ECONOMETRICS
EKONOMETRIA
Advances in Applied Data Analysis

Econometrics. Ekonometria.
Advances in Applied Data Analysis
Year 2024, Vol. 28, No. 1
ISSN 2449-9994

Isolation Forests for Symbolic Data as a Tool for Outlier Mining

Marcin Pełka

Wroclaw University of Economics and Business, Poland

e-mail: marcin.pelka@ue.wroc.pl

ORCID: 0000-0002-2225-5229

Andrzej Dudek

Wroclaw University of Economics and Business, Poland

e-mail: andrzej.dudek@ue.wroc.pl

ORCID: 0000-0002-4943-8703

Quote as: Pełka, M., and Dudek, A. (2024). Isolation Forests for Symbolic Data as a Tool for Outlier Mining. *Econometrics. Ekonometria. Advances in Applied Data Analysis*, *28*(1), 1-10.

DOI: 10.15611/eada.2024.1.01

JEL Classification: C80, C87, C30

Abstract

Aim: Outlier detection is a key part of every data analysis. Although there are many definitions of outliers that can be found in the literature, all of them emphasise that outliers are objects that are in some way different from other objects in the dataset. There are many different approaches that have been proposed, compared, and analysed for the case of classical data. However, there are only few studies that deal with the problem of outlier detection in symbolic data analysis. The paper aimed to propose how to adapt isolation forest for symbolic data cases.

Methodology: An isolation forest for symbolic data is used to detect outliers in four different artificial datasets with a known cluster structure and a known number of outliers

Results: The results show that the isolation forest for symbolic data is a fast and efficient tool for outlier mining.

Implications and recommendations: As the isolation forest for symbolic data appears to be an efficient tool for outlier detection for artificial data, further studies should focus on real data sets that contain outliers (i.e. credit card fraud dataset), and this approach should be compared with other outlier mining tools (i.e. DBCSAN). The authors recommend using the same initial settings for the isolation forest for symbolic data as the settings that are proposed for the isolation forest for classical data.

Originality/value: This paper is the first of its kind, focusing not only on the problem of outlier detection in general, but also extending the well-known isolation forest model for symbolic data cases.

Keywords: symbolic data analysis, isolation forest, outliers

1. Introduction

Outlier detection plays a very important role in statistical analysis and data mining. Outlier detection is also one of the opening steps in data analysis. A large variety of outlier detection techniques have been developed in different areas (see for example Aggarwal, 2017; Ayadi et al., 2017; Chandola et al., 2009). However, there are only a few methods and techniques applied for outlier detection for symbolic data analysis.

This paper presents an adaptation of isolation forests for outlier detection in symbolic interval-valued data and compares their effectiveness with the well-known DBSCAN algorithm that is also capable of detecting outliers (see e.g. Thang and Kim, 2011). This paper is organized as follows: the next section presents different outlier definitions and briefly describes outlier mining techniques; Section 3 presents symbolic objects and variables, with a special focus on symbolic interval-valued data, followed by an adaptation of isolation forests for symbolic data; Section 4 shows simulation results where isolation forests are compared to DBSCAN algorithm results. The final part of the paper contains the concluding remarks.

Outliers are also called anomalies, abnormalities, aberrations, contaminants, deviants, discordant observations, exceptions, peculiarities, or surprises in some applications (Aggrawal, 2017; Chandola et al., 2009). There are many different outlier definitions in the literature. In Ayadi et. al. (2017) there are twelve different interpretations of outliers. This shows that the outlier definition is a complex, and there is no single one that can be seen as 'the best'. Table 1 presents these definitions.

