**DOI: 10.5604/01.3001.0010.5216**

# ALTERNATIVE TERMINATION CRITERION FOR *K*-SPECIFIED CRISP DATA CLUSTERING ALGORITHMS

**Volodymyr Mosorov, Taras Panskyi, Sebastian Biedron**
Lodz University of Technology, Institute of Applied Computer Science

***Abstract.*** *In this paper the analysis of k-specified (namely k-means) crisp data partitioning pre-clustering algorithm's termination criterion performance is described. The results have been analyzed using the clustering validity indices. Termination criterion allows analyzing data with any number of clusters. Moreover, introduced criterion in contrast to the known validity indices enables to analyze data that make up one cluster.*

**Keywords**: pre-clustering algorithm, internal validity measures

# ALTERNATYWNY KRYTERIUM ZATRZYMANIA DLA *K*-OKREŚLONYCH TWARDYCH ALGORYTMÓW KLASTERYZACJI DANYCH

***Streszczenie.*** *W przedstawionym artykule została pokazana analiza wstępnej klasteryzacji danych w oparciu o partycjonowanie (algorytm k-średnich) w połączeniu z logiką dwuwartościową. Dodatkowo, zostało przedstawione kryterium zatrzymania klasteryzacji, które umożliwia analizowanie danych z dowolną liczbą klastrów. Otrzymane wyniki badań zostały przeanalizowane przy użyciu wewnętrznych indeksów walidacji. Wprowadzone kryterium w przeciwieństwie do znanych indeksów walidacji umożliwia analizę danych, które tworzą jeden klaster.*

**Słowa kluczowe**: algorytm wstępnej klasteryzacji, wewnętrzny indeksy walidacji

## Introduction and related work

Clustering refers to the process of the partition of a data set of objects into groups (clusters) so that the objects within a particular cluster have high similarity to each others, but are very dissimilar to objects in other clusters. Clustering methods have been classified into four types [15]: partitioning clustering, hierarchical clustering, density-based clustering and grid based clustering. Thus, basing on the relationship of each object to the cluster, we can distinguish *crisp* vs. *fuzzy* clustering.

The most fundamental version of cluster analysis is partitioning, which organizes objects of a data set into several mutually exclusive (no point in the data set belongs to more than one cluster), or jointly exhaustive (every point belongs to some cluster associated with other objects based on the membership levels) groups. This approach usually requires some background knowledge, namely an input parameter (number of clusters) as a starting point of a partitioning process. In the case of some partitioning algorithms (*k*-means, *k*-medoids, etc.) the user-defined initial parameter *k* (number of clusters) is simultaneously the stopping criterion of clustering performance.

Stopping criteria for optimal clustering have been a topic of discussion during the last decades and which caused an increase in research to confirm their usefulness [5, 8]. For partitioning clustering methods the stopping criteria are based on the predefined threshold or termination criterion including number of iteration, number of clusters, etc.

In order to quantify clustering optimality the procedure of estimating the results of clustering algorithm (cluster validity) has been used. In the case of partitioning clustering the only way to omit the strong user's influence on the clustering result is to use a pre-processing step (pre-clustering) or a post-processing (result validation). As a consequence, the resulting clustering configuration should be performed without a-priori understanding of the internal structure of data, but on the other hand it requires some sort of estimation related to its validity.

The distinctive feature of clustering is finding a structure in the investigated data, but its disadvantage is the introduction of an additional redundant structure into these data. Clustering allows finding structures even in the data which do not have it a priori (overclustering), which leads to the appearance of artifacts, that is, erratic results of cluster finding. In this case for finding the "best" number of clusters the pre-clustering is used. The most known pre-clustering algorithm is a canopy clustering algorithm [10]. The aim of this algorithm is finding the approximate number of clusters which make up the input information for further clustering algorithms. The disadvantage of this algorithm is a heuristic definition of two thresholds (distances $T_1$ and $T_2$). The only logical solution to the problem of receiving valid results and at the same time of elimination the user's influence on clustering results is the use of clustering validity indices.

