Learning Efficient Anomaly Detectors from K-NN Graphs

Jing Qian
Boston University

Jonathan Root
Boston University

Venkatesh Saligrama
Boston University

Abstract

We propose a non-parametric anomaly detection algorithm for high dimensional data. We score each datapoint by its average K-NN distance, and rank them accordingly. We then train limited complexity models to imitate these scores based on the max-margin learning-to-rank framework. A test-point is declared as an anomaly at α-false alarm level if the predicted score is in the α-percentile. The resulting anomaly detector is shown to be asymptotically optimal in that for any false alarm rate α, its decision region converges to the α-percentile minimum volume level set of the unknown underlying density. In addition, we test both the statistical performance and computational efficiency of our algorithm on a number of synthetic and real-data experiments. Our results demonstrate the superiority of our algorithm over existing K-NN based anomaly detection algorithms, with significant computational savings.

1 Introduction

Anomaly detection is the problem of identifying statistically significant deviations in data from expected normal behavior. It has found wide applications in many areas such as credit card fraud detection, intrusion detection for cyber security, sensor networks and video surveillance [Chandola et al., 2009, Hodge and Austin, 2004].

In classical parametric methods [Basseville et al., 1993] for anomaly detection, we assume the existence of a family of functions characterizing the nominal density (the test data consists of examples belonging to two classes–nominal and anomalous). Parameters are then estimated from training data by minimizing a loss function. While these methods provide a statistically justifiable solution when the assumptions hold true, they are likely to suffer from model mismatch, and lead to poor performance.

We focus on the non-parametric approach, with a view towards minimum volume (MV) set estimation. Given $\alpha \in (0, 1)$, the MV approach attempts to find the set of minimum volume which has probability mass at least $1 - \alpha$ with respect to the unknown sample probability distribution. Then given a new test point, it is declared to be consistent with the data if it lies in this MV set.

Approaches to the MV set estimation problem include estimating density level sets [Nunez-Garcia et al., 2003, Cuevas and Rodriguez-Casal, 2003] or estimating the boundary of the MV set [Scott and Nowak, 2006, Park et al., 2010]. However, these approaches suffer from high sample complexity, and therefore are statistically unstable using high dimensional data. The authors of [Zhao and Saligrama, 2009] score each test point using the K-NN distance. Scores turn out to yield empirical estimates of the volume of minimum volume level sets containing the test point, and avoids computing any high dimensional quantities. The papers [Hero, 2006, Sricharan and Hero, 2011] also take a K-NN based approach to MV set anomaly detection. The second paper [Sricharan and Hero, 2011] improves upon the computational performance of [Hero, 2006]. However, the test stage runtime of [Sricharan and Hero, 2011] is of order $O(dn)$, d being the ambient dimension and n the sample size. The test stage runtime of [Zhao and Saligrama, 2009] is of order $O(dn^2 + n^2 \log(n))$.

Computational inefficiencies of these K-NN based anomaly detection methods suggests that a different approach based on distance-based (DB) outlier methods (see [Orair et al., 2010] and references therein) could possibly be leveraged in this context. DB methods primarily focus on the *computational* issue of identifying a pre-specified number of L points (outliers) with largest K-NN distances in a database. Outliers are identified by pruning examples with small K-NN

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.
Copyright 2015 by the authors.

distance. This works particularly well for small L.

In contrast, for anomaly detection, we not only need an efficient scheme but also one that takes training data (containing no anomalies) and generalizes well in terms of AUC criterion on test-data where the number of anomalies is unknown. We need schemes that predict "anomalousness" for test-instances in order to adapt to any false-alarm-level and to characterize AUCs. One possible way to leverage DB methods is to estimate anomaly scores based only on the L identified outliers but this scheme generally has poor AUC performance if there are a sizable fraction of anomalies. In this context [Liu et al., 2008, Ting et al., 2010, Sricharan and Hero, 2011] propose to utilize ORCA [Bay and Schwabacher, 2003]. ORCA is a well-known *ranking DB* method that provides intermediate estimates for every instance in addition to the L outliers. They show that while for small L ORCA is highly efficient its AUC performance is poor. For large L ORCA produces low but somewhat meaningful AUCs but can be computationally inefficient. A basic reason for this AUC gap is that although such *rank-based* DB techniques provide intermediate KNN estimates & outlier scores that can possibly be leveraged, these estimates/scores are often too unreliable for anomaly detection purposes. Recently, [Wang et al., 2011] have considered strategies based on LSH to further speed up rank based DB methods. Our perspective is that this direction is somewhat complementary. Indeed, we could also employ Kernel-LSH [Kulis and Grauman, 2009] in our setting to further speed up our computation.

In this paper, we propose a ranking based algorithm which retains the statistical complexity of existing K-NN work, but with far superior computational performance. Using scores based on average KNN distance, we learn a functional predictor through the pair-wise learning-to-rank framework, to predict p-value scores. This predictor is then used to generalize over unseen examples. The test time of our algorithm is of order $O(ds)$, where s is the complexity of our model.

The rest of the paper is organized as follows. In Section 2 we introduce the problem setting and the motivation. Detailed algorithms are described in Section 3 and 4. The asymptotic and finite-sample analyses are provided in Section 5. Synthetic and real experiments are reported in Section 6.