Table 1. Outlier definitions in the literature

References	Outlier definitions
Anscombe and Guttman (1960)	An outlier is an observation which is suspected of being partially or wholly irrelevant because it is not generated by the stochastic model.
Grubbs (1969)	An outlying observation, or outlier, is one that appears to deviate markedly from other members of the sample in which it occurs.
Hawkings (1980)	An outlier is an observation, which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism.
Barnett and Lewis (1994)	An observation (or subset of observations) which appears to be inconsistent with the remainder of that set of data.
Breunig et al. (2000)	Outliers are points that lie in the lower local density with respect to the density of its local neighbourhood.
Jiang et al. (2001)	Outliers are points that do not belong to clusters of a data set or as clusters that are significantly smaller than other clusters.
Hawkins et. al. (2002)	Points that are not reproduced well at the output layer with high reconstruction error are considered as outliers.
Hu and Sung (2003)	A point can be considered as an outlier if its own density is relatively lower than its nearby high density pattern cluster, or its own density is relatively higher than its nearby low density pattern regularity.
Muthukrishnan et al. (2004)	If the removal of a point from the time sequence results in a sequence that can be represented more briefly than the original one, then the point is an outlier.
Aggarwal and Yu (2005)	A point is considered to be an outlier if in some lower-dimensional projection it is present in a local region of abnormal low density.

Cheng and Li (2006)	A spatial-temporal point whose non-spatial attribute values are significantly different from those of other spatially and temporally referenced points in its spatial or/and temporal neighbourhoods is considered as a spatial-temporal outlier.
Sadik and Gruenwald (2011)	An outlier is a data point which is significantly different from other data points, or does not conform to the expected normal behaviour, or conforms well to a defined abnormal behaviour.
Singh and Upadhyaya (2012)	Outliers are patterns in data that do not conform to a well-defined notion of normal behaviour.
Keller et al. (2012)	Outliers are objects that highly deviate from regular objects in their local neighbourhood.
Aguinis et al. (2013)	Data points that deviate markedly from others.
Branch et al. (2013)	Outliers are observations whose probability of occurrence is extremely small.
Smiti (2020)	Outliers are data instances that extremely deviate from well-defined norms of a data set or given concepts of expected behaviour.

Source: own elaboration based on (Aguinis et al., 2013; Ayadi et al., 2017, p. 321; Branch et al. 2013; Keller et al., 2012; Singh and Upadhyaya, 2012; Smiti, 2020).

Despite the differences, all definitions show that outliers can be seen as data points that are different from other data points and are not errors, mistakes or noise. Regardless of the definition of outlying objects, outlier detection in various areas is very important. Thus considerable research efforts in the survey of outlier detection have been made (see for example: Aggarwal, 2017; Aggarwal and Yu, 2005; Aguinis et al., 2013; Anscombe and Guttman, 1960; Ayadi et al., 2017; Barnett and Lewis, 1994; Branch et al., 2013; Breunig et al., 2000; Chandola et al., 2009; Wang et al., 2019).

Different methods of outlier detection can be seen as more similar than others, therefore there are the following groups of outlier detection methods (Wang et al., 2009, pp. 107964-108000):

- statistical-based,
- distance-based,
- density-based,
- clustering-based,
- graph-based,
- ensemble-based,
- learning-based.

In the next part of the paper focuses on the isolation forest method which is one of the ensemble--based methods used for outlier detection. Ensemble methods are used there to answer the question of whether an outlier should be linear-based, distance-based, density-based or any model-based. Ensemble methods are a useful tool when dealing with discriminant, regression or clustering problems.

In general, one can say that ensemble techniques combine (aggregate, summarise) the results obtained from different models in order to produce more robust models and reduce the dependency of one model. Nevertheless, the ensemble approach is quite difficult to use when dealing with outliers. Lazarevic and Kumar (2005) proposed an adaptation of well-known bagging for classification problems, whilst Liu et al. (2008) suggested isolation forest for parallel techniques. Rayana et al. (2016) described an outlier mining technique for sequential methods. Zhao and Hryniewicki (2018) examined extreme gradient boosting for outlier detection (XGBOD method), and Micenková et al. (2015) proposed bagged outlier representation ensemble (BORE) for the hybrid methods.

2. Symbolic Objects and Variables and Methodology of Isolation Forests for Symbolic Data

Unlike classical data where each object is being described by a set of qualitative or quantitative variables, symbolic data analysis allows that each object can be described not only by nominal, ordinal, interval or ratio variables, but also by symbolic interval-valued variables, symbolic multivalued variables, symbolic multivalued variables with weights and also symbolic histogram variables. What is more, symbolic objects and variables allow to take into account the relations between them, namely symbolic taxonomic variables (see e.g. Billard and Diday, 2006, pp. 7-30; Bock and Diday, 2000, pp. 2-3; Brito and Dias, 2022; Diday and Noirhomme-Fraiture, 2008, pp. 10-19). More details about symbolic variables and symbolic objects can be found in Billard and Diday (2006, pp. 7-66), Brito and Dias (2022, pp. 6-35), Bock and Diday (2000, pp. 2-8), Diday and Noirhomme-Fraiture (2008, pp. 3-30). Table 2 presents some examples of symbolic variables as well as their realisations.

