International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 3 (2011) pp.061-071
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Anomaly, Novelty, One-Class Classification: A Comprehensive Introduction

Anna M. Bartkowiak

Inst. of Computer Science, University of Wrocław (retired professor),
Joliot-Curie 15, Wroclaw 50-383,Poland
and Wrocław High School of Applied Informatics, Wrocław, Poland
Wejherowska 28, Wroclaw 54-239, Poland
aba@ii.uni.wroc.pl

***Abstract*: In data analysis and decision making we need frequently to judge whether the observed data items are normal or abnormal. This happens in banking, credit card use, diagnosing patient health state, fault detection in an engine or device like an off-shore oil platform or gearbox in an airplane motor. Sometimes the normal cases are boring and only the abnormal cases are of interest. In practice, it happens quite frequently that the normal state has a good representation, however the abnormal cases are rare and the abnormal class is ill-defined; in such a case we have to judge on the abnormality using information from the normal class only. The problem is called 'one-class classification' (OCC). The paper gives a survey of methods for performing the OCC. We show also an example: how to detect a masquerader (non-legitimate user) in a computer system – when observing a sequence of commands several thousands long.**

Keywords: Anomaly detection; One-class classification; Intrusion detection; Object classification and recognition; Schonlau's masquerade data.

I. Introduction

Search for anomalous observations (called outliers) is as old as the data analysis itself. There is no need to argue that data models deduced from data contaminated with outliers may yield very poor image of the structure of such data. Very early it has been noticed that the outlying values may have a damaging effect on the summarizing indices like the mean, the variance, the correlation coefficients, and other indices used for global description and future inference. In the early days the outliers were considered as anomalous, possibly erroneous observations, which should be identified and removed from the analysis, or - at least - given smaller weights, when building the model.

However, quite early it has been noticed that the found outliers – representing seemingly anomalous observations – might be true and that they might be an indication of a change of the population subjected to analysis, e.g. the appearance of a new species. More and more frequently it has been stated that the outliers may be of interest by themselves. Such a discussion was held during the 2003 ISI Session in Berlin [4] where two special sessions devoted to the theme 'outliers' were scheduled.

A not so rare attitude towards outliers is shown by Dan Pelleg: In his Ph.D. Thesis [23], pp. 109–111, he considers very large astronomical data where many outliers are found. Dan Pelleg muses upon the meaning of the found outliers: they may represent some classes of anomalies, both useful (true novelty) and useless (some artifacts). The human expert may flag them as 'interesting' or 'boring'. Pelleg states that the domain expert usually wants to find truly exotic rare events and not become swamped with uninteresting anomalies. Many researchers show similar attitude.

In the following I will consider the situation when the outliers constitute *de facto* the target of the analysis. This happens, e.g., in banking, fraud credit card use, diagnosing a patients' health state, fault detection in an engine or device like an off-shore oil platform or gearbox in an airplane motor, to naming a few. We want to know when the systems' behavior may be judged as normal, and when it starts to be abnormal. The essential question is then: where in the data space are the bounds permitting to distinguish between the normal and the novel (abnormal) items. The problem is called also one-class discrimination, or one-class classification, denoted also as OCC.

The different approach of the OCC method is clearly stated in [32]. We have a target class containing instance objects in normal state. Each object **z** is characterized in a multi-variate way and is given as a multivariate data vector. We imagine that all the objects constitute a multivariate data cloud in the data space. We are concerned with establishing some bounds for the normal objects, that is to mean, objects being in the 'normal' state. To find the bounds, we may consider the following two concepts:

• distance $d(\mathbf{z})$ or probability $p(\mathbf{z})$ (resemblance) of an object **z** to the target class represented by a training set $\mathbf{X}_{train}$

• threshold θ_d or θ_p put on the distance or probability (resemblance).

New objects are accepted when the distance to the target class (say, to the 'normal' class) is smaller than the threshold θ_d:

$$f(\mathbf{z}) = I(d(\mathbf{z}) < \theta_d) \tag{1}$$

or when the probability (resemblance) is larger than the threshold θ_p:

$$f(\mathbf{z}) = I(p(\mathbf{z}) > \theta_p) \tag{2}$$

where $I(.)$ is the indicator function. The one-class classification methods differ in their definition of $p(\mathbf{z})$ (or $d(\mathbf{z})$), in their optimization of $p(\mathbf{z})$ (or $d(\mathbf{z})$) and in their thresholds with respect to the training set $\mathbf{X}_{train}$.

The most important feature of one-class classifiers is the tradeoff between the fraction of the target class that is accepted, and the fraction of outliers that is rejected. The fraction of instances belonging to the target class and rejected by the classifier is called the first kind error (err1). This error can be evaluated using an independent test data sampled from the same target class.

The one-class approaches should be flexible enough to provide decision boundaries which are robust against outliers hidden in the training data. Although the one-class classification method should accept the most probable objects from the target class, it may happen that the hidden outliers are not rejected. Therefore it is desirable to have a bank of special algorithms which permit to find the delimiting boundary in presence of known outliers. For instance, when a Gaussian distribution is used as the model of the target class, the model and training procedure may be not flexible enough to reject a single outlier, and some robust estimates of the underlying Gaussian distribution are preferable. Other models, like the Parzen density, can easily incorporate some hidden outliers into their probability estimators.

The schedule of the paper is the following: The present section constitutes the introduction to the considered problem. Next section (II) is a kind of survey of the state-of-the-art in the theme OCC; I will give there a survey of the literature on the problem, including novel publications on banking and telecom frauds, also on bio-surveillance, which have appeared this year (2010). In section III a real problem: detecting illegitimate users in a computer network is considered. The problem will be illustrated with Schonlau's masqueraders data. I will show a detailed analysis of one sequence of system calls, 15000 calls long, issued by the user #24. It is known that the sequence of calls numbered 10001–15000 is contaminated with calls of an alien user being intruder to the system. I will show a simple statistical method which is able to detect the alien blocks of system calls. I will show also the OCC approach using classic Gauss, robust Gauss and Parzen density modelling of $p(\mathbf{z})$ appearing in formula (1) and point out some difficulties appearing for that data. Section IV contains short concluding remarks.

II. One-class classification: The state of the art

Outliers are as old as data analysis. A rich bibliography on outliers may be found in the book by Barnett and Lewis [1]; the book had three editions. More up-to-date surveys may be found in Hodge and Austin [16] and Bartkowiak [3]. The early trend has considered in the first place the problems: How to identify the outliers, how to assess their impact on the models, how to construct robust methods (estimators) resistant to outliers.

Outliers in the context of novelty detection or one-class classification are considered in recent surveys by Markou and Singh [18] (two separate papers constituting part 1 and 2 of the survey), also by Patcha and Park [22]. The mentioned reviews offer a rich help in bibliographical details, e.g. [18] part 1 and part 2 offer 64 and 91 references, and [22] as much as 100 references. The problem of OCC was formulated somewhere in the last decade of the XXth century, the mentioned survey papers cover the period till about 2007.

Now, subsection A will introduce shortly the developments of the OCC methodology when using statistical and neural networks (NN) approaches. Subsection B will be devoted to OCC in computer networks. Information on the most recent sources and developments is presented in subsection C.

A. Statistical and NN approaches to OCC

The topic is greatly covered in the papers by Markou and Singh [18], where both statistical and neural network (NN) methods in OCC are reviewed.

