LA-UR- 03-2206

Approved for public release;
distribution is unlimited.

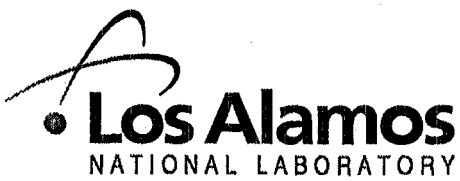
qq

Title: RESAMPLING APPROACH FOR ANOMALY DETECTION IN MULTISPECTRAL IMAGES

Author(s): James Theiler
D. Michael Cai

Nonproliferation and International Security Division
Los Alamos National Laboratory

Submitted to: Proc. SPIE 5093
(To be presented at Aerosense,
21-25 April 2003, Orlando, FL)



Resampling Approach for Anomaly Detection in Multispectral Images

James Theiler[1] and D. Michael Cai[2]

[1]Space and Remote Sensing Sciences Group and [2]Space Data Systems Group,
Los Alamos National Laboratory, Los Alamos, NM 87545

ABSTRACT

We propose a novel approach for identifying the "most unusual" samples in a data set, based on a resampling of data attributes. The resampling produces a "background class" and then binary classification is used to distinguish the original training set from the background. Those in the training set that are most like the background (*i.e.,* most unlike the rest of the training set) are considered anomalous. Although by their nature, anomalies do not permit a positive definition (if I knew what they were, I wouldn't call them anomalies), one can make "negative definitions" (I *can* say what does *not* qualify as an interesting anomaly). By choosing different resampling schemes, one can identify different kinds of anomalies. For multispectral images, anomalous pixels correspond to locations on the ground with unusual spectral signatures or, depending on how feature sets are constructed, unusual spatial textures.

Keywords: anomaly detection, machine learning, multispectral imagery

1. INTRODUCTION

The job of the professional image analyst is to find things in imagery. Often the analyst knows ahead of time what kinds of things to look for: landing strips, industrial facilities, soybean crops, *etc.* But sometimes, the analyst is confronted with the more open-ended task of finding "unusual" things in the images, without knowing ahead of time what those unusual things will be.

When the target of interest *is* known, and for high-value targets in particular, it may be worth the effort to develop specialized automated target recognition (ATR) systems to aid – or, in some optimistic scenarios, to replace – the analyst. A less expensive approach is to employ *supervised learning.* The analyst "marks up" pixels in a set of training imagery which contain the item of interest and also marks up as negative controls a sample of pixels which do not contain the item. A machine learning system uses these examples to "train" a classifier to identify the target in new imagery. (For one example of this approach, see Ref. [1].) This may work better for some targets than for others, but it does have the benefit of flexibility. The same system can be employed for a wide variety of target types – all that changes is the analyst's markup. Although it can be a somewhat laborious process to mark up on a pixel-by-pixel basis of just where the target is and where it is not for an adequate quantity of imagery, obtaining this markup is easier than developing a full-up ATR system from first principles. And the domain knowledge of the expert is directly exploited, by the the production of the markup, instead of indirectly elicited as a set of "fuzzy rules" in which the analyst tries to explain to the computer programmer how the targets can be identified.

But a different problem arises when examples of the target of interest are unavailable, or when the target of interest is just plain unknown. The analyst would like to mark up whole images as "normal" and use that for training. This is the anomaly detection problem: it is a kind of *unsupervised* classification in which the "learning by example" proceeds without any examples of the target itself.

One problem with this open-ended statement of the problem is that it is easy to provide a "solution" which optimizes some mathematical formulation but which the analyst nonetheless finds unsatisfactory. The pixels in an image that are brightest are, in some sense, anomalous (they are "unlike most of the other pixels"), but they may not be especially interesting. Because of what anomalies are, the analyst cannot point positively to certain kinds of features as anomalous, but it would still be useful for the analyst to at least rule out some kinds of anomalies that are known *a priori* to be uninteresting.

E-mail: {jt,dmc}@lanl.gov

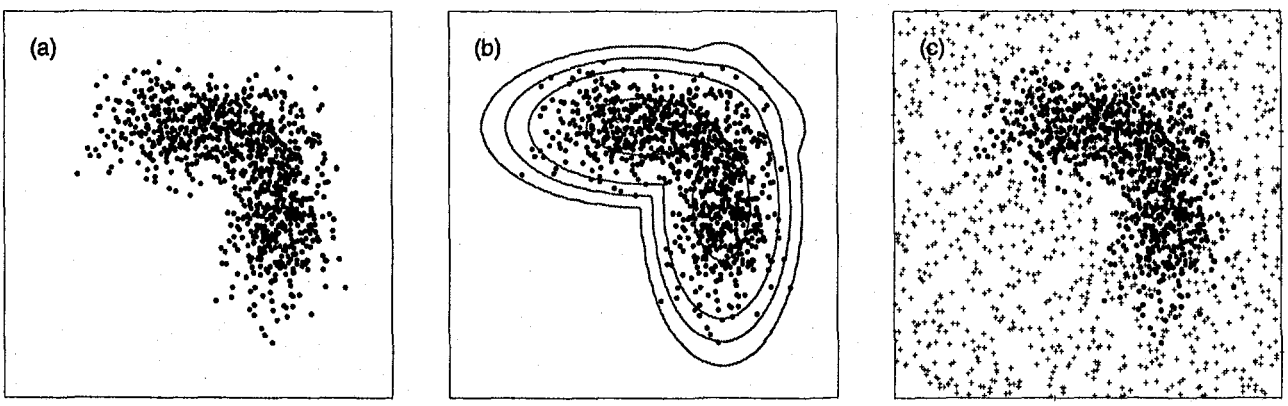


Figure 1. (a) $N = 1000$ spots, randomly selected from a distribution $P(x)$, which is a sum of two gaussians. (b) Contours around the spots at the levels of $\alpha = 0.001$, $\alpha = 0.05$, $\alpha = 0.1$, and $\alpha = 0.5$. These contours are based on the known underlying distribution $P(x)$, and represent the smallest area sets which enclose a fraction $1 - \alpha$ of the normal data. (c) A uniform background (indicated with + symbols) transforms the anomaly detection problem into a binary classification problem.

