DETECTING OUTLIERS IN MULTIVARIATE DATA BY USE OF KOHONEN's SELF-ORGANIZING MAPS

ANNA BARTKOWIAK, PIOTR PIOTROWICZ, ADAM SZUSTALEWICZ

Institute of Computer Science, University of Wrocław, Przesmyckiego 20, Wrocław 51–151 Poland, e-mail: {aba,asz}@ii.uni.wroc.pl

Abstract. We consider the role of *Kohonen's self-organizing maps (SOMs)* in detecting outliers in multivariate data. A case study of the Eurostat data on italian earnings analyzed previously by Morlini (1998) is presented. We conclude that in principle a SOM can detect outlying data points – however this method do not reveal the entire neighborhood structure and additional analysis is needed to confirm the true outlyingness of the identified data points. We advise as very usefull methods in this respect the *grand tour* and *parallel coordinates plots*.

Keywords. Identification of outliers, Self-Organizing Maps, Reduction of dimensionality, Visualization of multivariate data points, Knowledge discovery.

1. THE DATA AND THEIR CHARACTERISTICS

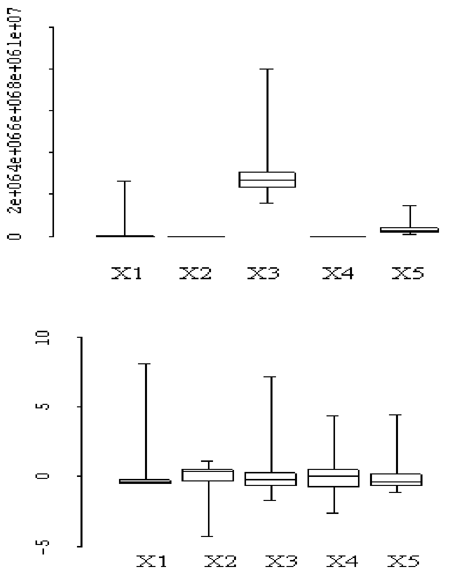

Fig.1. Boxplots constructed from original (top) and standardized (bottom) data on italian earnings for variables X1 – X5

We will illustrate our considerations by using a data set analyzed recently by Morlini [10]. The data were taken from the Eurostat New Cronos Data Base and can be presented in the form of the data matrix $\mathbf{X}_{n \times p}$, with $n = 158$ units of economical activities in Italy in 1995, each characterized by $p = 5$ variables:
$X1$ – "nbempl" –number of employees, $X2$ – "mwhour" – mean weekly hours, $X3$ – "mmearn" – mean gross monthly earnings, $X4$ – "cvarme" – coefficient of variation of monthly earnings, $X5$ – "bonus" – mean monthly value of periodic bonuses. Each data vector may be viewed as a point in 5-dimensional space R^p, the input space.

All the recorded values were measured in a real continuous scale with a very differentiated range, what can be seen in the boxplots exhibited in Fig. 1, upper plot. After standardizing the variables to mean=0 and standard deviation=1 we obtained a reasonable common scale, what can be seen in Fig. 1, lower plot.

Both plots exhibited in Fig. 1 show a high asymmetry in the distributions of all analyzed variables. The asymmetry could be reduced by using some transformations, e.g. taking *logarithm* of the observed values. We will not do it here.

In the following we will analyze the standardized raw data, without transformations reducing their skewness. By "standardization" we mean subtrahing from each variable value the appropriate mean, and next dividing the obtained difference by the appropriate standard deviation. To avoid negative numbers we added to each standardized value the number "5.0", thus obtaining values from the interval [0, 15].

2. THE GRAND TOUR METHOD

The grand tour, originally proposed by Asimov, was adapted by Bartkowiak & Szustalewicz [2] specifically for detecting and identifying outliers. It works as a proceeding sequence of rotations and projections of the considered data points (see [2] for further description and references).

The grand tour was performed on standardized variables.

One snapshot of the projections visible in the computer screen at time t_k is shown in Fig. 2. The momentary view at time t_k consists of a scatterplot with a concentration ellipse (upper plot) and an indexplot (called *count plot*) exhibiting counts telling how many times the given point observed up to time t_k was notified outside the ellipse (lower plot).