In [1, 16], three approaches to investigation of cluster validity are described. The first one is based on *external criteria,* which consist in comparing the results of cluster analysis to externally known results, such as externally provided class labels. The second approach is based on *internal criteria*, and serves to estimate the goodness of clustering results without reference to external information. The third approach (*relative criteria*) is based on the estimation of the clustering structure by comparing different input parameter values for the same algorithm, e.g., the number of clusters. Most of the validity indices require statistical sequential substitution of the input parameter and are based on finding the "best" index value. Different indices in different situations cause different results. However, this paper is focused on the mixed sample of *k*-specified data partitioning clustering indices proposed for the comparison purpose criterion for *k*-specified data partitioning clustering algorithms of the termination criterion of clustering performance. The termination criterion helps to perform partitioning up to the certain step for the optimal determination of the number of clusters and gives the chance to keep an important balance between underclustering and overclustering.

## 1. Termination criterion for the pre-clustering algorithm

The pre-clustering algorithm as opposed to other existing algorithms does not require input parameters or threshold values for the correct determination of the number of clusters. Pre-clustering is the procedure of checking the possibility of input data clustering. The published pre-clustering algorithm [12] and its main part – the decision rule – determines the existence of *one* or *two* clusters in the input data set. The decision rule has been implemented in the termination criterion [11] for the determination of *any* number of clusters.

In the following pre-clustering algorithm and its termination criterion we denote that:
$n$ is a number of objects,
$p$ is a number of attributes,
$k$ is a number of clusters,
$X = \{x_i, i=1,2,\ldots,n\}$ stands for data set containing $n$ objects, in a $p$-dimensional space,
$K_q$ is a sequential number of cluster, where $q = 1,2,...,k$,
$x_i^{(q)}$ is the $i$-th object of $K_q$ cluster.

**Algorithm:** The pre-clustering algorithm, with the termination criterion, where partitioning is based on the crisp *k*-means clustering.

**Input**: *X*: a data set containing *n* objects with *p* attributes.

**Output**: Number of clusters *k* in the form of an acyclic connected graph.

**Method**:

(1) assign the input data set to the general cluster *K*;

**repeat**

(2) perform *k*-means clustering (always with *k*=2), where general cluster *K* is partitioned into two pre-clusters $K_1$ and $K_2$. The pre-cluster is a group of objects which is not a single cluster, but can become one after checking;

(3) check the possibility of clusters existence with the use of the decision rule that is checking if two pre-clusters $K_1$ and $K_2$ are separate clusters. If $K_1$ and $K_2$ are separate clusters – continue checking – step (4), otherwise stop partitioning – step (6);

(4) reassign each cluster, found at the previous step to the general cluster. Split each cluster again with *k*-means algorithm (nota bene *k* = 2) and check the results of the partitioning by the decision rule – step (3);

(5) continue cluster partitioning for checking the possibility of the existence of a smaller separate cluster;

**until** all pre-clusters in every partition step should be analyzed. All pre-clusters should be checked by the decision rule;

(6) count the number of clusters, using the acyclic connected graph.

The advantage of the pre-clustering algorithm is that it does not require setting the initial parameter (number of clusters). The pre-clustering algorithm based on the application of crisp partitioning algorithms, in this case *k*-means. However, the *k*-means algorithm can be replaced by any other crisp partitioning algorithm. It should also be noted that the input parameter for partitioning (*k* – number of clusters) is not set by the user but at every step of partitioning it is set automatically on default being equal to the value *k* = 2.

## 2. Numerical results of pre-clustering algorithm validation

In this section the characteristics of data set are described. Thereafter, the validation of termination criterion of pre-clustering algorithm is presented.

*Artificial #1*: two-attribute data set containing 100 objects with Gaussian distribution, where all data objects make up one globular group.

*Artificial #2*: data set is similar to the previous one, but distinguished by the presence of three well separated groups at the equal distance from each other.

*Artificial #3*: data set is based on longitudinal distribution of objects in an elongated group.

*Iris*: all known four-attribute data set, where each group/cluster refers to the length and the width of the sepals and petals of iris flower.

*Artificial #4*: artificial two-attribute data set containing 100 objects generated with normal distribution and with three well separated globular form clusters.

*Artificial #5*: artificial two-attribute data set containing 500 objects in the form of three concentric ring clusters. Three classes labeled as “core”, “first ring” and “second ring”, accordingly.

Tested data sets are shown in Figure 1.

In this paper for the purpose of termination criterion validation, the internal validity measures [3] (Davies–Bouldin index, the Dunn index, index called “silhouette statistic”, average within cluster distance and cluster density) are used.

The Dunn [4] index defines the ratio between the minimal intracluster distance to maximal intercluster distance. The Dunn index is limited to the interval [0,∞] and should be maximized.