All happy families are alike, but unhappy families are all unhappy in their own way.

– Leo Tolstoy, *Anna Karenina*

2. DEFINE "ANOMALY"

The dictionary[2] provides two related definitions for the word "anomaly". The first is "deviation or departure from the normal or common order, form, or rule."

An anomaly detection algorithm, then, would be some kind of mathematical formula or model which describes the data. Data which fit this description are normal; data which do not fit are considered anomalies. Note that this is a negative definition: the anomalies are the data samples that do *not* conform to the rule.

By the second definition, an anomaly "is peculiar, irregular, abnormal, or difficult to classify." This definition highlights an important property of the kinds of anomalies that are usually sought. Anomalies are outliers, and are as different from each other as they are from normal cases. Like Tolstoy's unhappy families, anomalies tend to be anomalous in their own way.

We briefly remark that this point of view of anomalies as anomalous even to each other applies only for data sets in which the samples are truly independent. With images, for instance, pixels are very often highly correlated with neighboring pixels, and an anomalous object in a scene might correspond to several pixels which are unlike the rest of the image but are nonetheless close to each other.

3. MATHEMATICAL FORMULATIONS

As is generally the case with unsupervised learning problems, the mathematical formulation of the problem itself is nontrivial. First we will describe what the problem is:

> We are given a dataset with N samples, $\{x_1, \ldots, x_N\}$, with each $x_i \in \mathcal{R}^d$. Our goal is to find the "most anomalous" subset of these points. For instance, given a small scalar $\alpha \ll 1$, identify the αN samples that are most unlike the rest of the data.

This description of the problem is not yet a formulation; not only does it not tell us how to go about solving it, it does not even provide a criterion for knowing whether or not we have succeeded. If we chose αN samples at random and called them the anomalies, who could contradict us? Fig. 1(a) illustrates this situation: we have

N data points and nothing else – no labels, no parametric models, no underlying probability densities. To help clarify our thoughts, we will contrast this statement of the problem with an extremely idealized version:

> We are given a probability distribution[3] $P(x)$ from which normal data samples are drawn, and a distribution $Q(x)$ which describes the anomalies. We want to specify a set $\mathcal{S}_\alpha \subset \mathcal{R}^d$ which has the property if x is drawn from $P(x)$, then with probability at least $1-\alpha$, it will be in the set $\mathcal{S}_\alpha$. But if x is drawn from $Q(x)$ then it is *unlikely* to be in $\mathcal{S}_\alpha$. Here $x \notin \mathcal{S}_\alpha$ indicates the x will be labelled as an anomaly.

Thus, our goal is to optimize

$$\max \quad \int \mathcal{I}(x \notin \mathcal{S}_\alpha) Q(x)\, dx \tag{1}$$

$$\text{such that} \quad \int \mathcal{I}(x \notin \mathcal{S}_\alpha) P(x)\, dx \leq \alpha \tag{2}$$

where $\mathcal{I}$ is the indicator function; it is one if its argument is true, and is zero otherwise. Here, the first integral corresponds to the "detection rate" for anomalies, and the second integral corresponds to the "false alarm rate." And if $P(x)$ and $Q(x)$ are both known, then the solution is given by sets $\mathcal{S}_\alpha$ whose boundaries are contours of constant ratio $Q(x)/P(x)$.

3.1. Hypothesis testing

In the language of hypothesis testing, we would say that x being generated by $P(x)$ is the *null hypothesis.* We are looking for a discriminating *statistic* $s(x)$ with a threshold t_α that depends on α such that

$$\int \mathcal{I}(s(x) \leq t_\alpha) P(x)\, dx = 1 - \alpha \tag{3}$$

Thus, if we observe a sample value x, then if $s(x) > t_\alpha$ we can *reject* the null hypothesis with a *p-value* of α.

Here, the function $s(x)$ is a measure of how anomalous a data sample is, and the recipe for finding anomalies is to apply this function to the data and choose a threshold for which the fraction α of the data with the largest "anomaly rating" are identified as anomalous.

If you actually know what kind of anomaly you are looking for, then you devise $s(x)$ to incorporate this knowledge. In particular, if $Q(x)$ is known, then $s(x) = Q(x)/P(x)$ is an optimal[4] measure of "anomalousness."

3.2. Smallest volume approach

Since $Q(x)$ cannot be estimated from data, it must be directly specified. We can formalize our ignorance of what anomalies we expect to see by choosing $Q(x)$ to be as uninformative as possible. The usual choice is for $Q(x)$ to be a flat function over an area that is much larger than the support of the data. (In fact, we can take the limit as this area goes to infinity; then $Q(x)$ is no longer a probability, but it is still a measure – in this case, it is the Lesbegue measure – and that is adequate for our purposes.)