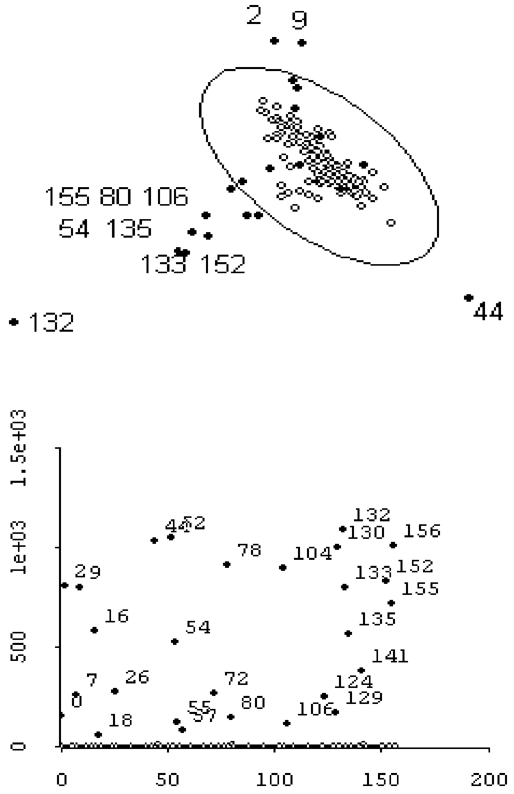

Fig. 2. One snapshot from the grand tour.
Top: A momentary projection seen on the screen exhibiting the point projections and a 99% robustified concentration ellipse.
Bottom: Corresponding count plot indicating how many times each data point was found – till the moment of the snapshot – outside the concentration ellipse.
Both plots are dynamically linked. 27 points with the highest counts are highlighted

The data points appearing outside the concentration ellipse are nominated as suspected outliers. All the nominated points have high counts indicated in the count plot. For the analyzed data these are the points (the numeration starts from 0): { 0 2 7 9 16 18 26 44 52 54 55 57 72 78 80 104 106 124 129 130 132 133 135 141 152 155 156}.

Morlini indicated additionally 74, 81, 83, 107, 109, however has not found from our set : 0, 2, 7, 16, 18, 106.

Generally, the nominated outliers may be located at various 'sides' of the multivariate data cloud $\subset R^p$. The proximity of some points noticed in the momentary projections does not mean that these points are really close in the R^p space. To assess their proximity we may use an agglomerative clustering with complete linkage based on angular distances between the points (for a definition of angular distance and an example of working in such way see [3, 4]).

3. SELF-ORGANIZING MAPS

Self-organizing maps, like Kohonen networks [7, 8, 12] permit to obtain a planar (2D) representation of points from the multivariate space. Usually this is a mapping from R^p to R^2 with the desirable property of topology preservation.

The method is frequently used for projection and for clustering of multivariate data. It has been applied in many practical problems (a.o. Kohonen [8] gives reference to a publication base, where about 3700 papers using this method are listed). There have been also attempts to apply the method for detection of atypical data points [9, 10].

The method is based on an unsupervised learning algorithm (competitive learning) producing a grid configuration of neurons, which adapts and adjusts itself in subsequent iterations (epochs) – with the aim to be in accordance with (to reflect) the true configuration of the data points in R^p. Each grid node h, ($h = 1, \dots m \times n$, with $m \times n$ denoting the dimensions of the grid) is in correspondence with a reference point (short: *ref* point) $\mathbf{m}_h \in R^p$.

At the end of the training the grid nodes (neurons) should reflect the configuration neighborhood of the reference points in R^p.

The training needs many (often: several thousands) of iterations. Then, the learning needs some tuning, which depends from such parameters, as the magnitude and shape of the neighborhood, the shrinking schedule of the neighborhood, and the learning rate – which should be specified by the user.

The method aims at mapping data points lying near each other in the R^p to nearby map units.

