# Anomaly Detection in Labeled Data

Girish Keshav Palshikar
Tata Research Development and Design Centre (TRDDC)
54B Hadapsar Industrial Estate
Pune 411013, India
gk.palshikar@tcs.com

Rohit Kelkar
Tata Research Development and Design Centre (TRDDC)
54B Hadapsar Industrial Estate
Pune 411013, India
rohit.kelkar@tcs.com

## Abstract

Noisy points in training data maybe due to incorrect class labels or erroneous recording of attribute values. These points greatly influence the orientation of the classification boundary. In this paper, we formalize two notions of noisy points: intrusive outliers and hard-to-classify points. We adapt two well-known distance-based notions of outliers in unlabeled data to formalize intrusive outliers and adapt the corresponding algorithms to detect them in labeled data. We propose a boosting-based algorithm to identify hard-to-classify points in labeled data. We empirically compare these two notions of noisy points and their influence on classification accuracy. Finally we experimentally prove that removal of noisy points improves the robustness of the classification boundary against future noisy data.

## 1. Introduction

In this paper we focus on the following two types of noisy points in the training dataset:

- Intrusive outliers, which are "too far out" of their respective class clusters and are "intruding" into (or overlapping with) the cluster of some other class.
- Hard-to-classify points, which are characterized by the fact that most classifiers would make an error in assigning the class label to such points. Such points have also been called class outliers by [1](He et al., 2004).

These two sets of points may not be disjoint; e.g., some points may be both intrusive outliers and hard-to-classify. As shown in Fig. 1 we ignore the class labels and apply the well-known RRS algorithm by [7] (Ramaswamy et al., 2000) for outlier detection, we call this approach as unlabeled outlier detection. Alternatively, we take only those points from a single class as shown in Fig 2 and apply any standard outlier detection algorithm to these points; we call this approach class-wise outlier detection. Both unlabeled and class-wise outlier detection approaches ignore the additional information provided by the class label and hence are not likely to perform well in identifying the points which affect the decision boundary. Our key idea is to make use of the class label to identify such points.

*International Conference on Management of Data*
*COMAD 2006, Delhi, India, December 14-16, 2006*


## 2. Related Work

[5] (Papadimitriou and Faloutsos, 2003) uses a cross outlier criterion based on local neighborhood for detection of outliers in labeled data. [1](He et al., 2004) defines several notions of outliers in labeled data, based on local outlier factors. [9](Tax and Duin, 1998) proposes the notion of instability of a classifier when classifying new objects to identify outliers in labeled data. The basic notion of distance-based outliers was introduced by Knorr [2](Knorr et al., 2000). A related notion of outliers based on distance to nearest neighbours was proposed in [7] (Ramaswamy et al., 2000). These approaches deal with identifying outliers in the data without considering the labels. See [3](Markou and Singh, 2003) for a review of outlier detection techniques. In this paper, we adapt both these approaches for detection of noisy points in labeled data.

## 3. Intrusive outliers in labeled datasets

### 3.1 Outliers in Labeled Datasets

Let C be a finite set of class labels. Consider a finite set D containing tuples of the form (x, c) where x is a point (or record) and $c \in C$ is the associated class label for that point. We assume that a distance metric is available to measure the distance (or dissimilarity) between any two points in D (class labels are not included in this distance computation).

### 3.2 Distance-based Notion of Intrusive Outliers

We adapt the well-known notion of distance-based outliers [2] (Knorr, 2000) to cover the concept of intrusive outliers in labeled data.

**Definition 1.** Let $\varepsilon$ be a given positive real number. Let $D$ be a given set of labeled points and let $(\boldsymbol{q}, c)$ be a given point having label $c$ ($(\boldsymbol{q}, c)$ may or may not be in $D$). Then the $\varepsilon$-neighbourhood of $\boldsymbol{q}$, denoted $N_\varepsilon(q)$, is the set of those

points in $D$ whose distance from $\boldsymbol{q}$ is less than or equal to $\varepsilon$ (class labels are not considered when computing this distance). Let $0 \le p \le 100.0$ be a real number. Labeled point $(\boldsymbol{q}, c)$ is $(\varepsilon, p)$-anomalous if the labels of at least $p\%$ points in $N_\varepsilon(\boldsymbol{q})$ (i.e., in the $\varepsilon$-neighbourhood of $\boldsymbol{q}$) are different from the label $c$ of $\boldsymbol{q}$.