2 Problem Setting & Motivation

Let $\mathbf{x} = \{x_1, x_2, ..., x_n\}$ be a given set of nominal d-dimensional data points. We assume $\mathbf{x}$ to be sampled i.i.d from an unknown density f_0 with compact support in $\mathbb{R}^d$. The problem is to assume a new data point, $\eta \in \mathbb{R}^d$, is given, and test whether η follows the distribution of $\mathbf{x}$. If f denotes the density of this new (random) data point, then the set-up is summarized in the following hypothesis test:

$$H_0 : f = f_0 \quad \text{vs.} \quad H_1 : f \neq f_0.$$

We look for a functional $D : \mathbb{R}^d \to \mathbb{R}$ such that $D(\eta) > 0 \implies \eta$ nominal. Given such a D, we define its corresponding acceptance region by $A = \{x : D(x) > 0\}$. We will see below that D can be defined by the p-value.

Given a prescribed significance level (false alarm level) $\alpha \in (0, 1)$, we require the probability that η *does not* deviate from the nominal ($\eta \in A$), given H_0, to be bounded below by $1 - \alpha$. We denote this distribution by P (sometimes written $P(\text{not } H_1 | H_0)$):

$$P(A) = \int_A f_0(x)\, dx \geq 1 - \alpha.$$

Said another way, the probability that η *does* deviate from the nominal, given H_0, should fall under the specified significance level α (i.e. $1 - P(A) = P(H_1|H_0) \leq \alpha$). At the same time, the false negative, $\int_A f(x)\, dx$, must be minimized. Note that the false negative is the probability of the event $\eta \in A$, given H_1. We assume f to be bounded above by a constant C, in which case $\int_A f(x)\, dx \leq C \cdot \lambda(A)$, where λ is Lebesgue measure on $\mathbb{R}^d$. The problem of finding the most suitable acceptance region, A, can therefore be formulated as finding the following minimum volume set:

$$U_{1-\alpha} := \arg\min_A \left\{ \lambda(A) : \int_A f_0(x)\, dx \geq 1 - \alpha \right\}. \quad (1)$$

In words, we seek a set A which captures at least a fraction $1 - \alpha$ of the probability mass, of minimum volume.

3 Score Functions Based on K-NNG

In this section, we briefly review an algorithm using score functions based on nearest neighbor graphs for determining minimum volume sets [Zhao and Saligrama, 2009, Qian and Saligrama, 2012]. Given a test point $\eta \sim f$, define the p-value of η by

$$p(\eta) := P\left(x : f_0(x) \leq f_0(\eta)\right) = \int_{\{x : f_0(x) \leq f_0(\eta)\}} f_0(x)\, dx.$$

Then, assuming technical conditions on the density f_0 [Zhao and Saligrama, 2009], it can be shown that p defines the minimum volume set:

$$U_{1-\alpha} = \{x : p(x) \geq \alpha\}.$$

Thus if we know p, we know the minimum volume set, and we can declare anomaly simply by checking

whether or not $p(\eta) < \alpha$. However, p is based on information from the unknown density f_0, hence we must estimate p.

Set $d(x, y)$ to be the Euclidean metric on $\mathbb{R}^d$. Given a point $x \in \mathbb{R}^d$, we form its associated K nearest neighbor graph (K-NNG), relative to $\mathbf{x}$, by connecting it to the K closest points in $\mathbf{x} \setminus \{x\}$. Let $D_{(i)}(x)$ denote the distance from x to its ith nearest neighbor in $\mathbf{x} \setminus \{x\}$. Set

$$G_{\mathbf{x}}(x) = \frac{1}{K} \sum_{j=1}^{K} D_{(j)}(x). \tag{2}$$

Now define the following score function:

$$R_n(\eta) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{G_{\mathbf{x}}(\eta) < G_{\mathbf{x}}(x_i)\}} \tag{3}$$

This function measures the relative concentration of point η compared to the training set. In [Qian and Saligrama, 2012], given a pre-defined significance level α (e.g. 0.05), they declare η to be anomalous if $R_n(\eta) \leq \alpha$. This choice is motivated by its close connection to multivariate p-values. Indeed, it is shown in [Qian and Saligrama, 2012] that this score function is an asymptotically consistent estimator of the p-value:

$$\lim_{n \to \infty} R_n(\eta) = p(\eta) \quad \text{a.s.}$$

This result is attractive from a statistical viewpoint, however the test-time complexity of the K-NN distance statistic grows as $O(dn)$. This can be prohibitive for real-time applications. Thus we are compelled to learn a score function respecting the K-NN distance statistic, but with significant computational savings. This is achieved by mapping the data set $\mathbf{x}$ into a reproducing kernel Hilbert space (RKHS), H, with kernel k and inner product $\langle \cdot, \cdot \rangle$. We denote by Φ the mapping $\mathbb{R}^d \to H$, defined by $\Phi(x_i) = k(x_i, \cdot)$. We then optimally learn a ranker $g \in H$ based on the ordered pair-wise ranking information,

$$\{(i, j) : G_{\mathbf{x}}(x_i) > G_{\mathbf{x}}(x_j)\}$$

and construct the scoring function as

$$\hat{R}_n(\eta) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{\langle g, \Phi(\eta) \rangle < \langle g, \Phi(x_i) \rangle\}}. \tag{4}$$

It turns out that $\hat{R}$ is an asymptotic estimator of the p-value (see Section 5) and thus we will say a test point η is anomalous if $\hat{R}(\eta) \leq \alpha$.

4 Anomaly Detection Algorithm

In this section we describe our rank-based anomaly detection algorithm (RankAD), and discuss several of its properties and advantages.

Algorithm 1: RankAD Algorithm

1. Input: Nominal training data $\mathbf{x} = \{x_1, x_2, ..., x_n\}$, desired false alarm level α, and test point η.

2. Training Stage:

(a) Calculate Kth nearest neighbor distances $G_{\mathbf{x}}(x_i)$, and calculate $R_n(x_i)$ for each nominal sample x_i, using Eq.(2) and Eq.(3).