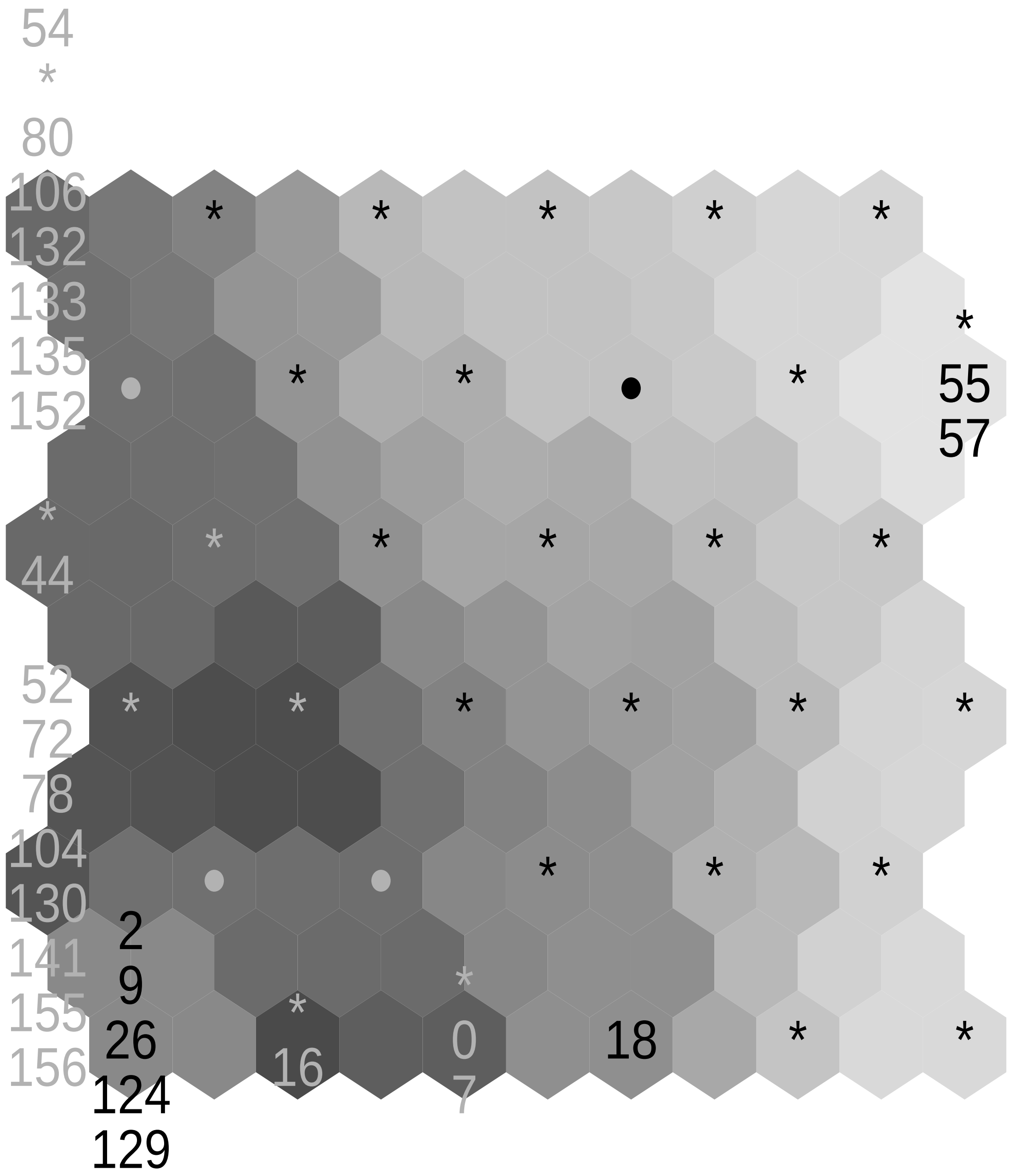

Fig. 3. Map of the italian earnings data - constructed on a grid of size 6×6 with a hexagonal topology. Deep gray indicates that the corresponding reference point $\mathbf{m}_h \in R^5$ is far from its neighboring (according the map) reference points.
Open circles denote empty regions. Asterisks (*) indicate regions containing points not suspected being outliers. Ordinals written in columns centered on some nodes of the map indicate data vectors suspected to be outliers.
*Please notify that all suspected outliers - except no. 55 and no. 57 – are in dark regions of the map. The region corresponding to the upper left corner grid-point contains the points: 54, 80, 106, 132, 133, 135, 152 and some other(s) denoted by an *; the region corresponding to the lower left, one row above the corner, node contains the points: 52, 72, 78, 104, 130, 141, 155, 156 only – all of them suspected as being outliers*

It may happen, that the mentioned parameters are chosen improperly, and then the result of the learning will not reflect truly the configuration in the input space R^p.

