Metric Anomaly Detection Via Asymmetric Risk Minimization

Aryeh Kontorovich[1,2], Danny Hendler[1,2], and Eitan Menahem[1,3]

[1] Deutsche Telekom Laboratories
[2] Department of Computer Science
Ben-Gurion University of the Negev
[3] Department of Information Systems Engineering
Ben-Gurion University of the Negev

Abstract. We propose what appears to be the first anomaly detection framework that learns from positive examples only and is sensitive to substantial differences in the presentation and penalization of normal vs. anomalous points. Our framework introduces a novel type of asymmetry between how *false alarms* (misclassifications of a normal instance as an anomaly) and *missed anomalies* (misclassifications of an anomaly as normal) are penalized: whereas *each* false alarm incurs a unit cost, our model assumes that a high *global cost* is incurred if *one or more* anomalies are missed.

We define a few natural notions of risk along with efficient minimization algorithms. Our framework is applicable to any metric space with a finite doubling dimension. We make minimalistic assumptions that naturally generalize notions such as *margin* in Euclidean spaces. We provide a theoretical analysis of the risk and show that under mild conditions, our classifier is asymptotically consistent. The learning algorithms we propose are computationally and statistically efficient and admit a further tradeoff between running time and precision. Some experimental results on real-world data are provided.

1 Introduction

Cost-sensitive learning [10, 38] is an active research area in machine learning. In this framework, different costs are associated with different types of misclassification errors. In general, these costs differ for different types of misclassification. Classifiers are then optimized to minimize the expected cost incurred due to their errors. This is in contrast with cost-insensitive learning, where classification algorithms are optimized to minimize their error rate — the expected fraction of misclassified instances, thus implicitly making the (often unrealistic) assumption that all misclassification errors have the same cost.

Cost-sensitive classification is often useful for binary classification, when the datasets under consideration are highly imbalanced and consist mostly of normal instances and with only a small fraction of anomalous ones [19, 23]. Since the terms “false positive” and “false negative” are confusing in the context of anomaly detection, we call a normal instance misclassified as an anomaly a *false alarm* and an anomaly misclassified as normal a *missed anomaly.* Typically, the cost of a missed anomaly is much higher than that of a false alarm.

We consider a cost-sensitive classification framework, in which learning is based on normal instances only and anomalies are never observed during training. Our framework introduces a novel type of asymmetry between how false alarms and missed anomalies are penalized: whereas *each* false alarm incurs a unit cost, our model assumes that a high *global cost* is incurred if *one or more* anomalies are missed.

As a motivating example for our framework, consider a warehouse equipped with a fire alarm system. Each false fire alarm automatically triggers a call to the fire department and incurs a unit cost. On the other hand, *any nonzero number* of missed anomalies (corresponding to one or more fires breaking out in the warehouse) cause a *a single* "catastrophic" cost corresponding to the warehouse burning down one or more times (only the first time "matters").

We define a natural notion of risk and show how to minimize it under various assumptions. Our framework is applicable to any metric space with a finite doubling dimension. We make minimalistic assumptions that naturally generalize notions such as *margin* in Euclidean spaces. We provide a theoretical analysis of the risk and show that under mild conditions, our classifier is asymptotically consistent. The learning algorithms we propose are efficient and admit a further tradeoff between running time and precision — for example, using the techniques of [15] to efficiently estimate the doubling dimension and the spanner-based approach described in [14] to quickly compute approximate nearest neighbors. Some experimental results on real-world data are provided.

Related work The majority of published cost-sensitive classification algorithms assume the availability of supervised training data, were all instances are labeled (e.g. [9, 12, 24, 32, 35, 38, 39]).

Some work considers semi-supervised cost-sensitive classification. Qin et al. [29] present cost-sensitive classifiers for training data that consists of a relatively small number of labeled instances and a large number of unlabeled instances. Their implementations are based on the expectation maximization (EM) algorithm [8] as a base semi-supervised classifier. Bennett et al. [4] present ASSEMBLE, an adaptive semi-supervised ensemble scheme that can be used to to make any cost-sensitive classifier semi-supervised. Li et al. [22] recently proposed CS4VM - a semi-supervised cost-sensitive support vector machine classifier. Other cost-sensitive semi-supervised work involves attempts to refine the model using human feedback (see, e.g., [16, 25, 27]).

Our framework falls within the realm of *one-class classification* [34] since learning is done based on normal instances only. Crammer and Chechik [7] consider the one-class classification problem of identifying a small and coherent subset of data points by finding a ball with a small radius that covers as many data points as possible. Whereas previous approaches to this problem used a cost function that is constant within the ball and grows linearly outside of it [3, 30, 33], the approach taken by [34] employs a cost function that grows linearly within the ball but is kept constant outside of it. Other papers employing the one-class SVM technique include [18, 26]. Also relevant is the approach of [31] for estimating the support of a distribution — although in this paper, the existence of a kernel is assumed, which is a much stronger assumption than that of a metric.