(b) Quantize $\{R_n(x_i),\, i = 1, 2, ..., n\}$ uniformly into m levels: $r_q(x_i) \in \{1, 2, ..., m\}$. Generate preference pairs (i, j) whenever their quantized levels are different: $r_q(x_i) > r_q(x_j)$.

(c) Set $\mathcal{P} = \{(i, j) : r_q(x_i) > r_q(x_j)\}$. Solve:

$$\begin{aligned} \min_{g, \xi_{ij}} : \quad & \frac{1}{2}||g||^2 + C \sum_{(i,j) \in \mathcal{P}} \xi_{ij} \\ s.t. \quad & \langle g,\, \Phi(x_i) - \Phi(x_j) \rangle \geq 1 - \xi_{ij}, \quad \forall (i, j) \in \mathcal{P} \\ & \xi_{ij} \geq 0 \end{aligned} \tag{5}$$

(d) Let $\hat{g}$ denote the minimizer. Compute and sort: $\hat{g}(\cdot) = \langle \hat{g}, \Phi(\cdot) \rangle$ on $\mathbf{x} = \{x_1, x_2, ..., x_n\}$.

3. Testing Stage:

(a) Evaluate $\hat{g}(\eta)$ for test point η.

(b) Compute the score: $\hat{R}_n(\eta) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{\hat{g}(\eta) < \hat{g}(x_i)\}}$. This can be done through a binary search over sorted $\{\hat{g}(x_i), i = 1, ..., n\}$.

(c) Declare η as anomalous if $\hat{R}_n(\eta) \leq \alpha$.

Remark 1: The standard learning-to-rank setup [Joachims, 2002] is to assume non-noisy input pairs. Our algorithm is based on noisy inputs, where the noise is characterized by an unknown, high-dimensional distribution. Yet we are still able to show the asymptotic consistency of the obtained ranker in Sec.5.

Remark 2: For the learning-to-rank step Eq.(5), we equip the RKHS H with the RBF kernel $k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{\sigma^2}\right)$. The algorithm parameter C and RBF kernel bandwidth σ can be selected through cross validation, since this step is a supervised learning procedure based on input pairs. We use cross validation and adopt the weighted pairwise disagreement loss (WPDL) from [Lan et al., 2012] for this purpose.

Remark 3: The number of quantization levels, m, impacts training complexity as well as performance. When $m = n$, all $\binom{n}{2}$ preference pairs are generated.

This scenario has the highest training complexity. Furthermore, large m tends to more closely follow rankings obtained from K-NN distances, which may or may not be desirable. K-NN distances can be noisy for small training data sizes. While this raises the question of choosing m, we observe that setting m to be $3 \sim 5$ works fairly well in practice. We fix $m = 3$ in all of our experiments in Sec.6. $m = 2$ is insufficient to allow flexible false alarm control, as will be demonstrated next.

Remark 4: Let us mention the connection with ranking SVM. Ranking SVM is an algorithm for the learning-to-rank problem, whose goal is to rank unseen objects based on given training data and their corresponding orderings. Our novelty lies in building a connection between learning-to-rank and anomaly detection:
(1) While there is no such natural "input ordering" in anomaly detection, we create this order on training samples through their K-NN scores.
(2) When we apply our detector on an unseen object it produces a score that approximates the unseen object's p-value. We theoretically justify this linkage, namely, our predictions fall in the right quantile (Theorem 3). We also empirically show test-stage computational benefits.

4.1 False alarm control

In this section we illustrate through a toy example how our learning method approximates minimum volume sets. We consider how different levels of quantization impact level sets. We will show that for appropriately chosen quantization levels our algorithm is able to simultaneously approximate multiple level sets. In Section 5 we show that the normalized score Eq.(4), takes values in $[0, 1]$, and converges to the p-value function. Therefore we get a handle on the false alarm rate. So null hypothesis can be rejected at different levels simply by thresholding $\hat{R}_n(\eta)$.

Toy Example:
We present a simple example in Fig. 1 to demonstrate this point. The nominal density $f \sim 0.5\mathcal{N}([4; 1], 0.5I) + 0.5\mathcal{N}([4; -1], 0.5I)$. We first consider single-bit quantization ($m = 2$) using RBF kernels ($\sigma = 1.5$) trained with pairwise preferences between p-values above and below 3%. This yields a decision function $\hat{g}_2(\cdot)$. The standard way is to claim anomaly when $\hat{g}_2(x) < 0$, corresponding to the outmost orange curve in (a). We then plot different level curves by varying $c > 0$ for $\hat{g}_2(x) = c$, which appear to be scaled versions of the orange curve. While this quantization appears to work reasonably for α-level sets with $\alpha = 3\%$, for a different desired α-level, the algorithm would have to retrain with new preference pairs. On the other hand, we also train rankAD with $m = 3$ (uniform quantization) and obtain the ranker $\hat{g}_3(\cdot)$. We then vary c for $\hat{g}_3(x) = c$ to obtain various level curves shown in (b), all of which surprisingly approximate the corresponding density level sets well. We notice a significant difference between the level sets generated with 3 quantization levels in comparison to those generated for two-level quantization. In the appendix we show that $\hat{g}(x)$ asymptotically preserves the ordering of the density, and from this conclude that our score function $\hat{R}_n(x)$ approximates multiple density level sets (p-values). Also see Section 5 for a discussion of this. However in our experiments it turns out that we just need $m = 3$ quantization levels instead of $m = n$ ($\binom{n}{2}$ pairs) to achieve flexible false alarm control and do not need any re-training.

