Baltic J. Modern Computing, Vol. 6 (2018), No. 4, 321-334
https://doi.org/10.22364/bjmc.2018.6.4.01

Multi-level Massive Data Visualization: Methodology and Use Cases

Jelena LIUTVINAVIČIENĖ, Olga KURASOVA

Vilnius University, Institute of Data Science and Digital Technologies

jelena.liutvinaviciene@mii.vu.lt, olga.kurasova@mii.vu.lt

Abstract. This research focuses on massive data visualization that is based on dimensionality reduction methods. We propose a new methodology, which divides the whole data visualization process into separate interactive steps. In each step, some part of data can be selected for further analysis and visualization. The different dimensionality method can be chosen/changed in each step. The decision which methods to be chosen depends on desirable accuracy measures and visualization samples. In addition, there are provided statistical measures of the identified clusters. We have developed a special tool, which implements the proposed methodology. R language and Shiny package were used for developing the tool. In the paper, the principles of the methodology and features of the tool are presented by describing the specific use case.

Keywords: massive data, dimensionality reduction, data visualization, data mining

1. Introduction

Big data analytics is the process of investigating big data to uncover hidden and useful information for better decisions. It involves a visual presentation of data that enables to see hidden relations between objects, which cannot be detected using conventional data analysis methods (Zubova et al., 2016). Recently this topic has been widely investigated by various researchers (Yongjie et al., 2018), (Khomtchouk et al., 2017), (Xuedi et al., 2018), (Domeniconi, 2004), (Diamond and Mattia, 2017).

Our main goal is to improve the data visual analysis process and to propose new effective ways to analyse and visualize massive data. In this paper, we present the multi-level methodology for massive data interactive visualization that is based on dimensionality reduction methods. Dimensionality reduction refers to the process of taking a data set with a usually large number of dimensions, and then creating a new data set with a fewer number of dimensions, that preserve as much of initial information as possible (Menon, 2007). In our case, we always reduce the initial number of dimensions to two (these dimensions are named by D1 and D2).

Our previous research (Zubova et al., 2018) has shown that the speed and accuracy of dimensionality reduction depend on the amount of analysed data items and the initial number of dimensions that describe them. The kind of data might also have influence. Therefore, here we present an interactive tool which allows changing various settings in different stages of visual data analysis. R language and its package Shiny were used for developing the tool.

We assume that at the beginning the computational speed is the most important factor for data visualization. In further steps of visualization process, the demand for accuracy gradually increases. This fact requires using more accurate, but possibly slower methods. During each step, the selected data set is divided into smaller clusters. In the end, the most accurate method processes the data. It would require too many resources at the beginning of dimensionality reduction, but in the end, the data set is small enough to be processed in the most accurate way. Therefore, visual samples together with accuracy measures are needed to decide which method should be applied in a particular case.

The remainder of this paper is organized as follows. In Section 2, we describe the principles of the proposed methodology in detail. Section 2 presents the use case, which reveals the features of the developed tool. Finally, conclusions are drawn in Section 4.

2. Multi-level Data Visualization Methodology

Here we propose and describe a visualization methodology, which divides the data visualization process into separate steps (Fig. 1). At each stage, a particular dimensionality reduction-based visualization method can be applied considering to data volume and type. The methods are selected according to their speed and accuracy. The more initial information dimensionality reduction method preserves, the more accurate it is. The possible accuracy measures are described in Subsection 2.2. When data are processed and visualized, there is ability to see the statistical measures of all features of each data cluster. The further analysis can be performed only for the selected data cluster.