So our choice for $\mathcal{S}_\alpha$ is the smallest volume set for which Eq. (2) holds. With this choice of $\mathcal{S}_\alpha$, the boundary of $\mathcal{S}_\alpha$ will be a contour of the density $P(x)$.

This leads us to our final formulation of the anomaly detection problem. Since neither $P(x)$ nor $Q(x)$ are known, we seek to estimate $P(x)$ from the data and to assert that $Q(x)$ is known, even though the usual choice of $Q(x)$ is deliberately uninformative. Thus, anomaly detection is cast as a data-versus-density problem:

> We are given N samples, $\{x_1, \ldots, x_N\}$, with each $x_i \in \mathcal{R}^d$ and each sample assumed to be drawn randomly from an unknown distribution $P(x)$. We are furthermore given a known distribution $Q(x)$. Our goal is to find sets $\mathcal{S}_\alpha$ for which Eqs. (1,2) are optimized. Points $x_i \notin \mathcal{S}_\alpha$ will be labelled anomalous.

The definition of anomalies in terms of the "smallest volume set" is mathematically well-defined, and although it does not impose explicit conditions on the nature of anomalies, it does requires that you define a metric on your space (that is, $Q(x)$) so that volume can be measured, and this implicitity imposes prejudices. It can depend, for instance, on choice of coordinate system. In Fig. 1(b), density contours have be plotted over the data points. The goal of the anomaly detection problem is to infer these contours just from the data.

4. ALGORITHMS

4.1. One-class classifiers

4.1.1. Direct estimation of $P(x)$

In the real problem, $P(x)$ is not known, but one can try to estimate it from the data. We remark that direct density estimation is, by itself, an ill-posed problem; the maximum-likelihood solution, for instance, is the sum of N delta functions centered on the data points. For any finite N, this is not a useful estimate for identifying anomalies. In this section we will describe two standard approaches for directly estimating density, and show how they can be used for anomaly detection.

The first, and most straightforward, begins with the assumption that $P(x)$ is a multivariate gaussian:

$$P(x) = \frac{1}{(2\pi)^{d/2}|K|^{1/2}} \exp\left[-\frac{1}{2}(x-x_o)^T K^{-1}(x-x_o)\right] \tag{4}$$

If this gaussian has centroid at x_o and covariance given by the matrix K, then the contours are ellipses given by constant values of

$$T^2(x) = (x-x_o)^T K^{-1}(x-x_o). \tag{5}$$

This is the Mahalanobis distance from the centroid, and is sometimes called the the Hotelling T^2 statistic.[5] In practice x_o is taken to be the sample mean of the data, and K is a regularized sample estimate of the covariance, that is:

$$K = \frac{1}{N}\sum_{i=1}^{N}(x_i - x_o)(x_i - x_o)^T + \lambda I. \tag{6}$$

The choice of regularization can in some cases be a delicate issue. Its purpose is both numerical (to ensure that the matrix K is invertible) and statistical (to reduce the effect of finite N sample error). If K is invertible in the limit of large N, then it is possible (for N large and d small) to get away with $\lambda = 0$. If $P(x)$ is indeed gaussian, then this method is asymptotically optimal; but for nongaussian $P(x)$, the method is not even consistent – that is, the $N \to \infty$ limit does not approach the true distribution. In Fig. 2(a), we illustrate the fit of this gaussian to artificial data that was generated from a mixture of two gaussians.[6]

A second approach is to use Parzen windows. Chapter 6 of Fukunaga's text[7] describes this method in some detail. The idea is to estimate the density with a regularized version of the sum of delta functions; the most popular choice is as a sum of gaussians centered on each data point. That is,

$$P(x) \propto \sum_{i=1}^{N} \exp(-\gamma\|x - x_i\|^2) \tag{7}$$

Here γ is a kind of smoothing parameter, and its choice is something of an art; as $\gamma \to \infty$, the estimate approaches a sum of delta functions. If $\gamma \sim N^{1/2}$, then the estimator is consistent in the $N \to \infty$ limit. See Fig. 2(b,c).

4.1.2. One-class support vector machines

Although direct estimation of the underlying distribution $P(x)$ from a finite sample of data is problematic, Ben-David and Lindenbaum[8] introduced a machine learning approach in which contours of $P(x)$ can be estimated with functions of bounded complexity. Theoretical bounds on the error were obtained for finite N (not just the $N \to \infty$ limit) and are independent of the underlying distribution $P(x)$.

This is the same mathematical framework that is the basis of support vector machines[9] (SVM), and SVM-based approaches to anomaly detection have been developed by Tax *et al.*[10–12] and Schölkopf *et al.*[13–15]