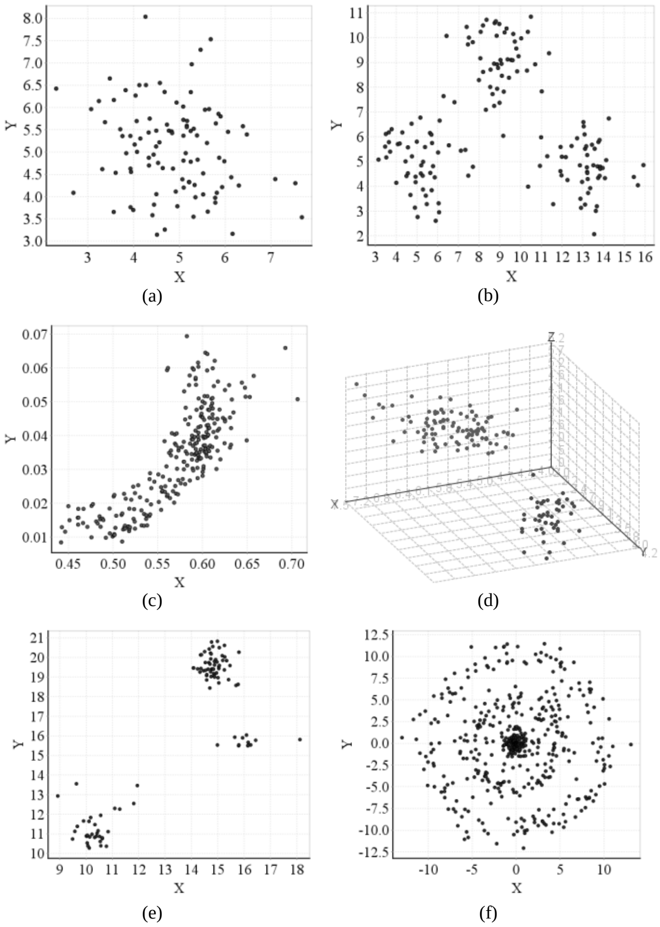

*Fig. 1. Artificial (a, b, c, e, f) two-attribute data sets, (d) real-life iris data set that contains 150 objects and three classes of iris*

Rousseeuw [14] introduced the Silhouette index. The maximum value of the index is used to determine the optimal number of clusters in the data. Silhouette index is not defined for *k* = 1 (only one cluster).

The average within cluster distance [13] is calculated by averaging the distance between the centroid and all examples of a cluster. As clusters get more compact, this measure reduces. Of course, as the number of clusters increases, the average distance will decrease naturally anyway and so this measure can be difficult to interpret.

Cluster density measure [7] considers each cluster in turn and finds the average of the distances between all the pairs of points in the cluster and multiplies by the number of points in the cluster. This results in a measure that is equivalent to a distance per point within the cluster and which is, therefore, similar to a density. This measure tends to zero as the number of clusters increases, but smaller values indicate more compact clusters.

The Gini index [6] for measuring class inequality is also used as a validation index. A Gini coefficient of 1 (or 100%) expresses maximal inequality among values.

Simulations were carried out on the basis of RapidMiner software. The scheme of the validation process of pre-clustering algorithm is shown in Figure 2. Operators *Data Set* generate the artificial data whilst *Iris* data set is read from the RapidMiner samples repository. It is straightforward to connect the input of Loop operator. Operator *Loop Parameters* generate clusters that makes multiple partitioned clusters with *k* = 1 up to a maximum number of clusters defined by the user (*k* = 6). The measure type is set to numerical Euclidean distance. The *Log* operator is a very important part of RapidMiner as it allows data to be recorded during the execution. The values returned in the log are converted to real values, where necessary, to make analysis easier later on.

