies) of the total values in the diagonal matrix $\mathbf{\Sigma}$. Additionally, we compute $\mathbf{U}$ via the eigendecomposition of $\mathbf{A}^T\mathbf{A}$ which has a lower dimensionality. Particularly, we can see that:

$$\mathbf{A}^T\mathbf{A} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T. \tag{8}$$

Then, taking the square of both sides and pre-multiplying with $\mathbf{\Sigma}^{-1}\mathbf{V}^T$ and post-multiplying with $\mathbf{V}\mathbf{\Sigma}^{-1}$, we obtain:

$$\begin{aligned} \mathbf{\Sigma}^{-1}\mathbf{V}^T\mathbf{A}^T(\mathbf{A}\mathbf{A}^T)\mathbf{A}\mathbf{V}\mathbf{\Sigma}^{-1} &= \mathbf{\Sigma}^2 \\ \mathbf{U}^T(\mathbf{A}\mathbf{A}^T)\mathbf{U} &= \mathbf{\Sigma}^2. \end{aligned} \tag{9}$$

This implies that columns in $\mathbf{U}$ are the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and they can be computed via the eigenvectors of the smaller matrix $\mathbf{A}^T\mathbf{A}$, i.e., $\mathbf{U} = \mathbf{A}\mathbf{V}\mathbf{\Sigma}^{-1}$. Thus, the final pseudo inversion $(\mathbf{A}\mathbf{A}^T)^\dagger$ can be simply approximated by $\mathbf{U}\mathbf{\Sigma}^{-2}\mathbf{U}^T$. Plugging this value into our objective function in Eq.(7), it is straightforward to see that the optimal direction for $\mathbf{w}$ is the first eigenvector of the matrix $\mathbf{U}\mathbf{\Sigma}^{-2}\mathbf{U}^T\mathbf{B}\mathbf{B}^T$ of which $J(\mathbf{w})$ achieves the maximum value as the largest eigenvalue of this matrix.

Given the optimal direction $\mathbf{w}$ uncovered by the technique developed above, the statistical distance between $\mathbf{o}$ and $R(\mathbf{o})$ can be calculated in terms of the standard deviation as follows:

$$AD(\mathbf{o}) = \max\left\{\sqrt{\frac{(\mathbf{w}^T\mathbf{o} - \sum_i \frac{\mathbf{w}^T\mathbf{x}_i}{N_\mathbf{o}})^2}{Var(\mathbf{w}^T R(\mathbf{o}))}}, \sqrt{Var(\mathbf{w}^T R(\mathbf{o}))}\right\} \tag{10}$$

where the second term in the max operation is added to ensure that the projection of $\mathbf{o}$ is not too close to the center of the projected neighboring instances (calculated in the first term). Notice that unlike most techniques that find multiple subspaces and have to deal with the problem of dimensionality bias [20], our approach naturally avoids this issue since it learns a 1-dimensional subspace and thus directly enables a comparison across objects. Therefore, with the objective of generating an outlier ranking over all objects, the relative difference between the statistical distance of an object $\mathbf{o}$ defined above and that of its neighboring objects is used to define its local anomalous degree:

$$LAD(\mathbf{o}) = AD(\mathbf{o}) \times \left(\sum AD(\mathbf{x}_i)/N_{\mathbf{o}} \right)^{-1}. \quad (11)$$

For this relative outlier measure, it is easy to see that if $\mathbf{o}$ is a regular object embedded in a cluster, its local anomaly degree is close to 1 whereas if it is a true outlier, the value will be greater than 1.

3.3 Outlier Interpretation

In interpreting the anomaly degree of an outlier, it is possible to rely on the correlation between the projected data in $\mathbf{w}$ and those in each of the original dimensions (i.e., $\mathbf{R}$'s rows). Features with highest absolute values can be used to interpret the anomaly degree of $\mathbf{o}$ since values of $\mathbf{o}$ and its referenced objects on these features are correlated to those projected onto $\mathbf{w}$. Nonetheless, this approach requires computing correlations with respect to all original features. A better and more direct approach is to exploit the optimal direction $\mathbf{w}$ directly. Recall that the projection of $R(\mathbf{o})$ over $\mathbf{w}$ is equivalent to the local linear combination of the original features. Consequently, coefficients within the eigenvector $\mathbf{w}$ are truly the weights of the original features. The feature corresponding to the largest absolute coefficient is the most important in determining $\mathbf{o}$ as an outlier. Analogously, the second important feature is the one corresponding to the second $\mathbf{w}$'s largest absolute component and so on. In this way, we are not only able to figure out which original features are crucial in distinguishing $\mathbf{o}$ but also show how important they are via the weights of the corresponding components in $\mathbf{w}$.