In Fig. 3 and Fig. 4 we show two maps ($SOMs$) obtained by using Kohonen's algorithm and Kohonen's software SOM_PAK [7, 6].

The maps were obtained with the following parameters:

- topology type of the map: hexagonal
- neighbor type: gaussian
- size of the map: 6×6 for the first and 8×8 for the second map
- nb of trials to obtain the starting configuration: 10
- nb of iterations in the first training period: 1500
- learning rate in the first training period: 0.4
- initial radius of neighborhood: 3.5 for the 1st and 5.5 for the 2nd map
- nb of iterations in the second training period: 4000
- learning rate in the second training period: 0.05
- radius of neighborhood in the 2nd stage: 2.2

The training proceeded in a number of iteration steps. At each step t a randomly chosen data vector $\mathbf{x}(t)$ was considered and its best matching *ref* vector $\mathbf{m}_c(t)$ was found, such that for every i

$$||\mathbf{x}(t) - \mathbf{m}_c(t)|| \leq ||\mathbf{x}(t) - \mathbf{m}_i(t)||.$$

Next this vector $(\mathbf{m}_c(t))$ and its correspondent neighbors from the map were updated according the formula

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{c(t),i}(t)(\mathbf{x}(t) - \mathbf{m}_i(t)) ,$$

where $h_{c(t),i}(t)$ was the neighborhood kernel around the winner unit $\mathbf{m}_c(t)$. For details see [7, 12, 6].

After the training period we obtained the quantization errors evaluated as the mean squared distance of each data point to its nearest *ref* point:

$err = 1.01$ for the first map (6×6),

$err = 0.83$ for the second map (8×8).

In such way, taking into account subsequent *ref* points $\mathbf{m}_h$ $(h = 1, \ldots, m \times n)$, the entire space R^p may be subdivided into $m \times n$ regions, each region R_h having as its representative the *ref* point $\mathbf{m}_h$.

Now also the data points $\{1, 2, \ldots, n\}$ can be subdivided into $m \times n$ subgroups according to the principle, that $\mathbf{x}_i$, the i-th data point, is assigned to the region R_h such that

$$h: \quad arg\ min||\mathbf{x}_i - \mathbf{m}_j||,\ j = 1, \ldots, m \times n.$$

In Figs 3 and 4 we see in first place a hexagonal grid of map units representing the neurons, called also nodes of the map. The nodes are in one–to–one correspondence to the *ref* points $\mathbf{m}_1, \ldots, \mathbf{m}_{m\times n}$. The hexagonal map units are in various shades of gray. The shade of the gray corresponds to the distance of the respective *ref* point from its neighbors; black standing for the furthest distance.

In the maps exhibited in Figs 3 and 4 the nodes are marked either by an asterisk (*) or by a small circle or by a column of ordinals. Wherefrom such differentiation?

The small circles indicate for *ref* points representing empty regions, what means that no data point was allotted to the regions represented by these *ref* points.

The remaining map units (nodes) represent non empty regions.

To make the presentation more clear we have concentrated only on the suspected outliers. Thus, if the region corresponding to the given node h did not contain any from the suspected outliers, we have marked that node by an asterisk * ; otherwise we have written – as a column centered on the given node — the identificators (ordinals) of the suspected outliers contained in the region R_h.

E.g. in the map 6×6 exhibited in Fig. 3 the node from the upper left corner of the map represents the region in R^5 containing the following suspected outliers: 54, 80, 106, 132, 133, 135, 152 and additionally some point (points) marked by the asterisk. Note, that the suspected outliers contained in that region are just exactly the points shown in Fig. 2 as located outside the concentration ellipse.

The maps show regions differentiated in colors (shades of gray): the dark regions correspond to *ref* points located far from their neighbors. Thus points located in dark regions may be outlying points.

To make the analysis easier, we have marked in the maps only points identified by the grand tour method. The remaining points are marked with an asterisk (*).

Looking at the two exhibited maps one may note that the configurations of the points are quite different, however they have something in common: The selected (by the Grand Tour) points are located at the borders of the maps, moreover nearly all of them are located in dark regions of the map.