Table 2. Examples of symbolic variables and their realizations

Variable name	Sample realisations	Variable type
Preferred car price (in EUR)	(10000, 30000); (15000, 45000); (12000, 22000)	symbolic interval-valued (non-disjoint intervals)
Engine capacity for insurance purposes (in ccm)	(up to 1000], (1000, 2000], (2000, 3000], (over 3000)	symbolic interval-valued (disjoint variables)
Preferred car colour	{orange, yellow, blue, red}	symbolic multi-valued
Preferred car brand	{Toyota (0.6), VW (0.4)}, {Audi (1.0)} {Skoda (0.5), Renault (0.4), Other (0.1)}	symbolic multi-valued with weights
Travel time (home-work) (minutes)	{[40, 60] (0.6), [60, 85] (0.4)}	symbolic histogram variable
Age	25, 45, …	ratio
Gender	F, M	nominal

Source: own elaboration.

The isolation forest, which is an ensemble-based outlier detection method, generates many random isolation trees to partition the data, and computes for each isolation tree the number of tree nodes required to isolate each object. Anomalies are detected as objects that have the smallest average path lengths for all considered isolation trees. The main assumption is that outliers differ from other data points (non-outliers), and thus on average one needs fewer nodes to find them. To build an isolation forest one needs an isolation tree. The isolation tree for symbolic data will follow the same rules as a decision tree for symbolic data.

The authors' proposal extends the classical isolation forest for symbolic interval-valued data cases, using the following elements:

- standard decision tree for symbolic-interval valued data,
- random value of a random variable as a cutting point,
- building the decision tree for symbolic data according to the rules for this data type,
- other elements of the isolation forest will be the same as for classical data cases (number of trees 100, sample size 256, depth of the tree 6, anomaly detection threshold 0.6).

For symbolic data analysis and decision tree models, there are classification trees requiring a nominal dependent variable and a set of explanatory variables, which can be either classical variables of any type or symbolic interval-valued variables, or symbolic multinominal variables without weights (see Gatnar and Walesiak, 2011, pp. 282-285). In the empirical part, symbolic interval-valued data were used, hence the article presents how to build an isolation tree for this type of symbolic data.

Table 3 shows all the steps needed to build a symbolic isolation tree for interval-valued symbolic variables.

Table 3. Symbolic isolation tree for interval-valued data

Step number	Step name	Key elements
1	preparation of a dataset	a) data collection and construction of symbolic data table b) calculating midpoints for symbolic interval-valued variables (future cutting values c) c) symbolic interval-valued variables can have different lengths (see Table 2) one can get a set of different cutting values c
2	maximum depth of a tree	select the maximum depth of the isolation tree (l)
3	random choice	a) random symbolic interval-valued variable is chosen b) a midpoint of this variable is being selected as cut criterion c in the following steps
4	estimating probabilities for split	a) for the left node: $p_k(l) = \frac{c - \underline{v}_{kj}}{\overline{v}_{kj} - \underline{v}_{kj}}$, where: $k = 1, \cdots, n$ – number of a symbolic object; $\overline{v}_{kj}$ – upper bound of a symbolic interval-valued variable; $\underline{v}_{kj}$ – lower bound of a symbolic interval-valued variable; c – cutting value, b) if $c \leq \underline{v}_{kj}$ then $p_k(l) = 0$; if $c > \underline{v}_{kj}$ then $p_k(l) = 1$, c) for the right node it is estimated as follows: $p_k(r) = 1 - p_k(l)$
5	split decision	an object is being assigned to right or left node according to the highest probability for a node
6	repeat steps 3-5	a) repeat steps 3 to5 until final nodes are obtained – in the case of isolation forest, these steps until a node contains unique data point or maximum tree depth level l is reached b) cutting values used in previous steps (nodes) are not used again

Source: own elaboration based on (Gatnar and Walesiak, 2011, pp. 282-285; Liu et al., 2008).