Generally, we can provide the user with a parameter λ, whose values are between $(0, 1)$, to control the number of features used to interpret the anomaly degree. We select $\{f_i\}_{i=1}^{d}$ as the set of features that correspond to the top d largest absolute coefficients in $\mathbf{w}$ and s.t. $\sum_{i=1}^{d} |w_i| \geq \lambda \times \sum_{j=1}^{D} |w_j|$. The degree of importance of each respective f_i can be further computed as the ratio $|w_i| / \sum_{j=1}^{D} |w_j|$. An object $\mathbf{o}$ therefore can be interpreted as an outlier in the d-subspace $\{f_1, ..., f_d\}$ with the corresponding feature importance degrees. An illustration is given in Figure 1 where $\mathbf{w}$ is plotted as the green line whose coefficients in the rightmost subgraph are $(0.11, 4.63, 5.12)$ (or in terms of importance degrees $(0.03, \mathbf{0.46}, \mathbf{0.51})$) which obviously reveals $\{f_2, f_3\}$ being two important features to explain $\mathbf{o}_1$ as an outlier. Note that the corresponding values of $\mathbf{w}$ (green lines) in Figures 1(a) and (b) are respectively $(6.12, 5.17, 0.59)$ and $(2.91, 4.72, 2.01)$, which tend to select $\{f_1, f_2\}$ and $\{f_1, f_2, f_3\}$ as the subspaces for $\mathbf{o}_1$ due to the influence of nearby *irrelevant* instances. The advantage of our entropy-based neighbor selection is thus demonstrated here where only the direction of $\mathbf{w}$ in Figures 1(d) is in parallel to the relevant subspace $\{f_2, f_3\}$ (compared to the slant lines of $\mathbf{w}$ shown in Figures 1(a) and (b)).

3.4 Algorithm Complexity

We name our algorithm LODI which stands for Local Outlier Detection with Interpretation and its computation complexity is analyzed as follows. LODI requires the calculation of the neighboring set as well as the local quadratic Renyi entropy. Both these steps take $O(DN \log N)$ with the implementation of

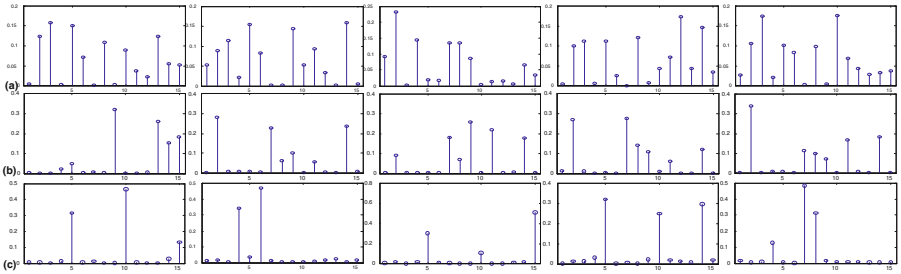


Fig. 2. Feature visualization over 5 top ranking outliers found in Syn1, Syn2 and Syn3 datasets (x- and y-axis are respectively the features' index and importance degree).

the k–d tree data structure. The size of the matrix $\mathbf{A}^T\mathbf{A}$ is $s \times s$ and thus its eigen-decomposition is $O(Ds \log s)$ using the Lanczos method [8]. Similarly, computing the eigen-decomposition of $\mathbf{U}\mathbf{\Sigma}^{-2}\mathbf{U}^T\mathbf{B}\mathbf{B}^T$ amounts to $O(D^2 \log D)$. We compute these steps for all instances to render the outlier ranking list so these computations take $O(DN(s \log s + D \log D))$. The overall complexity is thus at most $O(DN(\log N + s \log s + D \log D))$.