The process of data visualization and analysis is separated into several steps:

1. First of all, an initial data set is loaded. A detailed description of this action is presented in Subsection 2.1.
2. At the initial stage, the accuracy of chosen dimensionality reduction-based visualization method is not so important, therefore, the fastest one can be used. Thus, a 2-dimensional dataset is created, and data are visualised on the 2D scatter plot.
3. The decision maker can select a part of all data items on the plot for further visualization/analysis. If requested, the selected data are visualized by different methods. Accuracy measures and descriptive information are provided as well. These tool features are described in more details in Subsection 2.3.
4. Based on the provided plots and accuracy measures, the user can choose the best method for a particular case. We assume that the deeper we go, the more accurate, but possibly slower methods might be required.
    a. If a simple zoom of the selected plot area is chosen, then the selected items can be filtered from the 2-dimensional dataset, created in the previous step. In such a case, there is no need to execute dimensionality reduction process repeatedly. The selected items are presented on the plot.
    b. If the user chooses to apply a different dimensionality reduction method or re-apply the same method again, then the selected items are filtered from the initial dataset, which contains all initial dimensions. Before this action, there is also a possibility to add new additional items to initial dataset (from the selected source file). Afterwards, the user chooses the desired method, which is applied for dimensionality

reduction. "Working" 2-dimensional dataset is updated, and data is visualised on the 2D plot.

5. If the user chooses to perform further analysis and zoom the plot, then the process continues again from Step 3.

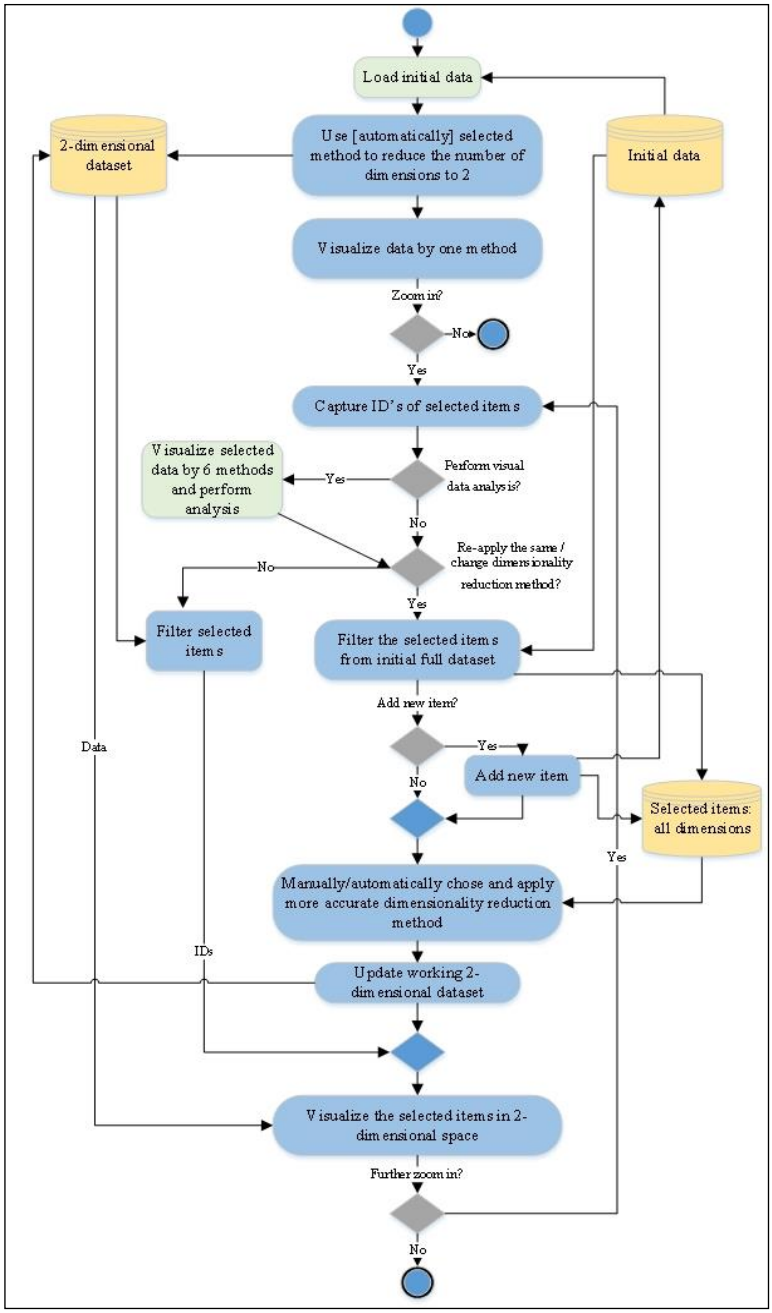

Fig. 1. Multi-level methodology for massive data visualization

2.1 Loading and analysing the initial data

The process of loading and analysing data is presented in Fig. 2.