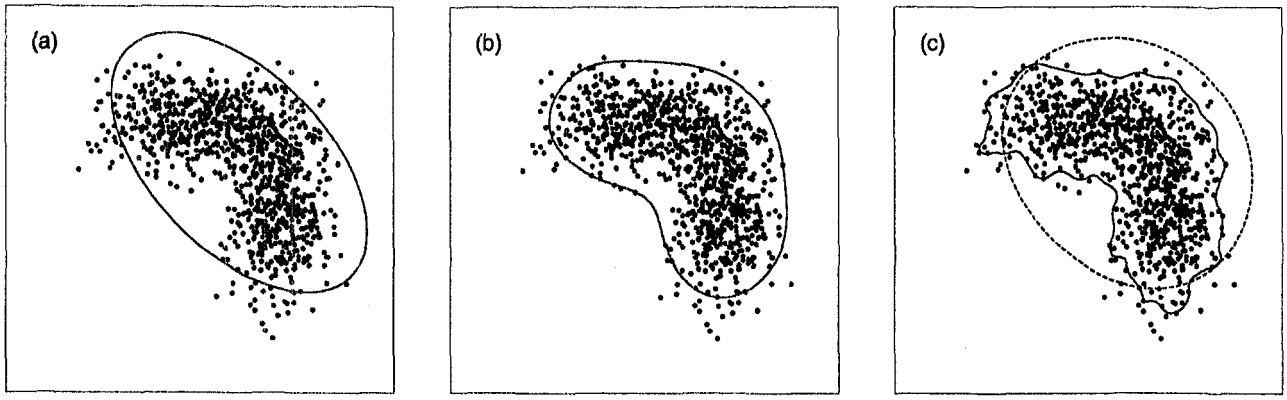


Figure 2. **(a)** A multivariate gaussian model produces an elliptical boundary, shown here corresponding to $\alpha = 0.05$. **(b)** Parzen window density estimation with boundary corresponding to $\alpha = 0.05$, shown here with $\gamma = 1$. Compared to the gaussian fit, the extra flexibility in the Parzen model produces a smaller volume set that still encloses the same amount of data. **(c)** Parzen estimators with $\gamma = 10$ (solid) produces a boundary that hugs the data even closer, but the jagged curve appears to be overfitting the data; on the other hand using $\gamma = 0.1$ produces a boundary that is much smoother, but also much larger area is needed to enclose the same amount of the data.

We describe here the algorithm of Tax and Duin.[10] The idea is to find the center x_o and radius R of the smallest sphere that encloses "most" of the data. This is specified in terms of an optimization problem.

$$\min_{R,x_o} R^2 + C \sum_i H(\|x_i - x_o\|^2 - R^2) \tag{8}$$

where H is a hinge function

$$H(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{otherwise.} \end{cases} \tag{9}$$

Thus, for points x inside the sphere, $\|x_i - x_o\|^2 < R^2$, the penalty term vanishes. For points outside the sphere, the penalty is proportional to how far outside the sphere they are. By appropriate choice of C, the sphere $\mathcal{S}_\alpha$ can be found to satisfy Eq. (2).

The use of a sphere is on its face a rather restrictive assumption, but this can be addressed by mapping into a "kernel space." For details on kernel methods in general, the reader is invited to read Schölkopf and Smola's illuminating and quite thorough text[15]; we provide here a very brief overview. A function $\phi : \mathcal{R}^d \to F$ maps data points x into a high-dimensional feature space, and dot products in that high-dimensional space are expressed in terms of kernels:

$$K(x, y) = \phi(x) \cdot \phi(y). \tag{10}$$

When the optimal sphere in the kernel space is mapped back to the data space, the solution can be expressed in terms of kernel functions.

$$f(x) = a_o + \sum_{i=1}^{N} a_i K(x_i, x) \tag{11}$$

and this defines the set $\mathcal{S}_\alpha = \{x \in R^d \ : \ f(x) > 0\}$. Two important properties of the optimization function in Eq. (8) are that it is convex (meaning that there are no local minima, just a single global minimum) and that it typically leads to sparse solutions in which $a_i = 0$ for most i; in fact, $a_i = 0$ for all data points x_i inside the sphere. Probably the most popular kernel function is the gaussian radial basis function:

$$K(x, y) = \exp(-\gamma \|x - y\|^2). \tag{12}$$

It is interesting to compare the kernelized one-class SVM estimator in Eq. (11) to the Parzen windows density estimator in Eq. (7). The form of the solution is the same, but where the Parzen estimator puts equal weight

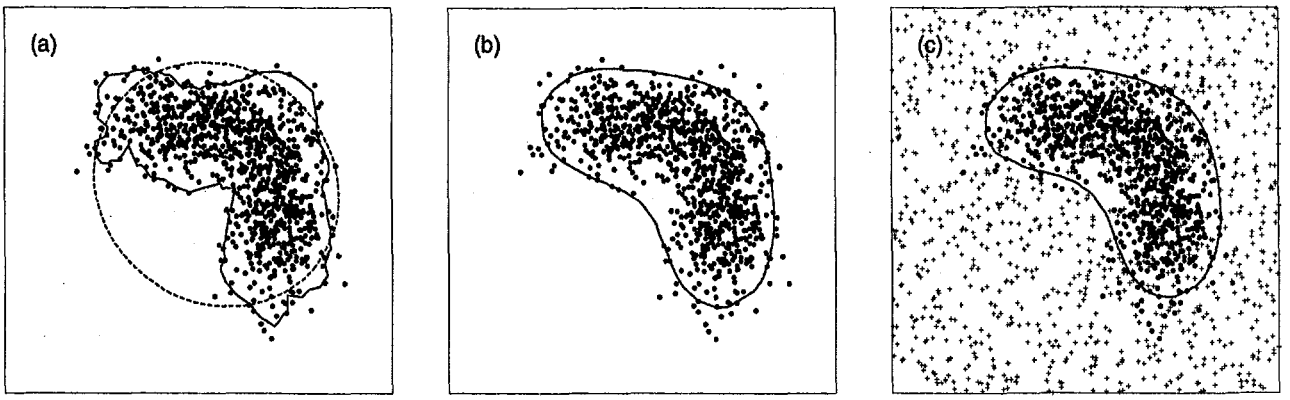


Figure 3. **(a)** The one-class SVM, at $\nu = 0.05$ (which corresponds roughly to $\alpha = 0.05$) employs a gaussian kernel with parameter γ that, in analogy with the Parzen window estimator, must be estimated. Shown here are $\gamma = 1.0$ (solid) and $\gamma = 0.01$ (dashed). **(b)** One-class SVM with $\nu = 0.05$ and $\gamma = 0.1$. **(c)** By adding the uniform background as a second class, an ordinary two-class SVM can be used to model the data.

