Data Perturbation for Outlier Detection Ensembles

Arthur Zimek
Ludwig-Maximilians-Universität
Munich, Germany
http://www.dbs.ifi.lmu.de
zimek@dbs.ifi.lmu.de

Ricardo J. G. B. Campello
University of São Paulo
São Carlos, Brazil
http://www.icmc.usp.br
campello@icmc.usp.br

Jörg Sander
University of Alberta
Edmonton, AB, Canada
http://www.cs.ualberta.ca
jsander@ualberta.ca

ABSTRACT

Outlier detection and ensemble learning are well established research directions in data mining yet the application of ensemble techniques to outlier detection has been rarely studied. Building an ensemble requires learning of diverse models and combining these diverse models in an appropriate way. We propose data perturbation as a new technique to induce diversity in individual outlier detectors as well as a rank accumulation method for the combination of the individual outlier rankings in order to construct an outlier detection ensemble. In an extensive evaluation, we study the impact, potential, and shortcomings of this new approach for outlier detection ensembles. We show that this ensemble can significantly improve over weak performing base methods.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data mining

Keywords

outlier detection; ensemble

1. INTRODUCTION

The main outlier methods proposed in the literature, aside from variants tackling efficiency issues, differ in the way they model outliers and, thus, in the assumptions they implicitly or explicitly rely on. Statistical methods for outlier detection (also: outlier identification or rejection) are based on assumptions on the nature of the distributions of objects. The classical textbook by Barnett and Lewis [6] discusses numerous tests for different distributions. The tests are optimized for each distribution and depend on: the specific parameters of the corresponding distribution; the number of expected outliers; and the space where to expect an outlier. A recent discussion of different statistical techniques is presented by Rousseeuw and Hubert [38]. A broader overview of outlier detection methods for modern applications has been provided by Chandola et al. [12].

In this paper, we focus on representative techniques based on distances and density estimates in Euclidean data spaces. The distance-based notion of outliers (DB-outlier) [27] was the first database-oriented approach. Variants consider the distances to the k nearest neighbors of each object and use these distances to rank the objects [36], or use the sum of distances to all points within the set of k nearest neighbors (called the "weight") as an outlier degree [5].

The so-called "local" approaches, e.g. LOF [10], consider ratios between the local density around an object and the local density around its neighboring objects. LDOF [47] is comparable in performance to classical kNN-based outlier detection but reported to be less sensitive to parameter values. LoOP [28] uses a density estimation based on the distance distribution of all nearest neighbors and defines the local outlier score as a probability. LOCI [34] is another variant on LOF [10], comparing the local density of an object to its neighbors at multiple radii.

DB-outlier and the variants using the k nearest neighbor distances are also known as "distance-based" methods, while LOF and its variants are known as "density-based" methods. This differentiation, however, is only superficial [41]. Both, "distance-based" and "density-based" approaches, basically aim at providing a more or less refined estimate of the density around some point.

Statistical approaches fit certain distributions to the data, and estimate the parameters of the distribution. However, these parameters (as, e.g., mean, standard deviation, or covariances) are rather sensitive to outliers. Possible effects of outliers on the parameter estimation are *masking* and *swamping*. Outliers *mask* their own presence by influencing the values of the distribution parameters (resulting in false negatives), or *swamp* inliers to appear as outlying due to the influenced parameters (resulting in false positives) [35, 7, 6, 21].

In contrast to statistical approaches, the density estimates used by "distance-based" and "density-based" algorithms are non-parametric. They do not assume a specific distribution but just estimate the density level. These estimates, however, are based on merely one sample (the data set) drawn from the (unknown) density distribution. Even for large data sets, especially in high-dimensional data spaces [50], the sample size can be too small since the data are often based on a complex mixture of distributions whereas different components are represented by only part of the data. Effects similar to masking and swamping can here occur, e.g., if the density estimates are misled by local errors in the density estimates induced by random variations such as "gaps" in the sample instead of being actually variations in the underlying density distribution. Re-sampling would be the method of choice if it only would be possible to draw new samples.

As a way to use the available data in the unsupervised scenario to obtain more reliable density estimates, we approximate re-sampling by bootstrapping i perturbed data sets (i.e., we change the attribute values of each point by adding a noise component of some small, randomized, amount), and run the outlier detection algorithm of

SSDBM '14, June 30 - July 02 2014, Aalborg, Denmark
Copyright 2014 ACM 978-1-4503-2722-0/14/06 ...$15.00.
http://dx.doi.org/10.1145/2618243.2618257.

our choice on all the i perturbed data sets. This way, we can keep track of the identity of each data point and aggregate the scores and the rank positions of each point. We then combine the resulting i outlier rankings (or scorings) to get an integrated, more stable and reliable outlier ranking (or scoring) of the data.

From the point of view of learning theory, this could also be seen as an ensemble approach to outlier detection [4], where diversity of individual voters is introduced by means of perturbation of data. Open questions for the use of ensemble techniques for outlier detection have been discussed recently [48], pointing out the issue of fundamental principles for creating diversity and the ongoing discussion regarding combinations of outlier rankings. Here, we add perturbation as a fundamentally new way of creating diversity for an ensemble compared to the established ways and discuss a new way of combining outlier rankings.

Both aspects, the approximation of improved density-estimates, and the theoretical findings of ensemble-learning when individual voters are diverse but still accurate, explain our findings that this method is improving, sometimes substantially, over the corresponding basic outlier detector. The proposed principle is very fundamental and flexible and can be combined with various conventional outlier detection techniques, as we will demonstrate in the experiments.

In the following, we discuss related work w.r.t. data perturbation and w.r.t. ensemble learning for outlier detection (Section 2). We reason about the technical implications of data perturbation and its potential to improve local density estimates (Section 3). We discuss the combination of single outlier detection instances obtained on perturbed data and introduce a novel rank combination procedure for outlier rankings to construct an outlier detection ensemble (Section 4). In an extensive evaluation we show the superiority of the new rank combination procedure as well as the potential of data perturbation for improvement of outlier ensembles but we also point out the risks of this technique (Section 5). We conclude the paper in Section 6.

2. RELATED WORK

2.1 Perturbation

As an application of his clustering validity index, Rand suggested to compare clustering results on perturbed data to clustering results on the original data [37]. This procedure can assess the stability or robustness of clustering results. If the original clustering and the clustering obtained from the perturbed data are very different, the original clustering is unlikely to be a valid, reliable result. Despite the obvious usefulness of this procedure, data perturbation has not been thoroughly explored in data mining. Rand's proposal has been used occasionally for evaluation in (mainly biological) clustering research [9, 23, 26, 31]. The impact of "accidental" data perturbation on spectral clustering has been studied, considering data preprocessing techniques such as filtering, quantization, or compression, i.e., techniques used to speed-up cluster computation but essentially resulting in a certain degree of perturbation [24]. Data perturbation after deriving the clustering on the original data has been used to assess how stable a classifier performs, that was trained on the clustering [16]. Perturbation has not been used for outlier detection in any way.

2.2 Outlier Detection Ensembles

The first approach to improve outlier detection by ensemble techniques was "feature bagging" [30], combining different results of the same algorithm (namely LOF [10]) applied to different feature subsets. Feature bagging is a common procedure to induce diversity of ensemble members in ensemble classification [11] or ensemble clustering [17, 44, 8]. For combination of the outlier scores obtained on different feature subsets, Lazarevic and Kumar [30] proposed two methods. First, they apply a normalization by ranking as a breadth-first traversal through all the outlier rankings obtained from the different feature subsets. Second, they compute the cumulative sum of the different scores. Both combination methods are straightforward and have some severe drawbacks. The breadth-first traversal rank combination ranks those objects ranked on a top position by any one of the n different ensemble members also on a top position. This way, the first positions of each individual ranking are strongly emphasized and errors of single ensemble members cannot be outweight by even all other detectors being correct. This drawback is rendering this combination method ineffective w.r.t. one of the most fundamental benefits that one can expect from an ensemble method at all: the correction of errors that are committed by single ensemble members. The second combination method, score aggregation, relies heavily on the scorings being comparable. This problem practically rules out the combination of different base methods or, for many methods, different parametrizations (e.g., different k for a kNN-distance-based method). Even when using the same method as base outlier detector and identical parametrization, outlier scores obtained from different subspaces could vary considerably if some subspaces have different scales. The ensemble could then be biased by just one of the feature bags.