4 Experimental Results

In this section, we provide experimental results on both synthetic and real-world datasets. We compare LODI against the following algorithms: LOF (density-based technique) [3], ABOD (angle-based) [15] and SOD (axis-parallel subspaces) [14]. The last two algorithms are adapted from the ELKI package[1] with some small changes in their output formats. Unless specified differently, we use $k = 20$ as the lower bound for the number of kNNs used in LODI. We also vary the number of neighbors, like $minPts$ in LOF or reference points in SOD, between 10 and 40 and report the best results. With SOD, we further set $\alpha = 0.8$ as recommended by the authors [14].

4.1 Synthetic Data

Data Description. We generate three synthetic datasets Syn1, Syn2 and Syn3, each consists of 50K data instances generated from 10 normal distributions. For each dimension ith of a normal distribution, the center μ_i is randomly selected from $\{10, 20, 30, 40, 50\}$ while variance σ_i is taken from either of two (considerably different) values 10 and 100. Such a setting aims to ensure that if the dimension ith of a distribution takes the large variance, its corresponding generated data will spread out in almost entire data space and thus an outlier close to this distribution can be hard to uncover in the ith dimension due to the strongly overlapping values projected onto this dimension. We set the percentage of the large variance to 40%, 60% and 80%, respectively, to generate Syn1, Syn2 and Syn3. For each dataset, we vary 1%, 2%, 5% and 10% of the whole data as the

[1] http://elki.dbs.ifi.lmu.de/

number of randomly generated outliers within the range of the data space and also vary the dimensionality of each dataset from 15 to 50.

Outlier Explanation. In Figure 2, we provide a feature visualization of the 5 top-ranked outliers returned by our LODI algorithm on the three datasets. For each graph in the figure, the x-axis shows the index of features while the y-axis shows their degree of importance. For the purpose of visualization, we plot the results where three datasets are generated with 5% outlier percentage and in 15 dimensions. The results for higher dimensionalities and other outlier percentages are very similar to those plotted here and thus were omitted to save space (yet, they are summarized in Table 1 and will be soon discussed). As observed from these graphs, the number of relevant features used to explain the anomalous property of each outlier is varied considerably across the three datasets. In Syn1 (Figure 2(a)), each identified outlier can be interpreted in a large number of dimensions since the percentage of the large variance used to generate this dataset is small, only 40%. When increasing the percentage to 60% in Syn2 and to 80% in Syn3 (Figures 2(b-c)), the number of relevant features reduces accordingly. In Syn3 dataset, generally only 3 features are needed to interpret its outliers. These results have been anticipated and quite intuitive since once the number of dimensions with large variance increases, the dimensionality of the subspaces in which an outlier can be found and explained will be narrowed down. This is due to the wide overlapping of outliers and regular objects projected onto these (large variance) dimensions.

For comparison against other techniques, we select the SOD algorithm. Recall that SOD is not directly designed for outlier interpretation, yet its uncovered axis-parallel subspaces might be used to select outliers' relevant features. For these experiments, we select Syn3 dataset and vary the outlier percentage from 1% to 10%, and the data dimensionality from 15 to 50 features. Table 1 reports the average subspace's dimensionality of LODI and SOD computed from their top ranking outliers. The first column shows the outlier percentages while D15, D30 and D50 denote the data dimensionality. We set $\lambda = 0.8$ (see Section 3.3) for LODI and $\alpha = 0.8$ for SOD to ensure their good performance. As one can observe, LODI tends to select subspaces with dimensionality close to the true one whereas the dimensionality of the axis-parallel subspaces in SOD is often higher. For example, at D15, LODI uses around 3 original features to explain each outlier, which is quite consistent with the percentage of 80% of the large variance while it is approximately 8 features for SOD. It can further be observed that the number of relevant features uncovered by SOD also greatly varies, which is indicated by the high standard deviation. Additionally, it tends to increase as the percentage of outliers increases. In contrast, our method performs better and the relevant subspace dimensionality is less sensitive to the variation of the outlier percentages as well as to the number of original features.

Outlier Detection. For comparison of outlier detection rates, we further include the angle-based ABOD and the density-based LOF techniques. The receiver operating characteristic (ROC) is used to evaluate the performance of all algorithms. It was observed that all methods performed quite competitively in

Table 1. Average dimensionality of the subspaces selected for outlier explanation in LODI and SOD in Syn3 dataset (values after ± are standard deviations).