If it is already classified data (items are assigned to a particular class), then the data file and class file (containing items assignment to a particular class) are loaded.

If it is not classified file, then only the needed data file is loaded. The desired clustering method and its parameters are chosen in the next step. The required parameters depend on which clustering method is chosen. When parameters are set, the initial data are clustered. If the results do not satisfy the user, then the parameters can be changed, and new clustering is made.

Finally, in both cases (classified data/not classified data), there is a possibility to get statistics (in tables and graphs) of the initial data.

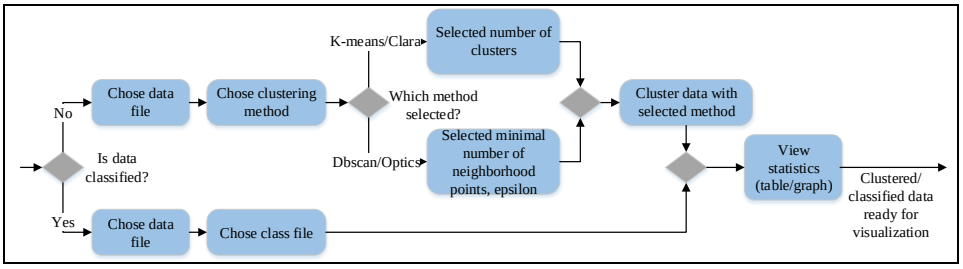

Fig. 2. Loading and analysing initial data

2.2. Data analysis

The detailed analysis of the selected items can be performed to make the proper choice of dimensionality reduction method, which will be applied in the next step of data visualization process.

We use the well-known methods for dimensionality reduction: Multidimensional Scaling (MDS), Principal Component Analysis (PCA), Independent Component Analysis (ICA), Principal Curves, Locally Linear Embedding (LLE), Isometric Mapping (Isomap) (Menon, 2007), (Domeniconi, 2004), (Fodor, 2002), (Mizuta, 2007), (Rosaria et.at., 2014) (Sorzano et al., 2014). The multidimensional data are processed by these dimensionality reduction methods, and six sets of the 2-dimensional data are obtained. They are presented in 2D scatter plots. Different methods visualize the same data in different ways. Therefore, the possibility to choose between several methods enables to find which one suits the best for particular kind of data.

The methods currently implemented in the tool are most appropriate for processing continuous data. In further stages there will be added methods applicable also for categorical data (e.g. CATPCA).

The accuracy measures (Stress, Spearman coefficient, Shannon entropy) for the results, obtained by different methods are calculated and presented as well:

- **Stress.** It shows the relative difference between distances in different spaces. It is obtained by solving the square loss function. The closer to 0 stress value is, the more accurate dimensionality reduction method is. For MDS method we use R function *mds()* from package ‘smacof’ to find the stress value (WEB, c).

To get the Stress value for other methods, the calculations by Stress formula are performed.

- **Spearman coefficient** (The Spearman's Rank Correlation Coefficient). It is a statistical measure used to discover the strength of a link between two sets of data (Hauke and Kossowski, 2011). This measure uses the ranks of variables instead of their values. Possible values range from –1 (strong negative relation) to 1 (strong positive relation). If the measure is equal to zero, this means there is no statistical link between datasets. To calculate this measure, R function cor() with method "spearman" was used.
- **Shannon entropy.** We used R function *entropy* from package 'entropy' that estimates the Shannon entropy of the random variable from the corresponding observed items (WEB, b), (Hausser and Strimmer, 2009).

Each case is individual, thus there can't be determined one best method in advance. It depends on data type, size etc. Therefore tool provides the accuracy measures of all methods. In further stages there will be added the measure of expected execution time. This will help in cases when several methods have similar accuracy measures and speed becomes a decisive factor.