The isolation forest was created by generating a set of t random isolation trees, expected path length $h(\boldsymbol{x})$ to isolate object $\boldsymbol{x}$ is computed using the mean of the path lengths required to isolate the point using each generated tree. Finally, the anomaly score was calculated as follows (Liu et al., 2008):

$$S(\boldsymbol{x}) = 2^{-\frac{E[h(x)]}{c(\psi)}}, \tag{1}$$

where: $c(n)$ is the average value of $h(\boldsymbol{x})$ for a dataset of size n, and this value can be computed as $c(n) = 2H(n-1) - \frac{2(n-1)}{n}$, in which $H(n)$ is the harmonic number (it can be estimated as $H(n) = \ln(n) + \gamma$ with $\gamma \approx 0.557$ is the Euler-Mascheroni constant.

If $E[h(\boldsymbol{x})] = c(n)$ the anomaly score of $\boldsymbol{x}$ is $s(\boldsymbol{x}, n)$ = 0.5. When $h(\boldsymbol{x}) \to +\infty$, for the points that are not outliers, the anomaly score tends to 0. When $h(\boldsymbol{x})$ is very small compared to $c(n)$, which is the case for outliers, the anomaly score tends to 1.

Thus, the anomaly threshold is $s_0 \in [0; 1]$ and can be defined in such a way that $\boldsymbol{x}$ is treated as an anomaly when $s(\boldsymbol{x}) > s_0$ and is a non-outlier in other cases. Liu et al. (2008) proposed the following parameters for an isolation forest:

- number of trees $(t) = 100$,
- sub-sample size $(\psi) = 256$,
- maximum depth of a tree $(l) = \text{ceil}(\log_2 \psi) = 8$,
- anomaly detection threshold $s_0 = 0.6$.

An isolation forest for classical data is a good tool for outliers' detection as it does not need any assumptions on the data distribution, number of anomalies in the dataset, and is computationally efficient. Nevertheless, the isolation forest algorithm can suffer from a bias due to the way the trees are created. To avoid problems of the isolation forest, Hariri et al. (2019) proposed the extended isolation forest (EIF). This approach allows the branching hyperplanes to take on any slope, as opposed

to hyperplanes which are only parallel to the coordinate frame. This extension in the algorithm completely resolves the bias introduced in the case of standard isolation forest. Lesouple et al. (2021) suggested another extension of the isolation forests, called the generalised isolation forest (GIF), namely to project all the data on the sampled normal unit vector, look for the minimum and maximum values of the projections, and to sample a split value uniformly between these two values.

3. Experiments with Outliers

To check how the proposed modification is dealing with outliers, artificial datasets with a known number of clusters and known number, or share, of outliers were generated with the `cluster.Gen` function from the `clusterSim` package (see Walesiak and Dudek 2023 for details). In the `cluster.Gen` function, to obtain symbolic interval data, the data were generated for each model twice into sets *A* and *B* and minimum (maximum) value of is treated as the beginning (the end) of an interval.

The outliers were generated independently for each variable for the whole data set from uniform distribution (the default range was [1, 10]). The generated values were randomly added to the maximum of *j*-th variable or subtracted from the minimum of *j*-th variable. Table 4 presents the details for each artificial dataset.

Table 4. Parameters for artificial datasets

Model	Clusters and variables	Means	Covariance matrix Σ	Size and outlier size or share
1	three elongated clusters in three dimensions	(1.5, 6, –3), (3, 12, –6), (4.5, 18, –9)	$\sigma_{jj} = 1 (1 \leq j \leq 3), \sigma_{12} = \sigma_{13} = -0.9$ and $\sigma_{23} = 0.9$	180 non-outliers 10 outliers
2	three elongated clusters in twodimensions	(0, 0), (1.5, 7), (3, 14)	$\sigma_{jj} = 1$ $\sigma_{jl} = -0.9$	150 non-outliers 10 outliers
3	five clusters in two dimensions that are not well separated	(5, 5), (–3, 3), (3, –3), (0, 0), (–5, – 5)	$\sigma_{jj} = 1$ $\sigma_{jl} = 0.9$	788 objects with 5% outliers
4	four clusters in three dimensions	(–4, 5, –4), (5, 14, 5), (14, 5, 14), (5, –4, 5),	$\sigma_{jj} = 1$ $(1 \leq j \leq 3)$ $\sigma_{jl} = 0$ $(1 \leq j \neq l \leq 3)$	280 non-outliers 20 outliers
5	four clusters in two dimensions	smiley dataset from `mlbench` package of R software[1] (the smiley consists of 2 Gaussian eyes, a trapezoid nose and a parabola mouth with vertical Gaussian noise)		500 non-outliers 10 outliers

Source: own elaboration.