	D15		D30		D50	
Outlier %	LODI	SOD	LODI	SOD	LODI	SOD
1%	3.12±0.84	8.35±1.61	6.34±1.27	16.05±2.46	10.92±2.15	26.50±3.95
2%	3.20±0.72	8.40±1.68	6.41±1.14	16.13±2.56	11.03±2.07	27.57±4.15
5%	3.15±0.81	8.16±1.69	6.70±1.18	16.20±2.69	10.87±2.21	26.62±4.31
10%	3.14±0.96	7.84±1.85	6.42±1.23	15.87±3.05	11.08±2.31	25.87±4.81

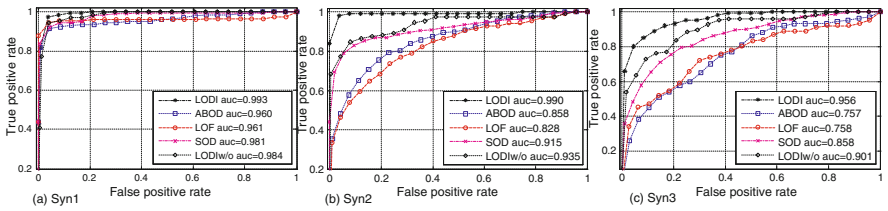
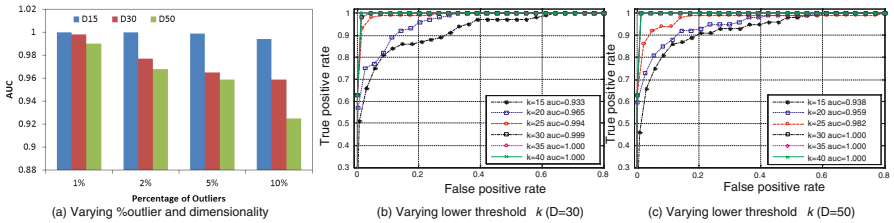


Fig. 3. Outlier detection rate of all algorithms on three synthetic datasets (D=50)

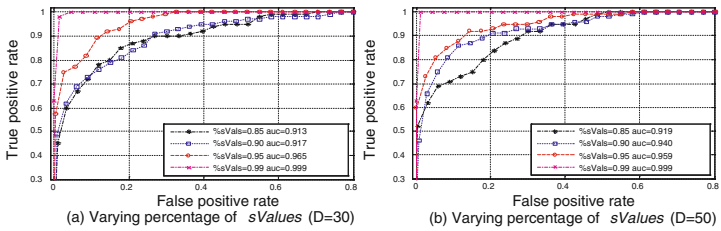
the low dimensionality yet their performances were more divergent on higher dimensional data. We hence report in Figure 3 the outlier detection rates of all methods in D50, setting for all 3 datasets. As observed from these graphs, the outlier detection performance of all algorithms is likely to be decreased as the the large variance percentage used to generate the data increases. However, while the detection rates decrease vastly for other methods, our technique LODI remains stable from Syn1 to Syn2 dataset, and only slightly reduces in Syn3. Nonetheless, its area under the ROC (AUC) is still around 96% for this dataset. Amongst other techniques, the AUCs of LOF are the lowest. This could be explained through its density-based approach which often makes LOF's performance deteriorated in high dimensional data. The performances of both ABOD and SOD are quite competitive yet their ROC curves are still lower than that of LODI for all three examined datasets. In Figure 3, we also report the performance of LODI not using the entropy-based approach in kNNs selection (denoted as LODIw/o). Instead, k is varied from 10 to 40 and the best result is reported. As seen in Figure 3, the AUC of LODIw/o in all cases are smaller than that of LODI, which highlights the significance of the entropy-based approach for kNNs selection. However, compared to other techniques, LODIw/o's outlier detection rate is still better, demonstrating the appealing approach of computing outlier degrees in subspaces learnt from the objective function developed in Eq.(6).

Parameters Sensitivity. To provide more insights into the performance of our LODI technique, we further test its detection rates with various parameter settings. In Figure 4(a), we plot its AUC performance on the Syn3 dataset when the data dimensionality increases from D15 to D50 and the outlier percentage varies from 1% to 10%. The lower threshold k for the neighboring set remains



(a) Varying %outlier and dimensionality (b) Varying lower threshold k (D=30) (c) Varying lower threshold k (D=50)

Fig. 4. Performance of LODI on Syn3 dataset with varying % outliers and threshold k



(a) Varying percentage of *sValues* (D=30) (b) Varying percentage of *sValues* (D=50)