4.2 Time Complexity

For training, the rank computation step requires computing all pair-wise distances among nominal points $O(dn^2)$, followed by sorting for each point $O(n^2 \log n)$. So the training stage has the total time complexity $O(n^2(d + \log n) + T)$, where T denotes the time of the pair-wise learning-to-rank algorithm. At test stage, our algorithm only evaluates $\hat{g}(\eta)$ on η and does a binary search among $\hat{g}(x_1), \ldots, \hat{g}(x_n)$. The complexity is $O(ds + \log n)$, where s is the number of support vectors. This has some similarities with one-class SVM where the complexity scales with the number of support vectors [Schölkopf et al., 2001]. Note that in contrast nearest neighbor-based algorithms, K-LPE, aK-LPE or BP-K-NNG [Zhao and Saligrama, 2009, Qian and Saligrama, 2012, Sricharan and Hero, 2011], require $O(nd)$ for testing one point. It is worth noting that $s \leq n$ comes from the "support pairs" within the input preference pair set. Practically we observe that for most data sets s is much smaller than n in the experiment section, leading to significantly reduced test time compared to aK-LPE, as shown in Table.1. It is worth mentioning that distributed techniques for speeding up computation of K-NN distances [Bhaduri et al., 2011] can be adopted to further reduce test stage time.

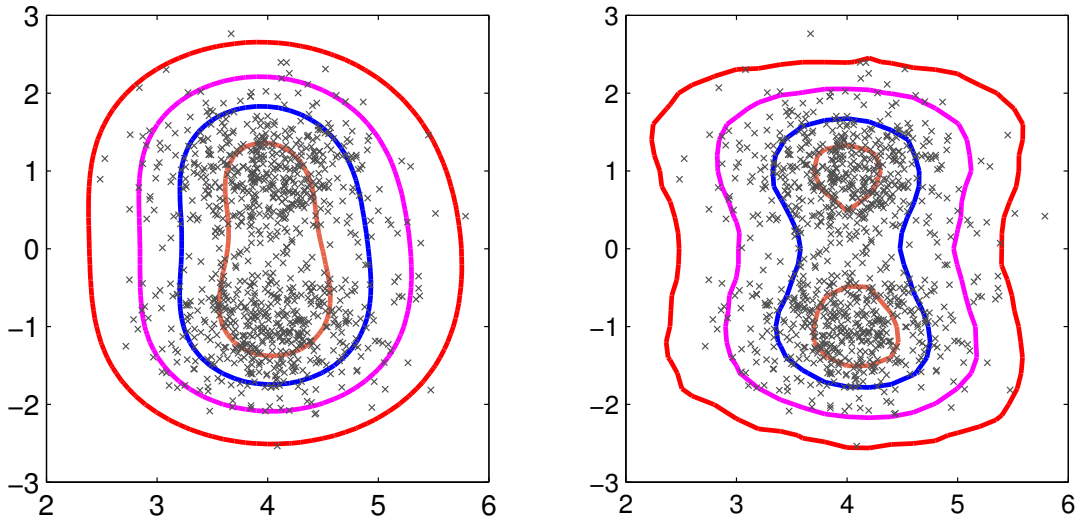
5 Analysis

In this section we present the theoretical analysis of our ranking-based anomaly detection approach.

5.1 Asymptotic Consistency

As mentioned earlier in the paper, it is shown in [Qian and Saligrama, 2012] that the average K-NN distance statistic converges to the p-value function:

Theorem 1. *With $K = O(n^{0.5})$, we have*



(a) Level curves ($m = 2$) (b) Level curves ($m = 3$)

Figure 1: Level curves of rankAD for different quantization levels. 1000 i.i.d. samples are drawn from a 2-component Gaussian mixture density. Left figure(a) depicts performance with single-bit quantization ($m = 2$). To learn rankAD we quantized preference pairs at 3% and $\sigma = 1.5$ in our RBF kernel. Right figure(b) shows rankAD with 3-levels of quantization and $\sigma = 1.5$. (a) shows level curves obtained by varying the offset c for $\hat{g}_2(x) = c$. Only the outmost curve ($c = 0$) approximates the oracle density level set well while the inner curves ($c > 0$) appear to be scaled versions of outermost curve. (b) shows level curves obtained by varying c for $\hat{g}_3(x) = c$. Interestingly we observe that the inner most curve approximates peaks of the mixture density.

$\lim_{n\to\infty} R_n(\eta) = p(\eta)$.

The goal of our rankAD algorithm is to learn the ordering of the p-value. This theorem therefore guarantees that asymptotically, the preference pairs generated as input to the rankAD algorithm are reliable. Note that the definition of G in [Qian and Saligrama, 2012] is slightly different than the one given in equation (2). However, for our purposes this difference is not worth detailing.

What we claim in this paper, and prove in the appendix, is the following consistency result of our rankAD algorithm. Note that the use of quantization (c.f. Section 4) does not affect the conclusion of this theorem, hence we assume there is none. Indeed, quantization is a computational tool. From a statistical asymptotic consistency perspective quantization is not an issue.