The proposed methodology also enables these additional features:

- A list of the selected items together with their features is formed, and it is shown in a data table. Labels (IDs) can be placed on the selected items;
- When a particular point is selected in one graph, the corresponding point is shown (highlighted) on other graphs;
- Outlier detection is enabled. Outliers are extreme values that deviate from other observations on data. They may indicate variability in measurement, experimental errors or a novelty (Xuedi et al., 2018).

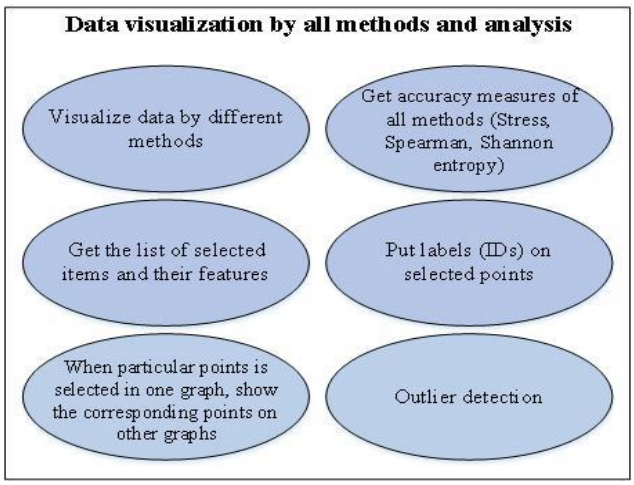

Fig. 3. Data analysis by various methods

3. Demonstration of tool prototype

The proposed multi-level large data visualization methodology has been implemented in a tool prototype. In this section, we describe a test dataset and a use case, which demonstrates how the designed tool prototype implements the features of the proposed methodology.

The tool prototype has been implemented in R language. The Shiny package was used to create an interactive user interface, therefore, the tool works as a web application (WEB, c).

3.1. Test dataset

Data objects are also called *items, instances, samples, observations*. Features are called *attributes, parameters, properties, variables, dimensions*. Objects described by the same features $x_1, x_2, \dots x_n$ form a data set. A combination of values of all features characterizes a particular object $X_i = (x_{i1}, x_{i2}, \dots, x_{in}), i \in \{1, \dots, m\}$, where n is the number of features, m is the number of objects.

If the objects are described by more than one feature, the data characterizing the objects are called *multidimensional data*. If the number of features is *n*, then $X_1, X_2, \dots, X_m$ are the *n*-dimensional data items.

If the data set consists of a lot of objects, i.e. the number m is lame enough, then the data set is called a large data set. If the number *n* is large, then the data set is called a high-dimensional data set.

We used a test dataset which contains the information about the frogs. It was previously used by others in several classifications tasks related to the challenge of anuran species recognition through their calls. This dataset was created segmenting 60 audio records belonging to 4 different families, 8 genus, and 10 species. Each audio corresponded to one specimen (an individual frog). The spectral entropy and a binary cluster method were used to detect audio frames belonging to each syllable. After the segmentation 7195 syllables were got (WEB, a). In this case, we use a smaller amount of items.

As the data are already classified (items are assigned to particular classes), there are two files:

- Data file (containing the data). It has 2610 items and 10 attributes (dimensions).
- Class file. It has 2610 items (for each corresponding item in the data file) and only one attribute that defines which particular class each item belongs to. All items are assigned to one of 4 clusters (Families). 68 items belong to the 1st cluster (Bufonidae), 542 items belong to the 2nd cluster (Dendrobatidae), 1000 items belong to the 3rd cluster (Hylidae) and 1000 items belong to the 4th cluster (Leptodactylidae).

3.2 Data loading and analysis

In the beginning, we must choose the type of data – classified / not classified. Here we present the case where not classified data is analysed, therefore, only one file (data) is loaded. At the first step, the tab for not classified data is selected. The next step requires choosing a clustering method (Fig. 4).

If *K-means* method is chosen, then it is enough to specify the number of clusters. The tool shows how many items are in each cluster: e.g. the first cluster has 606 items, the second one has 160 items, etc. (Fig. 5).

If *dbscan* method is chosen, then there is a need to specify the number of clusters, number of neighbourhoods and epsilon (Fig. 6). The epsilon value is selected according to the plot – we look for the point at which the biggest change of distances between neighbourhoods is observed. In this case, the epsilon value was set to 0.3.