Definitions and notation We use standard notation and definitions throughout. A *metric* d on a set $\mathcal{X}$ is a positive symmetric function satisfying the triangle inequality $d(x, y) \leq d(x, z)+d(z, y)$; together the two comprise the metric space $(\mathcal{X}, d)$. The diameter of a set $A \subseteq \mathcal{X}$ is defined by $\operatorname{diam}(A) = \sup_{x,y \in A} d(x, y)$. In this paper, we always denote $\operatorname{diam}(\mathcal{X})$ by Δ. For any two subsets $A, B \subset \mathcal{X}$, their "nearest point" distance $d(A, B)$ is defined by $d(A, B) = \inf_{x \in A, y \in B} d(A, B)$. The *Lipschitz constant* of a function $f : \mathcal{X} \to \mathbb{R}$ is defined to be the smallest $L > 0$ that makes $|f(x) - f(y)| \leq Ld(x, y)$ hold for all $x, y \in \mathcal{X}$. For a metric space (X, d), let λ be the smallest number such that every ball in $\mathcal{X}$ can be covered by λ balls of half the radius. The *doubling dimension* of $\mathcal{X}$ is $\mathrm{ddim}(\mathcal{X}) = \log_2 \lambda$. A metric is *doubling* when its doubling dimension is bounded. Note that while a low Euclidean dimension implies a low doubling dimension (Euclidean metrics of dimension k have doubling dimension $O(k)$ [17]), low doubling dimension is strictly more general than low Euclidean dimension.

Throughout the paper we write $\mathbb{1}_{\{\cdot\}}$ to represent the 0-1 truth value of the subscripted predicate.

Paper outline The rest of this paper is organized as follows. In Section 2 we present our theoretical results: first, for the idealized case where the data is well-separated by a known distance, and then for various relaxations of this demand. Some experimental results are provided in Section 3. We close with a discussion and ideas for future work in Section 4.

2 Theoretical results

2.1 Preliminaries

We define the following model of learning from positive examples only. The metric space $(\mathcal{X}, d)$ is partitioned into two disjoint sets, $\mathcal{X} = \mathcal{X}_+ \cup \mathcal{X}_-$, where $\mathcal{X}_+$ are the "normal" points and $\mathcal{X}_-$ are the "anomalous" ones. The normal set $\mathcal{X}_+$ is endowed with some (unknown) probability distribution P and the training phase consists of the learner being shown n iid draws of $X_i \in \mathcal{X}_+$ according to P. In the testing phase, the learner is asked to classify a new $X \in \mathcal{X}$ as *normal* or *anomalous.* By assumption, normal test points are drawn from P, but no assumption is made on the distribution of anomalous test points.

Further structural assumptions are needed to make the problem statement non-trivial. By analogy with common separability assumptions in supervised learning by hyperplanes, we make the following assumption:

$$d(\mathcal{X}_+, \mathcal{X}_-) \equiv \inf_{x \in \mathcal{X}_+, y \in \mathcal{X}_-} d(x, y) > \gamma \tag{1}$$

for some *separation distance* $\gamma > 0$.

We distinguish the two types of classification error: when a normal point is wrongly labeled as an anomaly, we call this a **false alarm**, and when an anomaly is wrongly classified as normal, we call this an **missed anomaly**.

2.2 Known separation distance

When the separation distance γ is known, we propose a simple classification rule $f : \mathcal{X} \to \{-1, 1\}$ as follows: given a sample $S \subset \mathcal{X}_+$, classify a new point x as normal (corresponding to $f(x) = 1$) if $d(x, S) \le \gamma$ and as anomalous ($f(x) = -1$) if $d(x, S) > \gamma$. Our assumption (1) implies that f will never make a missed anomaly error, and we can use the techniques of [14] to bound the false alarm rate of this classifier. Define the *false alarm rate* of f by

$$\mathrm{FA}(f) = \int_{\mathcal{X}_+} \mathbb{1}_{\{f(x) < 0\}} dP(x). \tag{2}$$

Theorem 1. *Given a training set $S = \{X_1, \ldots, X_n\}$ drawn from $\mathcal{X}_+$ iid under the distribution P, define the* proximity classifier $f_{n,\gamma}$ *as above:*

$$f_{n,\gamma}(x) = \mathbb{1}_{\{d(x,S) \le \gamma\}} - \mathbb{1}_{\{d(x,S) > \gamma\}}. \tag{3}$$

Then, with probability at least $1 - \delta$, this classifier achieves a false alarm rate that satisfies

$$\mathrm{FA}(f_{n,\gamma}) \le \frac{2\left(D \log_2(34en/D) \log_2(578n) + \log_2(4/\delta)\right)}{n}, \tag{4}$$

where

$$D = \lceil 8\Delta/\gamma \rceil^{\mathrm{ddim}(\mathcal{X})+1} \tag{5}$$

and $\mathrm{ddim}(\mathcal{X})$ *is the doubling dimension of $\mathcal{X}$.*

Proof. Consider the function $h : \mathcal{X} \to [-1, 1]$ satisfying

(i) $h(x) \ge 1$ for all $x \in S$
(ii) $h(x) < 0$ for all x with $d(x, S) > \gamma$
(iii) h has the smallest Lipschitz constant among all the functions satisfying (i) and (ii).