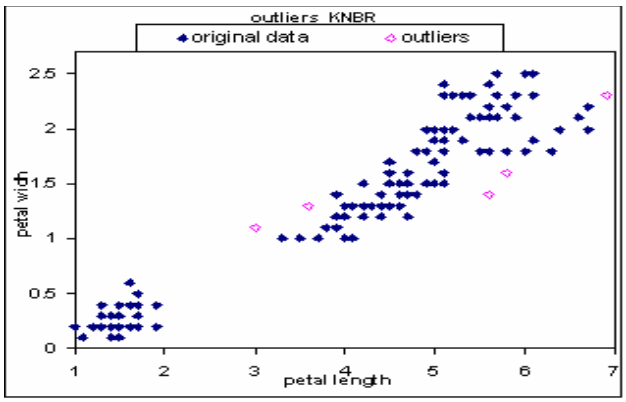


**Fig. 1. Unlabeled outlier detection in unlabeled Iris 2-D data with attributes 3 and 4.**

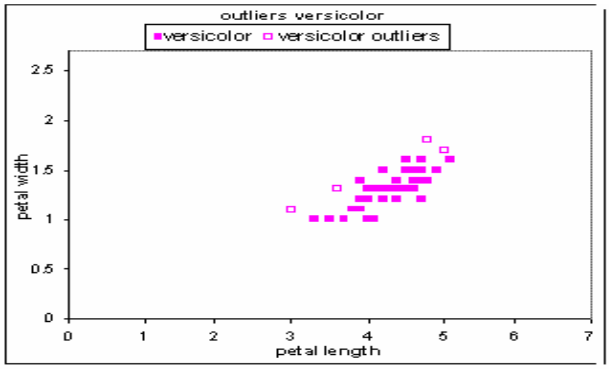


**Fig. 2. Class-wise outliers in unlabeled Iris 2-D data with attributes 3 and 4.**

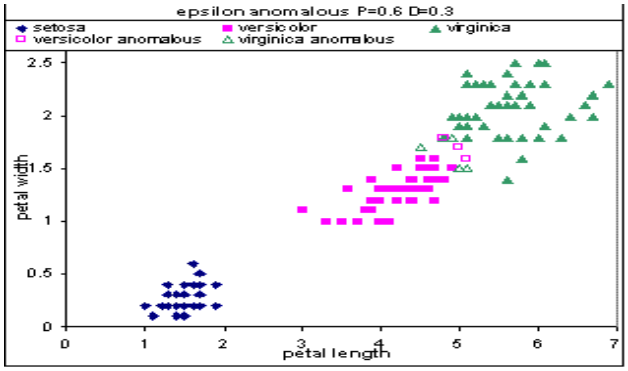


**Fig. 3. Anomalous points detected by knorr_class algorithm in labeled 2-D Iris dataset.**

**Algorithm** knorr_class

1. Given: $p$, $\varepsilon$, and set D
2. For each point $q_i$ in $D$ compute set $A$ of all points in $(D-p_i)$ which are within a distance $\varepsilon$ of $q_i$
3. Let $m$ = percentage of points in $A$ whose label does not agree with that of $q_i$
4. If $m \ge p$ then $q_i$ is an intrusive outlier

Fig. 3 shows the intrusive outliers detected by algorithm knorr_class in the iris 2 dimensional dataset considering only petal width and petal length.

### 3.3 kNN-based Notion of Intrusive Outliers

In this section, we extend the RRS notion of outliers [7] (Ramaswamy et al., 2000) to deal with labeled data.