on all data points and attempts to estimate the whole distribution $P(x)$ at once, the one-class SVM estimator has nonzero weight only on the data points that are on or outside the estimated boundary of $\mathcal{S}_\alpha$. Furthermore, the $f(x)$ in Eq. (11) is not attempting to estimate the entire probability distribution, just the boundary of $\mathcal{S}_\alpha$.

Fig. 3(a,b) illustrates the use of the one-class SVM on the example data introduced in Fig. 1(a). The user must specify both the choice of kernel, and (as in the case of the gaussian kernel) this often involves choices of kernel parameters as well. The one-class SVM algorithm is a clever and important contribution, but it is important to realize that it is optimizing a volume in a kernel space, which is different from optimizing the volume in data space. In a recent paper, Tax and Duin[11] address this issue by using a Monte-Carlo estimate of volume in the data space to choose the kernel parameter γ.

4.2. Recasting anomaly detection as a two-class problem

Since we already know how to solve two-class problems, it is useful to recast the anomaly detection problem in the two-class framework. In this case, the "normal" class is exemplified by the data. The "anomalous" class cannot be specified in terms of data, because we do not have examples of "typical anomalous" data. (Typical anomalous data is an oxymoron, after all.)

Instead, the anomalous class is defined by an underlying measure $Q(x)$, and the more direct two-class variant of anomaly detection can be achieved by producing artifical anomalies by randomly sampling from the $Q(x)$ distribution. This is illustrated in Fig. 1(c), and follows an approach suggested by Hastie *et al.*[16] (in particular, see Fig. 14.3 of that reference) in a different context. There are some obvious obstacles to this approach: a large number of artificial anomalies will lead to extra computational effort, and any finite number of anomalies will only approximate the actual underlying $Q(x)$.

But unlike the kernelized one-class support vector machines, random sampling directly implements the smallest volume condition. And having cast the problem as two-class classification, a wider variety of machine learning algorithms become available. Fig. 3(c) shows the application of an ordinary two-class SVM to the anomaly detection problem by embedding the data in a uniform "background" of artificial anomalies. By the way, the fact that $Q(x)$ might have infinite support is not really a problem. As long as the resampled data extends beyond the boundaries of $\mathcal{S}_\alpha$, then there is no need for it to extend far beyond those boundaries.

With this approach, there is often considerable overlap between the artificial anomalies and the normal data; this can be somewhat unintuitive at first (anomalies are supposed to be *different* from normal data!) but it is equivalent to the minimum volume formulation. This overlap can also be a burden for some classifier algorithms (SVMs, for instance, lose some of their sparseness properties); for this reason, we are investigating the use of

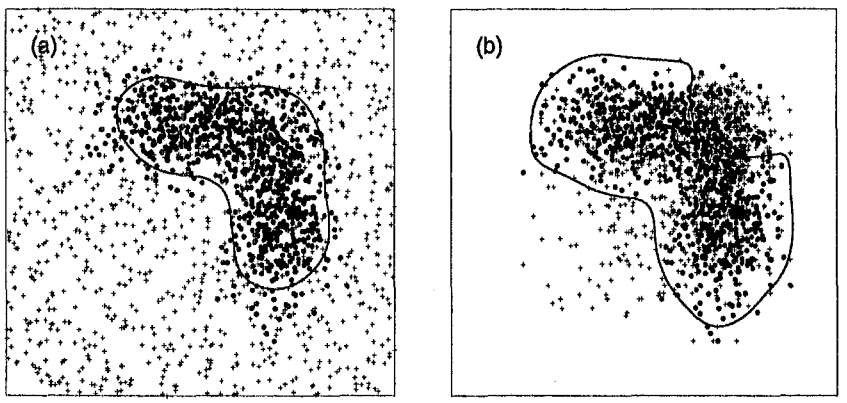


Figure 4. Two class "simple classifier" (linear classifier with gaussian radial basis kernel) applied to two different kinds of random backgrounds. **(a)** Using a uniform background as a second class, the simple classifier produced a result similar to that seen in Fig. 3(c). But the simple classifier is less sensitive than the SVM to the large amount of overlap between the two classes. **(b)** Using a background class obtained from a resampling of the original data in each of its coordinates, a different anomaly detector is obtained. Here there is even more overlap, and the detector takes note of the fact that the density in the upper-right quadrant of the data is much lower in the data than would be expected if the coordinates were independent; thus points in that area are also considered anomalous in that they are providing evidence against the null hypothesis that the coordinates are independent.

the "simple classifier" methodology introduced by Cannon *et al.*,[17–19] for this problem. This investigation is preliminary, but Fig. 4(a) illustrates the simple classifier applied to this problem.

Finally, the random sampling approach permits the analyst to explore other options for $Q(x)$ in a way that discounts some anomalies in favor of others. In particular, we will explore distributions of $Q(x)$ that are derived from the data set itself.