Fig. 5. Performance of LODI on Syn3 dataset with varying % singular values

at 20. One may see that LODI's performance slightly deteriorates as the number of outliers generated in the dataset increases. This happens since once the number of outliers increases, there are higher chances for them to be included in other instances' neighboring sets. Recall that LODI has alleviated this issue by excluding those with low information potential via the use of quadratic entropy. And in order to gain insights into this matter, we further test the case when the lower threshold for the neighboring set is varied. Figures 4(b-c) show the algorithm's ROC curves when k is changed from 15 to 40 for two cases of $D30$ and $D50$. As expected, once k increases, LODI has more capability in excluding irrelevant instances from the neighboring sets and its overall performance increases. As visualized from Figures 4(a-b), a general setting of k around 20 or 25 often leads to competitive results. We finally provide the impact of the total number of singular values used in our matrix approximation. In Figure 5, our algorithm's ROC curves are plotted as the percentage of keeping singular values is varied from 85%, 90%, 95% to 99%. We use Syn3 dataset for these experiments with the data dimensionality at 30 and 50. It is clearly seen that LODI performs better for higher percentages of singular values and in order to keep it at high performance, this parameter should be set around 90% or 95%.

4.2 Real World Data

In this section, we provide the experimental results of all algorithms on three real-world datasets selected from the UCI repository [1]. The first dataset is the image segmentation data which includes 2 310 instances of outdoor images {brickface, sky, foliage, cement, window, path, grass} classified into 7 classes. Each instance is a 3×3 region described by 19 attributes. However, we remove

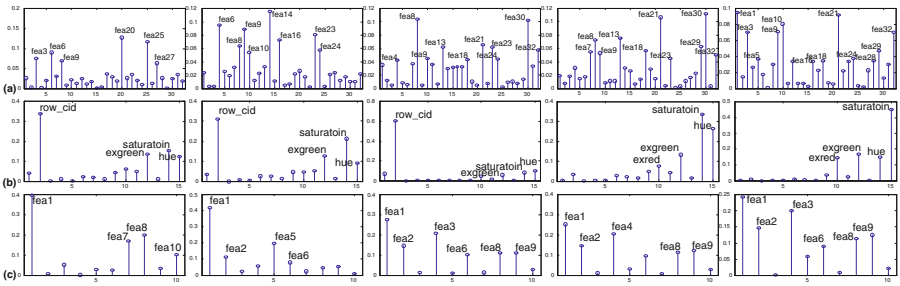


Fig. 6. Feature visualization over 5 top outliers found in: Ionosphere data (a), Image segmentation data (b) and Vowel data (c) (relevant features are shown with labels)

three features 5,7 and 9 from this data as they are known to be repetitive with the attributes 4,6 and 8 [1]. The second dataset is the vowel data consisting of 990 instances and is described by 11 variables (low pass filtered signals), of which the last one is the class label corresponding to 11 different English vowels {hid, hId, hEd, hAd, hYd, had, hOd, hod, hUd, hud, hed}. The third dataset is the ionosphere data containing 351 instances and being described by 32 features (electromagnetic signals). Instead of randomly generating artificial outliers and adding them to these datasets, it is more natural to directly downsample several classes and treat them as hidden outliers (as suggested in [19,11]). Specifically, we keep instances from two randomly selected classes of segmentation data as regular objects and downsample five remaining classes, each to 2 instances to represent hidden outliers. Likewise with the vowel dataset, we keep one class of regular objects and randomly sample 10 instances from the remaining classes to represent outliers. With the 2-class ionosphere data, we select instances from the second class as outliers since its number of objects is much lower than that of the first class.

Unlike the synthetic data where we can manage the data distributions and report the average subspace sizes for all outliers, it is harder to perform such analysis for the real-world datasets since different outliers may have relevant subspaces of different sizes. However, in an attempt to interpret the results of LODI, we plot in Figures 6(a-c) the original features' important degrees of 5 top-ranking outliers respectively selected from the ionosphere, image segmentation and vowel datasets. Figure 6(a) reveals that, for each outlier, there are only few features having high importance degrees and they can be selected as the subspace to interpret the abnormal property of the outlier. However, as this dataset has a large number of outliers, the subspaces do not have many features in common. It is thus more interesting to observe the feature visualization for the two other datasets. Looking at the the 5 top outliers of the image segmentation dataset in Figure 6(b), one can see that out of 15 original features, only a few are suitable to interpret the outliers. For example, the space spanned by $\{row\_icd, exgreen, saturatioin, hue\}$ attributes is suitable to interpret the exceptional property of the first 3 outliers while the space spanned by $\{exred, exgreen, saturatioin, hue\}$ is appropriate to explain the last 2 outliers.