It has been pointed out recently that some subspace outlier detection techniques could also be seen as ensemble techniques but that they also face similar challenges regarding the combination of scores from different subspaces [4, 50, 48].

Subsequent research on outlier detection ensembles focused on this very same issue of comparability of scores for combination. Sigmoid functions and mixture modeling have been applied to fit outlier scores provided by different detectors into comparable probability values [20], using different values for k of the kNN distance as an outlier score to induce diversity. Another method [33] uses scaling by standard deviation of outlier scores and induces diversity in the applied base detectors by feature bagging.

Statistical reasoning motivated normalization of scores of different outlier detection methods into a unified value range in $[0, 1]$, enabling the combination of different outlier detection methods into one ensemble [29]. Schubert et al. [39] proposed a similarity measure to appropriately compare different outlier rankings (based on scores) and to allow for the assessment of actual diversity of different outlier detectors. As an application, they propose a greedy ensemble approach demonstrating the importance of diversity for the performance of an ensemble. In all these papers, though outlier detection ensembles have been discussed and improved, no new method of inducing diversity has been pursued.

Using several instances of a randomized method was a principle of diversity, although not studied for its impact, in the study on isolation forests by Liu et al. [32]. Zimek et al. [49] used data subsamples to learn diverse outlier models for an ensemble. Other examples return to the approach of combining models learned with different parameters. Schubert et al. [40] use generalized kernel density estimates for adaptation to various datasets and combine results from different parametrization of their method. Likewise, Dang et al. [13] aggregate over different parameter values in their evaluation, hence essentially building an ensemble for the tested methods.

Aside from the simple breadth-first traversal strategy of the original feature bagging approach [30], all subsequent ensemble methods combined the individual outlier rankings by some aggregation (sum or average) of the individual scores.

As discussed in a recent position paper [48], methods to induce diversity and methods to combine outlier rankings are both open issues and research is in an early stage. Here, we contribute a new approach to each of these two fundamental open issues.

3. PERTURBATION FOR OUTLIER DETECTION

3.1 Motivation

At first sight, it may seem counterintuitive that, while much effort is spent to reduce or eliminate noise in data sets, adding noise to data should actually help in the data mining process. However, adding controlled noise is strongly motivated by theory on ensemble learning. In classification, building ensembles combining several single classifiers to gain an improved effectiveness has a rich tradition and a sound theoretical background [14, 45, 11]. The fundamental lessons learned w.r.t. ensemble learning in classification are that we have two basic requirements for an ensemble to improve over the contained base-classifiers: the base classifiers, i.e., the members of the ensemble, need to be (i) *accurate* (i.e., at least better than random) and (ii) *diverse* (i.e., making different errors on new instances). If several individual classifiers were not diverse, then all of them will be wrong whenever one of them is wrong. This way, nothing is gained by combining them. On the other hand, if the errors made by the ensemble members were uncorrelated, more members may be correct while some members are wrong. Therefore, a majority vote by an ensemble may be also correct. It is clear that each ensemble member should be at least somehow meaningful in order to get meaningful results out of their combination. Hence, a key for building good ensembles is to use ensemble members that are diverse in the sense that they make different (ideally: uncorrelated) errors (if any).

3.2 Theoretical Justification

In outlier detection, adding noise in a controlled way now has actually the potential to introduce diversity where the single outlier detection method is likely to make wrong decisions. The magnitude of the added noise should be small enough to avoid swamping of clear inliers or masking of clear outliers, but we argue that the borderline cases can be expected to benefit from diversified decisions. Considering the case of global outliers, let us assume we have $f(x)$ as the true, smooth probability density function underlying our dataset $\mathcal{D}$, and a density threshold θ, to discern inliers $\mathcal{I}$ and outliers $\mathcal{O}$ (see Figure 1), i.e.,

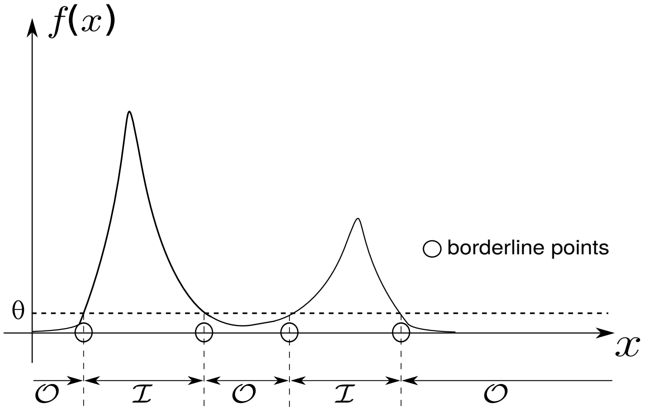


Figure 1: Relationship of Density and borderline points

$$f(x) < \theta \Rightarrow x \in \mathcal{O}, \text{ and } f(x) > \theta \Rightarrow x \in \mathcal{I}$$

(the case of $f(x) = \theta$ can be defined either way).

However, we do know only a sample X of the data and an estimate of $f(x)$ based on X, given by $\hat{f}_X(x) = f(x) + v_X(x)$, where $v_X(x)$ is a random variable describing the error of the estimate due to the finite sample. Assuming the error being independent of the single datapoint x, we have $\hat{f}_X(x) = f(x) + v_X$. Given a sample X and an inlier x ($x \in \mathcal{I}$), the probability that x is evaluated as a false positive according to $\hat{f}_X(x)$ is

$$P(\hat{f}_X(x) < \theta) = P(f(x) + v_X < \theta).$$

Hence, independent of the distribution $p(v_X)$,

$$P(\hat{f}_X(x_1) < \theta) \geq P(\hat{f}_X(x_2) < \theta)$$

if

$$f(x_1) - \theta \leq f(x_2) - \theta,$$

i.e., if $f(x_1) \leq f(x_2)$ for $x_1, x_2 \in \mathcal{I}$.

Analogously, the probability of an outlier $x_1 \in \mathcal{O}$ to be erroneously deemed an inlier (i.e., the probability of x_1 being a false negative) is greater or equal to that of another outlier x_2, i.e.,