It is shown in [14, 36] that h (a) has Lipschitz constant $1/\gamma$ and (b) the function $x \mapsto \operatorname{sgn} h(x)$ is realized by $f_{n,\gamma}$ defined in (3). Corollary 3 in [14] shows that the collection of real-valued $1/\gamma$-Lipschitz functions defined on a metric space $\mathcal{X}$ with doubling dimension $\mathrm{ddim}(\mathcal{X})$ and diameter Δ has a fat-shattering dimension at scale $1/16$ of at most $(8\Delta/\gamma)^{\mathrm{ddim}(\mathcal{X})+1}$. The claim follows from known generalization bounds for function classes with a finite fat-shattering dimension (e.g., Theorem 1.5 in [2]). □

Remark 1. Note that the approach via Rademacher averages in general yields tighter bounds than those obtained from fat-shattering bounds; see [36].

In the sequel, we will find it useful to restate the estimate in Theorem 1 in the following equivalent form.

Corollary 1. *Let $f_{n,\gamma}$ be the proximity classifier defined in Theorem 1, based on a sample of size n. Then, for all $0 \le t \le 1$, we have*

$$P(\mathrm{FA}(f_{n,\gamma}) > t) \le \exp((A_{n,\gamma} - t)/B_n)$$

where

$$A_{n,\gamma} = \left(2D_\gamma \log_2(34en/D_\gamma) \log_2(578n) + 2\log_2 4\right)/n$$

and

$$B_n = 2/(n \ln 2)$$

and $D = D_\gamma$ is defined in (5).

Proof. An equivalent way of stating (4) is that

$$\mathrm{FA}(f_{n,\gamma}) > A_{n,\gamma} - B_n \ln \delta$$

holds with probability less than δ. Putting $t = A_{n,\gamma} - B_n \ln \delta$ and solving for δ yields the claim. □

2.3 Definition of risk

We define risk in a nonstandard way, but one that is suitable for our particular problem setting. Because of our sampling assumptions — namely, that the distribution is only defined over $\mathcal{X}_+$ — there is a fundamental asymmetry between the false alarm and missed anomaly errors. A false alarm is a well-defined random event with a probability that we are able to control increasingly well with growing sample size. Thus, any classifier f has an associated false alarm rate $\mathrm{FA}(f)$ defined in (2). Since $f_{n,\gamma}$ itself is random (being determined by the random sample), $\mathrm{FA}(f_{n,\gamma})$ is a random variable and it makes sense to speak of $\mathbf{E}[\mathrm{FA}(f_{n,\gamma})]$ — the expected false alarm rate.

A missed anomaly is not a well-defined random event, since we have not defined any distribution over $\mathcal{X}_-$. Instead, we can speak of conditions ensuring that no missed anomaly will ever occur; the assumption of a separation distance is one such condition. If there is uncertainty regarding the separation distance γ, we might be able to describe the latter via a distribution $G(\cdot)$ on $(0, \infty)$, which is either assumed as a prior or somehow estimated empirically.

Having equipped γ with a distribution, our expression for the risk at a given value of γ_0 becomes

$$\mathrm{Risk}(\gamma_0) = \int_{\gamma_0}^{\infty} \mathbf{E}[\mathrm{FA}(f_{n,\gamma})] dG(\gamma) + C \int_0^{\gamma_0} dG(\gamma)$$

which reflects our modeling assumption that we pay a unit cost for each false alarm and a large “catastrophic” cost C for *any* nonzero number of missed anomalies.

2.4 Classification rule

As before, we assume a unit cost incurred for each false alarm and a cost C for any positive missed anomalies. Let $A_{n,\gamma}$ and B_n be as defined in Corollary 1 and assume in what follows that n is sufficiently large so that $A_{n,\gamma} < 1$ (the bounds are vacuous for smaller values of n).

When γ is known, the only contribution to the risk is from false alarms, and it decays to zero at a rate that we are able to control.

Theorem 2. *Suppose the separation distance γ is known. Let $f_{n,\gamma}$ be the proximity classifier defined in Theorem 1, based on a sample of size n. Then*

$$\mathrm{Risk}(\gamma) \le (A_{n,\gamma} + B_n)$$

where $A_{n,\gamma}$ and B_n are as defined in Corollary 1 and n is assumed large enough so that $A_{n,\gamma} < 1$.