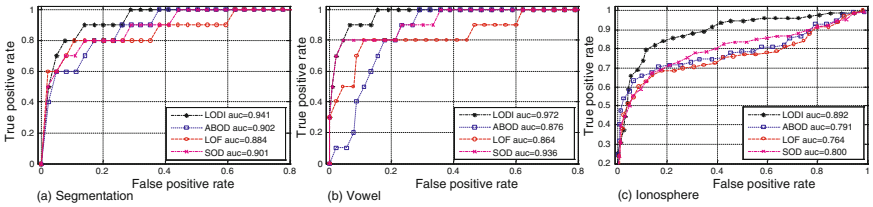


Fig. 7. Performance of all algorithms on three UCI real datasets.

Taking a closer look, we find out that these two types of outliers are indeed exceptional with respect to the 2 main distributions which correspond to the outdoor imaging instances of 2 classes (number 3rd and 6th) in the segmentation data. In the last dataset, vowel, shown in 6(c), few prominent features stand out for outlier interpretation, yet the features vary across different outliers (using $\lambda = 0.8$). Nevertheless, a common and interesting point is that the first attribute always has the highest value across all outliers, indicating it is the most important feature. Recall that for this dataset, we keep instances from only a single vowel (by random selection it is "hYd") as normal objects while randomly downsample one from each of the remaining vowels as hidden outliers. This might also justify the diversity of the other prominent features across the 5 outliers shown here.

We now compare the performance of LODI and the other algorithms through their outlier detection rates. In Figure 7, we report the ROC curves of all algorithms over the three datasets. As observed, LODI shows the best detection performance compared to all three techniques. In the segmentation data, LOF is less successful with its AUC value around 88% though we have tried to optimize its parameter $minPts$. The detection rates of ABOD and SOD are quite competitive and achieve 90% AUC which yet is still lower than LODI's 94%. Moreover, LODI is also likely to uncover all true outliers earlier than the other techniques. As observed in Figure 7, its false positive rate is only at 24% when all outliers are found compared to that of 43% for SOD or 60% for LOF. With the vowel dataset, we observe a similar behavior. Nevertheless, in the ionosphere where the number of outliers is considerably larger, none of the algorithms can discover all outliers before their false positive rate reaches 100%. However, it is seen that while both SOD and ABOD can uncover at most 70% of true outliers when the false positive rate is at 20%, LODI retrieves 86% at the same level. Its overall area under the curve is 89% which is clearly better than the other algorithms.

5 Conclusion

In this work, we developed the LODI algorithm to address outlier identification and explanation at the same time. In achieving this twin-objective, our method makes use of an approach firmly rooted from information theory to select appropriate sets of neighboring objects. We developed an objective function to learn

subspaces in which outliers are most separable from their nearby inliers. We showed that the optimization problem can be optimally solved from the matrix eigen-decomposition of which relevant features are obtained to understand exceptional properties of outliers. Our thorough evaluation on both synthetic and real-world datasets demonstrated the appealing performance of LODI and its interpretation form over outliers is intuitive and meaningful. Nonetheless, LODI has some limitations. First, its computation is rather expensive (quadratic in the dimensionality), making LODI less suitable for very large and high dimensional datasets. In dealing with this issue, approaches based on features' sampling [21] seem to be potential; yet they also lead to some information loss. The challenge is thus to compromise the trade-off between these two criteria. Second, LODI made an assumption that an outlier can be *linearly* separated from inliers. This assumption may not be practical if distributions of inliers exhibit non-convex shapes. Though several learning techniques based on nonlinear dimensionality reduction can be applied to uncover such outliers [5], this, however, still leaves open to the difficult question of what can be an appropriate form to interpret these "*nonlinear*" outliers. We consider these challenges as the immediate issues for our future work.

Acknowledgements. Part of this work has been supported by the Danish Council for Independent Research - Technology and Production Sciences (FTP), grant 10-081972.