$$P(\hat{f}_X(x_1) > \theta) \geq P(\hat{f}_X(x_2) > \theta),$$

if

$$\theta - f(x_1) \leq \theta - f(x_2),$$

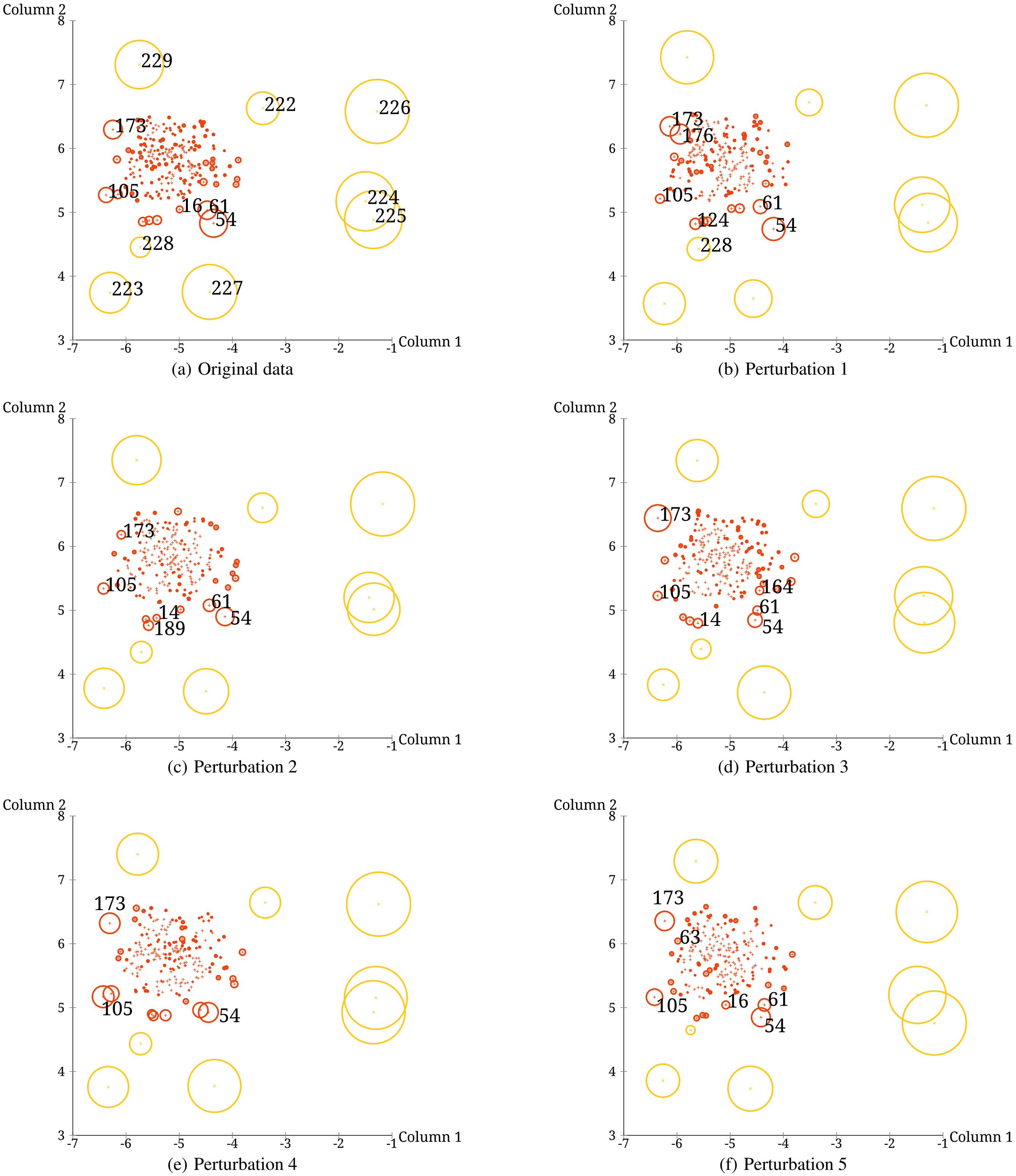
i.e., if $f(x_1) \geq f(x_2)$ for $x_1, x_2 \in \mathcal{O}$.

It follows that the probabilities of false positives and false negatives are maximal for "borderline" points, i.e., points that are near those values of x where $f(x)$ is close to the threshold θ.

This reasoning can be extended to local outlier models by allowing the constant threshold θ to adapt variably to local estimates.

As a consequence, the quality of the estimate $\hat{f}$ of f (where the true probability density function f is *not* known to us) decides over success and failure of the outlier detection. Density estimates as used by outlier detection algorithms in the area of database research are based on the available data, which can be seen as a sample of some underlying distribution. Even with a large data set the sample size can be just too small to allow for a reliable and stable density estimate in all regions of the data space. To obtain reliable and stable density estimates, what one would like to do would be to draw new and more samples, if one only had access to the true probability density function.

Perturbation is an alternative that simulates multiple samples, based on the one sample we have, although it imputes an error $v_X(x)$ that actually *cannot* be expected to be independent of x. However, it can be used if we are only interested in a good binary classification between $\mathcal{O}$ and $\mathcal{I}$ for a given θ, rather than in a good ranking inside these sets of outliers and inliers, respectively. Note that the threshold level θ is used only implicitly, as it is, e.g., defined for some data set by the annotated ground truth, or it remains actually unknown. It is not a parameter of any method, rather an evaluation procedure would evaluate the dichotomous problem if outliers (density below θ) are ranked before inliers (density higher than θ). We expect to rarely observe inversions in the original set of a point x ($\mathcal{O}$ or $\mathcal{I}$) for clear inliers and top outliers but to observe inversions more frequently for borderline points.



(a) Original data

(b) Perturbation 1

(c) Perturbation 2

(d) Perturbation 3

(e) Perturbation 4

(f) Perturbation 5

Figure 2: Effect of perturbation on outlier scores (LOF, $k = 10$), IDs of critical objects annotated.

Table 1: Example: effect of perturbation on ranking and rank accumulation

(a) Rankings (top-14) according to LOF on original data and on 5 instances of perturbation.

original		pert. 1		pert. 2		pert. 3		pert. 4		pert. 5	
ID	LOF	ID	LOF	ID	LOF	ID	LOF	ID	LOF	ID	LOF
226	5.50	226	5.40	226	5.63	226	5.78	226	5.61	225	5.54
224	5.16	225	5.04	225	4.79	225	5.53	225	5.55	226	5.35
225	5.05	224	4.81	224	4.64	224	5.35	224	5.53	224	5.05
227	4.85	229	4.65	229	4.55	227	4.98	227	4.81	227	4.18
229	4.39	223	3.88	227	4.27	229	4.14	229	4.01	229	4.09
223	3.84	227	3.57	223	3.91	223	3.35	223	3.94	222	3.39
222	3.27	222	2.83	222	3.14	222	3.02	222	3.22	223	3.35
54	2.92	228	2.59	228	2.56	173	2.98	228	2.59	173	2.37
228	2.38	54	2.57	54	2.30	228	2.48	105	2.56	54	2.33
173	2.27	176	2.36	61	1.87	54	2.07	173	2.47	105	2.16
61	2.25	173	2.33	105	1.79	14	1.7	54	2.41	61	1.87
105	2.01	61	1.96	189	1.70	61	1.67	25	2.13	228	1.65
14	1.57	124	1.75	173	1.60	105	1.66	61	2.08	16	1.55
25	1.56	105	1.62	14	1.50	164	1.60	14	1.78	63	1.40

(b) Rank accumulation for combination of the LOF ranks on the perturbed data

	among top-														$\sum$
ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	$_{1,\dots,14}$
14	0	0	0	0	0	0	0	0	0	0	1	1	1	3	6
16	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2
25	0	0	0	0	0	0	0	0	0	0	0	1	1	1	3
54	0	0	0	0	0	0	0	0	3	4	5	5	5	5	27
61	0	0	0	0	0	0	0	0	0	1	2	4	5	5	17
63	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
105	0	0	0	0	0	0	0	0	1	2	3	3	4	5	18
124	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2
164	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
173	0	0	0	0	0	0	0	2	2	3	4	4	5	5	25
176	0	0	0	0	0	0	0	0	0	1	1	1	1	1	5
189	0	0	0	0	0	0	0	0	0	0	0	1	1	1	3
222	0	0	0	0	0	1	5	5	5	5	5	5	5	5	41
223	0	0	0	0	1	4	5	5	5	5	5	5	5	5	45
224	0	0	5	5	5	5	5	5	5	5	5	5	5	5	60
225	1	5	5	5	5	5	5	5	5	5	5	5	5	5	66
226	4	5	5	5	5	5	5	5	5	5	5	5	5	5	69
227	0	0	0	3	4	5	5	5	5	5	5	5	5	5	52
228	0	0	0	0	0	0	0	3	4	4	4	5	5	5	30
229	0	0	0	2	5	5	5	5	5	5	5	5	5	5	52

Sampling has been used in ensemble *clustering* to induce diversity. There, different subsamples of the data set have been clustered and the resulting clusterings are combined to a consensus clustering [43, 15, 19, 22]. In these cases, however, each re-sampled dataset is smaller than the original data set or, if it is not smaller in case of sampling with replacement, at least does not contain more information to estimate the density. The new samples are just subsamples of the original sample, not re-drawn from the (unknown) underlying distribution. The fundamental difference of clustering and outlier detection in this respect is that clustering is interested in the overall structure of the data set while outlier detection is specifically interested in identifying and filtering out single and by definition rare data points that are not even guaranteed to be present. Outliers are most likely not contained in a sub-sample and would not even be expected to be reproduced by a re-sampling based on the statistical properties of the data set as a whole (e.g., as determined by sophisticated density estimators like kernel density estimates).

3.3 Practical Example

To illustrate how perturbation, i.e., adding noise, can improve outlier detection, consider the toy example in Figure 2. The original data set (Figure 2(a)) consists of a Gaussian cluster (red) and 8 uniformly distributed outliers (yellow). The outliers far from the cluster are easy to detect (we use LOF [10] and the visualization of ELKI [2], larger bubbles around the points signaling a higher outlier score).