**Definition 2.** Let $k \ge 1$ be a given integer. Let $kNN(\boldsymbol{q})$ denote the set of nearest $k$ neighbours of $\boldsymbol{q}$ (class labels are ignored when computing this set of points and $\boldsymbol{q}$ itself is not included in this set). Let $0 \le p \le 100.0$ be a given real number. Then $(\boldsymbol{q}, c)$ is an $(k, p)$–anomalous if at least $p\%$ points in $kNN(\boldsymbol{q})$ have label other than c.

We adapt the RRS outlier detection algorithm [7] (Ramaswamy et al., 2000) to identify $(k, p)$-anomalous points in the given labeled dataset (for given values of $\varepsilon$ and $p$).

**Algorithm** RRS_class

1. Given: $p$, $k$; and set D
2. For each point $q_i$ in $D$ compute set $A$ of all points in $(D-p_i)$ which are $k$- Nearest Neighbors of $q_i$
3. Let $m$ = percentage of points in $A$ whose label does not agree with that of $q_i$
4. If $m \ge p$ then $q_i$ is an intrusive outlier

Fig. 4 shows the 6 intrusive outliers detected by the algorithm RRS_class ($P = 0.8$ and $k = 8$) in the labeled Iris 2-D dataset (containing only attributes petal length and petal width). Note that all the 6 intrusive outliers detected by the RRS_class algorithm are also detected by the knorr_class algorithm. This gives us confidence that both these definitions do well to capture the notion of intrusive outliers.

## 4. Hard-to-classify points in labeled datasets

### 4.1 Hard-to-classify Points: General Scheme

Intuitively, a point is hard-to-classify if majority of classifiers fail to correctly classify this point. Removal of these points from the training dataset avoids distortion of the classification boundary and makes it more robust to noise in the attribute values.

**Definition 3.** Let $D$ be a given set of labeled points. Let $S = \{C_1, C_2, ..., C_m\}$ be an ensemble classifiers, trained on $D_1$, $D_2$, …, $D_m$, where each $D_i \subseteq D$. Let $0 \le p \le 100.0$ be a real number. A labeled point $(\boldsymbol{q}, c)$ is $p$-anomalous with respect to $S$ if at least $p\%$ of the classifiers in $S$ misclassify the point $q$.

In this paper we use boosting [8](Schapire, 1999), to construct an ensemble of SVM classifiers.

**Algorithm** hard_to_classify_using_boosting

1. Given: *p, s*et *S* (ensemble of *m* classifiers created using boosting algorithm) and set *D*
2. Classify each point $q_i$ in *D* using the m classifiers in *S*
3. Let *count* = percentage of classifiers in *S* that misclassified point $q_i$
4. If *count* ≥ *p* then $q_i$ is a hard-to-classify point

Fig. 5 shows the hard-to-classify points (*p* = 60%) detected by the algorithm hard_to_classify_using_boosting in the labeled 2-D Iris dataset. An ensemble of *m* = 100 SVM classifiers was created using boosting. Each of the 3 hard-to-classify points reported were misclassified by at least 60 SVM classifiers (out of 100). Note that these 3 hard-to-classify points are included in the ($\varepsilon$, *p*)-anomalous and (*k*, *p*)-anomalous points detected by the algorithms proposed in Section 3.

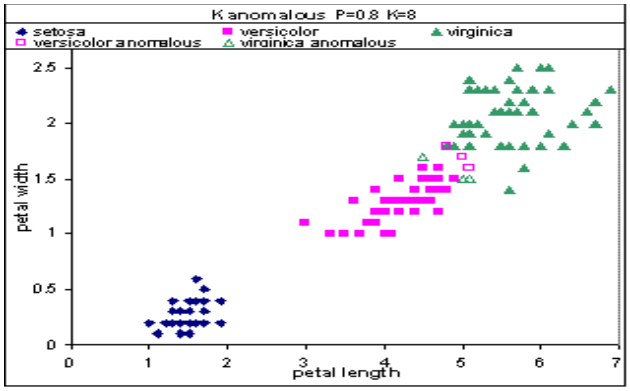


**Fig. 4. (*k*,*p*)-anomalous points detected by RRS_class algorithm in labeled 2-D Iris dataset.**

## 5. Experimental results

In this section, we empirically compare the notions of intrusive outliers and hard-to-classify points and support our hypothesis that removal of hard-to-classify points improves the robustness of the classification boundary against future noisy points.