The **statistical** approach attempts in first place to model the data by some densities or probability distributions in a parametric or a non-parametric way. Next a kind of acceptable bounds for the area of plausible observations is sought. Parzen density estimators, mixture models, hidden Markov models and hypothesis testing play here a great role. Concerning non-parametric approaches, the k-NN (k nearest neighbors) method is often used; it is preferable because it does not require a smoothing parameter (necessary when using Parzen windows or rbf kernels). Also clustering approaches (Bezdek's fuzzy C-means) belong here.

The string matching approaches launched by Stephanie Forrest and her group, also by DasGupta and collaborators, are gaining more and more attention. This is a biology-inspired approach, it includes also as a particular method the negative selection process. The methods proved to be successful in many industrial applications and analysis of sequential phenomena, e.g., computer intruders.

Generally, as stated in [18] part 1, there is no single best model which might be advised. The success depends not only on the type of the method used but also on statistical properties of the data.

Neural Networks (NNs) provide another methodology for OCC. The authors [18] state in part 2 of their review that ”compared to statistical methods, some issues for novelty detection are here more critical”, by which they mean the ability of the NNs to generalize and the computational expense while training the networks. In particular, they are critical about MLPs (Multi-Layer Perceptrons), which are the most frequently used NNs. The authors state that ”neural networks fail at automatic detection of novel classes because they are discriminators rather than detectors”. The classical MLP builds open plane boundaries, such as hyper planes, to separate targets from each other and fail to decide when a feature set does not represent any known class. Yet in novelty detection is is important to build bounds for the groups, to state the novel happenings. To do this, several researchers have tried to create new feed-forward networks for new class samples while keeping the earlier trained on previously known classes (like the cascade-correlation network, growing neural gas and their variants).

A very nice **combination of NNs and probability theory** was elaborated by C.M. Bishop [7]. He was concerned with monitoring of oil flow in multi-phase pipelines containing a mixture of gas, oil and water. The problem was to determine the magnitude of the proportions (F1, F2, F3) of these constituents, using a non-invasive beam monitoring yielding output on a densitometer. Bishop builds an unconditional probability density model using a standard Parzen window approach with kernel density estimation. For a given training sample, the parameters θ of the model (including the interesting proportions F1, F2, F3) may be estimated from the *Likelihood* $\mathcal{L}$ of the sample. The value $\mathcal{L}(\hat{\theta})$ is viewed as a kind of (not-normalized) index of fit of the assumed model to the analyzed data. Bishop shows that, when taking samples with different structure (different proportions), and computing for them the likelihood with parameters obtained from the training sample, one finds markedly different values of the likelihood. The task of training a MLP may be formulated as an optimization problem for a Bayesian model that uses $p(\mathbf{x})$, the unconditional density of the input data. Bishop suggests that this unconditional density (its estimate $\hat{p}(\mathbf{x})$) can provide an appropriate quantitative measure of novelty. If the input vector falls into a region of low $p(\mathbf{x})$, then the input data may be considered as novel. A variance factor of the form $\sigma_y(\mathbf{x}) = \{p(\mathbf{x})\}^{-1/2}$ might be used to assign **x**-dependent error bars to the network outputs $y_j(\mathbf{x})$ via the model $\hat{p}(\mathbf{x})$ (j denotes the jth output neuron). Bishop shows the scheme of a RBF network performing such tasks.

Other methodologies use Support Vector Machines (SVM), Adaptive Resonance Theory (ART), Radial Basis Function (RBF) networks and a general class of auto-associators, including various variants and modifications of self-organizing maps, also some hybrid methods. It is impossible to mention all the approaches which proved to be important and successful in real life applications. Let me emphasize below a few of them:

1. Outstanding results in the domain of OCC were obtained by the Pattern Recognition (PR) Group from Delft University of Technology, The Netherlands – under the guidance of R.P.W. Duin. The Group has developed special Matlab software (PRTools [13]) for solving various difficult PR problems.

D.M.J. Tax has written his PhD Thesis entitled ”One-class classification; Concept-learning in the absence of counter-examples” (thesis supervisor: R.P.W. Duin); he has also developed (and is still maintaining) the Matlab dd_tools [33] software devoted to data domain description and outlier detection. D.M.J. Tax and R.P.W. Duin [32] sought for a minimal sphere containing almost all ’normal’ points allowing for a few slack variables (SVM approach) and obtained in such a way the one-class SVM algorithm.

A. Ypma and R.P.W. Duin [37] have elaborated learning methods for machine vibration analysis and machine health monitoring, with special emphasis on learning temporal features with the ASSOM (Adaptive Subspace Self-Organizing Map) and ICA (Independent Component Analysis).

A crucial concept in novelty detection is the concept of similarity between objects. E. Pękalska and R.P.W. Duin [24] have worked on ”Dissimilarity representations in pattern recognition: Concepts, theory and applications”, which resulted in the book under the same title [24]. E. Pękalska, M. Skurichina and R.P.W. Duin have considered combining dissimilarity-based one class classifiers [25], which is important in situations, when the non-target (outlier) class is ill-defined and under-represented and various classifiers yield much differentiated results.

2. B. Schölkopf et al. (see references in part 2 of [18]) proposed an alternative approach to build a one-class classifier. Instead of the hyper-sphere with minimal radius to fit the data (the Tax-Duin approach) they proposed to separate the surface region containing the data from the region containing no data. This is achieved ”by constructing a hyper-plane which is maximally distant from origin with all data points lying on the opposite side from the origin and such that the margin is positive”. They proposed an algorithm performing this task.

3. There are very interesting approaches for constructing test sets for deviations from normality using the biologically inspired method of *negative selection* and distinguishing between *Self* and *nonSelf* [10, 14, 11, 31, 35].

B. Anomalies in functioning of computer networks

The topic is greatly covered in the survey paper by Patcha and Park [22]. The authors are mainly concerned with intrusion detecting systems. The development of networking technology increases instantaneously the treat from spammers, attackers and criminal enterprisers. Today’s commercially available detection systems are predominantly signature-based intrusion detection systems and are designed to detect known attacks by utilizing the signatures of known attacks. The techniques aiming at anomaly detection are subdivided into several groups:

G1. Statistical anomaly detection. Earliest examples of systems for statistical anomaly detection are: the Haystack and the SPADE (Statistical Packet Anomaly Detection Engine). The SPADE system observes the activity of subjects and generates profiles to represent their behavior. An anomaly score of a packet was proposed as the degree of strangeness based on recent past activity. However, it was stated that skilled attackers can train a statistical anomaly detection to accept abnormal behavior as normal.

G2. Machine learning based anomaly detection. The authors [22] have written: ”Machine learning aims to answer many of the same questions as statistics. However, unlike statistical approaches which tend to focus on understanding the process that generated the data, machine learning techniques focus on building a system that improves its performance based on previous results. ... We seek for systems that are based on the machine learning paradigm and have the ability to change their execution strategy on the basis of newly acquired information.” One finds in this group:

System based sequence analysis (sliding windows method) as proposed by Stephanie Forrest and Steven Hofmeyr,

Bayesian networks described as graphical models that encode probabilistic relationships among variables of interest; they may encode causal relationships and be used to predict the consequences of an action. Because Bayesian networks have both causal and probabilistic relations, they can be used to combine prior knowledge with data,

Principal component analysis – has many interesting aspects of application. One of them is the ability to reduce the dimensionality of the data without loos of the essential information hidden in the data. Reduction of dimensionality is often considered in the preprocessing stage of the analyzed data. We obtain then new features which summarize somehow the observed variables and constitute the essential message of the signal while leaving out the background noise.