Outliers near the cluster (i.e., true positives to report as outliers) and border points of the cluster (i.e., points in the tails of the Gaussian, that would be reported as false positives) are difficult to distinguish. Those are what we called “borderline” points. The scores of points of these two categories are often rather similar. Some inliers are even ranked higher than some outliers. We show some instances of LOF runs on perturbed data in Figure 2(b) to Figure 2(f). The perturbance leads especially to instability of the ranks in the borderline cases while clear outliers remain stably ranked at top.

Combining the ranks of several instances obtains an improved result especially for the borderline cases.

We inspect the top-14 ranked objects for the original data and the 5 perturbations in Table 1(a) for an illustration of the effects of perturbation, especially on the critical borderline points. There are 8 genuine outliers in this example, but the original LOF ranking does not include ID 228 among the top 8 due to a "false positive" (ID 54). The reason is that the outlier (ID 228) is close to the cluster and the inlier (ID 54) is relatively far out. Even though ID 228 is not always among the top-8 in the multiple perturbations either, the different perturbations commit different errors. Different cluster objects, not consistently ID 54, are taking a place among the top-8 instead of ID 228, and different cluster objects are occasionally ranked higher than ID 54.

We implement the perturbation by adding to each point an attribute-wise noise component sampled from $\mathcal{N}(0, \sigma_a)$ in each attribute a. We scale σ_a as a selected percentage of the range $\max_a(p \in \mathcal{D}) - \min_a(p \in \mathcal{D})$ of the corresponding attribute a. Rand proposed 0.01 for clustering evaluation, we experiment also with higher values. In the example of Figure 2 we use a percentage of 0.02.

4. PERTURBATION-BASED ENSEMBLE

Having derived a set of outlier rankings, building an ensemble requires a suitable combination of the different rankings. As the rankings are based on outlier scores, an aggregation (e.g., sum or average) of all scorings for each data object is straightforward and, in fact, is the strategy used in previous ensemble approaches [30, 33, 29, 39]. This combination of scorings, however, requires the different scorings to follow a comparable semantic and behavior. Here we apply, as one variant of a combination, the average score. As a prerequisite, we apply a linear scaling of all scores for each single outlier detector to the $[0, 1]$ interval to allow for *meaningful* combinations like the average score of an ensemble of individual scorings for each object.

Though more complex transformations have been proposed in the literature [29], the simple linear scaling is sufficient to guarantee a meaningful combination.

However, it is known that outlier scores are often not providing good contrast [29, 50]. As the perturbation induces additionally unforeseeable variants in the value range of the scorings (depending on the outlier detection method used as base detector), we propose as another variant of combination to use the rankings directly. For our rank combination, we count how often (i.e., how consistently) a data object is ranked within the top-n positions among the rankings of the i base detectors. However, we do not choose a specific value for n but vary n from 1 to the database size $N = |\mathcal{D}|$ and accumulate by summing up the counts for all n.

We expect several benefits from this rank accumulation procedure. First, the top positions will naturally contribute much more to the resulting combined ranking but, other than in the procedure of [30], erroneously top-ranked points in a single base ranking can be balanced by the remaining rankings. What we count is the consistency in the ranking. Second, as the ranking position for inliers is expected to vary much more (i.e., the ranking positions of inliers are much more inconsistent), this procedure should also help to widen the gap between outliers and inliers in the resulting score distribution. A significant gap typically does not exist in score distributions of most outlier detection methods [29]. Third, if no genuine outlier is present in the data, for many methods it is not obvious from the scores that the top-ranked data object is actually not an outlier. Here, however, we will obtain scores considerably lower than the theoretical maximum (i.e., $i \cdot N$) if the ranking is not consistent in most of the perturbed data sets.

As an example, consider the perturbation ranks of Table 1(a). We list in Table 1(b) the ranks accumulated for $n = 1, \ldots, 14$ over perturbations 1-5 for those objects in Table 1(a) that are involved in the top-14 of any of the 5 rankings on perturbed data. In each row, we count how many of the involved ensemble members report the object within the top-n, increasing n from column to column. For example, ID 225 has been seen at top-1 once and is on position 2 for the remaining 4 individual rankings, resulting in a count of 5 for all top-n ($n \geq 2$). We stop in the table at rank 14 for practical reasons, the algorithm would proceed until $N = |\mathcal{D}|$. This continued accumulation will not change the lead in the count accumulated until rank 14 but only disambiguate the ranking for the lower ranked points (that did not occur in the top-14 and so far have all the same count, namely 0). The points that occurred in all i (here: 5) rankings up to a certain position n (here: 14) have a fixed rank position among the top-n. For the two points that have been misranked by LOF on the original data, we see that the true outlier ID 228 on these few perturbations already accumulates enough evidence of being more consistently a top outlier than ID 54.

Normalizing the accumulated count with $\frac{1}{i \cdot N}$ (e.g., for combination with other methods) obtains a score in the interval $[0, 1]$. Values near 1 indicate a high probability of being an outlier as the corresponding object occurred in almost all rankings near the top position (the value is exactly 1 for an object that is always on position 1). In the example of Table 1(b), we could thus normalize the values in the last column dividing by $5 \cdot 14 = 70$, resulting in scores close to 1 for the top outliers ID 224, 225, and 226.

This specific benefit requires, however, the base outlier detection method to be unstable w.r.t. data perturbation and we will in fact show that different outlier detection methods seem differently suitable for our ensemble method.

5. EVALUATION

5.1 Methods

The canonical competitor is feature bagging [30]. Other ensemble methods [20, 33, 29, 39] are meta-methods and could be used on top of our perturbation method (or on top of feature bagging, as in [33, 29, 39]) but do not propose original means to induce diversity when using a selected base outlier detection method. Instead, they open up possibilities of a meaningful combination of the scores of different outlier detection methods as base methods for an ensemble or combinations of scores obtained by different parametrizations of some base method. This is a challenge per se, as the scores of different methods or different parametrizations of the same method (e.g., the kNN-distance) can differ widely in their magnitude. However, there is no fair way of comparing ensembles based on different methods to ensembles based on perturbation or on feature bagging.

As base methods we use kNN [36], weighted kNN (kNNw) [5], LOF [10], LDOF [47], and LoOP [28], all with a range of parameter settings for the size k of the neighborhood equal to 5, 10, and 50. We compare the results of the ensembles using the same base method and the same parametrization.

For the size of the feature bags (number of attributes) we chose $\left\lfloor \frac{2}{3} d \right\rfloor$; for the size of an ensemble we chose 25 ensemble members (i.e., 25 perturbations or 25 feature bags). For the average vote, we linearly scale the outlier scores provided by an individual outlier detector to the interval $[0, 1]$.

We report the area under the receiver operating characteristic curve (ROC AUC), which plots the true positive rate vs. the false positive rate, a common measure for evaluation of outlier detec-

Table 2: Comparison of breadth-first traversal vs. our rank accumulation on two batches of 30 datasets each

(a) Batch 1

	breadth-first traversal				rank accumulation			
k	μ	σ	min	max	μ	σ	min	max
5	0.872	0.022	0.826	0.922	0.951	0.009	0.926	0.972
10	0.884	0.023	0.833	0.928	0.952	0.009	0.931	0.969
20	0.876	0.027	0.82	0.924	0.95	0.009	0.933	0.969
50	0.859	0.032	0.775	0.908	0.944	0.01	0.924	0.961

(b) Batch 2

	breadth-first traversal				rank accumulation			
k	μ	σ	min	max	μ	σ	min	max
5	0.86	0.032	0.811	0.916	0.944	0.015	0.914	0.972
10	0.875	0.032	0.800	0.934	0.947	0.013	0.917	0.976
20	0.87	0.032	0.805	0.939	0.946	0.014	0.911	0.975
50	0.858	0.038	0.754	0.92	0.941	0.016	0.907	0.970

tion methods [30, 20, 33, 29, 39, 49, 48]. The experiments are performed using ELKI [3].