After analysis of the configuration of the indicated points we come to the conclusion: the maps show the outliers, although they do not show exactly their mutual position in the p-dimensional space.

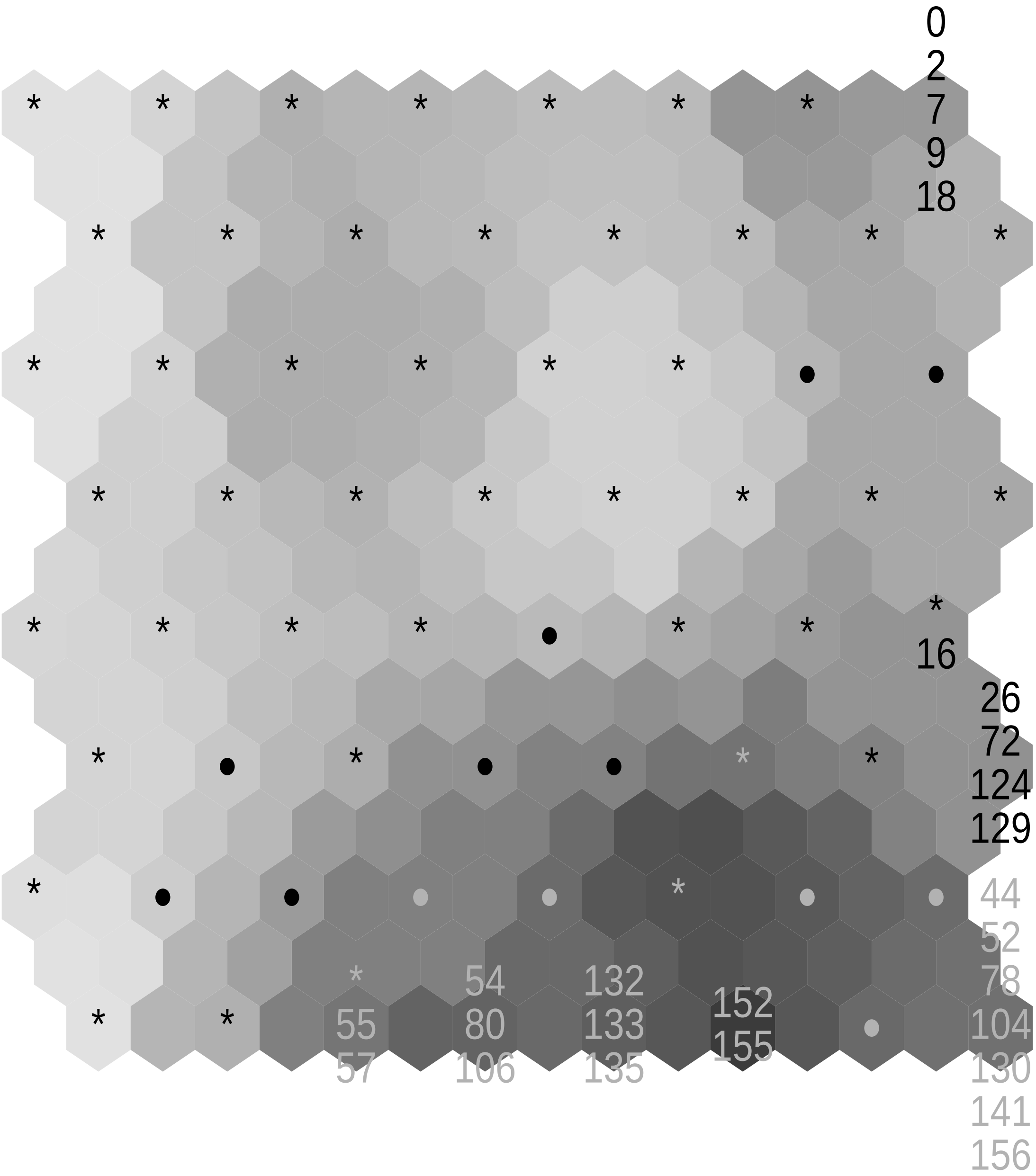

Fig. 4. Map of the italian earnings data - built on a grid of size 8×8. Legend – as in Fig. 3. Please notice, that all the suspected outliers are again in dark regions of the map, albeit the location of the dark regions in the map is somehow flipped and inverted – as compared to Fig. 3. Please notice also that despite this different physical position (coordinate position) on the map the neighborhood of the suspected points is - in great extend - preserved.
An essential difference is in location of points no. 55 and no. 57: now these points appear at the border of the dark regions containing the outliers. The region corresponding to the bottom right corner node contains the points: 44, 52, 78, 104, 130, 141, 156, all of them suspected being outliers