Theorem 2. *With $K = O(n^{0.5})$, as $n \to \infty$, $\hat{R}_n(\eta) \to p(\eta)$.*

The difficulty in this theorem arises from the fact that the score, $\hat{R}_n(\eta)$, is based on the ranker, $\hat{g}$, which is learned from data with high-dimensional noise. Moreover, the noise is distributed according to an *unknown* probability measure. For the proof of this theorem, we begin with the law of large numbers. Suppose for any $n \geq 1$, a function G is found such that $f(x_i) < f(x_j) \implies G(x_i) < G(x_j)$. Note that in Section 3 we use K-NN distance surrogates which reverses the order but the effect is the same and should not cause any confusion. Then it can be shown that

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{G(x_i)<G(\eta)\}} \to p(\eta).$$

Thus we wish to prove that the output of our rankAD algorithm is such a function.

The first step in our proof is to show that the solution to our rankAD algorithm, $\hat{g}$, is consistent [Steinwart, 2001]. Fix an RKHS H on the input space $X \subset \mathbb{R}^d$ with RBF kernel k. We denote by L the hinge loss. We may write $\hat{g}$ as the solution to the following regularized minimization problem,

$$\hat{g} = \arg\min_{f\in H} \mathcal{R}_{L,T}(f) + \lambda_n \|f\|_H^2,$$

where $\mathcal{R}_{L,T}(f) = \frac{1}{n^2}\sum_{i,j} L(f(x_i) - f(x_j))$. T denotes the pairs from the sample $\mathbf{x} = \{x_1, \ldots, x_n\}$, so this is a loss with respect to the empirical measure. The expected risk is denoted

$$\mathcal{R}_{L,P}(f) = E_{\mathbf{x}}[\mathcal{R}_{L,T}(f)].$$

Then consistency means that, under appropriate conditions as $\lambda_n \to 0$ and $n \to \infty$ (see appendix), we have

$$E_{\mathbf{x}}[\mathcal{R}_{L,T}(\hat{g})] \to \min_{f\in H} \mathcal{R}_{L,P}(f). \tag{6}$$

The proof of this claim requires a concentration of measure result relating $\mathcal{R}_{L,T}(f)$ to its expectation, $\mathcal{R}_{L,P}(f)$, uniformly over $f \in H$. The argument follows closely that made in [Cucker and Smale, 2001], except we make use of McDiarmid's inequality.

Finally we show that if $\hat{g}$ satisfies (6), then it ranks samples according to their density: $f(x_i) > f(x_j) \implies \hat{g}(x_i) > \hat{g}(x_j)$.

5.2 Finite-Sample Generalization Result

Based on a sample $\{x_1, \ldots, x_n\}$, our approach learns a ranker g_n, and computes the values $g_n(x_1), \ldots, g_n(x_n)$. Let $g_n^{(1)} \leq g_n^{(2)} \leq \cdots \leq g_n^{(n)}$ be the ordered permutation of these values. For a test point η, we evaluate $g_n(\eta)$ and compute $\hat{R}_n(\eta)$. For a prescribed false alarm level α, we define the decision region for claiming anomaly by

$$\begin{aligned} R_\alpha &= \{x : \hat{R}_n(x) \leq \alpha\} \\ &= \{x : \sum_{j=1}^{n} \mathbf{1}_{\{g_n(x) \leq g_n(x_j)\}} \leq \alpha n\} \\ &= \{x : g_n(x) < g_n^{\lceil \alpha n \rceil}\} \end{aligned}$$

where $\lceil \alpha n \rceil$ denotes the ceiling integer of αn.

We give a finite-sample bound on the probability that a newly drawn nominal point η lies in R_α. In the following Theorem, $\mathcal{F}$ denotes a real-valued function class of kernel based linear functions equipped with the ℓ_∞ norm over a finite sample $\mathbf{x} = \{x_1, \ldots, x_n\}$:

$$\|f\|_{\ell_\infty^{\mathbf{x}}} = \max_{x \in \mathbf{x}} |f(x)|.$$

Note that $\mathcal{F}$ contain solutions to an SVM-type problem, so we assume the output of our rankAD algorithm, g_n, is an element of $\mathcal{F}$. We let $\mathcal{N}(\gamma, \mathcal{F}, n)$ denote the covering number of $\mathcal{F}$ with respect to this norm (see appendix for details).

Theorem 3. *Fix a distribution P on $\mathbb{R}^d$ and suppose $x_1, \ldots, x_n$ are generated iid from P. For $g \in \mathcal{F}$ let $g^{(1)} \leq g^{(2)} \leq \cdots \leq g^{(n)}$ be the ordered permutation of $g(x_1), \ldots, g(x_n)$. Then for such an n-sample, with probability $1 - \delta$, for any $g \in \mathcal{F}$, $1 \leq m \leq n$ and sufficiently small $\gamma > 0$,*

$$P\left\{x : g(x) < g^{(m)} - 2\gamma\right\} \leq \frac{m-1}{n} + \epsilon(n, k, \delta),$$

where $\epsilon(n, k, \delta) = \frac{2}{n}(k + \log \frac{n}{\delta})$, $k = \lceil \log \mathcal{N}(\gamma, \mathcal{F}, 2n) \rceil$.

Remarks

(1) To interpret the theorem notice that the LHS is precisely the probability that a test point drawn from the nominal distribution has a score below the $\alpha \approx \frac{m-1}{n}$ percentile. We see that this probability is bounded from above by α plus an error term that asymptotically approaches zero. This theorem is true irrespective of α and so we have shown that we can simultaneously approximate multiple level sets.
(2) A similar inequality holds for the event giving a lower bound on $g(x)$. However, let us emphasize that lower bounds are not meaningful for our context. The ranks $g^{(1)} \leq g^{(2)} \leq \cdots \leq g^{(n)}$ are sorted in increasing order. A smaller $g(x)$ signifies that x is more of an outlier. Points below the lowest rank $g^{(1)}$ correspond to the most extreme outliers.