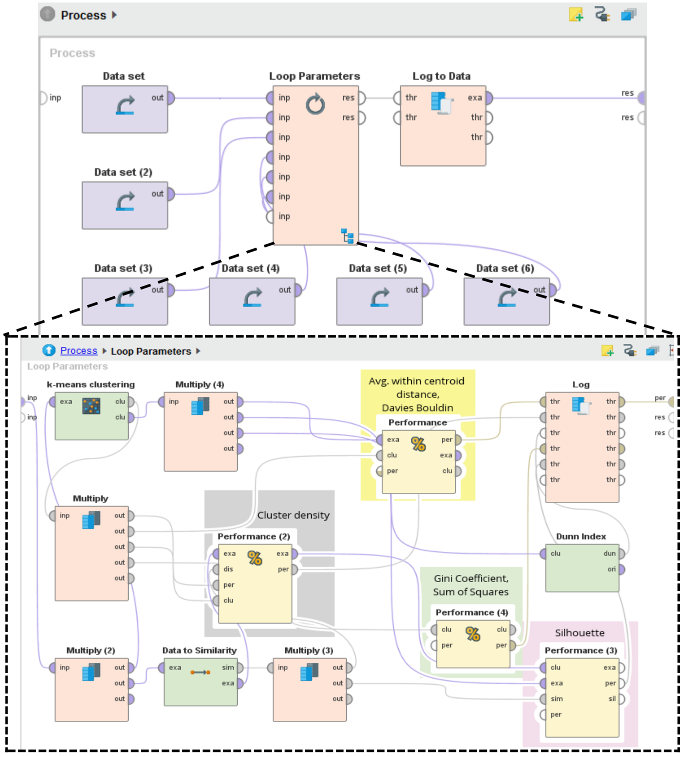

*Fig. 2. Scheme of the validation of pre-clustering algorithm*

The results of the validation process of *Iris* data set using the pre-clustering algorithm based on the crisp *k*-means algorithm are shown in Figure 3.

The graph presented in Figure 3 shows how internal validity measures vary as different clusterings are compared. All of the validity measures together indicate that $k = 2$ is a strong candidate for the best clustering. This is encouraging since in this case, the Dunn index was not told the correct result.

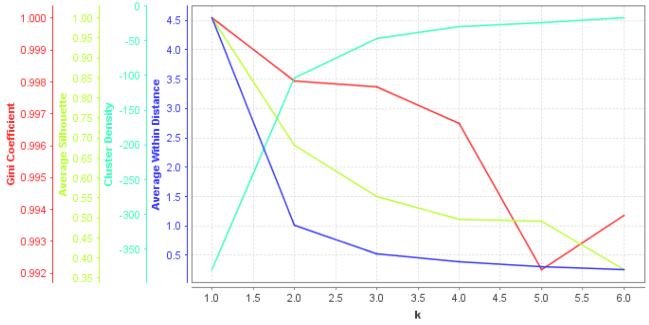

*Fig. 3. Internal validity measures as a function of k for Iris data set. The x axis is the value of k and the "best" number of clusters is estimated using the elbow method. This graph was produced using the Series Multiple plotter and consequently, the y axes are normalized to make the ranges of each series match*

The idea of the elbow method [9] is to choose the *k* at which the validity of indices decreases or increases abruptly. This produces an "elbow effect" in the graph. The number of clusters is chosen at this point, hence defined as "elbow criterion". The Elbow method is a heuristic and, as such, it may or may not work well in particular case. Sometimes, there is more than one elbow, or no elbow at all. In those situations user usually end up calculating the best *k* by evaluating how well partitioning algorithm performs clustering.

*Table 1. The "best" number of cluster is determined from the labeled with a red color validity index*

Artificial #1 data set

| k | D.B. | Dunn | Sil. | Av.D. | Cl.D. | Gini | $k_p$ |
|---|---|---|---|---|---|---|---|
| 1 | x | x | x | 1.91 | -172 | 1 | 1 |
| 2 | 1.07 | 0.033 | 0.35 | 1.17 | -68 | 0.998 | 2 |
| 3 | 0.94 | 0.06 | 0.35 | 0.83 | -36 | 0.999 | 3 |
| 4 | 0.91 | 0.06 | 0.33 | 0.63 | -24 | 0.997 | 4 |
| 5 | 0.88 | 0.059 | 0.34 | 0.52 | -18 | 0.991 | 5 |
| 6 | 0.77 | 0.088 | 0.36 | 0.43 | -15 | 0.984 | 6 |

Artificial #2 data set

| k | D.B. | Dunn | Sil. | Av.D. | Cl.D. | Gini | $k_p$ |
|---|---|---|---|---|---|---|---|
| 1 | x | x | x | 25.2 | -922 | 1 | 1 |
| 2 | 0.62 | 0.257 | 0.57 | 10.6 | -331 | 0.997 | 2 |
| 3 | 0.34 | 0.595 | 0.75 | 1.95 | -87.2 | 1.0 | 3 |
| 4 | 0.76 | 0.042 | 0.60 | 1.71 | -69.2 | 0.995 | 4 |
| 5 | 1 | 0.042 | 0.46 | 1.48 | -51.8 | 0.993 | 5 |
| 6 | 0.84 | 0.037 | 0.48 | 1.32 | -47.6 | 0.989 | 6 |

