Outliers analysis and one class classification approach for planetary gearbox diagnosis

Anna Bartkowiak\(^1\) and Radoslaw Zimroz\(^2\)

\(^1\) University of Wroclaw, Inst. of Computer Science, Joliot-Curie 15, Wroclaw 50-383, PL
\(^2\) Wroclaw University of Technology, Diagnostics and Vibro-Acoustic Laboratory, Poland

E-mail: aba@ii.uni.wroc.pl, Radoslaw.Zimroz@pwr.wroc.pl

Abstract.

An application of outlier analysis for diagnosis of gearboxes working under non-stationary operating conditions is considered. The analysis is performed using quite large data characterizing (by 15 power spectrum amplitudes) two gearboxes, one in bad and the other in good condition. Analysis of Mahalanobis Distances in the two data sets shows that both of them are highly heterogeneous and non-Gaussian. A mapping of the data from the set 'good' to two dimensions by the non-linear Neuroscale method permits to visualize the data in a plane. Using the derived mapping, some decision boundaries (DBs) – permitting to identify a given fraction \(\alpha = 0.05\) of low density outliers – are constructed for the 'good' set. This is done using three methods: 1. Parzen kernel density, 2. mixture of Gaussians and 3. Support Vector Data Description (SVDD). Each of the obtained DBs was tested using the bad data. For all 3 methods more than 98% of the bad data points were found outside the DBs constructed for the 'good' set.

1. Introduction

Finding abnormal items in data is an important topic. The outliers, if not identified properly, may largely influence the results of the performed analysis. In particular, outliers hidden in the data may have a great impact on the process of monitoring mechanical devices [12, 11, 4, 3]. When monitoring the condition of mechanical devices, usually the normal and abnormal states of the device are of interest. The abnormal state is termed also 'novelty'. The topic 'outliers' or 'novelty' was already considered in some studies [12, 11, 4, 3, 2], however this was done usually with the aim of finding abnormal functioning of the device without paying much attention to the distributional properties of the data serving for the diagnosis. Also, the analyzed data matrices were rather small, often based on simulations using a few real points taken as starting points. On the opposite, we got quite large data: about 1000 multivariate data vectors with \(d=15\) attributes each, for both the good and bad condition. In this connection, we saw a real opportunity to examine more closely the probability distribution of the recorded diagnostic data and find out what kind of outliers are there.

In the following, we consider vibration signals from two gearboxes, one in good (GOOD data), the other in bad condition (BAD data). The gears were mounted on two bucket wheel excavators. Detailed description of technical details and data acquisition is given in a companion paper [14], see also [13]. Paper [14] contains a separate section on Experiment description and Feature extraction. One may find there a detailed description of the experiment recording the
vibration signals by use of the Bruer&Kjaer Pulse system. In such a way, from vibration signals emitted by two gearboxes – after preprocessing – n = 951 segments for the GOOD and n = 1232 segments for the BAD machine were obtained. Each of the segments was subjected to the Discrete Fourier Transformation (function PSD from Matlab), which yielded power spectrum components. From these, 15 power spectrum components, (variables called in the following pp1, ... , pp15) were selected – for each segment. Records of these variables formed data matrices named GOOD and BAD data sets. The data were already analyzed in [1] with the aim of building an algorithm permitting to predict for a new data vector its set membership (GOOD or BAD).

The following questions will be considered:

(i) What is the distribution of the obtained multivariate data: is it multi-variate normal (Gaussian) or abnormal (non-Gaussian)?
(ii) Are there any serious outliers in the obtained data? If yes, then how to found them and identify?
(iii) Are the ‘good’ and ‘bad’ data separable?
(iv) Is it possible to build around the ‘good’ data a decision boundary with the property that beyond the constructed boundary the found data points may be judged as outliers?

To answer these questions, our approach will be the following:
Ad (i) and (ii) To state whether the distributions are normal (Gaussian), so called Mahalanobis Distances (DDs) from the centers of the respective data sets will calculated. The obtained DDs will be analyzed by considering index-plots and quantile-quantile (q-q) plots. Inspecting these plots (in particular the confidence bounds of the DDs and shapes of the q-q plots) will indicate whether the considered data are (or: are not) Gaussian. Let us emphasize that traditional inference methods are based on normality assumption; if this is not true, one should look for other methods suitable for abnormal data. Analysis of the normality of the data is realized in Section 2.