5.2 Datasets

For a statistical assessment, we generate two independent sets of 30 synthetic datasets (batch 1 and batch 2). For each dataset, we choose randomly values for the following parameters in the given range: dimensionality $d \in [20, \ldots, 40]$, number of clusters $c \in [2, \ldots, 10]$, for each cluster independently the number of points $n_{c_i} \in [600, \ldots, 1000]$. For each cluster, the points are generated following a Gaussian model as follows: For each cluster c_i, and each attribute a, we choose a mean $\mu_{c_i,a}$ from a uniform distribution in $[-10, 10]$ and a standard deviation $\sigma_{c_i,a}$ from a uniform distribution in $[0.1, 1]$. Then for the cluster c_i, n_{c_i} cluster objects (points) are generated attribute-wise by the Gaussians $\mathcal{N}(\mu_{c_i,a}, \sigma_{c_i,a})$. The resulting cluster is rotated by a series of random rotations and the covariance matrix Σ corresponding to the theoretical model is computed by the corresponding matrix operations [42]. Then, we compute for each point the Mahalanobis distance to its corresponding cluster center, using the covariance matrix Σ of the cluster. For a dataset dimensionality d, the Mahalanobis distances for each cluster follow a χ^2 distribution with d degrees of freedom. We label as outliers those points that exhibit a distance to their cluster center larger than the theoretical 0.975 quantile, independently of the actually occurring Mahalanobis distances of the sampled points. This results in an expected amount of 2.5% outliers per dataset.

As real datasets we chose from the UCI machine learning repository [18]: pendigits, Wisconsin breast cancer (WBC), Waveform Database Generator (waveform), and Cardiotocography (cardio). We prepared these datasets for outlier detection by downsampling one class (‘4’, ‘malignant’, ‘0’, and ‘5’, respectively) to obtain a small amount of outliers in the resulting data set (20, 10, 100, and 42 objects, respectively). Criteria for dataset selection were a suitably high dimensionality for the applicability of feature bagging, and a promissing class structure to allow for the existence of outliers when downsampling a class. The resulting datasets consist of 6734, 367, 5000, and 2126 objects, in 16, 30, 21, and 21 numerical dimensions, respectively. With this method of using classification data for evaluation of outlier detection methods we are conform with the literature [30, 1, 46, 47, 25, 49].

5.3 Rank Accumulation

To discern between the feature bagging as a method of inducing diversity on the one hand and the ranking combination procedure on the other hand, we first evaluate feature bagging with its original setup, the breadth-first traversal for rank aggregation, versus feature bagging with our proposed rank accumulation method for rank aggregation. We test this for LOF as base method as proposed in the original paper [30], for values of $k \in \{5, 10, 20, 50\}$. Using the two independent batches of 30 synthetic datasets each, we compare the ROC AUC of the original feature bagging based on LOF with different values of k, averaged over the 30 datasets of each batch.

The results are captured in Table 2(a) and Table 2(b) for batch 1 and batch 2, respectively. The figures for both batches of 30 datasets are similar and support our claim that the rank accumulation method we propose is superior to the breadth-first traversal strategy of the original feature bagging method [30].

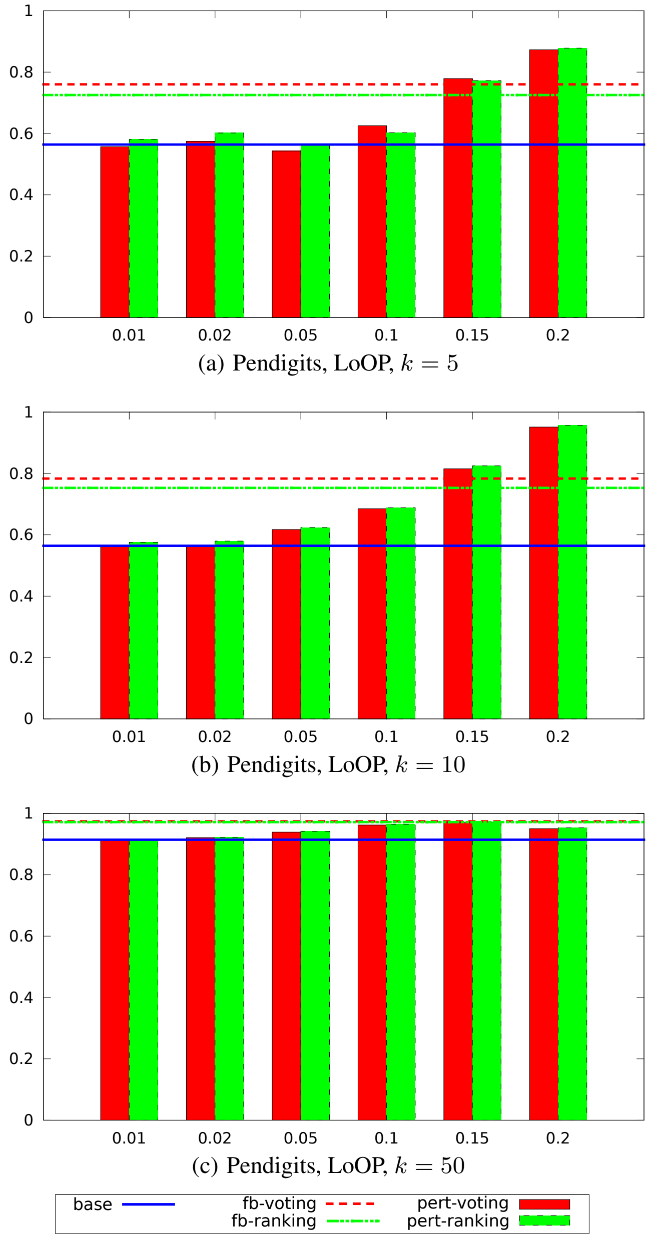
5.4 Perturbation

In the following, we study the potential of data perturbation as a means to induce diversity among individual outlier detectors independent of the rank combination method. We therefore use the competitor feature bagging in an improved variant by applying our rank accumulation method instead of the original breadth-first traversal. Thus the reader should consider that the results we report for feature bagging in the ranking variant are better results than the original method would achieve.

5.4.1 Impact of Magnitude of Noise

The impact of the choice of the standard deviation σ for the noise perturbation is demonstrated in Figure 3 for LoOP as base method (the other methods having similar results) with different parameter values for k on the Pendigits data, and in Figure 4 on the WBC data. The plots show the performance (ROC AUC) of the perturbation ensembles with mean score voting and our rank accumulation method (“pert-voting” and “pert-ranking”) using different values of σ against the performance of the base method and the two feature bagging variants (using mean score voting and our rank accumulation method, noted as “fb-voting” and “fb-ranking”, respectively). The base method and the feature bagging variants are shown as constant lines as they are independent of the data perturbation.

We essentially see on the pendigits data (Figures 3(a) to 3(c)) that a value for σ chosen too small has not much impact on the