In the case of *dbscan* method, there are 3 clusters: the first cluster has 2120 items, the second one has 371 items, and the third one 13.

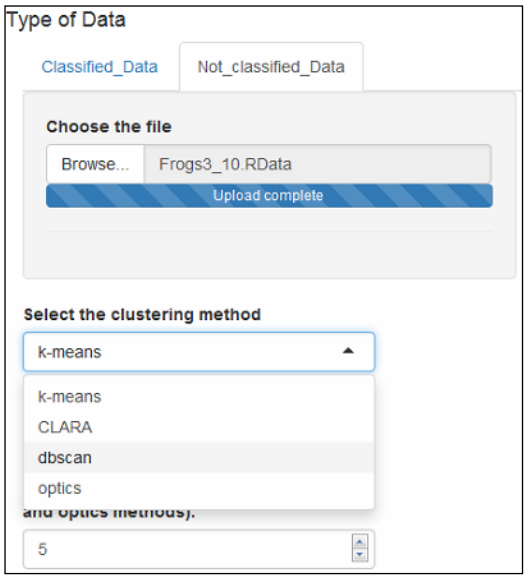

Fig. 4. Loading initial data

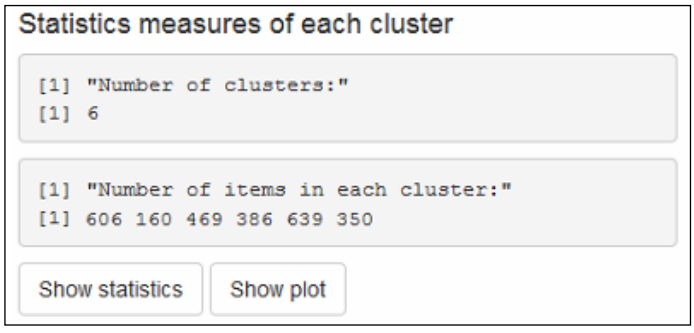

Fig. 5. Statistics of clusters obtained by *K-means* method

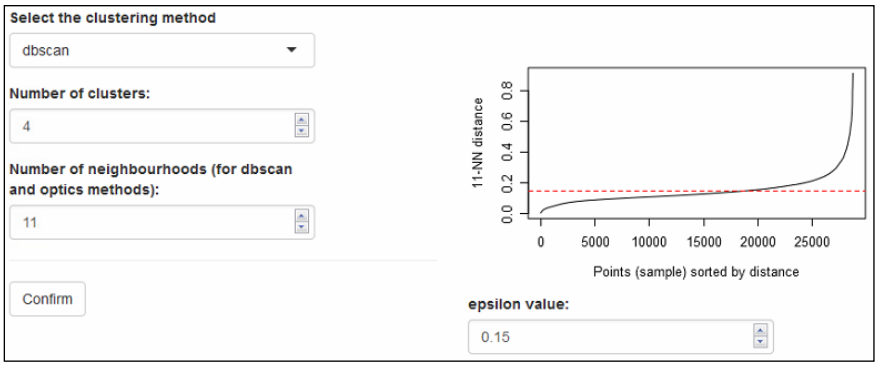

Fig. 6. Parameters of *dbscan* method

Nevertheless, which clustering method is chosen, it is always possible to get the statistics of each cluster. The average, standard deviation, minimum and maximum values of each data feature are shown (Fig. 7). The button "Show plot" enables to see the statistics in a graphical way (Fig. 8).