6 Experiments

In this section, we carry out point-wise anomaly detection experiments on synthetic and real-world data sets. We compare our ranking-based approach against density-based methods BP-K-NNG [Sricharan and Hero, 2011] and aK-LPE [Qian and Saligrama, 2012], and two other state-of-art methods based on random sub-sampling, isolated forest [Liu et al., 2008] (iForest) and massAD [Ting et al., 2010]. One-class SVM [Schölkopf et al., 2001] is included as a baseline.

6.1 Implementation Details

In our simulations, the Euclidean distance is used as distance metric for all candidate methods. For one-class SVM the lib-SVM codes [Chang and Lin, 2011] are used. The algorithm parameter and the RBF kernel parameter for one-class SVM are set using the same configuration as in [Ting et al., 2010]. For iForest and massAD, we use the codes from the websites of the authors, with the same configuration as in [Ting et al., 2010]. For aK-LPE we use the average k-NN distance Eq.(2) with fixed $k = 20$ since this appears to work better than the actual K-NN distance of [Zhao and Saligrama, 2009]. Note that this is also suggested by the convergence analysis in Thm 1 [Qian and Saligrama, 2012]. For BP-K-NNG, the same k is used and other parameters are set according to [Sricharan and Hero, 2011].

For our rankAD approach we follow the steps described in Algorithm 1. We first calculate the ranks $R_n(x_i)$ of nominal points according to Eq.(3) based on aK-LPE. We then quantize $R_n(x_i)$ uniformly into m=3 levels $r_q(x_i) \in \{1, 2, 3\}$ and generate pairs $(i, j) \in \mathcal{P}$ whenever $r_q(x_i) > r_q(x_j)$. We adapt the routine from [Chapelle and Keerthi, 2010] and extend it to a kernelized version for the learning-to-rank step Eq.(5). The trained ranker is then adopted in Eq.(4) for test stage prediction. We point out some implementation details of our approach as follows.

(1) Resampling: We follow [Qian and Saligrama, 2012] and adopt the U-statistic based resampling to compute aK-LPE ranks. We randomly split the data into two equal parts and use one part as "nearest neighbors" to calculate the ranks (Eq.(2, 3)) for the other part and vice versa. Final ranks are averaged over 20 times of resampling.

(2) Quantization levels & K-NN For real experiments with 2000 nominal training points, we fix $k = 20$ and $m = 3$. These values are based on noting that the detection performance does not degrade significantly with smaller quantization levels for synthetic data. The k parameter in K-NN is chosen to be 20 and is based on Theorem 1 and results from synthetic experiments (see below).

(3) Cross Validation using pairwise disagreement loss: For the rank-SVM step we use a 4-fold cross validation to choose the parameters C and σ. We vary $C \in \{0.001, 0.003, 0.01, \ldots, 300, 1000\}$, and the RBF kernel parameter $\sigma \in \Sigma = \{2^i \tilde{D}_K,\ i = -10, -9, \ldots, 9, 10\}$, where $\tilde{D}_K$ is the average 20-NN distance over nominal samples. The pair-wise disagreement indicator loss is adopted from [Lan et al., 2012] for evaluating rankers on the input pairs:

$$L(f) = \sum_{(i,j)\in\mathcal{P}} \mathbf{1}_{\{f(x_i)<f(x_j)\}}$$

Reported AUC performances are averaged over 5 runs.

6.2 Synthetic Data sets

We first apply our method to a Gaussian toy problem, where the nominal density is:

$$f_0 \sim 0.2\mathcal{N}\left([5;0], [1,0;0,9]\right) + 0.8\mathcal{N}\left([-5;0], [9,0;0,1]\right).$$

Anomaly follows the uniform distribution within $\{(x, y) : -18 \le x \le 18, -18 \le y \le 18\}$. The goal here is to understand the impact of different parameters (k-NN parameter and quantization level) used by RankAD. Fig.2 shows the level curves for the estimated ranks on the test data. As indicated by the asymptotic consistency (Thm.2) and the finite sample analysis (Thm.3), the empirical level curves of rankAD approximate the level sets of the underlying density quite well. We vary k and m and evaluate the AUC performances of our approach shown in Table 1. The Bayesian AUC is obtained by thresholding the likelihood ratio using the generative densities. From Table 1 we see the detection performance is quite insensitive to the k-NN parameter and the quantization level parameter m, and for this simple synthetic example is close to Bayesian performance.

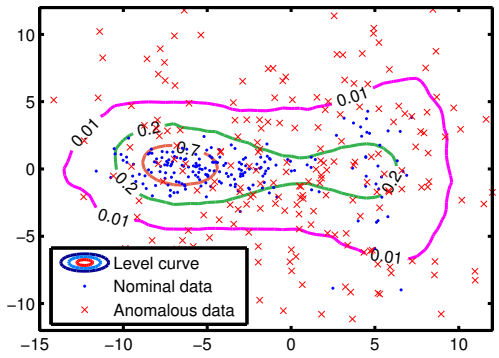


Figure 2: Level sets for the estimated ranks. 600 training points are used for training.

Table 1: AUC performances of Bayesian detector, aK-LPE, and rankAD with different values of k and m. 600 training points are used for training. For test 500 nominal and 1000 anomalous points are used.