4. CONFIRMATIVE EVALUATION OF THE FOUND OUTLIERS – THE PARALLEL COORDINATE PLOT

The *parallel coordinate plot* [5, 13] is a special technique permitting to see in one rectangular picture the values of all variables for every data point (individual data vector).

This technique draws p vertical equidistant and equal lines representing the axes of the succeeding analysed variables $X_1, \ldots, X_p$. Each data point (data vector) $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$, $i = 1, \ldots, n$ is represented in the plot as a $(p-1)$-element segment line crossing the axes at the values of the variables. A convenient software is described in the book by Tierney [11] and in the paper by Alexakis et. al. [1].

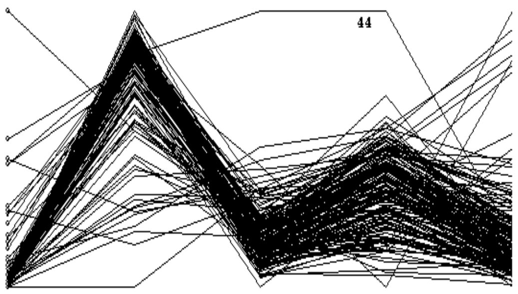

Fig. 5. Parallel coordinate plot for the entire data

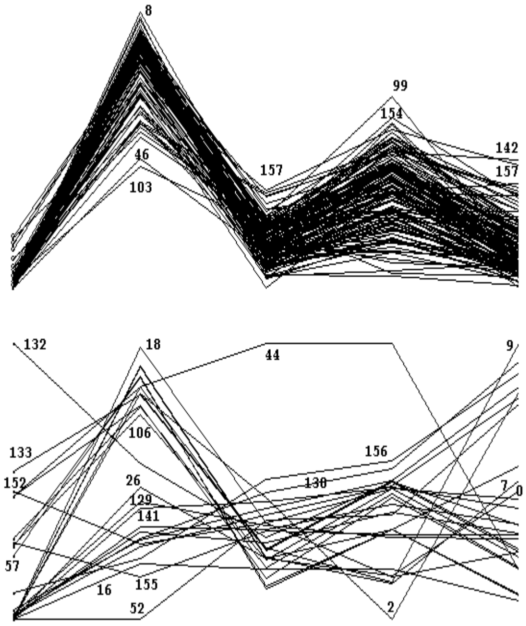

Fig. 6. Parallel coordinate plots: for the 'cleaned' data set (top) and for the outliers removed (bottom)

Individual data points can be selected (highlighted, or given special color) by clicking with a mouse the segment line representing the chosen data vector. We can select also a group of data vectors.

Looking at the position of the crossings of the line segments with the variable axes, one can see, which are the values of the variables for the selected individuals. In particular, one can see whether these values differ much, or not at all, from the values exhibited by the not selected data vectors.

The parallel plot needs operating with colors; it is difficult to appreciate its valor using only gray shades or different patterns in black and white colors.

Looking at the representation of individual data vectors (like those shown in Figs 5 – 7) one is able to say in more detail why the given data vectors were indicated as 'outliers'.

In our analysis we have split the data firstly into two parts: one containing the cleaned data set with outliers removed, and the other only the outliers. The latter was very diversified, therefore we have split it further to obtain some more homogeneous subgroups.

Parallel coordinate plots drawn for our data are shown in Figs 5 – 7.

Fig. 5 shows the entire data set. One can see there the main bulk of segment lines which follows a similar pattern, and some additional lines with different, individual pattern. Strongly atypical segment line visible in the upper part of the plot represents the lonely individual 44.

Fig. 6 was obtained after splitting the entire data set into two groups: one of them containing only the 'cleaned' data set, with suspected outliers removed, and the other – only the outliers. Parallel coordinate plots for these two subgroups are shown in Fig. 6.