Statistcs

```
, , 1

          [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
Average   0.33  0.20  0.23  0.10  0.16  0.17  0.01 -0.09  0.01  0.05
SD        0.13  0.18  0.16  0.09  0.10  0.14  0.11  0.13  0.12  0.12
Min      -0.20 -0.29 -0.47 -0.15 -0.21 -0.02 -0.20 -0.42 -0.37 -0.59
Max       0.97  0.62  0.83  0.41  0.65  0.70  0.55  0.26  0.23  0.34

, , 2

          [,1] [,2] [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
Average   0.69 0.93 0.37 -0.15  0.30  0.10 -0.05  0.14 -0.05 -0.04
SD        0.17 0.09 0.11  0.12  0.11  0.11  0.12  0.11  0.15  0.11
Min      -0.09 0.63 0.04 -0.41 -0.02 -0.22 -0.58 -0.20 -0.52 -0.48
Max       1.00 1.00 0.78  0.31  0.85  0.48  0.33  0.38  0.45  0.28

, , 3

          [,1]  [,2] [,3] [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
Average   0.25  0.24 0.53 0.18  0.04 -0.10  0.02  0.24  0.04 -0.26
SD        0.17  0.13 0.10 0.09  0.06  0.06  0.07  0.07  0.09  0.09
Min      -0.50 -0.19 0.25 0.01 -0.31 -0.37 -0.30 -0.06 -0.15 -0.75
Max       0.73  0.64 1.00 0.70  0.24  0.16  0.34  0.45  0.42  0.05
```

Fig. 7. Statistics of 3 out of 6 clusters

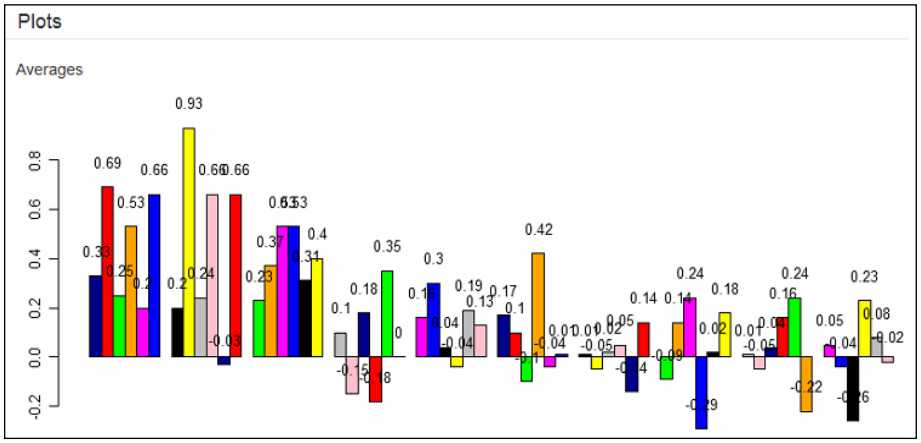

Fig. 8. Average values of each dimension in all clusters

3.3 Multi-level data visualization

In this case, from 6 possible methods, the PCA is chosen for an initial data visualization:

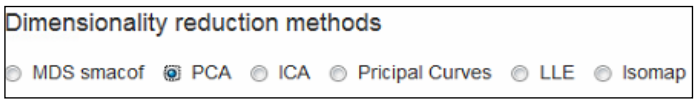

Fig. 9. Dimensionality reduction methods

Fig. 10 presents the initial data (containing all items), visualized by the PCA method. Different clusters are distinguished by different colours. Some clusters have more items (e.g. blue, brown, black), others – less (e.g. red, yellow, green). Some clusters also overlap each other. It is possible to select a part (or if needed - all) of the data points for further analysis (Fig. 10, right).

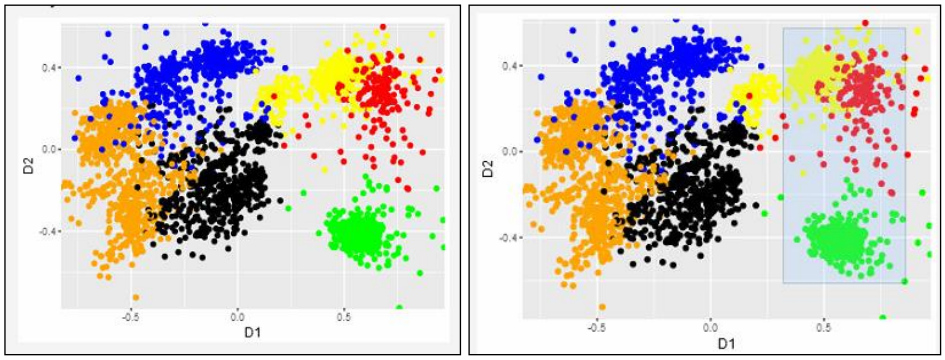

Fig. 10. Initial data visualization by using PCA method

The selected data can be processed by other dimensionality reduction method than in the previous step. In this case, the MDS method was applied to visualization of the selected data (Fig. 11). Such visualization of the selected items by the chosen method can be used as many times as needed. The data, selected in the previous step, are visualized by one more method - ICA (Fig. 12).