Markov models, in particular Markov chains and hidden Markov models. They have been employed intensively for anomaly detection, modelling normal system behavior, profiling system call sequences and shell command sequences, detecting anomalies in the usage of network protocols by inspecting packet headers.

G3. Data mining based anomaly detection. Methods used: *classification-based intrusion detection*, *inductive rule generation algorithms*, *fuzzy logic techniques*, *genetic algorithms*, and special kind of neural networks.

G4. Clustering and search for distance-based outliers. This is a largely exploited domain with many references.

G5. Association rule discovery. Example: The ADAM (Audit Data Analysis and Mining) system. It performs anomaly detection in such a way that it firstly filters out most of the normal traffic; then it uses some classification techniques to determine the exact nature of the remainder.

G6. Hybrid systems from the above mentioned. Example: the EMERALD system. The importance of this group seems to be growing: see the series of the *HAIS* conferences.

The road ahead – as viewed in [22] – looks dark and there are many open challenges. The traditional intrusion systems are unable to adapt adequately to new network paradigms like wireless and mobile networks and to meet the requirements posed by high-speed (gigabit and terabit) networks. Today's intrusion detection approaches will not be able to protect adequately tomorrow's networks against intrusions and attacks. This was the opinion expressed in 2007 [22]. The paper by Tom Rowan [26] published in 2010 confirms that opinion.

C. *Very recent approaches*

The interest in OCC is growing. The journal *Technometrics* has published this year (2010) several invited papers on novelty/fraud detection and bio-surveillance [29, 6, 15, 30], written a.o. by researchers from Bank of America, A.T.&T Labs-Research, and the Dept. of Decision and Decision Systems of University of Maryland. The papers show the extent of the problems in the context of money laundering (credit cards, debit cards, online banking or checks, especially in the retail banking sector), fraud in telecommunications (subscription fraud, intrusion fraud, fraud based on loopholes in technology, on new technology, on new regulation) and contemporary realistic treatises of terroristic bio-attacks.

Another important stream of research is concerned with detecting novelty in mechanical devices, with novelty meaning abnormality, that is to say, bad functioning due to some damages. How to describe the domain for the normal state of a mechanical device? Let us consider a gearbox. Such a gearbox, incorporated into a mechanical machine, emits vibration signals which contain information on normal (or abnormal) functioning of the gearbox. The essential question arises: is the given working gearbox still in good state (described as 'normal') or does it starts to be worn out and have some damages which are not seen directly? Is it possible to make a diagnosis of the state of the machine on the base of vibration sounds it is emitting? The problem has appeared frequently in specialistic journals like *Journal of Sound and Vibration* and *Mechanical Systems and Signals Processing*, where interesting elaborations on the topic may be found. OCC with its decision boundaries may work only with one-class data, however a reasonable number of samples from the normal state is needed. The decision boundary may be also constructed when using not only the normal data, but also counter-examples, that is some data from the abnormal class. Generally, it may be not possible to get an adequately representative sample from the abnormal class; however it is always possible to generate some counter-examples at random in the outer space of the normal data.

A recent paper [31] elaborated an algorithm inspired by the human immune system whose task is to differentiate between the antigens and the body itself and protect the body from invasion of external microorganisms like bacteria or viruses. The process is described as *negative selection.* The authors [31] have elaborated a similar procedure (based on the immune system metaphor) to study the behavior of an offshore structure with a steel platform and to model damages in an aircraft wing. Both devices were working in changing environmental conditions. A novelty index was proposed to measure the deviations in time as a result of changes of some conditions.

Nowadays it is stated that the difficulty and specificity of problems lies also in the growing size of observed data streams, their heteroscedasticity and mixed-attributes content. Some simple, yet promising methods have been elaborated in [21, 36]. We feel also the need of constructing autonomous robots working in an unknown environment and being able to adapt to that environment, as considered in [20].

III. Real example: Search for abnormal patterns in system calls

This section shows an example of detecting masqueraders in system calls. The data were specially prepared by Schonlau and are publicly available at www.schonlau.net [27, 28]; at the same URL one may find also references of papers whose authors were trying to identify the masqueraders (illegitimate users) in the provided data.

The data set contains 50 users; for each of them a sequence of 15000 system calls was recorded. The entire sequence of 15000 calls for each user was subdivided into 50+100=150 blocks, each block containing 100 calls. Thus we have two periods to analyze. They constitute part A and part B of the data. The first 50 blocks (part A) are intact, however some of next 100 blocks (part B) were exchanged by blocks from 20 extra users (aliens, playing the role of masqueraders). The information on the block exchange is given separately, so one can check whether one's alien block identification was right or not. The task is to recognize the alien blocks in the last 100 blocks of each of the 50 legitimate users.

Many sophisticated methods have been applied to identify the masqueraded blocks, most of them rather unsuccessful (see [27]).

In this paper I will analyze data for one user only – out of 50 users whose data were possibly contaminated with some alien blocks. I will show a more detailed analysis for the **user no. 24** (#24). This user has alien blocks located at

the position 119–139 (in the original numeration of blocks counted as 1, 2, . . . , 150). Generally, the information on the location of the alien blocks is not given a priori – I will use it for illustration only, to show how the OCC method works.
The following tasks will be considered:

1. A general statistical characterization of the data available for user #24, including some characteristics of the incidence matrix (blocks × frequency of commands).
2. Reducing the dimensionality of the analyzed data by expressing it by new uncorrelated features called principal components (PCs).
3. Analysis of part A of the data: How many PCs are sufficient to reproduce the blocks from part A of the data? Say, this happens for K PCs. The number K is called *intrinsic dimensionality* (dim) of the data.
4. Say, we found that the intrinsic dimensionality is K. Are K PCs sufficient to model the data of user #24 in a satisfactory way? Are these PCs also sufficient to model for that user some data recorded in a later period?
5. Building decision boundaries (DBs) for the data of user #24 using pairs of selected PCs. The DB will be constructed with the aim of making the first kind error (false positive) of order α, where α is a small number contained in the interval [0, 1], for example, $\alpha = 0.05$. Because pairs of two PCs constitute 2-dimensional data, it is possible to illustrate the constructed DBs in a plane, together with the training data. Moreover, it is possible to project to the same plane some other (say, test) data, and see the efficacy of the constructed DB; it is also possible to calculate the classification errors.
6. Constructing DBs and evaluation of classification errors for data based on ensembles composed from $K > 2$ PCs.

A. Statistical characterization of system calls by user #24

We know already that there were altogether 15000 system calls subdivided into 150 blocks, with 100 commands each. The 150 blocks were further subdivided into part A with 50 blocks and part B with 100 blocks. Next, part B was further subdivided into sub-parts B$_{nc}$ (non-contaminated) and B$_{cm}$ (contaminated) containing commands of the legitimate user and of the masquerader respectively.
We found how many unique commands were used in the first and second part of the data; also: how many unique commands were used in the B$_{nc}$ and B$_{cm}$ subgroups.
It was stated that

- All 15000 calls have used u=119 different commands.
- First 5000 calls used u1=90 unique commands.
- Next 10000 calls used u2=111 unique commands.
- Alien blocks contained in set B$_{cm}$ used u3=31 unique commands. There were 21 alien blocks composing this set.