Ad (iii) Some visualization of the data may be very helpful, therefore we will look for a way of reducing the dimensionality of the data, allowing to visualize the data points (data vectors) in the plane. In our previous investigation [13] we have shown that it is possible to separate the considered two groups of data nearly perfectly when displaying the data points in the reduces space spanned by the first two principal components. However, the separation was not ideal. Perhaps some non-linear method, performing a mapping of high-dimensional data to two dimensions, will separate the data completely.

We decided to explore the possibilities of a non-linear method called the Neuroscale. The method permits to construct mappings of data located in high dimensional data space $\mathbb{R}^d$ to a lower dimensionality data space $\mathbb{R}^q$, $q < d$ (in our case $d = 15$, and $q = 2$). The mapping performed by Neuroscale uses a criterion demanding that the inter-point distances between pairs of data vectors in $\mathbb{R}^d$ should be preserved as much as possible for their mappings in $\mathbb{R}^q$. The applied method allows for an independent look at the shape of the data clouds and the mutual distances of points-vectors constituting the data. The Neuroscale algorithm and its performance are elaborated in Section 3.

Ad (iv) We will use here the One-Class-Classification (OCC) approach [8, 2, 10, 12]. However, the task is not simple. The probability distribution of the gear data is not normal and we have to model it by using some more complex distributions. After reducing the dimensionality by constructing new features we will model the distribution of the obtained features using Parzen kernels, mixture of Gaussians and SVDD (Support Vector Data Description). We appreciate here the Matlab dd-toolbox by J.D.M. Tax [8]. The description of this task and its results are described in Section 4.

Conclusions and final remarks are presented in Section 5.
2. Mahalanobis distances

Let \( \mathbf{x} = (x_1, \ldots, x_d)^T \) denote a data vector. Let \( \mathbf{m} = (m_1, \ldots, m_d)^T \) denote a fixed point (usually the center) of the data. Let \( S_{d \times d} \) be the positive definite covariance matrix of the observed variables \( X_1, \ldots, X_d \). The squared Mahalanobis distance between the points \( \mathbf{x} \) and \( \mathbf{m} \) is then defined as:

\[
DD = (\mathbf{x} - \mathbf{m})^T S^{-1} (\mathbf{x} - \mathbf{m}).
\]

(1)

It is known that for data vectors coming from a MV (multivariate) Gaussian distribution with expected value \( \mathbf{m} \) and full rank covariance matrix \( S \) the quadratic form defined by (1) is distributed like a \( \chi^2 \) variate with \( d \) degrees of freedom.

Figures 1 and 2 show the squared Mahalanobis distances DD computed both for the 'good' and the 'bad' data. Each DD was calculated taking the centroid (vector of means) of the respective group as the reference vector \( \mathbf{m} \) in formula (1).

![Figure 1](image.png)

Figure 1. Squared Mahalanobis distances computed in the GOOD (left) and the BAD (right) data sets, with boundaries established at the alpha=0.05, 0.01 and 0.001 significance level. Notice, how many data vectors surpass all the boundaries.

Figure 1 shows exhibits containing index plots (called sometimes historic plots). For each item (vibration segment of recorded data) the corresponding DD-value is displayed. Successive items appear in time order - as they were recorded. Significance bounds at the level \( \alpha = 0.05 \), \( \alpha = 0.01 \) and \( \alpha = 0.001 \) are marked by horizontal lines. Looking at the two exhibits in figure 1, it becomes immediately apparent that neither the GOOD nor the BAD data set is normal (Gaussian). There are too many outliers surpassing in figure 1 the upper boundary established for the \( \alpha = 0.001 \) significance level. Moreover, there is a time dependence. The bad data appear in clumps. Eventually, one may see three outliers far beyond the \( \alpha = 0.001 \) significance level. The 'good' data with item numbers 400-500 (approximately) show a specific dip containing similar low values of \( DD \approx 10 \).