Figure 1 shows the interval-valued plots for all the datasets with outliers. In each model the outliers are presented as navy rectangles, whereas the clusters are denoted by green, yellow, red, and light blue.

[1] In order to generate symbolic interval-valued data, initial smiley data points were generated twice and a minimum and maximum values were selected as the lower and upper bounds of symbolic interval-valued variable.

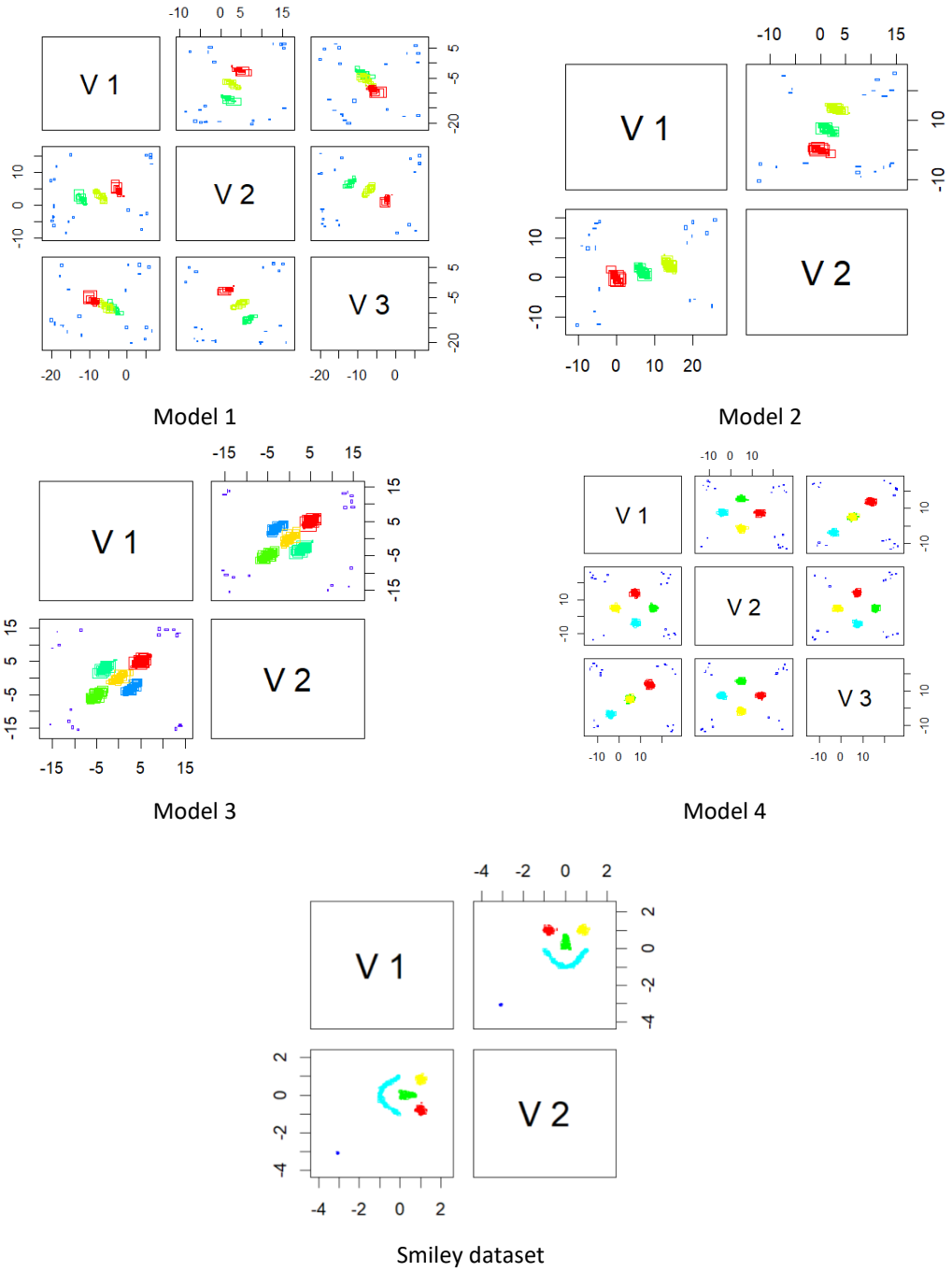

Fig. 1. All datasets with outliers

Source: own elaboration with R software.

4. Results

For each model, 20 simulation runs were performed and the average value of the adjusted Rand index for outliers and the number of outliers were obtained. The same simulations were carried out for the DBSCAN algorithm, which is capable of outlier detection (see e.g. Aggarwal, 2017; Hahsler et al., 2019; Schubert et al., 2017). Table 5 presents the simulation results.

Table 5. Simulation results

Model	Size and outlier size or share	Methods	Number of outliers	Average adjusted rand index	Calculation time for 20 loops
1	180 non-outliers 10 outliers	isolation forest	**10**	**1**	**0.245647 s**
		DBSCAN	29.3	0.4188352	0.1307719 s
2	150 non-outliers 10 outliers	isolation forest	**10.05**	**0.9971117**	**0.2196631 s**
		DBSCAN	19.4	0.6126875	0.1015229 s
3	368 objects with 5% outliers	isolation forest	**17.95 (5.128%)**	**0.9983548**	**0.5078101 s**
		DBSCAN	29.2 (7.935%)	0.8340508	0.156868 s
4	280 non-outliers 20 outliers	isolation forest	**19.95**	**0.9984436**	**0.6292601 s**
		DBSCAN	179	-0.02471935	0.16311 s
5	500 non-outliers 10 outliers	isolation forest	**10.06**	**0.9998371**	**0.5702689 s**
		DBSCAN	3.5	0.5921260	0.6023978 s

Source: own elaboration.

Both the computational times and the number of outliers found appeared to be promising. The isolation forest for symbolic data was able to detect outliers in each case (in most cases the number of outliers was correct). In each case the isolation forest achieved better results than the well-known DBSCAN algorithm.