### 5.1 Experiment 1

In this experiment we empirically compare the sets of anomalous points obtained using knorr_class, RRS_class and hard_to_classify_using_boosting algorithms on datasets from UCI machine learning repository [4](Merz and Murphy, 1996). We created an ensemble of 100 SVM classifiers using the boosting algorithm in order to identify the hard-to-classify points in a dataset. The kernel parameters and cost of misclassification for a single classifier were optimized for best accuracy using k-fold cross-validation on each dataset. The same kernel parameter values were used for all SVM classifiers subsequently created using boosting. We used the SVMlight toolset version 6.0.1 (http://svmlight.joachims.org/) for these experiments. Table 1 shows that there is a significant overlap between hard-to-classify points (set A), the ($\varepsilon$, *p*)-anomalous points (set B) and the (*k*, *p*)-anomalous points (set C) confirming our intuition that these 3 notions of anomalous points are closely related, though of course, not identical.

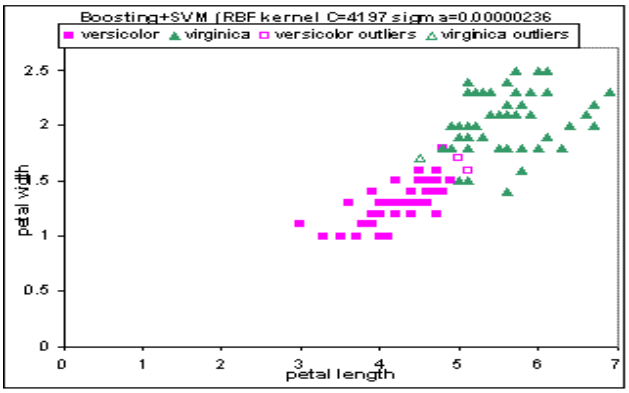


**Fig. 5. Anomalous points (p = 60%) detected by hard_to_classify_using_boosting algorithm in labeled 2-D Iris dataset.**

### 5.2 Experiment 2

To support our hypothesis that the removal of the hard to classify points improves the robustness of the classification boundary we designed the following experiment.

In a given dataset *D*, the set of hard to classify points, denoted by A, are identified. We randomly sample 20% of the data from *D* and keep that as the test set and denote it by $D_{test}$. The remaining 80% of the data is denoted by $D_{train}$. We then remove all the hard to classify points that appear in *D* and denote the resulting set as $D'_{train}$. Note that the test set $D_{test}$ still contains a few of the hard to classify points.

This results in two training sets, $D_{train}$ which includes the hard to classify points, and $D'_{train}$ which does not include the hard to classify points. Decision Tree [6](Quinlan, 1993) classifiers *C* and *C*′ are trained on $D_{train}$ and $D'_{train}$ respectively. Both classifiers *C* and *C*′ are tested on $D_{test}$. The corresponding errors $\varepsilon$ and $\varepsilon'$ are recorded.

This experiment is repeated 50 times each time choosing a different random sample Dtest.

This supports our hypothesis that the hard to classify points have an adverse influence on the classification boundary and removal of these points improves the performance of the classifier by making it more robust to noisy points in the dataset.

## 6. Conclusions and further work

We formalized two notions of noisy points: intrusive outliers and hard-to-classify points. An intrusive outlier is "too far out" of its own class cluster and "intrudes" into the cluster of another class. A point is hard-to-classify if it is misclassified by many classifiers. We have empirically compared these two notions of noisy points and found that there is significant overlap among them. We have found that removing hard-to-classify points from the training dataset makes the classification boundary robust and

increases the classification accuracy over the remaining unseen points.