The abnormality of both data set may also be seen in qq-plots shown in Figure 2. It is evident that the data are asymmetric and heavy-tailed.

In the following we will concentrate on the the GOOD data and investigate what is their probability distribution and what kind of outliers does it contain. To answer this question, it is necessary to construct a probability model describing the multivariate data vectors constituting the set GOOD. It follows from the analysis of Mahalanobis distances shown above that the
sought distribution is certainly not Gaussian, thus we have to seek for other distribution. The result obtained in [13], stating that the intrinsic dimensionality of the data is 2 or at most 3, may be helpful here. In the following we will seek for a mapping of the 15-dimensional data to 2-dimensional plane using the Neuroscale method. Next we will model the representation in two dimensions by some 2D density models and find its low-density boundaries.

### 3. The Neuroscale algorithm

The Neuroscale algorithm [6, 7, 4] permits to project multivariate data to lower dimensions. It is aimed at to keep the distances between the projected pairs of data points similar to those in the original multivariate space. Let \( x_i \in \mathbb{R}^d \) denote a generic data point located in the observed data space with \( d \) dimensions. Let \( d^*_{ij} \) denote the distance (usually in Euclidean metric) between two data points \( x_i \) and \( x_j \), both from \( \mathbb{R}^d \). Say, we project the points from \( \mathbb{R}^d \) to a lower subspace \( \mathbb{R}^q \), \( q < d \) (usually \( q = 2 \) or \( q = 3 \), the projection will be defined below). Let \( y_i \) and \( y_j \) denote the projections of \( x_i \) and \( x_j \), and let \( d_{ij} \) denote the distance (in the same metric) between the projections \( y_i \) and \( y_j \). There are \( N \) data points. The quality of their projection is measured by the Sammon stress metric (quoted after [7]; some authors add here denominators for each term):

\[
E_{\text{sam}} = \sum_{i=1}^{N} \sum_{j>i}^{N} (d_{ij} - d^*_{ij})^2. \tag{2}
\]

The smaller the stress, the more closely the distances between the \( y_i \)-s match the distances in the original space between the \( x_i \)-s, and hence the better the structure of the original data space is preserved in the reduced data space. The metric for measuring the distances may be arbitrary, however, if the main purpose of the projection is visualization of the data in lower dimensions, it is worth remembering that our visual system is highly tuned to discriminate patterns based on Euclidean distances, and so it makes most sense for the metric in the projection space to be Euclidean (quoted from Ian Nabney [7], p. 264). On the other hand, the metric should be the same in both spaces, otherwise some spurious structure may appear.
The concept of mapping points from a high dimensional data space $R^d$ to lower dimension $R^q$, $q < d$, using the criterion given by eq. (2), was formulated in 1967 by J.W. Sammon Jr., who solved it as an optimization problem in the variables $y_i$, $i = 1, ..., N$ for given $N$. The solution does not provide any generative formula for mapping points from $R^d$ to $R^q$. One obtains a mapping only for the given $N$ points of data; it is not possible to make a mapping for an additional data point $x^*$ not accounted for during the previous optimization process; to do this, it is necessary to carry out the whole optimization process anew with $N + 1$ data points.

Lowe and Tipping [6] proposed a complete different approach which is generative in the meaning explained above. They proposed to use the RBF neural network with weights designated from the criterion (2), and apply an optimization algorithm called by them shadow targets. The algorithm is described in the book by Nabney [7], where also an implementation in Matlab is provided. Using Nabney’s software, we have constructed the mapping for the set GOOD (N=951 items). This was performed twice: 1) for raw data, and 2) for data standardized statistically (each variable has mean=0 and standard deviation=1). The obtained mappings are shown in figures 3 and 4.

Figure 3. The Neuroscale mapping applied to the non-standardized data set ’GOOD’. Top row: two projections, obtained from random start of the algorithm. Bottom row: Other two projections obtained starting from the PCA plane. 30 RBF centers were used in each simulation. Notice that the displayed clusters have similar shapes, although each cluster is located in its coordinate system in a different way.
Each RBF network contained 30 centers (radial basis functions) defined as TPS (thin plate splines, see [7]). The algorithm places firstly the randomly chosen centers according to density of the data; next the weights of the neural network have to be found so as to provide at the output the desired projections $y_i$-s satisfying the Sammon stress criterion given by eq. (2).