What is more important, is that the adjusted Rand index was very high. It can be even exactly at 1 if the clusters in the data are well separated and there are not too many outliers. Where the clusters are not too well separated, the isolation forest is also performing well as far as the adjusted Rand index is concerned. As with all the other data analysis methods, there are key elements that must be set in advance – in this case the number of trees (isolation trees), sample size, maximum depth of a tree and anomaly threshold.

5. Discussion and Conclusions

Isolation forests can be quite easily adapted for symbolic data cases, and they proved to be an effective tool for outlier mining when dealing with artificial sets with a known data structure. Further analysis should include also real data sets with outliers (i.e. credit fraud detection data) and comparisons with other methods that can be applied for symbolic data cases (i.e. decision trees, neural networks and k nearest-neighbour method for symbolic data).

The computational times were promising for all the datasets, and all the cluster shapes' computational times for 20 loops were very similar (usually it took around 6 seconds to compute the final results). The proposed algorithm has the same drawbacks as the isolation forests for classical data; if there are no outliers in the dataset, some objects could be labelled as outliers. The isolation forest achieved better results in terms of outlier detection, when no outliers were present while building the initial isolation forest and its trees. However, if outliers were present in the data when preparing the isolation forest, the final results were slightly worse.

The only problem for the isolation forest was the selection of the critical value for the anomaly score. In this paper the authors used the same idea proposed for classical data, namely if the anomaly score was greater than 0.6, the object was an anomaly, and if not – the object was considered to be a non--outlier.

References

Aggarwal, C. C., and Yu, P. S. (2005). An Effective and Efficient Algorithm for High-Dimensional Outlier Detection. *The VLDB Journal*, *14*, 211-221.

Aggarwal, C. (2017). *Outlier Analysis*. Springer.

Aguinis, H., Gottfredson, R. K., and Joo, H. (2013). Best-practice Recommendations for Defining, Identifying, and Handling Outliers. *Organizational Research Methods*, *16*(2), 270-301.