4.3. Resampling schemes

We have described the artificial background of anomalies in terms of a uniform sampling of the state space, but once the idea of using a random background is suggested, a number of possibilities arise in terms of how that background might be resampled. These possibilities can be interpreted in terms of alternative $Q(x)$, but instead of specifying $Q(x)$ directly, it is specified in terms of the input data.

Fig. 4(b) illustrates a different choice for random background. Here, a random point x is produced so that each of its coordinates is chosen randomly from the coordinate values that are in the data. For instance, the first coordinate is given by the first coordinate of the data point x_i for some randomly chosen $i \in \{1, \ldots, N\}$, the second coordinate is given by the second coordinate of the data point x_j for some other randomly chosen $j \in \{1, \ldots, N\}$, and so on. The effective $Q(x)$ is then the outer product of the marginal distributions of $P(x)$ for each of the coordinate directions. Compared to the uniform background, this choice is somewhat more robust to changes of coordinate system. A nonlinear change of coordinates along any of the axes will be reflected in the background as well as in the data. Further, it addresses the issue of different scaling along each axis. Particularly for learning problems where different axes correspond to qualitatively different aspects of the data, the choice of scaling along these axes can have an important influence on the final results.

For images, we have also considered various spatial resampling schemes, but these are beyond the scope of this investigation.

5. EXAMPLE IMAGE

We will illustrate the application of this approach to the multispectral image shown in Fig. 5. This is a 200×200 chip from a four-channel Ikonos scene of Los Alamos, New Mexico. We will use three methods for identifying

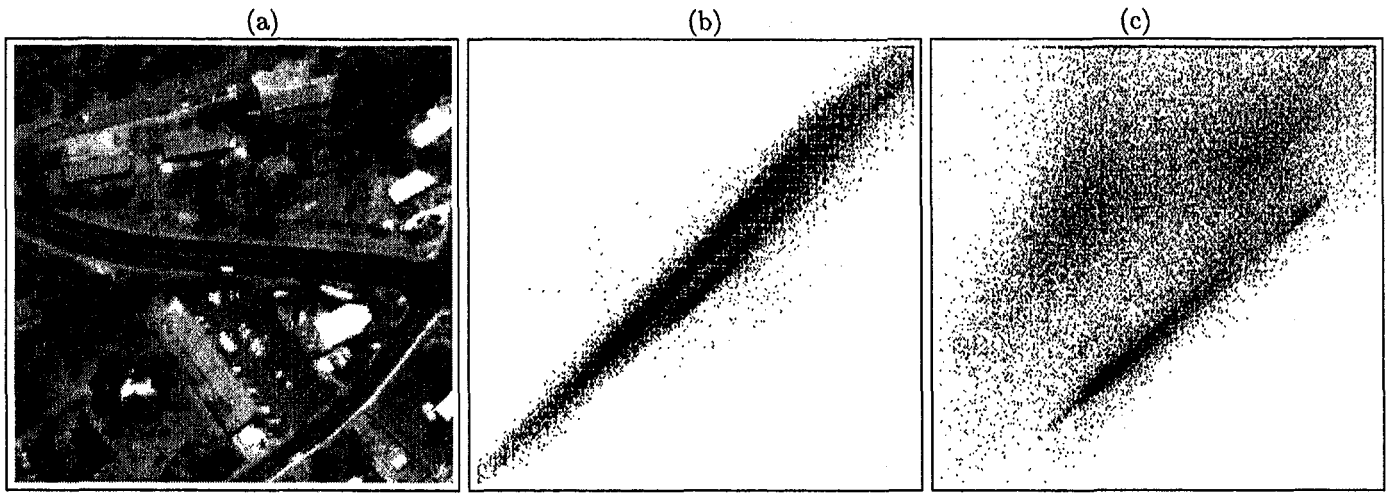


Figure 5. **(a)** Intensity image: sum of the first three (blue, green, and red)channels of a four-channel Ikonos image. **(b)** Scatterplot: Blue channel vs. Green channel. These channels are highly correlated, but there are a few pixels which are some distance from the main diagonal. **(c)** Scatterplot: Red channel vs. Near Infrared channel.

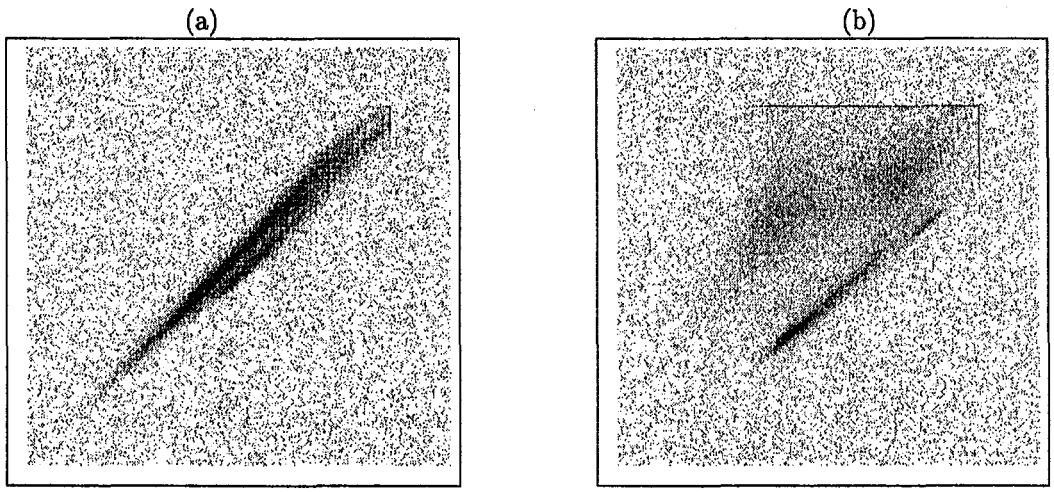


Figure 6. Scatterplots based uniform background; shown are both the uniform background, and superimposed on the background, the original data: **(a)** Blue vs. Green, and **(b)** Red vs. Near IR.