Proof. We compute

$$\begin{aligned}
\mathrm{Risk}(\gamma) &= \mathbf{E}[\mathrm{FA}(f_{n,\gamma})] \\
&= \int_0^{\infty} P(\mathrm{FA}(f_{n,\gamma}) > t) dt \\
&\le \int_0^1 \min\{1, \exp((A_{n,\gamma} - t)/B_n)\}\, dt \\
&= \left[\int_0^{A_{n,\gamma}} dt + \int_{A_{n,\gamma}}^1 \exp((A_{n,\gamma} - t)/B_n) dt \right] \\
&= [A_{n,\gamma} + B_n - B_n e^{(A_{n,\gamma}-1)/B_n}] \\
&\le (A_{n,\gamma} + B_n),
\end{aligned}$$

where the first inequality is an application of Corollary 1. □

When the exact value of the separation distance γ is unknown, we consider the scenario where our uncertainty regarding γ is captured by some known distribution G (which might be assumed a priori or estimated empirically).

In this case, the risk associated with a given value of γ_0 is:

$$\begin{aligned}\mathrm{Risk}(\gamma_0) &= \int_{\gamma_0}^{\infty} \mathbf{E}[\mathrm{FA}(f_{n,\gamma})] dG(\gamma)\gamma + C\int_0^{\gamma_0} dG(\gamma) \\ &\leq \int_{\gamma_0}^{\infty} (A_{n,\gamma} + B_n) dG(\gamma) + C\int_0^{\gamma_0} dG(\gamma) \\ &=: R_n(\gamma_0),\end{aligned}$$

where the inequality follows immediately from Theorem 2.

Our analysis implies the following classification rule: compute the minimizer γ^* of $R_n(\cdot)$ and use the classifier f_{n,γ^*}. As a sanity check, notice that $A_{n,\gamma}$ grows inversely with γ (at a rate proportional to $1/\gamma^{\mathrm{ddim}(\mathcal{X})+1}$), so γ^* will not be arbitrarily small. Note also that $R_n(\gamma_0) \to 0$ as $n \to \infty$ for any fixed γ_0.

2.5 No explicit prior on γ

Instead of assuming a distribution on γ, we can make a weaker assumption. In any discrete metric space $(\mathcal{S}, d)$, define the quantity we call *isolation distance*

$$\rho = \sup_{x \in \mathcal{S}} d(x, \mathcal{S} \setminus \{x\});$$

this is the maximal distance from any point in S to its nearest neighbor. Our additional assumption will be that $\rho < \gamma$ (in words: the isolation distance is less than the separation distance). This means that we can take ρ — a quantity we can estimate empirically — as a proxy for γ.

We estimate $\rho = \rho(\mathcal{X}_+, d)$ as follows. Given the finite sample $X_1, \ldots, X_n$ drawn iid from $\mathcal{X}_+$, define

$$\hat{\rho}_n = \max_{i \in [n]} \min_{j \neq i} d(X_i, X_j). \tag{6}$$

It is obvious that $\hat{\rho}_n \leq \rho$ and for countable $\mathcal{X}$, it is easy to see that $\hat{\rho}_n \to \rho$ almost surely. The convergence rate, however, may be arbitrarily slow, as it depends on the (possibly adversarial) sampling distribution P.

To obtain a distribution-free bound, we will need some additional notions. For $x \in \mathcal{X}$, define $B_\epsilon(x)$ to be the ϵ-ball about x:

$$B_\epsilon(x) = \{y \in \mathcal{X} : d(x, y) \leq \epsilon\}.$$

For $S \subset \mathcal{X}$, define its ϵ-envelope, S_ϵ, to be

$$S_\epsilon = \bigcup_{x \in S} B_\epsilon(x).$$

For $\epsilon > 0$, define the ϵ-covering number, $N(\epsilon)$, of $\mathcal{X}$ as the minimal cardinality of a set $E \subset \mathcal{X}$ such that $\mathcal{X} = E_\epsilon$. Following [5], we define the *ϵ-unseen mass* of the sample $S = \{X_1, \ldots, X_n\}$ as the random variable

$$U_n(\epsilon) = P(\mathcal{X}_+ \setminus S_\epsilon). \tag{7}$$

It is shown in [5] that the expected ϵ-unseen mass may be estimated in terms of the ϵ-covering numbers, uniformly over all distributions.

Theorem 3 ([5]). *Let $\mathcal{X}$ be a metric space equipped with some probability distribution and let $U_n(\epsilon)$ be the ϵ-unseen mass random variable defined in (7). Then for all sampling distributions we have*

$$\mathbf{E}[U_n(\epsilon)] \le \frac{N(\epsilon)}{en},$$

where $N(\epsilon)$ is the ϵ-covering number of $\mathcal{X}$.