**Table 1. Comparison of the 3 algorithms for detection of anomalous points**

| DATASETS | Size | γ | Cost of mis-classification C | Error | \|A\| \|A\|/Size | \|B\| \|B\|/Size % | \|C\| \|C\|/Size % | \|A∩B\| \|A∩B\|/A % | \|A∩C\| \|A∩C\|/A % | \|B∩C\| \|B∩C\|/B % | \|A∩B∩C\| |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IRIS(versiocolor and virginica) k=3 p=0.9 ε=0.5 p=0.8 | 100 | 1.73 | 1.7393 | 5.0% | 3 3.00% | 3 3.00% | 5 5.00% | 1 33.33% | 2 66.67% | 1 33.33% | 1 |
| LIVER k=3 p=1.0 ε=2.3 p=0.9 | 345 | 961.6 | 961.68 | 24.63% | 15 4.35% | 16 4.64% | 21 6.09% | 9 60.00% | 7 46.67% | 2 12.50% | 2 |
| WISC-BC k=7 p=0.7 ε=2.7 p=0.9 | 699 | 1.26 | 1.26 | 2.71% | 16 2.29% | 18 2.58% | 17 2.43% | 15 93.75% | 14 87.50% | 16 88.89% | 14 |
| WISC-DIAG k=9 p=0.8 ε=3.2 p=0.8 | 569 | 14.27 | 14.27 | 1.58% | 4 0.70% | 4 0.70% | 7 1.23% | 3 75.0% | 4 100.0% | 4 100.0% | 3 |
| WISC-PROG k=15 p=1.0 ε=3.8 p=1.0 | 198 | 29.82 | 29.82 | 19.1% | 1 0.5% | 2 1.01% | 1 0.5% | 1 50.0% | 1 100% | 1 50% | 1 |

A = set of hard-to-classify points output by hard_to_classify_using_boosting using SVM classifiers
B = set of ($\varepsilon$, $p$)-anomalous points output by algorithm knorr_class
C = set of ($k$, $p$)-anomalous points output by algorithm RRS_class

This technique can be further extended for detecting noisy points in labeled time-series data. We are also applying the proposed techniques on several large real-life labeled datasets, to check their effectiveness in practical applications.

**Table 5. Effects of removing the hard-to-classify points on the accuracy of the classifier.**

| Dataset | ε | ε′ |
|---|---|---|
| LIVER | 36.00±5.85 | 32.16±6.39 |
| WISC-BC | 6.27±1.87 | 6.18±4.66 |
| WISC-DIAG | 18.55±22.71 | 12.63±16.84 |
| WISC-PROG | 32.36±13.62 | 29.22±10.71 |

## 7. References


[1] He Z., Xu Z., Huang J.Z., Deng S., 2004. Mining class outliers: concepts, algorithms and applications in CRM, Expert Systems with applications, 27, pp. 681 – 697.

[2] Knorr E.M., Ng R.T., Tucakov V., 2000. Distance-based outliers: algorithms and applications, VLDB journal, 8 (3 – 4), pp. 237 – 253.

[3] Markou M., Singh S., 2003. Novelty detection: a review – part 1: statistical approaches, Signal Processing, 83, 2481 – 2497.

[4] Merz G., Murphy P., 1996. UCI repository of machine learning databases Technical Report, University of California: Department of Information and Computer Science http://www.ics.uci.edu/mlearn/MLRepository.html.

[5] Papadimitriou S., Faloutsos C., 2003. Cross-outlier detection, Proceedings of the SSTD03, pp. 199–213.

[6] Quinlan, J. R. (1993), C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, California.

[7] Ramaswamy S., Rastogi R., Kyuseok S., 2000. Efficient algorithms for mining outliers from large data sets, Proc. of the SIGMOD00, pp. 93–104.

[8] Schapire R.E., 1999. A brief introduction to boosting, in Proceedings of the 16th International Joint Conference on Artificial Intelligence, pp. 1401 – 1406.

[9] Tax D.M.J., Duin R.P.W., 1998. Outlier detection using classifier instability, In Advances in Pattern Recognition, Joint IAPR International Workshops, pp. 593-601.