AUC	k=5	k=10	k=20	k=40
m=3	0.9206	0.9200	0.9223	0.9210
m=5	0.9234	0.9243	0.9247	0.9255
m=7	0.9226	0.9228	0.9234	0.9213
m=10	0.9201	0.9208	0.9244	0.9196
aK-LPE	0.9192	0.9251	0.9244	0.9228
Bayesian	0.9290	0.9290	0.9290	0.9290

Table 2: Data characteristics of the data sets used in experiments. N is the total number of instances. d the dimension of data. The percentage in brackets indicates the percentage of anomalies among total instances.

data sets	N	d	anomaly class
Annthyroid	6832	6	classes 1,2
Forest Cover	286048	10	class 4 vs. class 2
HTTP	567497	3	attack
Mamography	11183	6	class 1
Mulcross	262144	4	2 clusters
Satellite	6435	36	3 smallest classes
Shuttle	49097	9	classes 2,3,5,6,7
SMTP	95156	3	attack

6.3 Real-world data sets

We conduct experiments on several real data sets used in [Liu et al., 2008] and [Ting et al., 2010], including 2 network intrusion data sets HTTP and SMTP from [Yamanishi et al., 2000], Annthyroid, Forest Cover Type, Satellite, Shuttle from UCI repository [Frank and Asuncion, 2010], Mammography and Mulcross from [Rocke and Woodruff, 1996]. Table 2 illustrates the characteristics of these data sets.

We randomly sample 2000 nominal points for training. The rest of the nominal data and all of the anomalous data are held for testing. Due to memory limit, at most 80000 nominal points are used at test time. The

Table 3: Anomaly detection AUC performance and test stage time of various methods.

Data Sets		rankAD	one-class svm	BP-K-NNG	aK-LPE	iForest	massAD
AUC	Annthyroid	0.844	0.681	0.823	0.753	**0.856**	0.789
	Forest Cover	**0.932**	0.869	0.889	0.876	0.853	0.895
	HTTP	**0.999**	0.998	0.995	**0.999**	0.986	0.995
	Mamography	**0.909**	0.863	0.886	0.879	0.891	0.701
	Mulcross	**0.998**	0.970	0.994	**0.998**	0.971	0.998
	Satellite	**0.885**	0.774	0.872	0.884	0.812	0.692
	Shuttle	**0.996**	0.975	0.985	0.995	0.992	0.992
	SMTP	**0.934**	0.751	0.902	0.900	0.869	0.859
test time	Annthyroid	0.338	0.281	2.171	2.173	1.384	0.030
	Forest Cover	1.748	1.638	8.185	13.41	7.239	0.483
	HTTP	0.187	0.376	2.391	11.04	5.657	0.384
	Mamography	0.237	0.223	0.981	1.443	1.721	0.044
	Mulcross	2.732	2.272	8.772	13.75	7.864	0.559
	Satellite	0.393	0.355	0.976	1.199	1.435	0.030
	Shuttle	1.317	1.318	6.404	7.169	4.301	0.186
	SMTP	1.116	1.105	7.912	11.76	5.924	0.411

time for testing all test points and the AUC performance are reported in Table 3.

We observe that while being faster than BP-K-NNG, aK-LPE and iForest, and comparable to one-class SVM during test stage, our approach also achieves superior performance for all data sets. The density based aK-LPE and BP-K-NNG has somewhat good performance, but its test-time degrades with training set size. massAD is very fast at test stage, but has poor performance for several data sets.

One-class SVM Comparison The baseline one-class SVM has good test time due to the similar $O(dS_1)$ test stage complexity where S_1 denotes the number of support vectors. However, its detection performance is pretty poor, because one-class SVM training is in essence approximating one single α-percentile density level set. α depends on the parameter of one-class SVM, which essentially controls the fraction of points violating the max-margin constraints [Schölkopf et al., 2001]. Decision regions obtained by thresholding with different offsets are simply scaled versions of that particular level set. Our rankAD approach significantly outperforms one-class SVM, because it has the ability to approximate different density level sets.

aK-LPE & BP-K-NNG Comparison: Computationally RankAD significantly outperforms density-based aK-LPE and BP-K-NNG, which is not surprising given our discussion in Sec.4.3. Statistically, RankAD appears to be marginally better than aK-LPE and BP-K-NNG for many datasets and this requires more careful reasoning. To evaluate the statistical significance of the reported test results we note that the number of test samples range from 5000-500000 test samples with at least 500 anomalous points. Consequently, we can bound test-performance to within 2-5% error with 95% confidence ($< 2\%$ for large datasets and $< 5\%$ for the smaller ones (Annthyroid, Mamography, Satellite)) using standard extension of known results for test-set prediction [Langford, 2005]. After accounting for this confidence RankAD is marginally better than aK-LPE and BP-K-NNG statistically. For aK-LPE we use resampling to robustly ranked values (see Sec. 6.1) and for RankAD we use cross-validation (CV) (see Sec. 6.1) for rank prediction. Note that we cannot use CV for tuning predictors for detection because we do not have anomalous data during training. All of these arguments suggests that the regularization step in RankAD results in smoother level sets and better accounts for smoothness of true level sets (also see Fig 6.2) in some cases, unlike NN methods.

7 Conclusions

We presented a novel anomaly detection framework based on combining statistical density information with a discriminative ranking procedure. Our scheme learns a ranker over all nominal samples based on the k-NN distances within the graph constructed from these nominal points. This is achieved through a pairwise learning-to-rank step, where the inputs are preference pairs (x_i, x_j) and asymptotically models the situation that data point x_i is located in a higher density region relative to x_j. We then show the asymptotic consistency of our approach, which allows for flexible false alarm control during test stage. We also provide a finite-sample generalization bound on the empirical false alarm rate of our approach. Experiments on synthetic and real data sets demonstrate our approach has state-of-art statistical performance as well as low test time complexity.