Each of the figures 3 and 4 contains four exhibits arranged in two rows. The upper rows show mappings obtained when using random starts of the centers. The bottom rows shows results when starting from random centers located in the plane composed from the first two principal components of the data.

Comparing all the exhibits one may state that generally the projections look consistent:

- all clusters of projected points show a topographic similarity, and have similar shape,
- the exhibits in the lower rows, which were started from the $<PC1, PC2>$ plane, have the same shape and location with respect of the constructed $Y_1$ and $Y_2$ axes,
- the exhibits in the upper rows have varying position with respect of the the constructed $Y_1$ and $Y_2$ axes, however they look topologically similar, i.e. after a rotation and reflection they could be superposed each other,
- generally, all the points in the exhibits look like composed from one major and one minor cluster. The clusters obtained from standardized data look more rounded (especially the smaller one) and more scattered.

Figure 4. As in previous figure, however considering standardized data.
In the following, one of the obtained 2D Neuroscale projections (standardized data, starting points taken from the <PC1,PC2> plane) will be analyzed in more detail. This will be done by (A) building three models of the probability density of the 2D points, (B) constructing for each model a decision boundary delimiting high and low-density areas of the distribution, and (C) checking the efficacy of the constructed bounds in rejecting bad data from the BAD data set.

4. Modelling the data and performing one-class-classification (OCC)

A. After inspecting the 2-D data shown in figures 3 and 4 it is evident that the data should be modelled using very flexible distributions allowing for a big variety in shapes. We have chosen the following three ones:

(i) **Parzen kernel** method: has one smoothing parameter \( h \),
(ii) **mixture of Gaussians**: each Gaussian needs parameters characterizing its mean, covariance matrix and proportion of contribution to the mixture; the user has to declare \( K \), the number of Gaussians in the mixture,
(iii) **one-class Support Vector Data Description (SVDD)**: needs one slack variable \( C \).

The used **Parzen density** function with Gaussian kernels was taken as proportional to \( f(x) \):

\[
f(x) = \sum_{i=1}^{n} \exp(- (x - x_i)^T h^{-2} (x - x_i)).
\]

The free parameter \( h \) was optimized by maximizing the likelihood on the training data using leave-one-out method.

The **mixture of Gaussians** density had the form:

\[
f(x) = \sum_{i=1}^{K} P_i \exp\left(- (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)\right)
\]

The parameters \( P_i, \mu_i, \Sigma_i \) were optimized internally in the software using Maximum Likelihood and EM method; the parameter \( K \) was found by trial and error.

The **one-class SVDD** method was inspired by the support vectors classifier (used in SVM, Support Vector Machines method) tending to put a largest margin between two classes of data. It needs a difficult optimization. The OCC formulation was given originally by Tax and Duin [10], see also [9, 8]. Basically, it fits a hypersphere around the target class (in our case: the GOOD data) and may be combined with kernels, which transform the hyper-sphere to more complicated shapes. We used RBF kernels.

B. The goal was finding a decision boundary (DB) delimiting an alpha=0.05 fraction of the observed data points from the main bulk of the data, with the aim to find low-density data points (outliers) outside the boundary. We have used for this purpose the algorithms described in [9, 8, 10], see also [2]. The obtained DBs are shown in figure 5.