Anscombe, F. J., and Guttman, I. (1960). Rejection of Outliers. *Technometrics, 2*(2), 123-147.

Ayadi, A., Ghorbel, O., Obeid, A. M., and Abid, M. (2017). Outlier Detection Approaches for Wireless Sensor Networks: A Survey. *Comput. Netw.*, (129), 319-333.

Barnett, V., and Lewis, T. (1994). *Outliers in Statistical Data* (vol. 3, no. 1). Wiley.

Bock, H.-H., and Diday, E. (eds.) (2000). *Analysis of Symbolic Data. Explanatory Methods for Extracting Statistical Information from Complex Data.* Springer Verlag.

Billard, L., and Diday, E. (2006). *Symbolic Data Analysis. Conceptual Statistics and Data Mining*. John Wiley & Sons.

Branch, J. W., Giannella, C., Szymanski, B., Wolff, R., and Kargupta, H. (2013). In-network Outlier Detection in Wireless Sensor Networks. *Knowledge and Information Systems*, *34*, 23-54.

Breunig, M. M., Kriegel, H. P., Ng, R. T., and Sander, J. (2000, May). *LOF: Identifying Density-Based Local Outliers* (Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, pp. 93-104).

Brito, P., and Dias, S. (Eds.). (2022). *Analysis of distributional data*. CRC Press.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly Detection: A Survey. *ACM Comput. Surv*., (41), 15:1-15:58

Cheng, T., and Li, Z. (2006). A Multiscale Approach for Spatio-Temporal Outlier Detection. *Transactions in GIS*, *10*(2), 253-263.

Hawkins, D. M. (1980). *Identification of Outliers* (Vol. 11). Chapman and Hall.

Hahsler, M., Piekenbrock, M., and Doran, D. (2019). dbscan: Fast Density-based Clustering with R. *Journal of Statistical Software*, (91), 1-30.

Ghosh, D., and Vogt, A. (2012). Outliers: An Evaluation of Methodologies. *Joint Statistical Meetings*, *12*(1), 3455-3460.

Grubbs, F. E. (1969). Procedures for Detecting Outlying Observations in Samples. *Technometrics*, *11*(1), 1-21.

Gatnar E., and Walesiak M. (Ed.). (2011). *Analiza danych jakościowych i symbolicznych z wykorzystaniem programu R*. C.H. Beck.

Hariri, S., Kind, M. C., and Brunner, R. J. (2019). Extended Isolation Forest. *IEEE Transactions on Knowledge and Data Engineering*, *33*(4), 1479-1489.

Hawkins, D. M. (1980). *Identification of Outliers* (Vol. 11). London: Chapman and Hall.

Hawkins, S., He, H., Williams, G., & Baxter, R. (2002). Outlier Detection Using Replicator Neural Networks. In: *Data Warehousing and Knowledge Discovery: 4th International Conference, DaWaK 2002 Aix-en-Provence, France, September 4-6, 2002 Proceedings 4* (pp. 170-180). Springer Berlin Heidelberg.

Hu, T., and Sung, S. Y. (2003). Detecting Pattern-based Outliers. *Pattern Recognition Letters*, *24*(16), 3059-3068.

Jiang, M. F., Tseng, S. S., and Su, C. M. (2001). Two-phase Clustering Process for Outliers Detection. *Pattern Recognition Letters*, *22*(6-7), 691-700.

Keller, F., Muller, E., and Bohm, K. (2012, April). *HiCS: High Contrast Subspaces for Density-Based Outlier Ranking* (2012 IEEE 28th International Conference on Data Engineering, pp. 1037-1048). IEEE.

Lazarevic, A., and Kumar, V. (2005, August). *Feature bagging for Outlier Detection* (Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 157-166).

Lesouple, J., Baudoin, C., Spigai, M., and Tourneret, J. Y. (2021). Generalized Isolation Forest for Anomaly Detection. *Pattern Recognition Letters*, *149*, 109-119.

Liu, F. T., Ting, K. M., and Zhou, Z.-H. (2008). *Isolation Forest* (*2008 Eighth IEEE International Conference on Data Mining*, Pisa, Italy, 2008, pp. 413-422). doi: 10.1109/ICDM.2008.17.