References

M. Basseville, I.V. Nikiforov, et al. *Detection of abrupt changes: theory and application*, volume 104. Prentice Hall Englewood Cliffs, NJ, 1993.

Stephen D. Bay and Mark Schwabacher. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, pages 29–38, New York, NY, USA, 2003. ACM. ISBN 1-58113-737-0. doi: 10.1145/956750.956758. URL http://doi.acm.org/10.1145/956750.956758.

K. Bhaduri, B. L. Matthews, and C. R. Giannella. Algorithms for speeding up distance-based outlier detection. In *ACM SIGKDD*, pages 859–867, 2011.

V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41 (3):15:1–15:58, July 2009. ISSN 0360-0300. doi: 10.1145/1541880.1541882. URL http://doi.acm.org/10.1145/1541880.1541882.

C. Chang and C. Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011. ISSN 2157-6904. doi: 10.1145/1961189.1961199. URL http://doi.acm.org/10.1145/1961189.1961199.

O. Chapelle and S. S. Keerthi. Efficient algorithms for ranking with svms. In *Information Retrieval*, volume 81, pages 201–215, 2010.

F. Cucker and S. Smale. On the mathematical foundations of learning. In *Bull. Amer. Math. Soc.*, pages 1–49, 2001.

A. Cuevas and A. Rodriguez-Casal. Set estimation: An overview and some recent developments. In *Recent advances and trends in nonparametric statistics*, pages 251–264, 2003.

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

A.O. Hero. Geometric entropy minimization (gem) for anomaly detection and localization. In *Neural Information Processing Systems Conference*, volume 19, 2006.

V. Hodge and J. Austin. A survey of outlier detection methodologies. In *Artificial Intelligence Review*, volume 22, pages 85–126, 2004.

T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM. ISBN 1-58113-567-X. doi: 10.1145/775047.775067. URL http://doi.acm.org/10.1145/775047.775067.

Brian Kulis and Kristen Grauman. Kernelized locality-sensitive hashing for scalable image search. In *IEEE International Conference on Computer Vision (ICCV*, 2009.

Y. Lan, J. Guo, X. Cheng, and T. Liu. Statistical consistency of ranking methods in a rank-differentiable probability space. In *Advances in Neural Information Processing Systems*, pages 1241–1249, 2012.

John Langford. Tutorial on practical prediction theory for classification. *J. Mach. Learn. Res.*, 6:273–306, December 2005.

F. T. Liu, K. M. Ting, and Z. Zhou. Isolation forest. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, 2008.

J. Nunez-Garcia, Z. Kutalik, K.-H.Cho, and O. Wolkenhauer. Level sets and minimum volume sets of probability density functions. In *Approximate Reasoning*, volume 34, pages 25–47, 2003.

G. H. Orair, Carlos H. C. Teixeira, Wagner Meira, Jr., Ye Wang, and Srinivasan Parthasarathy. Distance-based outlier detection: Consolidation and renewed bearing. *Proc. VLDB Endow.*, 3(1-2):1469–1480, September 2010. ISSN 2150-8097. doi: 10.14778/1920841.1921021. URL http://dx.doi.org/10.14778/1920841.1921021.

C. Park, J. Z. Huang, and Y. Ding. A computable plug-in estimator of minimum volume sets for novelty detection. *Operations Research*, pages 1469–1480, 2010.

J. Qian and V. Saligrama. New statistic in p-value estimation for anomaly detection. In *Statistical Signal Processing Workshop, IEEE*, pages 393 –396, Aug. 2012. doi: 10.1109/SSP.2012.6319713.

D. M. Rocke and D. L. Woodruff. Identification of outliers in multivariate data. In *Journal of the American Statistical Association*, pages 1047–1061, 1996.

B. Schölkopf, J.C. Platt, J. Shawe-Taylor, A.J. Smola, and R.C. Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

C.D. Scott and R.D. Nowak. Learning minimum volume sets. *The Journal of Machine Learning Research*, 7:665–704, 2006.

K. Sricharan and A. O. Hero. Efficient anomaly detection using bipartite k-nn graphs. In *Neural Information Processing Systems*, 2011.

I. Steinwart. Consistency of support vector machines and other regularized kernel machines. In *IEEE Trans. Inform. Theory*, pages 67–93, 2001.

K. M. Ting, G. Zhou, F. T. Liu, and J. S. C. Tan. Mass estimation and its applications. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 989–998, New York, NY, USA, 2010. ACM.

Ye Wang, Srinivasan Parthasarathy, and Shirish Tatikonda. Locality sensitive outlier detection: A ranking driven approach. In Serge Abiteboul, Klemens Bhm, Christoph Koch, and Kian-Lee Tan, editors, *ICDE*, pages 410–421. IEEE Computer Society, 2011. ISBN 978-1-4244-8958-9. URL `http://dblp.uni-trier.de/db/conf/icde/icde2011.html#WangPT11`.

K. Yamanishi, J.-I. Takeuchi, G. Williams, and P. Milne. Online unsupervised outlier detection using finite mixtures with discounting learning algorithms. In *Proceedings of the ACM SIGKDD*, pages 320–324, 2000.

M. Zhao and V. Saligrama. Anomaly detection with score functions based on nearest neighbor graphs. In *Neural Information Processing Systems Conference*, volume 22, 2009.

Acknowledgment: This work is supported by the U.S. DHS Grant 2013-ST-061-ED0001 and US NSF under awards 1218992, 1320547 respectively. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the agencies.