The DBs were constructed using the GOOD data only. The entire set (n=951) was subdivided at random (Matlab function randperm) into two parts containing 400 and 551 data vectors appropriately. The first part has served as training sample and the remaining (second) part as test sample. For each distribution model, the respective DBs were constructed using the training sample. Next, the constructed DBs were tested for their effectiveness using the remaining part of the data (called 'remn'). The left column of exhibits in figure 5 shows the DBs (together with the data points) for the training sample, and the right column the same (i.e. the DBs and data points) for the test sample. Each exhibit shows three DBs obtained for different values of parameters (h, K, C) of the respective model. One may notice that the results (DBs) obtained for the probabilistic models (that is, parzen and mog) look much alike, with a predominant
Figure 5. Decision boundaries for the GOOD data constructed so as to yield the first kind error $\alpha=0.05$. Left column: DBs constructed from training sample: $n=400$. Right column: DBs constructed from test sample: $n=551$. Top row: Parzen window (parzen) method. Middle row: mixture-of-Gaussians (mog) method. Bottom row: SVDD method.
circular shape. We have observed in other experiments (not shown here) that the shape of the DBs remains practically the same over a quite large range of parameters. Concerning the model SVDD which works not on densities but on margins, the shape tends to be triangular; with a larger value of C (the slack parameter), the entire data becomes subdivided into two disjoints parts, one of them smaller and circular. The general conclusion from visual inspection of the graphs could be that the methods are relatively stable and the same data points appear outside the constructed boundary.

C. The accuracy of the constructed DBs was measured by two indices:

err – 1st kind error, defined as proportion of GOOD (normal) data located beyond the DB,
rec – recall, defined as proportion of data enclosed by the DB.

For one-class classification, err and rec are complementary with the restraint: err + rec = 1.

Another check of accuracy in recognizing BAD (abnormal) data is provided by
TN – true negatives, defined as proportion of BAD data recognized as BAD,
FN – false negatives, defined as proportion of BAD data recognized as good.

The accuracy measured for the training sample (with n=400) is nominal, that is to mean, err=0.05 and rec=0.95. It is shown in Table 1 as err1 and rec1.

The accuracy measured for the test sample (with n=551) is shown in Table 2 as err2 and rec2. One may notice there that, compared to the nominal level, a larger fraction of data points is positioned outside the DB, and consequently, a smaller fraction of data is recognized as normal.

The accuracy of recognizing the abnormal data (n=1232) when using the BAD set is shown in Table 1 as TN and FN. The BAD data contain ex definitio only bad (negative) samples, which all should be found outside the DB. One may notice that the DB constructed using the GOOD set is surprisingly good in classifying the BAD data. Barely 1-2% of the bad data is falsely classified as ‘good’, and this happens for all the evaluated variants of the constructed DBs. The repeatability of these results needs further investigation.

Table 1. Error and Recall evaluated for (a) a sample from the GOOD data, (b) its remaining part ‘remn’, and (c) the BAD data as test sample. Density models: parzen – Parzen, mog – mixture of Gaussians, svdd – Support Vector Data Description.

|               | (a) GOOD sample | (b) GOOD remn | (c) BAD data |
|---------------|-----------------|---------------|--------------|
|               | err1 | rec1 | err2 | rec2 | TN   | FN   |
| parzen k=1    | 0.0500 | 0.9500 | 0.0726 | 0.9274 | 0.9870 | 0.0130 |
| parzen k=1.5  | 0.0500 | 0.9500 | 0.0617 | 0.9383 | 0.9878 | 0.0122 |
| parzen k=2.0  | 0.0500 | 0.9500 | 0.0690 | 0.9310 | 0.9911 | 0.0089 |
| mog k=2       | 0.0500 | 0.9500 | 0.0472 | 0.9528 | 0.9889 | 0.0211 |
| mog k=10      | 0.0500 | 0.9500 | 0.0708 | 0.9292 | 0.9886 | 0.0114 |
| mog k=20      | 0.0500 | 0.9500 | 0.0744 | 0.9256 | 0.9894 | 0.0106 |
| svdd C=3.0    | 0.0500 | 0.9500 | 0.1125 | 0.8875 | 0.9797 | 0.0203 |
| svdd C=2.0    | 0.0500 | 0.9500 | 0.1180 | 0.8820 | 0.9838 | 0.0162 |
| svdd C=2.5    | 0.0500 | 0.9500 | 0.1089 | 0.8911 | 0.9894 | 0.0106 |

5. Discussion and closing remarks
The aim of this analysis was to find outliers in the data and use them for condition monitoring of planetary gearboxes working under time varying external load. Our concern was diagnosis of a multistage gearbox being part of a complex mining machine, namely bucket wheel excavator. There are not many such machines, they have unique design and it is difficult to get data showing the degradation (abnormal state) of such device; in particular when the abnormal state
may manifest itself in very different modes of malfunctioning. However, it is relatively easy to obtain data characterizing the normal state of the device. In such a case the OCC (One-Class Classification), a not widely recognized method, seems to be a good and relatively simple solution for troubles with diagnosing the state of a given machine.