Corollary 2. *Let $U_n(\epsilon)$ be the ϵ-unseen mass random variable defined in (7). Then*

$$\mathbf{E}[U_n(\epsilon)] \le \frac{1}{en}\left(\frac{\Delta}{\epsilon}\right)^{\mathrm{ddim}(\mathcal{X})+2}.$$

Proof. For doubling spaces, it is an immediate consequence of [21] and [1, Lemma 2.6] that

$$N(\epsilon) \le \left\lceil \frac{\Delta}{\epsilon} \right\rceil^{\mathrm{ddim}(\mathcal{X})+1} \le \left(\frac{\Delta}{\epsilon}\right)^{\mathrm{ddim}(\mathcal{X})+2}.$$

Substituting the latter estimate into Theorem 3 yields the claim. □

Our final observation is that for any sample $X_1, \ldots, X_n$ achieving an ϵ-net, the corresponding $\hat{\rho}_n$ satisfies

$$\hat{\rho}_n \le \rho \le \hat{\rho}_n + 2\epsilon.$$

We are now in a position to write down an expression for the risk. The false-alarm component is straightforward: taking $\hat{\gamma} = \hat{\rho}_n + 2\epsilon$, the only way a new point X could be misclassified as a false alarm is if it falls outside of the ϵ-envelope of the observed sample. Thus, this component of the risk may be bounded by

$$\frac{1}{en}\left(\frac{\Delta}{\epsilon}\right)^{\mathrm{ddim}(\mathcal{X})+2}.$$

On the other hand a missed anomaly can only occur if $\hat{\gamma} > \gamma$. Unfortunately, even though $\hat{\gamma} = \hat{\rho}_n + 2\epsilon$ is a well-defined random variable, we cannot give a non-trivial bound on $P(\hat{\gamma} > \gamma)$ since we know nothing about how close ρ is to γ. Therefore, we resort to a "flat prior" heuristic (corresponding roughly to the assumption $\Pr[\rho + t\Delta > \gamma] \approx t$), resulting in the missed-anomaly risk term of the form

$$\frac{2C\epsilon}{\Delta}. \tag{8}$$

Combining the two terms, we have

$$R_n(\epsilon) = \frac{1}{en}\left(\frac{\Delta}{\epsilon}\right)^{\mathrm{ddim}(\mathcal{X})+2} + \frac{2C\epsilon}{\Delta}$$

which is minimized at

$$\epsilon_n = \frac{\Delta^{\mathrm{ddim}(\mathcal{X})+3}}{2Cen}.$$

Note that as $n \to \infty$, we have $\epsilon_n \to 0$ and $R_n(\epsilon_n) \to 0$, implying an asymptotic consistency of the classifier $f_{n,\hat{\rho}_n+2\epsilon_n}$ for this type of risk. Observe also that analogous asymptotically consistent estimators are straightforward to derive for risk bounded by

$$R_n(\epsilon) = \frac{1}{en}\left(\frac{\Delta}{\epsilon}\right)^{\mathrm{ddim}(\mathcal{X})+2} + \frac{2C\epsilon^a}{\Delta}$$

for any $a > 0$.

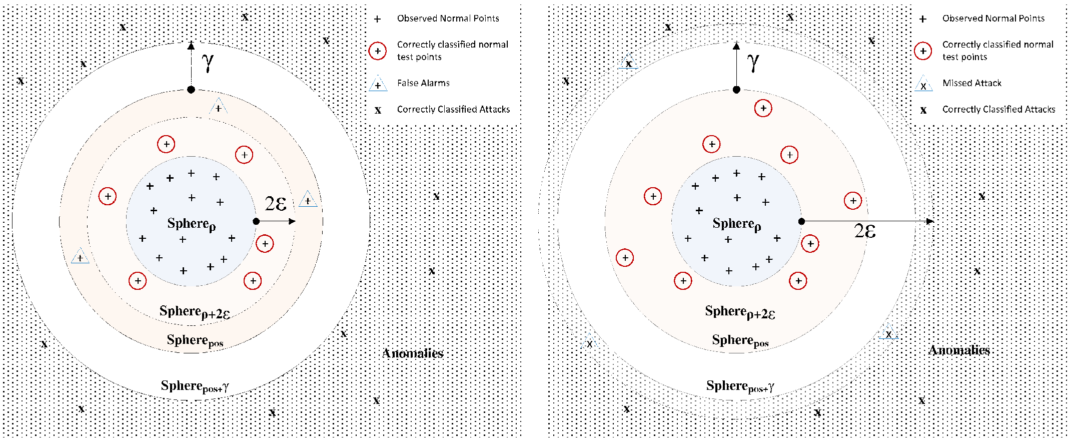


Fig. 1. A schematic presentation of the various quantities defined in Section 2.5. In the left diagram, ϵ is too small, resulting in false alarms. On the right, a too-large value of ϵ leads to missed attacks.

3 Experiments

3.1 Methodology

We experimented with several datasets, both synthetic and real-world. The Euclidean metric $d(x, x') = \|x - x'\| = \sqrt{\sum(x_i - x'_i)^2}$ was used in each case. For each dataset, a false alarm incurs a unit cost and any number of missed anomalies incurs a catastrophic cost C. The value of C is strongly tied to the particular task at hand. In order to obtain a rough estimate in the case of an attack on a computer network, we consulted various figures on the damage caused by such events [13, 37] and came up with the rough estimate of $300,000$ for C; this was the value we used in all the experiments. The diameter Δ is estimated as the largest distance between any two sample points and the doubling dimension $\mathrm{ddim}(\mathcal{X})$ is efficiently approximated from the sample via the techniques of [15]. The figures presented are the averages over 10 trials, where the data was randomly split into training and test sets in each trial.