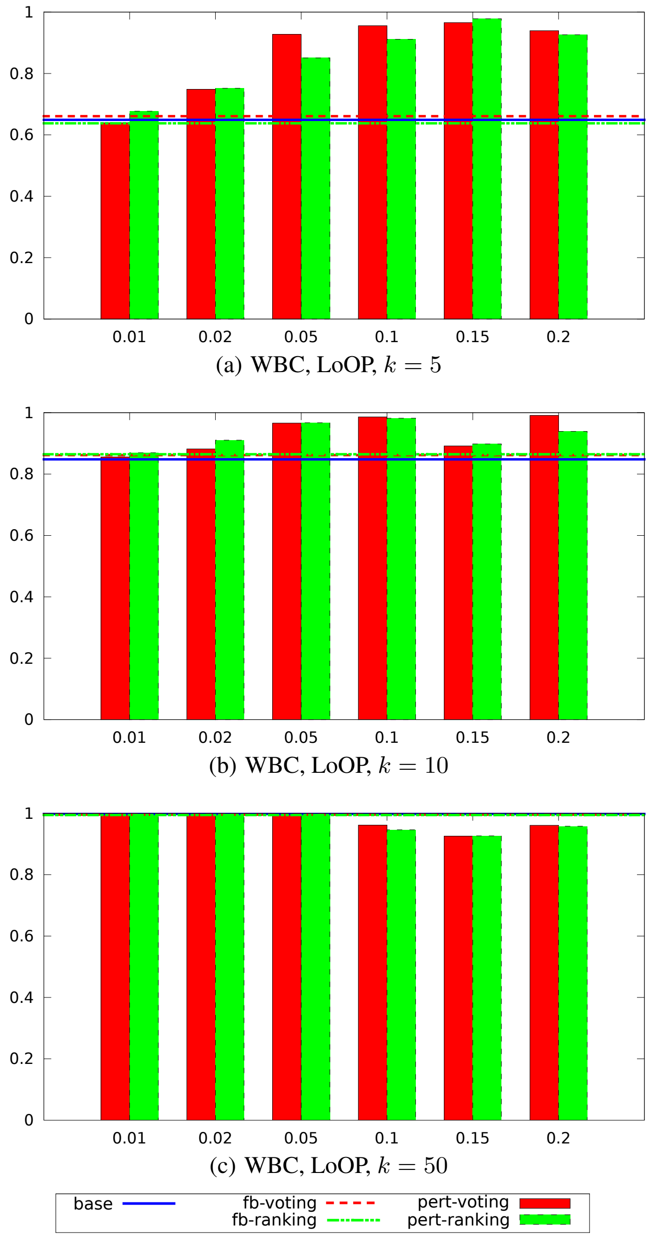


(a) Pendigits, LoOP, $k = 5$

(b) Pendigits, LoOP, $k = 10$

(c) Pendigits, LoOP, $k = 50$

Figure 3: ROC AUC of the perturbation ensemble (mean score voting and rank accumulation) over Pendigits data, based on LoOP against LoOP as baseline and feature bagging (mean score voting and rank accumulation) for different magnitudes of perturbance (x-axis: scaling factor of σ).



(a) WBC, LoOP, $k = 5$

(b) WBC, LoOP, $k = 10$

(c) WBC, LoOP, $k = 50$

Figure 4: ROC AUC of the perturbation ensemble (mean score voting and rank accumulation) over WBC data, based on LoOP against LoOP as baseline and feature bagging (mean score voting and rank accumulation) for different magnitudes of perturbance (x-axis: scaling factor of σ).

ensemble performance as compared to the base method. If σ is chosen large enough, the impact is significant, especially in case of a relatively bad performing base method. For higher parameters k the results are similar, LoOP is performing already almost perfect and the room for further improvement of the ensemble becomes small (see Figure 3(c) for $k = 50$).

For the WBC data (Figures 4(a) to 4(c)), the results are similar. These data are easier, however, and the performance of the base method (we show LoOP again, the results for the other methods are similar) is almost at a ROC AUC value of 1 for a parameter value $k = 50$ (Figure 4(c)). For such a good performance of the base method, the ensemble cannot improve the results anymore and shows a very similar performance to the base method. Again, however, for a worse choice of k for LoOP (Figures 4(a), 4(b)), the performance gain of the perturbation ensemble over the base method can be substantial, also when compared to the feature bagging ensemble which here, actually, does not improve over the base method.

5.4.2 Performance for Different Parameter Values

For the next plots, we vary parameter values of the base method for a constant magnitude of perturbation. On the WBC data, we show LDOF for a small (Figure 5(a)) and a large (Figure 5(b)) value of σ for the perturbation. Clearly, for a bad parametrization of the base method, a larger perturbance has a higher potential of improvement, but is also more likely to deteriorate in case of a good parametrization of the base method where LDOF already reaches ROC AUC values near 1 ($k = 50$).

Similar are the results in this respect for LOF (Figures 5(c), 5(d)), except that LOF performs better than LDOF on these data with the same parameter values.

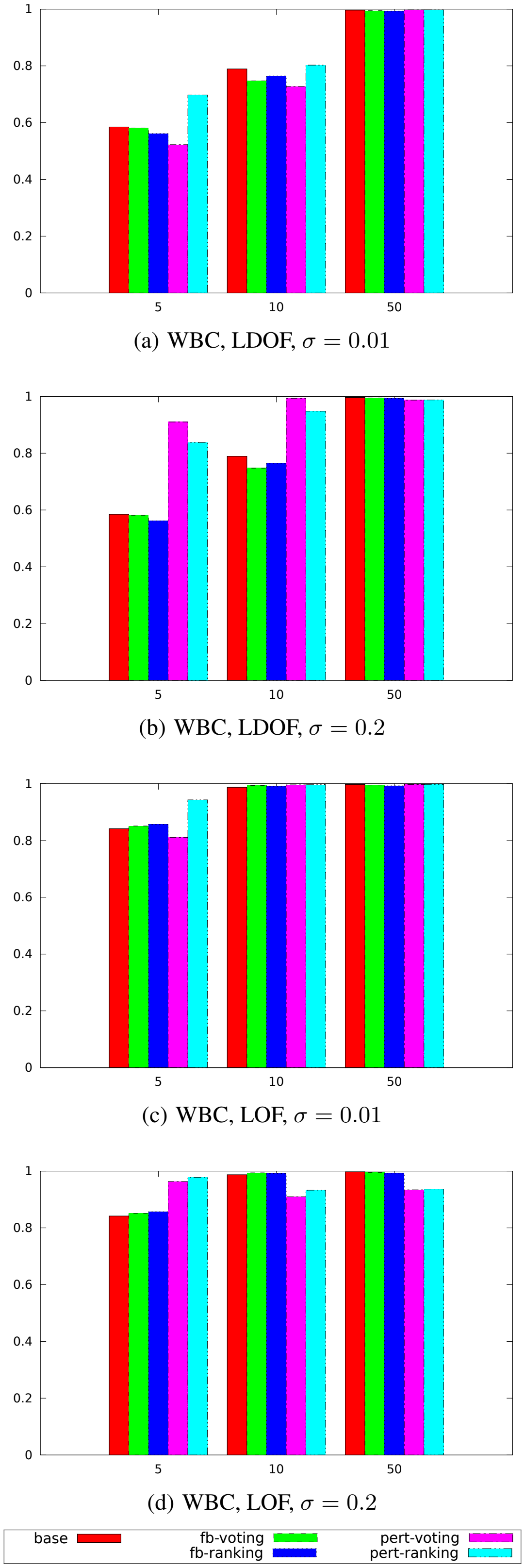
For the waveform data (Figures 6(a) to 6(d)), we confirm again that the perturbation ensemble improves more distinctly for bad parameter choices (neighborhood too small for these data) but for better parameter choices for the base method, the perturbation ensemble performs on a comparable level w.r.t. both the base method and the feature bagging variants (that do not improve here either).

On the cardio data, we see a strong potential of the perturbation ensemble to improve especially very bad results (Figures 7(a) to 7(d), examples are kNN, LDOF, LOF, and LoOP, the figures for the other methods are similar). The original methods for small parameter values all have ROC AUC values near 0.5, i.e., almost random performance. The perturbation ensemble can recover from that to ROC AUC values around 0.8 or even 0.9. Given the variance of the performance in the single perturbed data instances, these performance boosts exhibit z-scores in the range of 5 to 7 in these cases.

6. CONCLUSION

We proposed data perturbation as a means to induce diversity among individual outlier detectors and to build an ensemble with suitable combinations of the resulting outlier scores or ranks. We introduced a rank accumulation method that is far more suitable for outlier rank combination than the breadth-first traversal rank combination used in the feature bagging [30] method. Although there is no clear superiority of the rank accumulation over the average score combination, or vice versa, this result shows the potential of refined rank combination strategies for outlier detection ensembles.