A data set called GOOD with $n=951$ data vectors (with $d=15$ attributes each), characterizing the normal operating of a planetary gearbox, was considered. Using the Neuroscale method, the 15 attributes were reduced to two new features that can be visualized on the plane. The distribution of the reduced data appeared to be highly non-Gaussian and was synthesized by 3 methods: a) Parzen, b) mixture of Gaussians, c) SVDD.

For all the 3 methods, appropriate one-class decision boundaries (DBs) with nominal first kind error $\alpha=0.05$ were constructed; they were tested using an independent data set (BAD, $n=1232 \times 15$) obtained from a different device known to be in a bad condition. It appeared that all the DBs constructed for all the 3 models work well and recognize more than 98% of the BAD data as 'bad'. The data vectors indicated falsely as 'good', constitute less than 2% of all the BAD data, correspond to the operational state 'no load', when it is really difficult to judge, what is the state of the device (see Bartelmus [1]).

Acknowledgements
The authors thank to one anonymous referee for helpful comments and suggestions. RZ’s research is partially supported by (Polish) State Committee for Scientific Research in 2009 - 2012 as research project (R. Zimroz).

References
[1] Bartelmus W and Zimroz R 2009 A new feature for monitoring the condition of gearboxes in non-stationary operating conditions, Mechanical Systems and Signal Processing 23 1528-1534
[2] Bartkowiak A 2010 Anomaly, novelty, one-class classification: A short introduction, Proc. Computer Information Systems and Industrial Management Applications 2010 Ostrava ed A Ajith et al 1-6 (IEEE Xplore DOI 10.1109/CISIM.2010.5643699)
[3] Clifton D A, Clifton L A and Bannister P R 2008 Automated novelty detection in industrial systems. Studies in Computational Intelligence (SCI) 116 (www.springerlink.com) 269–296
[4] Clifton D A, Bannister P R and Tarassenko L A 2006 Application of an intuitive novelty metric for jet engine condition monitoring. M. Ali and R. Dapoigny (Eds): ISEA/AIE LNAI 4031 1149–1158
[5] Duin R P W et al. 2007 PRTools4, A Matlab Toolbox for Pattern Recognition 2007, (Delft Pattern Recognition Research, Faculty EWI Delft University of Technology, Delft The Netherlands) http://www.prttools.org
[6] Lowe D and Tipping M E 1997. NeuroScale: Novel Topographic Feature Extraction using RBF Networks. Advances in Neural Processing Systems (Cambridge MA: MIT Press) 543–549
[7] Nabney I 2001 Netlab: Algorithms for Pattern Recognition (Springer Series: Advances in Pattern Recognition)
[8] Tax D.M.J. 2008 Data description toolbox dd_tools 1.6.3. A Matlab toolbox for data description, outlier and novelty detection, (EWI Delft UT Sept 2008)
[9] Tax D M J 2001 One-class classification (PhD dissertation Delft University of Technology) http://www.ict.etw.tudelft.nl/papers.html
[10] Tax D M J and Duin R P W 1999 Support vector domain description Pattern Recognition Letters 20 1999 1191–1199
[11] Surace C and Worden K 2010 Novelty detection in a changing environment: A negative selection approach Mechanical Systems and Signal Processing 24 1114–1128
[12] Worden K, Manson G and Allman D 2003 Experimental validation of a structural health monitoring methodology: Part I. Novelty detection on laboratory structure Journal of Sound and Vibration 259 (2) 323–343
[13] Zimroz R and Bartkowiak A 2011 Two simple multivariate procedures for monitoring planetary gearboxes working in nonstationary operating conditions (Manuscript pp. 1–14)
[14] Zimroz R and Bartkowiak A 2010 Investigation on spectral structure of gearbox vibration signals by principal component analysis for condition monitoring purposes (Submitted to Damas 2011, under revision, 1–10)