The alien 21 blocks contained diffrrr=15 new commands, which have not appeared in the remaining blocks attributed to the legitimate users (that is: they did not appear neither in part A nor in the subset B$_{nc}$). The new alien commands are shown in Table 1.

Table 1: Specific commands in alien blocks

’emacs-20’	’faces’	’head’	’ksh’	’movemail’
’netscape’	’netstat’	’popper’	’rsh’	’showps’
’tail’	’top’	’uniq’	’wc’	’which’

From the sequence of the 150 blocks gathered for user #24 the incidence matrix $\mathbf{X}_{all}$ of size 150×119 was calculated. It was obtained by calculating the frequencies of the 119 commands in each block. After doing that, each element of the incidence matrix was divided by 100. The data matrix $\mathbf{X}_{all}$ obtained in such a way yielded the basis for further considerations. It was split into two parts: the matrix ’dat50’ obtained from the first 50 blocks of the data and ’dat100’ obtained from the remaining part of the data (100 blocks):

$$\mathbf{X}_{all} = \underbrace{\text{dat 50}}_{Part\ A} + \underbrace{\text{dat 100}}_{Part\ B}$$

It is sure that the data matrix ’dat50’ was composed from commands issued only by user #24. This set ’dat50’ was used for training purposes to establish decision boundaries for the ’normal’ commands issued by user #24. The data matrix ’dat100’ contains, in principle, also commands issued by user #24, however it is contaminated: some blocks are for sure alien, which means that they were issued by an alien unknown user. The task is to recognize, which ones are own (those of user #24) and which ones are the aliens.
In explorative data analysis it is advised to work with standardized or at least with with centered data. We decided to work only with centered data. Thus the matrix ’dat50’ was centered to 0, using the vector ’mu50’ of means calculated from that matrix. This yielded the data matrix ’dat50C’ (with the property that each of its column has mean equal to 0).
To center the data matrix dat100 we have used the ’mu50’ vector of means derived from the set dat50. The centered matrix was denoted as ’dat100c’, the lower case ’c’ indicating that the centering for that matrix was carried out using not own centers.

In the following we will find some intrinsic characteristics of the data contained in the data matrix dat50C. Next we will look whether the discovered intrinsic characteristics are preserved for all data vectors contained in the ’dat100c’ data matrix. We hope to find in dat100c some data vectors which are really outstanding – that is, are outliers, when compared to the main ’dat100c’ bulk of data.

B. Reduction of dimensionality of the data

We consider the following question: Is it possible to reduce the dimensionality of the data, that is, to construct of fewer number of new variables (features) representing the information (energy, inertia) contained in the data? The problem may be solved by computing eigenvalues of the covariance matrix calculated from the analyzed data matrix [17].
The respective eigenvalues (e-vals) were calculated from the training set dat50C. Looking at the values of subsequent e-vals (not shown here) one states that they are decaying fast to zero. It is known that the subsequent e-vals are equal to the variance of the principal components (PCs) obtained from the eigenvectors connected with the respective e-vals([17]). The

sum of the e-vals is equal to the sum of variances (called total inertia) of all original variables from the data matrix X_{all}. The cumulative sums of the eigenvalues, normalized so that they show how large fraction of total inertia is explained by subsequent k e-vals ($k = 1, \ldots, 50$), are shown in Figure 1.

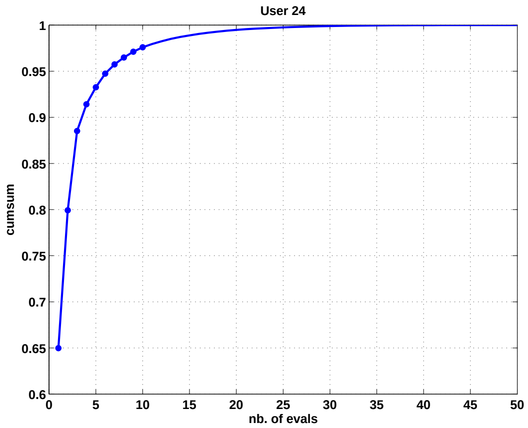


Figure. 1: Normalized cumulative sums of eigenvalues obtained from the centered incidence matrix 'dat50C' calculated from blocks 1–50 for user #24

One may notice in Figure 1 that the first three e-vals have markedly large contributions to the total inertia, the next ones contribute less and less. With $k = 20$ e-vals a fraction 0.995 of Total inertia is explained and indeed, there is no need to consider more features. Exact values of the (normalized) cumulative sums of the first k=20 e-vals are given below (to be read row-wise):

0.650	0.799	0.885	0.914	0.933
0.947	0.957	0.965	0.971	0.976
0.979	0.982	0.985	0.987	0.989
0.990	0.992	0.993	0.994	0.995

Each subsequent e-value permits to construct one new feature called principal component (PC). The constructed features are mutual orthogonal and span the principal component space (see, for example, [17] for exact formulations and explanations).

C. *Analysis of parts A and B of the data: How many PCs are necessary? The reconstruction method*

By inspecting Figure 1, one obtains an idea, how many PCs are essential for modelling the data for the considered user. It is seen that already k=3 PCs explain most of the inertia (exactly: an 0.885 part) of the data. This might provide explanations for the rough interrelation among the system calls occurring during the first 5000 calls of that user. However, probably not all specificities are accounted for. These specificities may occur more often during the next 10000 calls constituting part B of the data.

The number of retained e-vals will be denoted as ILE. They designate the number of essential (intrinsic) dimensions of the data. We decided to investigate the effect when retaining 5, 10, and 16 e-vals:

ILE = 5, 10, 16.

The effectiveness of the representation will be expressed by the correlation coefficients between the original data vectors $\mathbf{x}$ and the reconstructed data vectors $\hat{\mathbf{x}}^{ILE}$ when retaining ILE e-vals and constructing for them ILE PCs. We call such a correlation coefficient *r-stacked*. It acts as a similarity index. Exact formulae how to reconstruct the data vectors from a fewer number of principal components may be found, for example, in [17, 2], they are also given at the end of this subsection as eq. (3) and (4).

Figure 2 shows the correlation coefficients obtained for the 50 data vectors contained in part A of the data. The figure contains 3 panels, corresponding to ILE=5 (top exhibit), ILE=10 (middle exhibit) and ILE=16 (bottom exhibit). One may notice that with increase of the parameter ILE the similarity index *r-stacked* is also increasing. One does not notice specific outliers; the increase happens more or less uniformly for all the considered 50 data vectors. Taking ILE=10 and next ILE=16, one obtains all values of *r-stacked* greater then 0.90 and 0.95, which means a great similarity.

An analogous figure constructed from data set B using eigenvectors from set A is shown in Figure 3. We show there the behavior of the similarity index *r-stacked* applied to the data set dat100c. Let's say clearly: The reconstruction shown in that figure was done using eigenvectors derived from set dat50C. The results of the reconstruction are shown in the form of three panels. They illustrate the reconstruction obtained from ILE=5 (top), ile=10 (bottom) and ILE=16 eigenvectors.

The reconstructed items based on data vectors belonging to the legitimate user are denoted as black open circles. Comparing these with the exhibits in previous figure, one may notice that generally the reproduction is worse, although it ameliorates with increase of ILE, that is, with the number of eigenvalues used for reconstruction.