Before we list the classifiers that were tested, a comment is in order. For a fair comparison to our proposed method, we need a classifier that is both (i) cost-sensitive and (ii) able to learn from positive examples only. Since we were not able to locate such a classifier in the literature, we resorted to adapting existing techniques to this task. The following classifiers were trained and tested on each dataset:

- Asymmetric Anomaly Detector (AAD) is the classifier $f_{n,\hat{\rho}_n+2\epsilon_n}$ proposed in Section 2.5 of this paper.
- Peer Group Analysis (PGA) is an unsupervised anomaly detection method proposed by Eskin et al. [11] that identifies the low density regions using nearest neighbors. An anomaly score is computed at a point x as a function of the distances from x to its k nearest neighbors. Although PGA is actually a ranking technique applied to a clustering problem, we implemented it as a one-class classifier with $k = 1$. Given the training sample S, a test point x is classified as follows. For each $x_i \in S$, we pre-compute the distance to x_i's nearest neighbor in S, given by $d_i = d(x_i, S \setminus \{x_i\})$. To classify x, the distance to the nearest neighbor of x in S, $d_x = d(x, S)$ is computed. The test point x is classified as an anomaly if $d_x = d(x, S)$ appears in a percentile α or higher among the $\{d_i\}$; otherwise it is classified as normal. We set the parameter $\alpha = 0.01$ (obviously, it should depend on the value of C but the dependence is not at all clear).
- Global Density Estimation (GDE), proposed by [20] is also an unsupervised density-estimation technique using nearest neighbors. Given a training sample S and a real value r, one computes the anomaly score of a test point x by comparing the number of training points falling within the r-ball $B_r(x)$ about x to the average of $|B_r(x_i) \cap S|$ over all $x_i \in S$. We set r to be twice the sample average of $d(x_i, S \setminus \{x_i\})$ to ensure that the average number of neighbors is at least one. In order to convert GDE into a classifier, we needed a heuristic for thresholding anomaly scores. We chose the following one, as it seemed to achieve a low classification error on the data: x is classified as normal if $\exp(-((N_r(x) - \bar{N}_r)/\sigma_r) > 1/2$, where N_r is the number of r-neighbors of x in S, $\bar{N}_r$ is the average number of r-neighbors over the training points, and σ_r is the sample standard deviation of the number of r-neighbors.

Each classifier is evaluated based on the cost that it incurred on unseen data: c units were charged for each false alarm and an additional cost of C for one or more missed anomalies. As an additional datum, we also record the cost-insensitive classification error.

3.2 Data sets

We tested the classifiers on the following three data sets.

2D-Single-Cluster. This is a two-dimensional synthetic data set. As shown in Figure 2, the normal data points are concentrated along a thin, elongated cluster in the middle of a square, with the anomalies spread out uniformly. A total of 363 points were generated, of which 300 were normal with 63 anomalies. For the normal points, the x-coordinate was generated uniformly at random and the y-coordinate was a function of x perturbed by noise. A positive separation distance was enforced during the generation process.

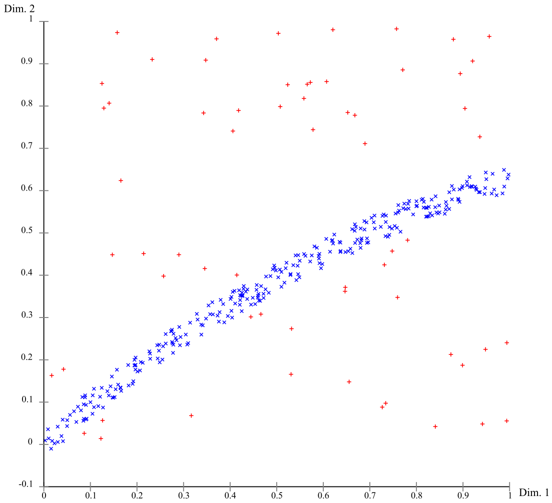


Fig. 2. The 2D-Single-Cluster dataset.

9D-Sphere. This is a 9-dimensional synthetic data set containing 550 instances. The coordinates are drawn independently from mean-zero, variance-35 Gaussians. Points with Euclidean norm under 90 were labeled as "normal" and those whose norm exceeded 141 were labeled "anomalies". Points whose norm fell between these values were discarded, so as to maintain a strong separation distance.

BGU ARP. The abbreviation ARP stands for "Address Resolution Protocol", see [28]. This is a dataset of actual ARP attacks, recorded on the Ben-Gurion University network. The dataset contains 9039 instances and 23 attributes extracted from layer-2 (link-layer) frames. Each instance in the dataset represents a single ARP packet that was sent through the network during the recording time. There were 173 active computers on the network, of which 27 were attacked. The attacker temporarily steals the IPv4 addresses of its victims and as a result, the victim's entire traffic is redirected to the attacker, without the victim's knowledge or consent. Our training data had an anomaly (attack) rate of 3.3%. The training instances were presented in xml format and their numerical fields induced a Euclidean vector representation.