The fundamental and flexible technique of data perturbance can be combined with a variety of conventional outlier detection techniques. For any of these methods, the parametrization is difficult and different parameters can lead to results of highly varying



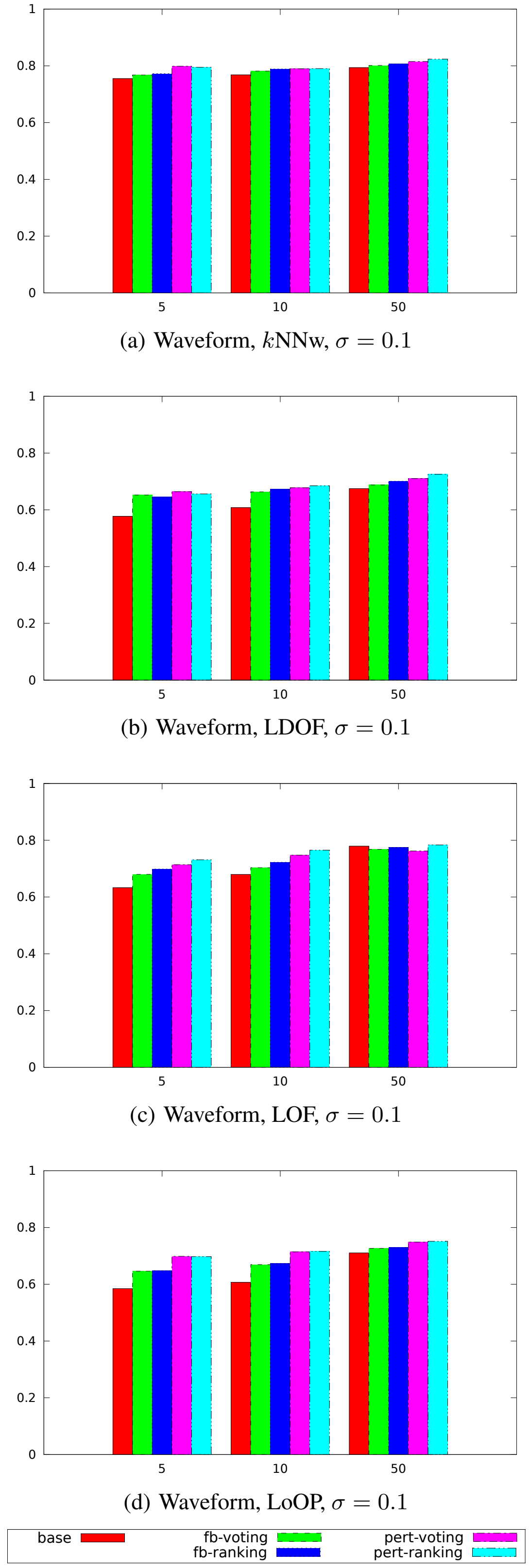
(a) WBC, LDOF, $\sigma = 0.01$

(b) WBC, LDOF, $\sigma = 0.2$

(c) WBC, LOF, $\sigma = 0.01$

(d) WBC, LOF, $\sigma = 0.2$

Figure 5: ROC AUC over WBC data, different base methods with different parametrizations of the base methods (x-axis: k) and performance of ensemble methods based on the respective base method: feature bags vs. perturbed data.



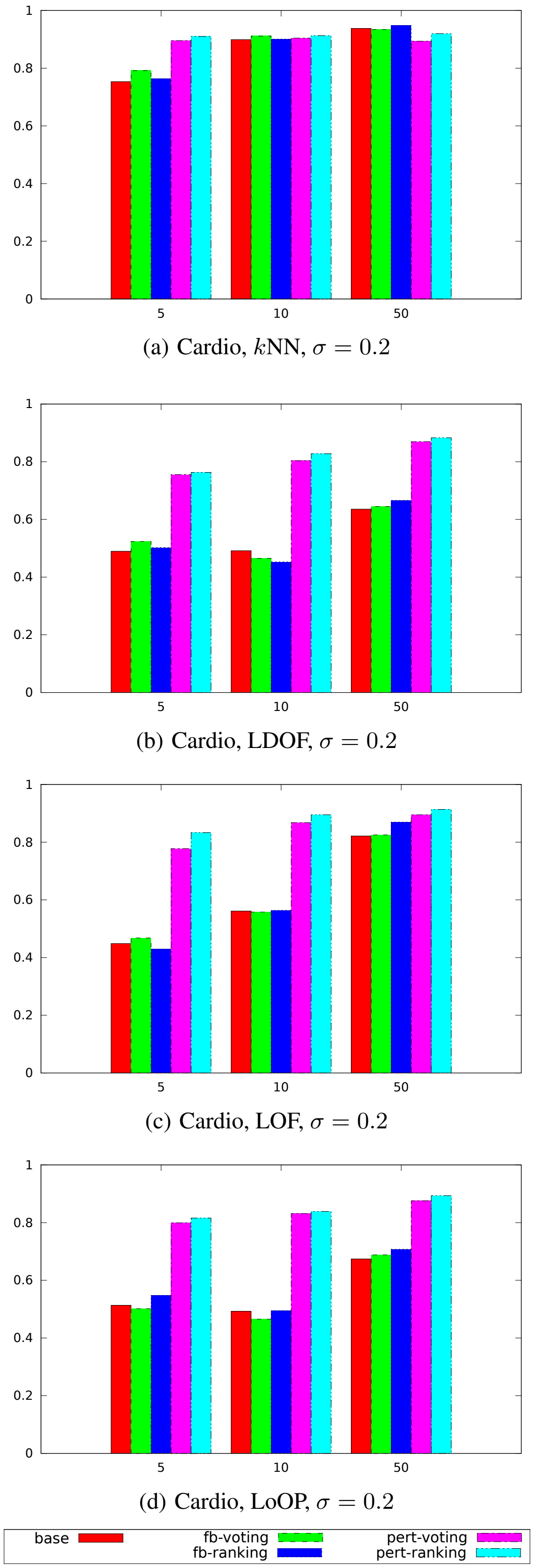
(a) Waveform, kNNw, $\sigma = 0.1$

(b) Waveform, LDOF, $\sigma = 0.1$

(c) Waveform, LOF, $\sigma = 0.1$

(d) Waveform, LoOP, $\sigma = 0.1$

Figure 6: ROC AUC over Waveform data, different base methods with different parametrizations of the base methods (x-axis: k) and performance of ensemble methods based on the respective base method: feature bags vs. perturbed data.



(a) Cardio, kNN, $\sigma = 0.2$

(b) Cardio, LDOF, $\sigma = 0.2$

(c) Cardio, LOF, $\sigma = 0.2$

(d) Cardio, LoOP, $\sigma = 0.2$

Figure 7: ROC AUC over Cardio data, different base methods with different parametrizations of the base methods (x-axis: k) and performance of ensemble methods based on the respective base method: feature bags vs. perturbed data.

quality (we demonstrated results varying between random performance and almost perfect performance). Outlier detection ensembles based on perturbance exhibit a remarkable potential for recovering from almost random performance of a base method (e.g., slightly above 0.5 ROC AUC for a poor choice of parameters) to values around 0.9 ROC AUC of the perturbation ensemble (feature bagging does not show this behavior).

We note that this positive potential comes along with the risk of *slightly* deteriorating an ensemble performance if the base method is already working extremely well. Our experiments show, however, that in most cases the perturbation ensemble on top of already strong base learners performs comparably.

As pointed out recently [48], research on ensemble methods for outlier detection is in an early stage but already showed promising results. Two fundamental issues for ensembles for outlier detection are methods or principles used to create diversity among outlier models and meaningful ways to combine outlier rankings produced by such diverse outlier models. Here, we contributed a new approach to both of these fundamental questions.

7. REFERENCES

[1] N. Abe, B. Zadrozny, and J. Langford. Outlier detection by active learning. In *Proceedings of the 12th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Philadelphia, PA*, pages 504–509, 2006.

[2] E. Achtert, H.-P. Kriegel, L. Reichert, E. Schubert, R. Wojdanowski, and A. Zimek. Visual evaluation of outlier detection models. In *Proceedings of the 15th International Conference on Database Systems for Advanced Applications (DASFAA), Tsukuba, Japan*, pages 396–399, 2010.

[3] E. Achtert, H.-P. Kriegel, E. Schubert, and A. Zimek. Interactive data mining with 3D-Parallel-Coordinate-Trees. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD), New York City, NY*, pages 1009–1012, 2013.

[4] C. C. Aggarwal. Outlier ensembles [position paper]. *ACM SIGKDD Explorations*, 14(2):49–58, 2012.

[5] F. Angiulli and C. Pizzuti. Fast outlier detection in high dimensional spaces. In *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discoverys (PKDD), Helsinki, Finland*, pages 15–26, 2002.

[6] V. Barnett and T. Lewis. *Outliers in Statistical Data*. John Wiley&Sons, 3rd edition, 1994.