Looking at the upper plot one can see the main bulk of segment lines which is narrow for the first variable and getting wider for the remaining ones. Generally, all the data vectors exhibited here show a similar dependency pattern.

Looking at the lower plot one can see a great diversity of the segment lines. To obtain more homogeneous sets we have further subdivided the outliers from the bottom plot of Fig. 6 into 3 separate subgroups containing data vectors with more similar pattern. These subgroups are shown in the three plots depicted in Fig. 7.

The upper plot shows segment lines representing twelve individuals {16, 26, 52, 72, 78, 104, 124, 129, 130, 141, 155, 156} from the darker region in the bottom right corner of the 8×8 map in Fig. 4. The data vector 155 has been included as well because of its location (in the map 6×6) listed above.

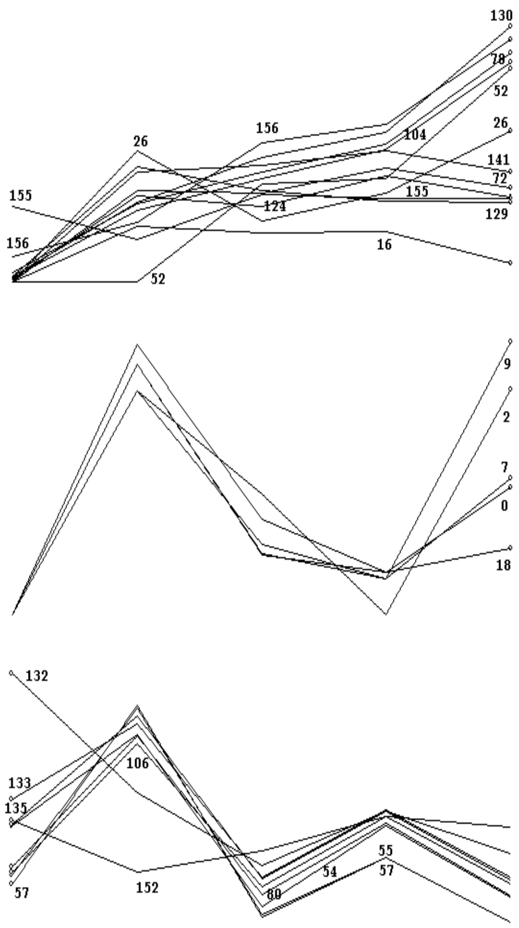

Fig. 7. Parallel coordinate plots for three subsets of suspected outliers selected from the Kohonen's maps

In the center plot we see five segment lines representing the data points {0, 2, 7, 9, 18} assembled together in the dark region in the upper right corner of the 8×8 map from Fig. 4. As we observe, this subgroup contains data vectors very similar with respect to the variables X1 – X4, but differentiated in the variable X5. Let us mention, that these data vectors appear also differentiated in the bottom darker region in the 6×6 map shown in Fig. 3.

And in the bottom plot of Fig. 7 we see nine segment lines representing the data vectors {54, 55, 57, 80, 106, 132, 133, 135, 152} from the bottom dark region of the map in Fig. 4. These data points are also visible outside the concentration ellipse on the snapshot in Fig. 2.

Looking at the plots from Fig. 7 one can state that really the indicated outliers are atypical in one or another way.

5. FINAL REMARKS AND CONCLUSIONS

Analyzing the data we have met several problems:

Firstly, there is the problem of scaling the data. It seems to us that before starting the analysis, the data should be firstly normalized, otherwise the results will reflect the scales of the variables and not necessarily their interdependency structure.

Secondly: How define the outliers for extremely skew distributions? Should the data (its variables values) be firstly transformed to reduce their skewness?

In the paper we have used two graphical methods for detecting multivariate outliers: *The Grand Tour* and Kohonen's Self-Organizing Maps (*SOM*s). Both methods gave reasonable results.

The Grand Tour seems to us simpler in concept and in explanation. The results from various runs are in principle the same. Except for very special structures, the method shows the true shape of the data. However, at the moment, the method can not be directly applied for analyzing large data sets, in particular with many variables (>20, curse of the dimensionality).

Kohonen SOMs can be applied (and were applied) to huge data sets. They work very fast and the algorithm is simple. However, they can not show the true shape of the data. Also, the final result in the map (grid of neurons) depends on many parameters which should be set by the user before starting the evaluations.