3.3 Results

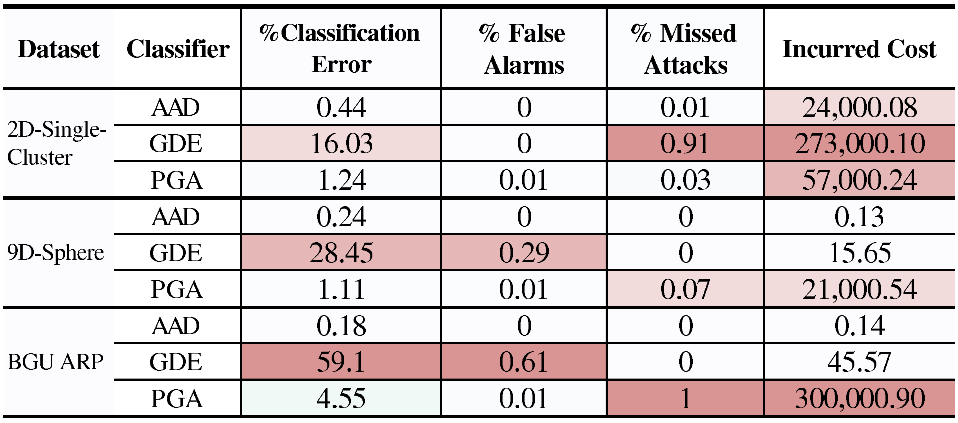
Our basic quantities of interest are the number of false alarms (FA), the number of missed anomalies (MA), and the number of correctly predicted test points (CP). From these, we derive the classification error

$$\mathrm{err} = \frac{\mathrm{FA} + \mathrm{MA}}{\mathrm{FA} + \mathrm{MA} + \mathrm{CP}}$$

and the incurred cost

$$\mathrm{Cost} = \mathrm{FA} + C \cdot \mathbb{1}_{\{\mathrm{MA}>0\}}.$$

Although in this paper we are mainly interested in the incurred cost, we also keep track of the classification error for comparison. The results are summarized in Figure 3. Notice that our classifier significantly outperforms the others in the incurred cost criterion. Also interesting to note is that a lower classification error does not necessarily imply a lower incurred cost, since even a single missed attack can significantly increase the latter.



Dataset	Classifier	%Classification Error	% False Alarms	% Missed Attacks	Incurred Cost
2D-Single-Cluster	AAD	0.44	0	0.01	24,000.08
	GDE	16.03	0	0.91	273,000.10
	PGA	1.24	0.01	0.03	57,000.24
9D-Sphere	AAD	0.24	0	0	0.13
	GDE	28.45	0.29	0	15.65
	PGA	1.11	0.01	0.07	21,000.54
BGU ARP	AAD	0.18	0	0	0.14
	GDE	59.1	0.61	0	45.57
	PGA	4.55	0.01	1	300,000.90

Fig. 3. The performance of the classifiers on the datasets, averaged over 10 trials.

4 Discussion and future work

We have presented a novel (and apparently first of its kind) method for learning to detect anomalies in a cost-sensitive framework from positive examples only, along with efficient learning algorithms. We have given some preliminary theoretical results supporting this technique and tested it on data (including real-world), with encouraging results.

Some future directions naturally suggest themselves. One particularly unrealistic assumption is the "isotropic" nature of our classifier, which implicitly assumes that the density has no preferred direction in space. Directionally sensitive metric classifiers already exist [6] and it would be desirable to extend our analysis to these methods. Additionally, one would like to place the heuristic missed-anomaly risk term we proposed in (8) on a more principled theoretical footing. Finally, we look forward to testing our approach on more diverse datasets.

Acknowledgments

We thank Lee-Ad Gottlieb for helpful discussions.

References

1. Noga Alon, Shai Ben-David, Nicolò Cesa-Bianchi, and David Haussler. Scale-sensitive dimensions, uniform convergence, and learnability. *Journal of the ACM*, 44(4):615–631, 1997.