The reconstructed items based on data vectors belonging to the masquerader are shown as big red squares. One may see that all of them are clearly outstanding, the discrepancy between them and the bulk of the black open circles is quite large. This means that the covariance structure in the data set composed from data items of the legitimate user is quite different from the covariance structure of data items produced by the masquerader. Notice also that the location of the big red squares is barely influenced when using 5, 10, or 16 eigenvectors for reconstruction.

The reconstruction was performed using the following formulae. Let $\mathbf{X}$ denote the centered data matrix, and let $\mathbf{A}^{(K)}$ be the matrix containing column-wise the first K eigenvectors (obtained from the covariance matrix of $\mathbf{X}$):

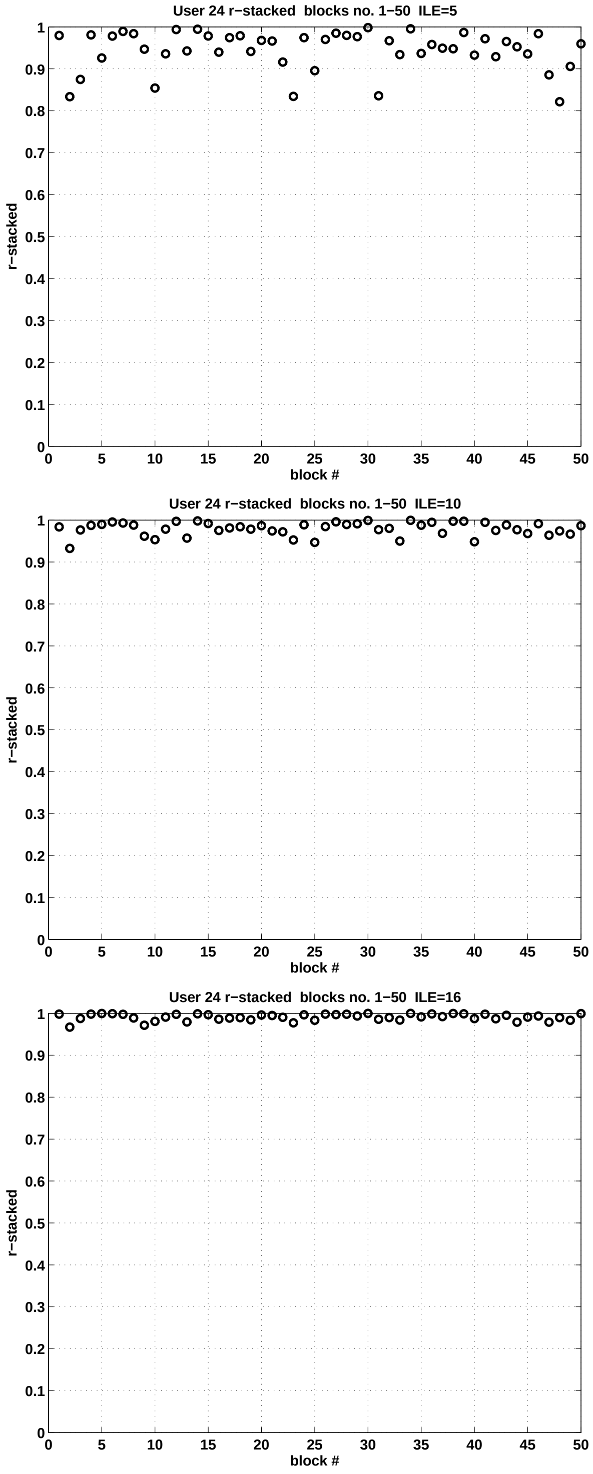
$$\mathbf{A}^{(K)} = (\mathbf{a}_1, \ldots, \mathbf{a}_K).$$

Then the new features, called PCs, are obtained as linear combinations of the original features appearing in subsequent columns of $\mathbf{X}$:

$$\mathbf{Y}^{(K)} = \mathbf{X}\mathbf{A}^{(K)}. \tag{3}$$

The matrix $\hat{\mathbf{X}}$, reconstructed from K principal components, is computed by the reverse formula

$$\hat{\mathbf{X}}^{(K)} = \mathbf{Y}^{(K)}(\mathbf{A}^{(K)})^T. \tag{4}$$



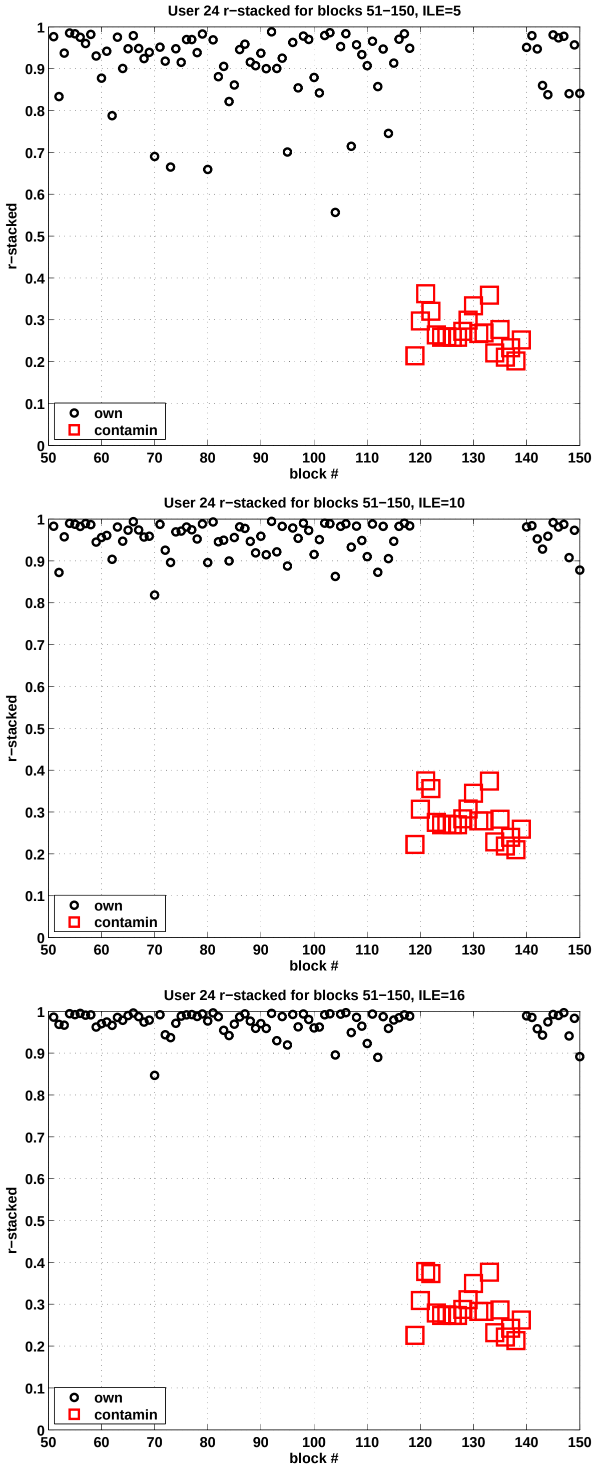


Figure. 2: User #24, data part A. Similarity between the original data vectors and their reconstructions from ILE=5 (top), ILE=10 (middle), and ILE=16 (bottom) exhibit. The black open circles indicate data vectors belonging to the legitimate user #24

Figure. 3: As Figure 2, however constructed from data part B, using eigenvectors obtained from part A of the data. Notice that generally the reproducibility of the items of the legitimate user is poorer as in Figure 2. The big red squares indicate masqueraded blocks

D. 2D data. Building decision boundaries using the OCC approach

We will now illustrate the OCC approach in which the concept of density modelling of the analyzed data is used. To illustrate graphically the approach, we will use pairs of principal components (PCs) constructed from the training set dat50.