Generally, our experience is that to reveal the atypical structure in the data by use of Kohonen SOMs, we have to perform a series of runs, varying the starting configuration and the size of the grid on which the map is based. In principle, the atypical structure may then be found. However the analysis of the obtained results takes much time and a procedure formalizing this phase is needed.

In the special case – when seeking for outliers – both methods need a confirmatory stage, which may be provided either by a repeated run of the grand tour, or by the parallel coordinate plot.

ACKNOWLEDGEMENTS

The research was partially sponsored by the (Polish State) KBN grant *no.* 8 T11 C031 16, and the authors acknowledge gratefully receiving this grant.

The authors acknowledge also gratefully having the possibility of using for their calculations the public domain software:

SOM_PAK by Teuvo Kohonen et al. [6], package used for training and presenting graphically the self-organizing maps,

XLISP-STAT by Luke Tierney [11], package used in remaining analysis, except the Grand Tour.

6. REFERENCES

[1] Alexakis A., Deftereos M., Samiotakis Y., Visualization techniques in statistics. The parallel coordinates visualizer (PARCOVI), In: Ph. Nanopoulos, P. Garonna, C. Lauro (Eds), *NTTS'98 International Seminar on New Techniques and Technologies for Statistics, Sorrento, Italy 4-6 November 1998*, Contributed Papers, 19–24.

[2] Bartkowiak A., and Szustalewicz A. (1997). Detecting outliers by a grand tour, *Machine Graphics & Vision* 6, 486–505.

[3] Bartkowiak A., and Szustalewicz A. (1998). Outliers – which genuine and which spurious. In: Ph. Nanopoulos, P. Garonna, C. Lauro (Eds), *NTTS'98 International Seminar on New Techniques and Technologies for Statistics, Sorrento, Italy 4-6 November 1998*, Contributed Papers, 61–66.

[4] Bartkowiak A., and Szustalewicz A. (1999). Some modern techniques for viewing multivariate data – a comparative look. *Intelligent Information Systems VIII. Proceedings of the Workshop held in Ustroǹ, Poland, June 14–18, 1999*, 7–11.

[5] Inselberg A., Parallel Coordinates; a guide for the perplexed Plot, *Hot Topics Sect. of IEEE Comp. Soc. Conf. on Visualization '96*, San Francisco, Proc., 35–38.

[6] Kohonen T., Hynninen J., Kangas J., and Laaksonen J. (1996): *SOM_PAK, The Self-Organizing Map Program Package*, Report A31. Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland.
The SOM_PAK package is available from the Internet at the address:
`http://www.cis.hut.fi/nnrc/nnrc-programs.html`

[7] Kohonen T. (1995). *Self-Organizing Maps.* Series in Information Science, Vol. 30. Heidelberg, Springer. Second ed. 1997.

[8] Kohonen T. (1999). Exploration of statistical and textual information by means of self-organizing maps. *Bulletin of the International Statistical Institute, Tome LVIII, Proceedings ISI 99, Book 1, Invited Papers* Helsinki 1999, 307–310.

[9] Maruzábal J., Muñoz A. (1997). On the visualization of outliers via self organizing maps. *Journal of Computational and Graphical Statistics,* 6, 355–382.

[10] Morlini I. (1998). Multivariate outlier detection with Kohonen networks: an useful tool for routine exploration of large data sets. In: Ph. Nanopoulos, P. Garonna, C. Lauro (Eds), *NTTS'98 International Seminar on New Techniques and Technologies for Statistics, Sorrento, Italy 4-6 November 1998*, Contributed Papers, 345–350.

[11] Tierney, L. (1990). *LISP–STAT, an Object–Oriented Environment for Statistical Computing and Dynamic Graphics.* Wiley, New York, 1990.
XLISP-STAT software is available from the Internet at the address:
`http://www.stat.umn.edu/~luke/xls/xlsinfo/`

[12] Vesanto J. (1999). SOM-based data visualization methods. *Intelligent Data Analysis* V.3, Nb. 2, 111–126.

[13] Wegman E.J. (1990). Hyperdimensional data analysis using parallel coordinates. JASA Nb.85, 664–675.