References

1. Asuncion, A., Newman, D.: UCI machine learning repository (2007)
2. Barnett, V., Lewis, T.: Outliers in statistical data, 3rd edn. John Wiley & Sons Ltd. (1994)
3. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: Identifying density-based local outliers. In: SIGMOD (2000)
4. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys 41(3) (2009)
5. Dang, X.H., Micenková, B., Assent, I., Ng, R.T.: Outlier detection with space transformation and spectral analysis. In: SIAM-SDM (2013)
6. de Vries, T., Chawla, S., Houle, M.E.: Finding local anomalies in very high dimensional space. In: ICDM, pp. 128–137 (2010)
7. Foss, A., Zaïane, O.R., Zilles, S.: Unsupervised class separation of multivariate data through cumulative variance-based ranking. In: ICDM (2009)
8. Golub, G., Loan, C.: Matrix Computations, 3rd edn. The Johns Hopkins University Press (1996)
9. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques, 3rd edn. Morgan Kaufmann Publishers Inc. (2012)
10. He, Z., Deng, S., Xu, X.: A unified subspace outlier ensemble framework for outlier detection. In: Fan, W., Wu, Z., Yang, J. (eds.) WAIM 2005. LNCS, vol. 3739, pp. 632–637. Springer, Heidelberg (2005)
11. Keller, F., Müller, E., Böhm, K.: Hics: High contrast subspaces for density-based outlier ranking. In: ICDE (2012)

12. Knorr, E.M., Ng, R.T.: Algorithms for mining distance-based outliers in large datasets. In: VLDB (1998)
13. Knorr, E.M., Ng, R.T.: Finding intensional knowledge of distance-based outliers. The VLDB Journal 8, 2111–2222 (1999)
14. Kriegel, H.-P., Kröger, P., Schubert, E., Zimek, A.: Outlier detection in axis-parallel subspaces of high dimensional data. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 831–838. Springer, Heidelberg (2009)
15. Kriegel, H., Schubert, M., Zimek, A.: Angle-based outlier detection in high-dimensional data. In: SIGKDD (2008)
16. Kriegel, H.-P., Kröger, P., Schubert, E., Zimek, A.: Outlier detection in arbitrarily oriented subspaces. In: ICDM, pp. 379–388 (2012)
17. Kriegel, H.-P., Kröger, P., Zimek, A.: Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. TKDD 3(1) (2009)
18. Lazarevic, A., Kumar, V.: Feature bagging for outlier detection. In: SIGKDD, pp. 157–166 (2005)
19. Müller, E., Schiffer, M., Seidl, T.: Statistical selection of relevant subspace projections for outlier ranking. In: ICDE, pp. 434–445 (2011)
20. Nguyen, H.V., Gopalkrishnan, V., Assent, I.: An unbiased distance-based outlier detection approach for high-dimensional data. In: Yu, J.X., Kim, M.H., Unland, R. (eds.) DASFAA 2011, Part I. LNCS, vol. 6587, pp. 138–152. Springer, Heidelberg (2011)
21. Olken, F., Rotem, D.: Random sampling from databases - a survey. Statistics and Computing 5, 25–42 (1994)
22. Papadimitriou, S., Kitagawa, H., Gibbons, P.B., Faloutsos, C.: Loci: Fast outlier detection using the local correlation integral. In: ICDE, pp. 315–326 (2003)
23. Parsons, L., Haque, E., Liu, H.: Subspace clustering for high dimensional data: a review. SIGKDD Explorations 6(1), 90–105 (2004)
24. Renyi, A.: On measures of entropy and information. In: Proc. Fourth Berkeley Symp. Math., Statistics, and Probability, pp. 547–561 (1960)
25. Schubert, E., Wojdanowski, R., Zimek, A., Kriegel, H.-P.: On evaluation of outlier rankings and outlier scores. In: SDM (2012)
26. Schubert, E., Zimek, A., Kriegel, H.-P.: Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection. In: Data Mining and Knowledge Discovery, pp. 1–48 (2012)
27. Tao, Y., Xiao, X., Zhou, S.: Mining distance-based outliers from large databases in any metric space. In: SIGKDD (2006)
28. Tibshirani, R., Hastie, T.: Outlier sums for differential gene expression analysis. Biostatistics 8(1), 2–8 (2007)
29. Zimek, A., Schubert, E., Kriegel, H.-P.: A survey on unsupervised outlier detection in high-dimensional numerical data. Statistical Analysis and Data Mining 5(5), 363–387 (2012)