2. Peter Bartlett and John Shawe-Taylor. Generalization performance of support vector machines and other pattern classifiers. pages 43–54, 1999.
3. Asa Ben-Hur. Support vector clustering. *Scholarpedia*, 3(6):5187, 2008.
4. Kristin P. Bennett, Ayhan Demiriz, and Richard Maclin. Exploiting unlabeled data in ensemble methods. In *KDD*, pages 289–296, 2002.
5. Daniel Berend and Aryeh Kontorovich. The missing mass problem, in preparation. 2011.
6. Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: identifying density-based local outliers. *SIGMOD Rec.*, 29:93–104, May 2000.
7. Koby Crammer and Gal Chechik. A needle in a haystack: local one-class optimization. In *ICML*, 2004.
8. A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B*, 39(1):1–38, 1977. With discussion.
9. Pedro Domingos. Metacost: A general method for making classifiers cost-sensitive. In *KDD*, pages 155–164, 1999.
10. Charles Elkan. The foundations of cost-sensitive learning. In *IJCAI*, pages 973–978, 2001.
11. Eleazar Eskin, Andrew Arnold, Michael Prerau, Leonid Portnoy, and Sal Stolfo. A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. In *Applications of Data Mining in Computer Security*. Kluwer, 2002.
12. Wei Fan, Salvatore J. Stolfo, Junxin Zhang, and Philip K. Chan. Adacost: Misclassification cost-sensitive boosting. In *ICML*, pages 97–105, 1999.
13. CEI figures: Computer Economics Inc. Security issues: Virus costs are rising again., 2003.
14. Lee-Ad Gottlieb, Leonid (Aryeh) Kontorovich, and Robert Krauthgamer. Efficient classification for metric data. In *COLT*, 2010.
15. Lee-Ad Gottlieb and Robert Krauthgamer. Proximity algorithms for nearly-doubling spaces. In *APPROX-RANDOM*, pages 192–204, 2010.
16. Russell Greiner, Adam J. Grove, and Dan Roth. Learning cost-sensitive active classifiers. *Artif. Intell.*, 139(2):137–174, 2002.
17. Anupam Gupta, Robert Krauthgamer, and James R. Lee. Bounded geometries, fractals, and low-distortion embeddings. In *FOCS*, pages 534–543, 2003.
18. Katherine A. Heller, Krysta M. Svore, Angelos D. Keromytis, and Salvatore J. Stolfo. One class support vector machines for detecting anomalous windows registry accesses. In *ICDM Workshop on Data Mining for Computer Security (DMSEC)*, 2003.
19. Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intell. Data Anal.*, 6(5):429–449, 2002.
20. Edwin M. Knorr and Raymond T. Ng. A unified notion of outliers: Properties and computation. In *KDD*, pages 219–222, 1997.
21. R. Krauthgamer and J. R. Lee. Navigating nets: Simple algorithms for proximity search. In *15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 791–801, January 2004.
22. Yu-Feng Li, James T. Kwok, and Zhi-Hua Zhou. Cost-sensitive semi-supervised support vector machine. In *AAAI*, 2010.
23. Charles X. Ling and Victor S. Sheng. Cost-sensitive learning. In *Encyclopedia of Machine Learning*, pages 231–235. 2010.
24. Charles X. Ling, Qiang Yang, Jianning Wang, and Shichao Zhang. Decision trees with minimal costs. In *ICML*, 2004.
25. Alexander Liu, Goo Jun, and Joydeep Ghosh. A self-training approach to cost sensitive uncertainty sampling. In *ECML/PKDD (1)*, page 10, 2009.
26. Jun Luo, Li Ding, Zhisong Pan, Guiqiang Ni, and Guyu Hu. Research on cost-sensitive learning in one-class anomaly detection algorithms. In Bin Xiao, Laurence Yang, Jianhua Ma, Christian Muller-Schloer, and Yu Hua, editors, *Autonomic and Trusted Computing*, volume 4610 of *Lecture Notes in Computer Science*, pages 259–268. Springer Berlin / Heidelberg, 2007.
27. Dragos D. Margineantu. Active cost-sensitive learning. In *IJCAI*, pages 1622–1613, 2005.
28. David C. Plummer. Rfc 826: An ethernet address resolution protocol – or – converting network protocol addresses to 48.bit ethernet address for transmission on ethernet hardware, 1982. Internet Engineering Task Force, Network Working Group.

29. Zhenxing Qin, Shichao Zhang, Li Liu, and Tao Wang. Cost-sensitive semi-supervised classification using CS-EM. In *Computer and Information Technology, 2008. CIT 2008. 8th IEEE International Conference on*, pages 131 –136, july 2008.
30. Bernhard Schölkopf, Chris Burges, and Vladimir Vapnik. Extracting support data for a given task. In *KDD*, pages 252–257, 1995.
31. Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.
32. Yanmin Sun, Mohamed S. Kamel, Andrew K. C. Wong, and Yang Wang 0007. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*, 40(12):3358–3378, 2007.
33. David M. J. Tax and Robert P. W. Duin. Data domain description using support vectors. In *ESANN*, pages 251–256, 1999.
34. David Martinus Johannes TAX. *One-class classification*. PhD thesis, Deltf University of Technology, 2001.
35. Peter D. Turney. Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm. *J. Artif. Intell. Res. (JAIR)*, 2:369–409, 1995.
36. Ulrike von Luxburg and Olivier Bousquet. Distance-based classification with lipschitz functions. *Journal of Machine Learning Research*, 5:669–695, 2004.
37. Richard Waters. When will they ever stop bugging us?, 2003. Financial Times, special report.
38. Bianca Zadrozny and Charles Elkan. Learning and making decisions when costs and probabilities are both unknown. In *KDD*, pages 204–213, 2001.
39. Zhi-Hua Zhou and Xu-Ying Liu. On multi-class cost-sensitive learning. *Computational Intelligence*, 26(3):232–257, 2010.