Artificial #3 data set

| k | D.B. | Dunn | Sil. | Av.D. | Cl.D. | Gini | $k_p$ |
|---|---|---|---|---|---|---|---|
| 1 | x | x | x | 0.003 | -14 | 1 | 1 |
| 2 | 0.48 | 0.03 | 0.64 | 0.001 | -4.5 | 0.998 | 2 |
| 3 | 0.62 | 0.033 | 0.51 | 0 | 2.4 | 0.997 | 3 |
| 4 | 0.74 | 0.022 | 0.40 | 0 | 1.5 | 0.996 | 4 |
| 5 | 0.64 | 0.038 | 0.48 | 0 | 1.2 | 0.993 | 5 |
| 6 | 0.72 | 0.025 | 0.41 | 0 | 0.8 | 0.992 | 6 |

Iris data set

| k | D.B. | Dunn | Sil. | Av.D. | Cl.D. | Gini | $k_p$ |
|---|---|---|---|---|---|---|---|
| 1 | x | x | x | 4.53 | -379 | 1 | 1 |
| 2 | 0.40 | 0.076 | 0.68 | 1.01 | -103 | 0.998 | 2 |
| 3 | 0.66 | 0.098 | 0.55 | 0.52 | -46 | 0.998 | 3 |
| 4 | 0.77 | 0.136 | 0.49 | 0.38 | -30 | 0.996 | 4 |
| 5 | 0.81 | 0.082 | 0.49 | 0.31 | -24 | 0.992 | 5 |
| 6 | 0.92 | 0.085 | 0.36 | 0.25 | -17 | 0.994 | 6 |

Artificial #4 data set

| k | D.B. | Dunn | Sil. | Av.D. | Cl.D. | Gini | $k_p$ |
|---|---|---|---|---|---|---|---|
| 1 | x | x | x | 20 | -495 | 1 | 1 |
| 2 | 0.24 | 0.61 | 0.80 | 2.34 | -94 | 0.996 | 2 |
| 3 | 0.23 | 0.69 | 0.80 | 0.67 | -39 | 0.991 | 3 |
| 4 | 0.56 | 0.06 | 0.71 | 0.488 | -31 | 0.984 | 4 |
| 5 | 0.75 | 0.02 | 0.50 | 0.367 | -15 | 0.986 | 5 |
| 6 | 0.78 | 0.03 | 0.45 | 0.318 | -13 | 0.980 | 6 |

Artificial #5 data set

| k | D.B. | Dunn | Sil. | Av.D. | Cl.D. | Gini | $k_p$ |
|---|---|---|---|---|---|---|---|
| 1 | x | x | x | 43.4 | -4032 | 1 | 1 |
| 2 | 1.25 | 0.03 | 0.36 | 30.51 | -1881 | 0.999 | 2 |
| 3 | 1 | 0.22 | 0.38 | 21.37 | -1145 | 0.998 | 3 |
| 4 | 0.89 | 0.01 | 0.41 | 15.8 | -578 | 0.998 | 4 |
| 5 | 0.79 | 0.01 | 0.44 | 11.71 | -415 | 0.997 | 5 |
| 6 | 0.80 | 0.02 | 0.46 | 9.273 | -317 | 0.996 | 6 |

Validation results can also be displayed in numerical form (see Table 1), where best indices performance and accordingly the number of possible clusters is labeled with a red color. The determination of $k_p$ causes finding a number of clusters using the pre-clustering algorithm with the termination criterion. Due to the limitation on the article size, additional metrics (accuracy, classification error, f-measure) as well as the results of pre-clustering algorithm based on other crisp partitioning algorithms (*k*-medoid, Kernel *k*-means, etc.) cannot be represented.

Also the external validity measures that compare clusters that are previously known with the clusters produced by the clustering algorithm are not presented. In this paper the data is presented in visual form in 2 or 3 dimensional space, however external validity measures (Rand, Jaccard, Fowlkes-Mallow and adjusted Rand indexes) could be used as ground truth to refer to the known clusters.