The very first PCs obtained from set dat50 (the training set) summarize some general characteristics of the data vectors contained in that set. These characteristics should be valid also for other data vectors obtained for that user and included into its test set (called 'dat100'). However, the test set contains also a small lot of some alien data vectors belonging to an unknown masquerader. We suppose that the alien data vectors have a specific interdependency structure which will manifest itself only in higher order PCs, which are destined to illustrate some specific relations occurring in small subsets of data.

The analyzed data (for user #24) had 21 alien blocks which were implanted – in unknown positions – into the set dat100 after removal of the original blocks. We found some indications that the alien blocks might manifest themselves in the PCs no. 9, 10, 11, 14, 15.

Below we show illustrations of constructed DBs when taking as data the following pairs of principal components: (PC1, PC2), (PC9, PC10) and (PC13, PC14). They confirm our supposition that lower order PCs express some general relations between the features, while higher order PCs may express more specific characteristics owned by few data vectors.

The constructed DBs will be based on

a) classic Gaussian distributions, with parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ estimated from the training data,

b) robust Gaussian distributions, with parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ estimated iteratively in a robust way by an algorithm which is down-weighting in subsequent steps the data vectors with big distance from the center $\boldsymbol{\mu}$,

c) Parzen-window classifier, where the respective density distribution is estimated as the sum of Gaussian kernels [12, 33]:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \exp(-(\mathbf{x} - \mathbf{x}_i)^T h^{-2} (\mathbf{x} - \mathbf{x}_i))\,.$$

The free parameter h is optimized by maximizing the likelihood on the training data using leave-one-out.

From the mentioned three classifiers the Parzen classifier is the most flexible one, however it needs a reasonable training sample, especially in the low density area.

For a given distribution we estimate firstly its empirical probability density. Next we find areas of low density. This done, we are able to depict a contour delimiting areas of low density from those of high density. We use here additional condition that the first kind error - when discarding observations falling into the low density area - should be equal to an assumed significance level α, for example $\alpha = 0.05$. The obtained delimiting contour is called *decision boundary* with significance α.

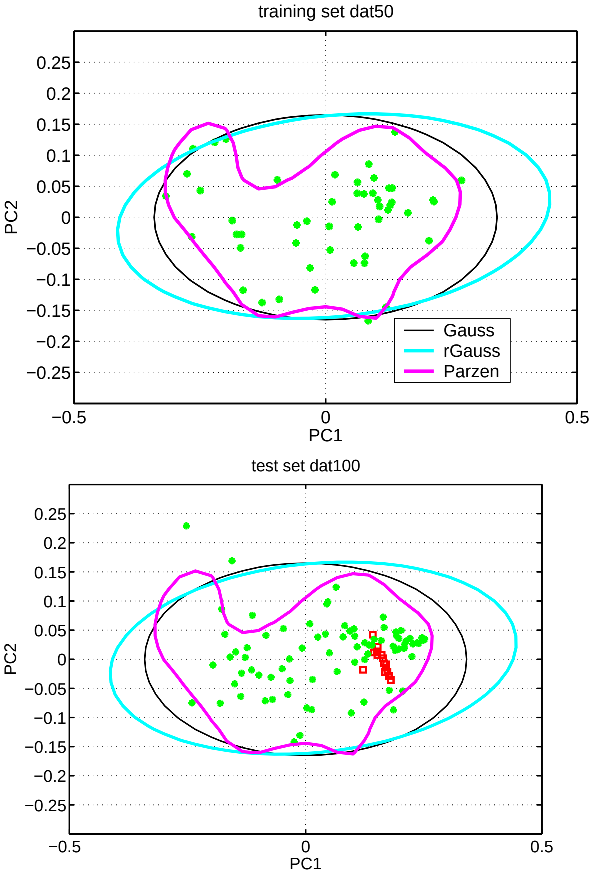


Figure. 4: Training data dat50 and test data dat100 (green stars) depicted in the coordinate system <PC1, PC2>, with decision boundaries established from dat50. Red squares denote alien blocks

We will investigate the efficacy of the constructed decision boundary for each pair of the analyzed PCs separately. The results are shown in Figures 4, 5, 6 – for the pairs <PC1, PC2>, <PC9, PC10>, <PC13, PC14> respectively.

Analysis of the pair <PC1, PC2>

This is the pair of principal components which extracts most of the common overall structure of the data. The first PC provides a new feature which looks often like an average of all the analyzed variables. As shown in Figure 1, the first two PCs explain (and are able to reproduce) 79.9% of total inertia of the data.

The data vectors belonging to the legitimate user have similar distribution both in the 'dat50' and the 'dat100' data set. This sounds optimistic. However the masqueraded blocks fall into the same area. Thus the conclusion: the first two PCs cannot identify the masqueraded blocks.

Analysis of pairs <PC9, PC10> and <PC13, PC14>

These pairs of PCs are responsible only for an 0.08 and an 0.05 part of total inertia. This means that they explain only some specificities of the data. The masqueraded blocks have moved to the border of the data cluster containing data vectors coming from the legitimate user; however they are not

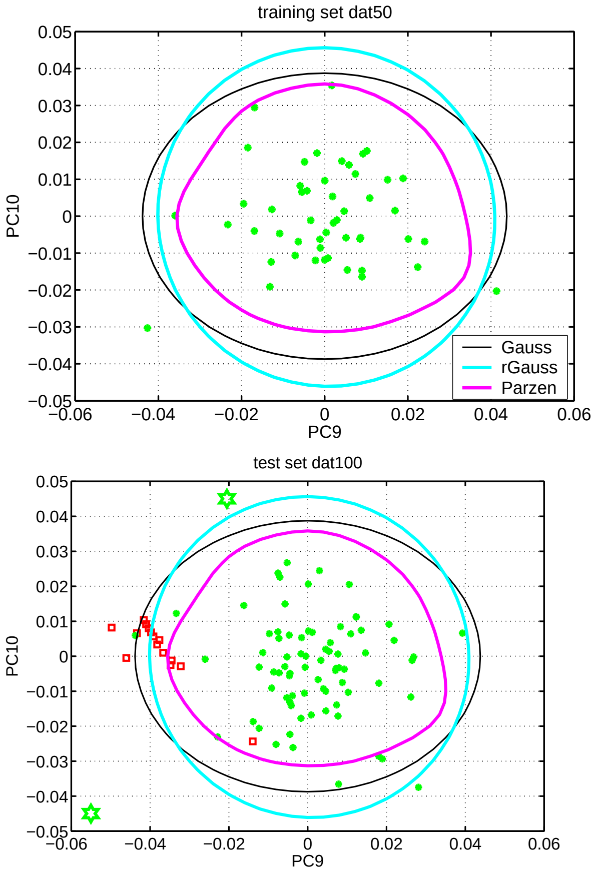


Figure. 5: The same representation as in Figure 4, however in the coordinate system <PC9, PC10>. There are two big outliers marked as stars