anomalies in this image: a one-class SVM model, a two-class SVM applied to data over a uniform random background, and two-class SVM applied to a background that is coordinate-wise resampled from the image data. All of the SVM models (one-class and two-class) were provided by the `libsvm` package.[20]

Fig. 6 and Fig. 7 show the two different backgrounds that were considered: uniform and coordinate-wise resampling. Using two-class classification to distinguish image data from background data produced formulas that could be applied to the image data to identify the anomalies. Fig. 8 shows the results of both the one-class SVM (which does not explicitly use a background) and the two two-class SVM classifiers. The differences between the one-class SVM and the two-class SVM with a uniform backgroun are visible, but both give qualitatively similar results, identifying anomalous pixels primarily on the basis of their intensity. However, Fig. 8(c) shows that for the two-class SVM with the data-resampled background, it was not unusual intensities but unusual colors that were identified as anomalous.

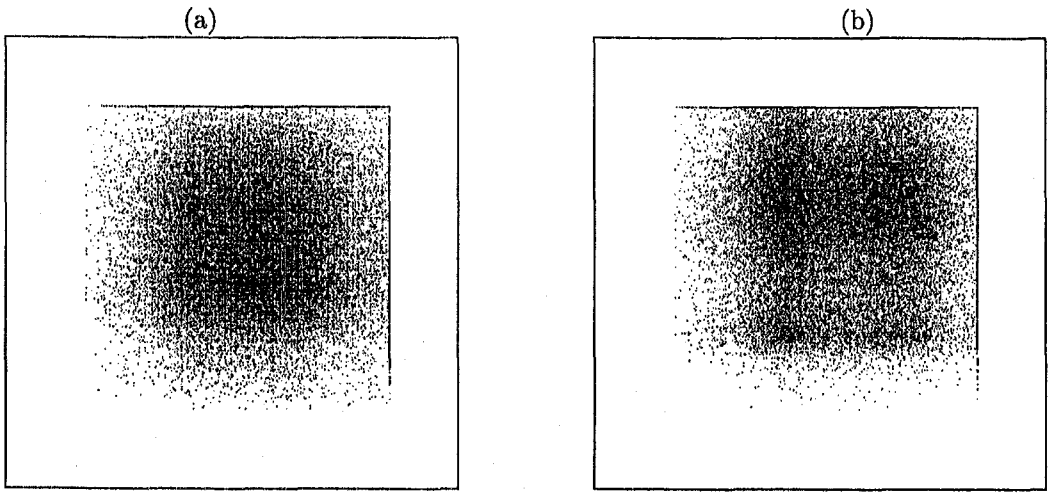


Figure 7. Scatterplots based on the data resampling scheme; unlike the previous figure (with the uniform background), these plots to *not* include a superposition of the original data. **(a)** Blue vs. Green, and **(b)** Red vs. Near IR.