## Conclusion

Briefly summarizing, the pre-clustering algorithm with the termination criterion is a good alternative for well-known clustering validity indices. Its considerable advantage is the ability to analyze data that make up one cluster. This pre-clustering algorithm has its disadvantages. One of them is the dependence of the parameters on calculated distances. When objects are significantly scattered, there are possibilities for existing anomalies or isolated clusters and, accordingly, the difficulties in obtaining adequate results, which can be seen in Table 1, from *Artificial* #5 data set.

## References

[1] Charrad M., Ghazzali N., Boiteau V., Niknafs A.: NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set. Journal of Statistic Software, 61(6), 2014, 1–36.

[2] Davies D.L., Bouldin D.W.: A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1, no. 2, 1979, 224–227.

[3] Desgraupes B.: Clustering indices. University Paris Ouest, Lab Modal'X, 2013.

[4] Dunn J.: Well separated clusters and optimal fuzzy partitions. Journal of Cybernetics 4, 1974, 95–104.

[5] Fraley C., Raftery A.E.: How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. The Computer Journal, 41(8), 1998, 578–588.

[6] Gini, C.: Variabilita e mutabilita Reprinted in Memorie di metodologica statistica (Ed. Pizetti E, Salvemini, T). Rome: Libreria Eredi Virgilio Veschi, 1912, Bologna: Tipogr. Di P. Cuppini.

[7] Halkidi M., Batistakis Y., Vazirgiannis M.: On clustering validation techniques. J. Intell. Inf. Syst., 17(2-3), 2001, 107–145.

[8] Jung Y., Park H., Du D-Z., Drake B.L.: A Decision Criterion for the Optimal Number of Clusters in Hierarchical Clustering. Journal of Global Optimization, 25(1), 2003, 91–111

[9] Ketchen Jr. Dj, Shook Cl.: The Application Of Cluster Analysis In Strategic Management Research: An Analysis And Critique, Strategic Management Journal, 17(6), 1996, 441–458.

[10] McCallum A., Nigam K., Ungar L.H.: Efficient Clustering of High Dimensional Data Sets with Application to Reference Matching, Sixth ACM SIGKDD international conference on Knowledge discovery and data mining, 2000.

[11] Mosorov V., Panskyi T., Biedron S.: Development of a stopping rule of clustering performance by using the connected acyclic graph. Eastern-European Journal of Enterprise Technologies, 5, 9(77), 2015, 24–30.

[12] Mosorov V., Tomczak L.: Image Texture Defect Detection Method UsingFuzzy C-Means Clustering for Visual Inspection Systems. Arabian Journal for Science and Engineering, 39(4), 2014, 3013–3022.

[13] RapidMiner GmbH: Cluster distance performance – RapidMiner documentation. http://docs.rapidminer.com/studio/operators/validation/performance/segmentation/cluster_distance_performance.html

[14] Rousseeuw P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20, 1987, 53–65.

[15] Sheikholeslami C., Chatterjee S., Zhang A.: WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Database. The International Journal on Very Large Data Bases, 8(3-4), 2000, 289–304.

[16] Theodoridis S., Koutroubas K.: Pattern Recognition 4th Edition, Academic Press, 2008.

**D.Sc. Volodymyr Mosorov**
e-mail: w.mosorow@kis.p.lodz.pl

Volodymyr Mosorov received his Ph.D. in 1998 from the State University of Lviv, Ukraine. V.Mosorov was awarded the title of Doctor of Science from AGH University of Science and Technology Krakow Poland in 2009. He is now an associate professor at the Institute of Applied Computer Science of Lodz University of Technology, Poland. His research interests include data mining and clustering. He has published more than 80 technical articles.

**M.Sc. Eng. Taras Panskyi**
e-mail: tpanski@kis.p.lodz.pl

Graduated from the Department of Theoretical Radio Engineering and Radio Measurement at Lviv Polytechnic National University, Ukraine. Since 2013, he has been a Ph.D. student at the Institute of Applied Computer Science of Lodz University of Technology, Poland. His research interests include data clustering, reliability and availability indexes of embedded systems, educational migration.

**M.Sc. Eng. Sebastian Biedron**
e-mail: sbiedron@iis.p.lodz.pl

Graduated from the Department of Science and Mathematics at Lodz University. Since 2012, he has been a court expert at the District Court at the Prague. Since 2013, he has been a Ph.D. student at the Institute of Applied Computer Science of Lodz University of Technology. The supervisor of his Ph.D. thesis is Volodymyr Mosorov, D.Sc. (dr hab. inż.), prof. PL.

*otrzymano/received: 01.02.2017* *przyjęto do druku/accepted: 14.08.2017*