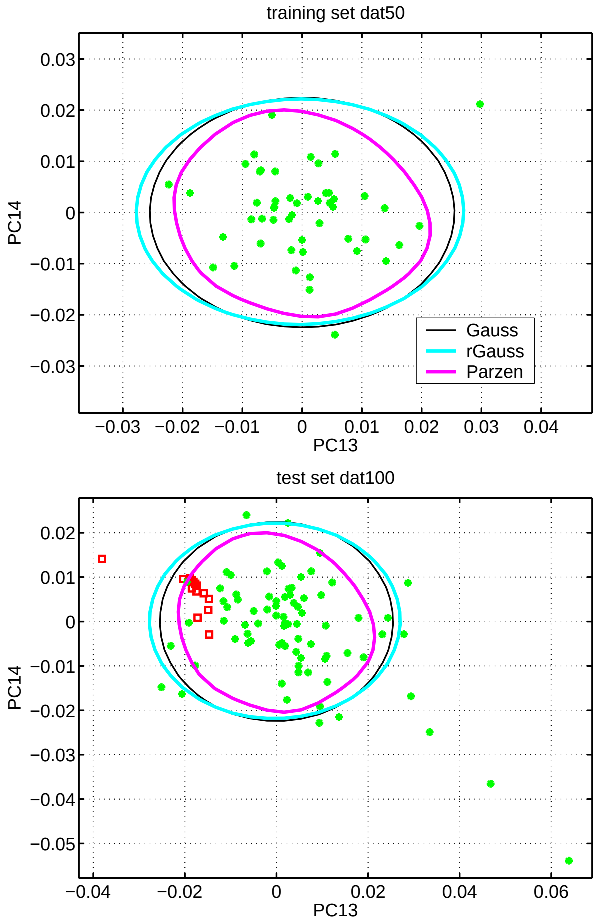


Figure. 6: The same representation as in Figure 4, however in the coordinate system <PC13, PC14>

outstanding and we are not allowed to point them out as alien data vectors.

On the opposite, some data vectors coming from the legitimate user, appear located outside the decision boundary (two big outliers) and may be pointed out as aliens.

Thus the general conclusion from this part of analysis is: working only with pairs of PCs we are not able to recognize the alien blocks coming from a masquerader.

E. Investigating first kind errors when working with ensembles of PCs

It was found in previous subsection that pairs of PCs are not able to discriminate the contaminated blocks. May this be possible when working not with pairs but with ensembles of PCs, composed from first K PCs? We have investigated the problem for K=2, 3, 4, 5, 10, 16 and 20. The value K denotes also dimension (dim) of the data. The DBs were constructed under the assumption that the first kind error α is set equal to 0.05. The methods for construction of the DBs were the same as in previous subsection.

The results of the investigation are shown in Table 2.

For each K (denoted in Table 2 as ’dim’) the following actions were performed:

1. The 3 classifiers based on the 3 chosen methods were trained using the first K PCs obtained from the training data set dat50. This resulted in three functions designating the respective DBs. Only a fraction $\alpha = 0.05$ of all the legitimate data was allowed to be outside the decision boundary. For some discretization reasons the nominal level 0.05 could not be kept exactly by the constructed DBs: both Gaussians allowed a fraction 0.04 to be outside the DB; the Parzen method did that for a fraction equal to 0.06. This happened for all sizes of the investigated ensembles.

2. The performance of the obtained DBs was tested on the set called ’dat100nc’ (’nc’ stands for *non contaminated*). It is expected here that the proportions of data points outside the respective DBs will be similar (may be a little worse) than the respective proportions shown for dat50. Table 2 shows that – generally – the DB from robust Gauss performs well for the dat100nc data. Ordinary Gauss methods starts to be worse for higher dimensions, especially for dim=16 and dim=20. The Parzen method performs badly: it yields a high percentage of improperly recognized target data.

3. Performing the test on the dat100cm data, containing only contaminated blocks, one may notice that both Gaussian methods fail for dimensions lower than 20: they do not recognize these data as alien. On the opposite, the Parzen method (except dim=2) works well for these data and recognizes them properly.

Let us say that the decision boundaries were obtained from

Table 2: Proportions of data sets located outside the decision boundary obtained when using three densities: ordinary Gauss (oG), robust Gauss (rG) and Parzen Kernels (Pz). 'dim' means dimensionality of the data

	dim=2	dim=5	dim=10	dim=16	dim=20
dat50					
oG	0.0400	0.0400	0.0400	0.0400	0.0400
rG	0.0400	0.0400	0.0400	0.0400	0.0400
Pz	0.0600	0.0600	0.0600	0.0600	0.0600
dat100nc					
oG	0.0253	0.0127	0.0633	0.1013	0.1519
rG	0.0253	0	0.0253	0.0253	0.0380
Pz	0.0633	0.2405	0.4430	0.5316	0.5443
dat100cm					
oG	0	0	0	0.0476	1.0000
rG	0	0	0	0	0.6190
Pz	0	0.9524	1	1.0000	1.0000

training data given by an incidence matrix of size 50×119, thus the number of observed variables is more then twice as large as the number of rows constituting the data matrix, which means enormous over-fitting. Therefore it was absolutely necessary to preprocess the data and reduce them to a much smaller number of new features, called PCs. The new features were obtained by a spectral decomposition of the data matrix; moreover the features were extracted sequentially. With growing number of features the instability of the classifiers is expected to rise.

The essence of the performed OCC method is to find areas of low probability. However, to find these, we need large samples of data, which can not be done for the analyzed example. Despite this, we got results which are reasonable. The applied Gaussians need estimates of the their parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, thus for larger dimensions (> 10) there are no free parameters left, and we should not expect any generalization abilities of the derived boundaries. To our surprise, we got coherent results. The problem may be elaborated in another way [34], but this is beyond the scope of this paper.

IV. Concluding remarks

The specificity of finding and identifying novel unexpected abnormal phenomena was reviewed. We have explained, what is specific in the one-class-classification named OCC. In the last decade it became obvious that one-class classifiers are needed in many important domains of our life environment. We need monitoring systems using non-invasive measurements able to signalize that something abnormal starts to happen. Abnormal events to be prevented include: an unexpected eruption of a volcano, a leakage of an off-shore platform in the Mexican Gulf, an outbreak of an atypical flue decimating the population, a plane crash. There is also the need of designing autonomous robots working in an vision-based environment, being able to detect novelty and concentrate to explore further this novelty.

Which is the preferable method for designing the one-class classifier and finding the unusual observations? It depends on the data. There is no single omnibus method for all data.

As an example, we have analyzed the data for user #24 from Schonlau's data, where the main task is to recognize the blocks implanted by an intruder. The reconstruction method proved to work fine: it has recognized all the implanted blocks as not belonging to the legitimate user. An analogous analysis for a different user (# 26, not shown here) did not work so good: the reconstruction method has discovered only half of the implanted blocks. This is quite plausible: the intruder must not for the entire time behave differently: partially he may use similar commands as the legitimate user.

The OCC approach needs an assumption on the data distribution or density. Three models were considered: Ordinary Gauss, Robust Gauss, SVM. It appeared that our data (50 or 100 blocks of data vectors with 119 variables), even after a transformation to 50 PCs, are not suitable to obtain reliable results.

We did not consider the problem, how accurate are the decision boundaries constructed from relative small samples compared to relative large number of features. Some indication how to find instability of classifiers may be found in a paper by Tax and Duin [34].