[7] R. J. Beckman and R. D. Cook. Outlier..........s. *Technometrics*, 25(2):119–149, 1983.

[8] A. Bertoni and G. Valentini. Ensembles based on random projections to improve the accuracy of clustering algorithms. In *16th Italian Workshop on Neural Nets (WIRN), and International Workshop on Natural and Artificial Immune Systems (NAIS), Vietri sul Mare, Italy*, pages 31–37, 2005.

[9] M. Bittner, P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, M. Hendrix, M. Radmacher, R. Simon, Z. Yakhini, A. Ben-Dor, N. Sampas, E. Dougherty, E. Wang, F. Marincola, C. Gooden, J. Lueders, A. Glatfelter, P. Pollock, J. Carpten, E. Gillanders, D. Leja, K. Dietrich, C. Beaudry, M. Berens, D. Alberts, V. Sondak, N. Hayward, and J. Trent. Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature*, 406:536–540, 2000.

[10] M. M. Breunig, H.-P. Kriegel, R. Ng, and J. Sander. LOF: Identifying density-based local outliers. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD), Dallas, TX*, pages 93–104, 2000.

[11] G. Brown, J. Wyatt, R. Harris, and X. Yao. Diversity creation methods: a survey and categorisation. *Information Fusion*, 6:5–20, 2005.

[12] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):Article 15, 1–58, 2009.

[13] X. H. Dang, I. Assent, R. T. Ng, A. Zimek, and E. Schubert. Discriminative features for identifying and interpreting outliers. In *Proceedings of the 30th International Conference on Data Engineering (ICDE), Chicago, IL*, 2014.

[14] T. G. Dietterich. Ensemble methods in machine learning. In *First International Workshop on Multiple Classifier Systems (MCS), Cagliari, Italy*, pages 1–15, 2000.

[15] S. Dudoit and J. Fridlyand. Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19(9):1090–1099, 2003.

[16] V. Estivill-Castro. The instance easiness of supervised learning for cluster validity. In *Proceedings of the PAKDD Workshop on Quality Issues, Measures of Interestingness and Evaluation of Data Mining Models (QIMIE)*, pages 197–208, 2011.

[17] X. Z. Fern and C. E. Brodley. Random projection for high dimensional data clustering: A cluster ensemble approach. In *Proceedings of the 20th International Conference on Machine Learning (ICML), Washington, DC*, pages 186–193, 2003.

[18] A. Frank and A. Asuncion. UCI machine learning repository. `http://archive.ics.uci.edu/ml`, 2010.

[19] A. L. N. Fred and A. K. Jain. Robust data clustering. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), Madison, WI*, pages 128–136, 2003.

[20] J. Gao and P.-N. Tan. Converting output scores from outlier detection algorithms into probability estimates. In *Proceedings of the 6th IEEE International Conference on Data Mining (ICDM), Hong Kong, China*, pages 212–221, 2006.

[21] A. S. Hadi, A. H. M. Rahmatullah Imon, and M. Werner. Detection of outliers. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1):57–70, 2009.

[22] S. T. Hadjitodorov, L. I. Kuncheva, and L. P. Todorova. Moderate diversity for better cluster ensembles. *Information Fusion*, 7(3):264–275, 2006.

[23] J. Handl, J. Knowles, and D. B. Kell. Computational cluster validation in post-genomic data analysis. *Bioinformatics*, 21(15):3201–3212, 2005.

[24] L. Huang, D. Yan, M. I. Jordan, and N. Taft. Spectral clustering with perturbed data. In *Proceedings of the 22nd Annual Conference on Neural Information Processing Systems (NIPS), Vancouver, BC*, pages 705–712, 2008.

[25] F. Keller, E. Müller, and K. Böhm. HiCS: high contrast subspaces for density-based outlier ranking. In *Proceedings of the 28th International Conference on Data Engineering (ICDE), Washington, DC*, 2012.

[26] M. K. Kerr and G. A. Churchill. Bootstrapping cluster analysis: Assessing the reliability of conclusions from microarray experiments. *Proceedings of the National Academy of Sciences of the United States of America*, 98(16):8961–8965, 2001.

[27] E. M. Knorr and R. T. Ng. A unified notion of outliers: Properties and computation. In *Proceedings of the 3rd ACM International Conference on Knowledge Discovery and Data Mining (KDD), Newport Beach, CA*, pages 219–222, 1997.

[28] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. LoOP: local outlier probabilities. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM), Hong Kong, China*, pages 1649–1652, 2009.

[29] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. Interpreting and unifying outlier scores. In *Proceedings of the 11th SIAM International Conference on Data Mining (SDM), Mesa, AZ*, pages 13–24, 2011.

[30] A. Lazarevic and V. Kumar. Feature bagging for outlier detection. In *Proceedings of the 11th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Chicago, IL*, pages 157–166, 2005.

[31] C. Li and W. H. Wong. Model-based analysis of oligonucleotide arrays: model validation, design issues and standard error application. *Genome Biology*, 2(8:research0032):1–11, 2001.

[32] F. T. Liu, K. M. Ting, and Z.-H. Zhou. Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(1):3:1–39, 2012.

[33] H. V. Nguyen, H. H. Ang, and V. Gopalkrishnan. Mining outliers with ensemble of heterogeneous detectors on random subspaces. In *Proceedings of the 15th International Conference on Database Systems for Advanced Applications (DASFAA), Tsukuba, Japan*, pages 368–383, 2010.

[34] S. Papadimitriou, H. Kitagawa, P. Gibbons, and C. Faloutsos. LOCI: Fast outlier detection using the local correlation integral. In *Proceedings of the 19th International Conference on Data Engineering (ICDE), Bangalore, India*, pages 315–326, 2003.

[35] E. S. Pearson and C. Chandra Sekar. The efficiency of statistical tools and a criterion for the rejection of outlying observations. *Biometrika*, 28(3/4):308–320, 1936.

[36] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD), Dallas, TX*, pages 427–438, 2000.

[37] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[38] P. J. Rousseeuw and M. Hubert. Robust statistics for outlier detection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):73–79, 2011.

[39] E. Schubert, R. Wojdanowski, A. Zimek, and H.-P. Kriegel. On evaluation of outlier rankings and outlier scores. In *Proceedings of the 12th SIAM International Conference on Data Mining (SDM), Anaheim, CA*, pages 1047–1058, 2012.

[40] E. Schubert, A. Zimek, and H.-P. Kriegel. Generalized outlier detection with flexible kernel density estimates. In *Proceedings of the 14th SIAM International Conference on Data Mining (SDM), Philadelphia, PA*, 2014.

[41] E. Schubert, A. Zimek, and H.-P. Kriegel. Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection. *Data Mining and Knowledge Discovery*, 28(1):190–237, 2014.

[42] T. Soler and M. Chin. On transformation of covariance matrices between local Cartesian coordinate systems and commutative diagrams. In *ASP-ACSM Convention*, pages 393–406, 1985.

[43] A. Strehl and J. Ghosh. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.

[44] A. Topchy, A. Jain, and W. Punch. Clustering ensembles: Models of concensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1866–1881, 2005.

[45] G. Valentini and F. Masulli. Ensembles of learning machines. In *Proceedings of the 13th Italian Workshop on Neural Nets, Vietri, Italy*, pages 3–22, 2002.

[46] J. Yang, N. Zhong, Y. Yao, and J. Wang. Local peculiarity factor and its application in outlier detection. In *Proceedings of the 14th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Las Vegas, NV*, pages 776–784, 2008.

[47] K. Zhang, M. Hutter, and H. Jin. A new local distance-based outlier detection approach for scattered real-world data. In *Proceedings of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), Bangkok, Thailand*, pages 813–822, 2009.

[48] A. Zimek, R. J. G. B. Campello, and J. Sander. Ensembles for unsupervised outlier detection: Challenges and research questions [position paper]. *ACM SIGKDD Explorations*, 15(1):11–22, 2013.

[49] A. Zimek, M. Gaudet, R. J. G. B. Campello, and J. Sander. Subsampling for efficient and effective unsupervised outlier detection ensembles. In *Proceedings of the 19th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Chicago, IL*, 2013.

[50] A. Zimek, E. Schubert, and H.-P. Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining*, 5(5):363–387, 2012.