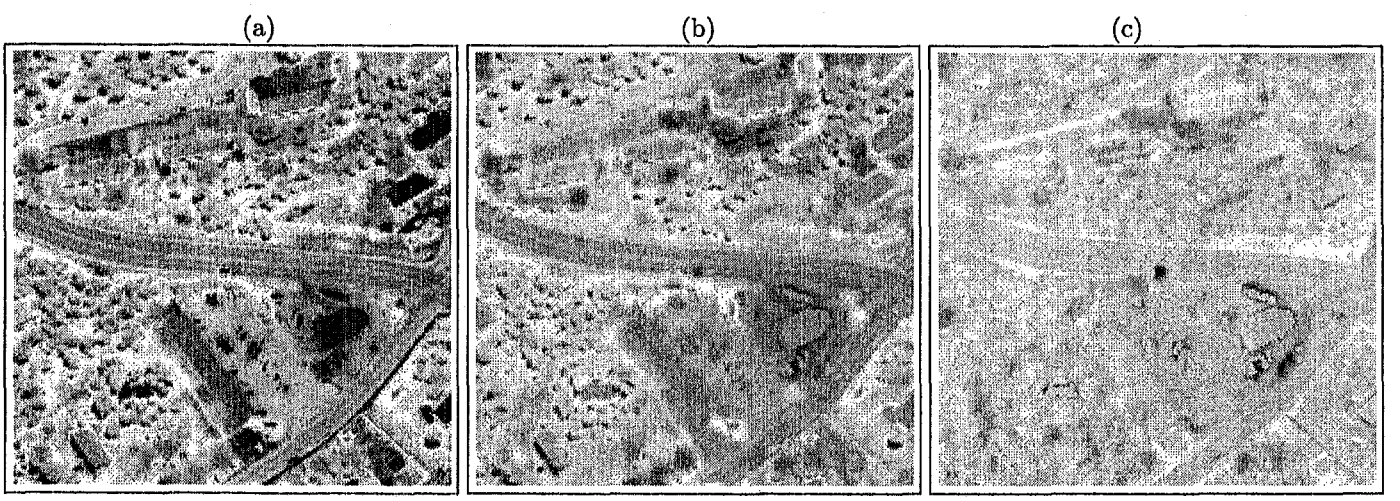


Figure 8. Result of anomaly detection using various anomaly detection schemes; in these images, the darker pixels are the more anomalous. **(a)** Using the one-class SVM, pixels which in the original image are unusually bright (*e.g.*, some of the rooftops) or unusually dark (*e.g.*, shadows of the trees) show up as anomalous. **(b)** Using an ordinary two-class SVM with a uniform background (as seen in Fig. 6) produces a similar result, though the darker pixels are considered more anomalous than the bright pixels. **(c)** Anomalies found in this image used the resampled background seen in Fig. 7. Here, it is anomalous "colors" that are identified; unusually dark or unusually bright pixels are not identified as anomalous. The main anomaly (seen here in the center of the image) shows up in the color image as a yellow-green glow just behind a truck on the roadway.

ACKNOWLEDGMENTS

This work was supported by the Laboratory Directed Research and Development program at Los Alamos. We are grateful to our colleagues on the Real World Machine Learning project for discussion and encouragement. We in particular thank James Howse for making available to us his implementation of simple classifiers.

REFERENCES

1. N. R. Harvey, J. Theiler, S. P. Brumby, S. Perkins, J. J. Szymanski, J. J. Bloch, R. B. Porter, M. Galassi, and A. C. Young, "Comparison of GENIE and conventional supervised classifiers for multispectral image feature extraction," *IEEE Trans. Geosci. and Remote Sens.* **40**, pp. 393–404, 2002.
2. *The American Heritage Dictionary of the English Language, Fourth Edition*, Houghton Mifflin Company, 2000. (`www.dictionary.com`).
3. We use $P(x)$ to denote a probability distribution, but it is not necessarily the case that $P(x)$ is at all smooth or even that it is a well-defined function; properly, a probability distribution needs to be described in terms of a measure μ on a σ-algebra of measureable subsets of the domain $\mathcal{R}^d$. But this level of formality does not serve our purposes.
4. Actually, it is only necessary that $s(x) = h(Q(x)/P(x))$ where h is a monotonic function. This looseness in the definition of optimal $s(x)$ suggests a strategy for designing sub-optimal anomaly detectors when – as is the case – both $P(x)$ and $Q(x)$ are not known. One chooses $s(x)$ to measure some quantity, like brightness, or red-ness, or smoothness, that describes a data point x and then calibrates $s(x)$ against normal data. This provides an "anomaly" detector that is sensitive to the desired property. Of course, the problem with this approach is that the very nature of an anomalies makes the identification of their properties formally impossible. Informally, however, there may be times when this approach is useful.
5. *Matlab Statistical Toolbox User's Guide*, MathWorks, 2001.
6. We remark that the estimate of a distribution with a mixture of gaussians is another time-honored approach, and a popular one in remote sensing. Finite mixture models have even more parameters than single gaussians – which makes overfitting even more of a concern – but are still unable to claim consistency in the $N \to \infty$ limit. Still they provide an often useful middle ground between single gaussian models and Parzen windows.
7. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Morgan Kauffman, San Francisco, 2nd ed., 1990.
8. S. Ben-David and M. Lindenbaum, "Learning distributions by their density levels: A paradigm for learning without a teacher," *J. Computer and System Sciences* **55**, pp. 171–182, 1997.
9. C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery* **2**, pp. 121–167, 1998.
10. D. Tax and R. Duin, "Data domain description by support vectors," in *Proc. ESANN99*, M. Verleysen, ed., pp. 251–256, D. Facto Press, (Brussels), 1999.
11. D. Tax and R. Duin, "Uniform object generation for optimizing one-class classifiers," *J. Machine Learning Res.* **2**, pp. 155–173, 2002.
12. D. Tax and P. Juszczak, "Kernel whitening for one-class classification," in *Pattern Recognition with Support Vector Machines*, S.-W. Lee and A. Verri, eds., vol. 2388 of *Lecture Notes in Computer Science*, pp. 40–52, Springer Verlag, (Berlin), 2002.
13. B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method of novelty detection," in *Advances in Neural Information Processing Systems*, S. A. Solla, T. K. Leen, and K.-R. Mller, eds., vol. 12, pp. 582–588, MIT Press, 2000.
14. B. Schölkopf, J. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation* **13**, pp. 1443–1471, 2001.
15. B. Schölkopf and A. J. Smola, *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond.*, MIT Press, Cambridge, MA, 2002.
16. T. Hastie, R. Tibshirani, and J. Friedman, *Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer-Verlag, New York, 2001.

17. A. Cannon, J. M. Ettinger, D. Hush, and C. Scovel, "Machine learning with data dependent hypothesis classes," *J. Machine Learning Res* **2**, pp. 335–358, 2002.
18. A. Cannon, J. Howse, D. Hush, and C. Scovel, "Simple classifiers," Submitted to: *IEEE Trans. Neural Networks* , 2003.
19. A. Cannon, J. Howse, D. Hush, and C. Scovel, "Simple classifiers from data dependent hypothesis classes," Submitted to: *International Conference on Machine Learning* , 2003.
20. C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines." Software made available by the authors from `http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html`.