Micenková, B., McWilliams, B., and Assent, I. (2015). *Learning Representations for Outlier Detection on a Budget*. arXiv preprint arXiv:1507.08104.

Muthukrishnan, S., Shah, R., and Vitter, J. S. (2004, June). *Mining Deviants in Time Series Data Streams* (Proceedings. 16th International Conference on Scientific and Statistical Database Management, pp. 41-50). IEEE.

Rayana, S., Zhong, W., and Akoglu, L. (2016, December). *Sequential Ensemble Learning for Outlier Detection: A Bias-Variance Perspective* (2016 IEEE 16th International Conference on Data Mining (ICDM), pp. 1167-1172). IEEE.

Singh, K., and Upadhyaya, S. (2012). Outlier Detection: Applications and Techniques. *International Journal of Computer Science Issues (IJCSI)*, *9*(1), 307.

Sadik, S., and Gruenwald, L. (2011, September). *Online Outlier Detection for Data Streams* (Proceedings of the 15th Symposium on International Database Engineering & Applications, pp. 88-96).

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., and Xu, X. (2017). DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Transactions on Database Systems*, *42*(3), 1-21.

Smiti, A. (2020). A Critical Overview of Outlier Detection Methods. *Computer Science Review*, *38*(100306).

Thang, T. M., and Kim, J. (2011, April). *The Anomaly Detection by Using Dbscan Clustering with Multiple Parameters* (2011 International Conference on Information Science and Applications, pp. 1-5). IEEE.

Zhao, Y., and Hryniewicki, M. K. (2018, July). *Xgbod: Improving Supervised Outlier Detection with Unsupervised Representation Learning* (2018 International Joint Conference on Neural Networks (IJCNN), pp. 1-8). IEEE.

Walesiak, M., and Dudek, A. (2023). *clusterSim: Searching for Optimal Clustering Procedure for a Data Set*. Retrieved from www.r-project.org

Wang, H., Bah, M. J., and Hammad, M. (2019). Progress in Outlier Detection Techniques: A Survey. *Ieee Access*, *7*, 107964--108000.

Lasy separujące dla danych symbolicznych jako narzędzie wykrywania obserwacji odstających

Streszczenie

Cel: Identyfikacja obserwacji odstających stanowi kluczowy element w analizie danych. Pomimo że w literaturze funkcjonuje wiele różnych definicji, czym są obserwacje odstające, to ogólnie można stwierdzić, że są to obiekty różniące się od pozostałych obserwacji ze zbioru danych. Literatura przedmiotu wskazuje wiele różnorodnych metod, które można wykorzystać w przypadku danych klasycznych. Niestety w przypadku danych symbolicznych brakuje takich analiz. Celem artykułu jest zaproponowanie modyfikacji lasów separujących (*isolation forests*) dla danych symbolicznych.

Metodyka: W artykule wykorzystano lasy separujące dla danych symbolicznych do identyfikacji obserwacji odstających w sztucznych zbiorach danych o znanej strukturze klas i znanej liczbie obserwacji odstających.

Wyniki: Otrzymane wyniki wskazują, że lasy separujące dla danych symbolicznych są efektywnym i szybkim narzędziem w identyfikacji obserwacji odstających.

Implikacje i rekomendacje: Ponieważ lasy separujące dla danych symbolicznych okazały się skutecznym narzędziem w identyfikacji obserwacji odstających, celem przyszłych badań powinno być przeanalizowanie skuteczności tej metody w przypadku rzeczywistych zbiorów danych (np. zbioru dotyczącego oszustw z użyciem kart kredytowych), a także porównanie tej metody z innymi metodami, które pozwalają odnaleźć obserwacje odstające (np. DBSCAN). Autorzy sugerują, by w przypadku lasów separujących dla danych symbolicznych stosować te same parametry, jakie zwykle stosuje się w przypadku lasów losowych dla danych klasycznych.

Oryginalność/wartość: Artykuł nie tylko stanowi ujęcie teorii w zakresie obserwacji odstających, ale jednocześnie proponuje, jak zastosować lasy separujące w przypadku danych symbolicznych.

Słowa kluczowe: analiza danych symbolicznych, lasy separujące, obserwacje odstające