References

[1] V. Barnett and T. Lewis. Outliers in Statistical Data. 3rd ed. Wiley, Chichester, 1994.

[2] A. Bartkowiak. Outliers in some Faces and nonFaces data. Int. J. of Biometrics, Vol. 2, No. 1, pp. 2–18, 2010.

[3] A. Bartkowiak. Outliers in biometrical data, what's old, what's new. Int. J. of Biometrics, Vol. 2, No. 3, pp. 203–221, 2010.

[4] A. Bartkowiak and J. Zdziarek. Outliers: a continuing problem. Bulletin of the International Statistical Institute, 54th Session, Proceedings. Berlin August 2003. http://isi.cbs.nl/iamamember/CD3/index.html

[5] A. Bartkowiak and A. Szustalewicz. Outliers – finding and classifying which genuine and which spurious. Computational Statistics, vol. 15, no 1, pp. 3–12, 2000.

[6] R.A. Becker, C. Volinsky, and A.R. Wilks. Fraud detection in telecommunications: History and lessons learnt. Technometrics, February 2010, vol. 52, No. 1, pp. 20–33.

[7] C.M. Bishop. Novelty detection and neural network validation. Proceedings of IEE Conference on Vision and Image Signal Processing, pp. 217–222, 1994.

[8] H.S. Burkom, S.P. Murphy and G. Shmueli. Automated time series forcasting for biosurveillance. Statistics in Medicine, vol. 26, pp. 4202–4218, 2007.

[9] U.G. Daphne *et al.*. Algorithm for statistical detection of peaks – syndromic surveillance system for the Athens 2004 Olympic Games. MMWR, September 24, pp. 36–94, 2004.

[10] D. DasGupta, S. Yu, and N.S. Majumdar. MILA - Multilevel immune learning algorithm, Gecco 2003, LNCS 2723, pp. 183–194, 2003.

[11] D. Dasgupta and S. Forrest. Artificial immune systems in industrial applications. Proc. IPMM, IEEE Press, pp. 257–267, 1999.

[12] R.O. Duda, P.E. Hart and D.G. Stork. Pattern Classification, 2nd Edition. Wiley, New York, 2001.

[13] R.P.W. Duin *et al.* PRTools4, A Matlab Toolbox for Pattern Recognition. Delft Pattern Recognition Research, Faculty EWI - ICT, Delft University of Technology, Delft The Netherlands, Version 4.1, August 2007. http://www.prtools.org

[14] S. Forrest, A.S. Perelson, L. Allen, and R. Cherukuri. Self - nonself discrimination in a computer. Proc. of the 1994 IEEE Symposium on Research in Security and Privacy, IEEE Press, Los Alamitos, CA, pp. 202–212, 1994.

[15] D.J. Hand. Fraud detection in telecommunications and banking: Discussion of Becker, Volinsky, and Wilks (2010) and Sudjianto et al. (2010). Technometrics, Vol. 52, No. 1, pp. 34–38, 2010.

[16] V.J. Hodge and J. Austin. A Survey of Outlier Detection Metodologies Artificial Intelligence Review, Kluwer Academic Publishers, Vol. 22, 85–124, 2004.

[17] I.T. Jolliffe. Principal Component Analysis, 2nd ed., Springer, New York, 2002.

[18] M. Markou and S. Singh. Novelty detection: a review. Part 1: statistical approaches. Signal Processing, Vol. 83, pp. 2481–2497, 2003.

[19] M. Markou and S. Singh. Novelty detection: a review. Part 2: Neural network based approaches. Signal Processing, Vol. 83, pp. 2499–2521, 2003.

[20] S. Marsland, U. Nehmzow, and J. Shapiro. On-line novelty detection for autonomous mobile robots. Robotics and Autonomous Systems, Vol. 51, pp. 191–206, 2005.

[21] M.E. Otey, A. Ghoting, and S. Parthasarathy. Fast distributed outlier detection in mixed-attribute data sets. Data Mining and Knowledge Discovery, Vol. 12, pp. 203–228, 2006.

[22] A. Patcha and J-M Park. An overview of anomaly detection techniques: Existing solutions and latest technology trends. Computer Networks, Vol. 51, pp. 3448–3470, 2007.

[23] D. Pelleg. Scalable and Practical Probability Density Estimators for Scientific Anomaly Detection. PhD Thesis. CMU Pittsbourg, 2004.

[24] E. Pekalska and R.P.W. Duin. The Dissimilarity Representation for Pattern Recognition, Foundations and Applications. World Scientific, Singapore, 2005.

[25] E. Pekalska, M. Skurichina and R.P.W. Duin. Combining dissimilarity-based one-class classifier. MCS 2004, LNCS 3077, Springer Berlin Heidelberg, pp. 122–133, 2004.

[26] T. Rowan. Negotiating WIFI security Network Security, February 2010, pp. 8-12.

[27] M. Schonlau. Masquerading used data, web page at: http://www.schonlau.net

[28] M. Schonlau *et al.* Computer intrusion: Detecting Masquerades. Statistical Science, Vol. 16, pp. 1–17, 2001.

[29] A. Sudjianto *et al.*. Statistical methods for fighting financial crimes. Technometrics, Vol. 52, No. 1, pp. 5–19, 2010.

[30] G. Shmueli and H. Burkom. Statistical challenges facing early outbreak detection in biosurveillance. Technometrics, Vol. 52, No. 1, pp. 39–51, 2010.

[31] C. Surace and K. Worden. Novelty detection in a changing environment: A negative selection approach. Mechanical Systems and Signal Processing, Vol. 24, pp. 1114–1128, 2010.

[32] D.M.J. Tax. One-class classification. PhD dissertation, Delft University of Technology, Delft, The Netherlands, Juni 2001, available at: http://www-ict.ewi.tudelft.nl/papers.html

[33] D.M.J. Tax. Data description toolbox dd_tools 1.6.3. A Matlab toolbox for data description, outlier and novelty detection, EWI, Delft UT, Sept 2008.

[34] D.M.J. Tax and R.P.W. Duin. Outlier detection using classifier instability. In: A. Amin, D. Dori, P. Pudil and H. Freeman (Eds), Advances in Pattern Recognition, Proc. Joint IAPR Int. Workshops SSPR'98 and SPR'98, Sydney, Australia. LNCS 1451, Springer, Berlin, 593–601, 1998.

[35] S.T. Wierzchon. Artificial Immunological Systems (in Polish). Akademicka Oficyna Wydawnicza Exit, Warsaw, 2001.

[36] M. Ye, X. Li, and M.E. Orlowska. Projected outlier detection in high-dimensional mixed attributes data set. Expert Systems with Applications Vol. 36, pp. 7104–7113, 2009.

[37] A. Ypma. Learning methods for machine vibration analysis and health monitoring. PhD dissertation, Faculty EWI, Delft University of Technology, 2001, http://www-ict.ewi.tudelft.nl/papers.html

Author Biography

Anna M. Bartkowiak received her MSc and PhD degrees in Applied Mathematics from the University of Wrocław, Poland, and her DSc degree (habilitation) from the Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland. She is presently Professor Emeritus in the Institute of Computer Science, University of Wrocław, and a lecturer in the Wrocław High School of Applied Informatics. She is a Fellow of the Royal Statistical Society, London, also a member of IBS, IASC ISI, and ENNS. Her scientific interests are: algorithms of computational statistics and multivariate analysis, in particular: graphical visualization of observational data, pattern recognition and neural networks.