The visualization results reveal that items belonging to *green* cluster clearly separate from the rest of the data. However, some items belonging to *yellow* and *red* clusters overlap.

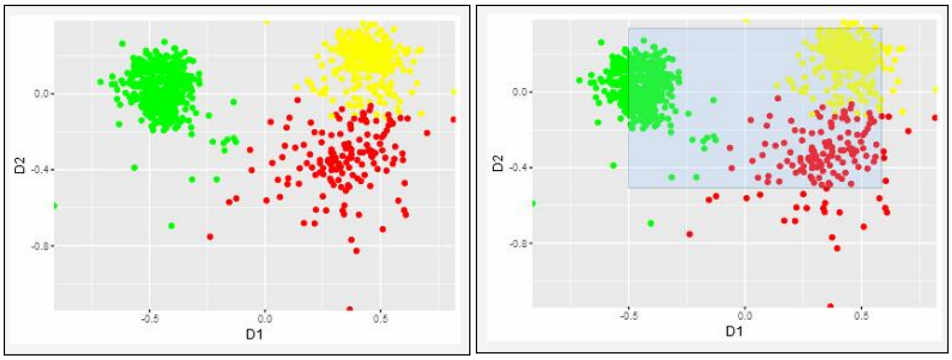

Fig. 11. Data visualization by using the MDS method

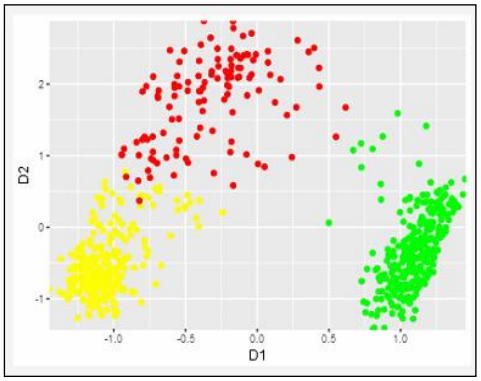

Fig. 12. Data visualization by using the ICA method

If we are not sure, which method should be used, the selected data can be visualized by all 6 methods in order to choose the best (preferred) method. Figures 13, 14 and 15 present a part of the data, selected in the previous step, visualized by the MDS, PCA, ICA, Principal Curves, LLE and Isomap methods. We also get the accuracy measures of each method (Fig.16). Fig. 17 presents the accuracy measures in a graphical way.

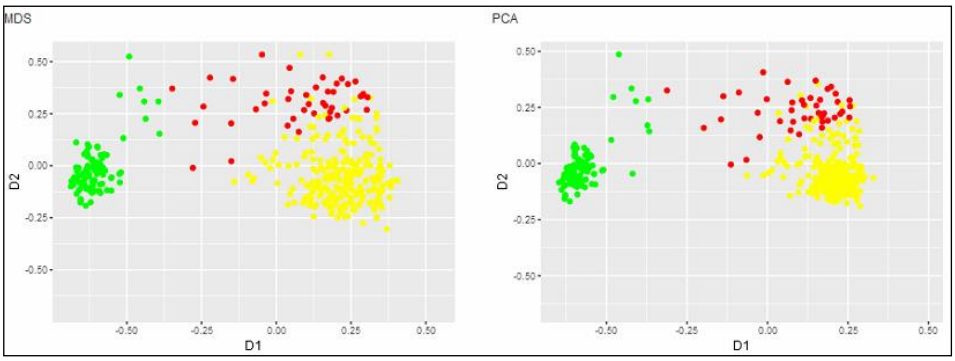

Fig. 13. Data visualization by using the MDS and PCA methods

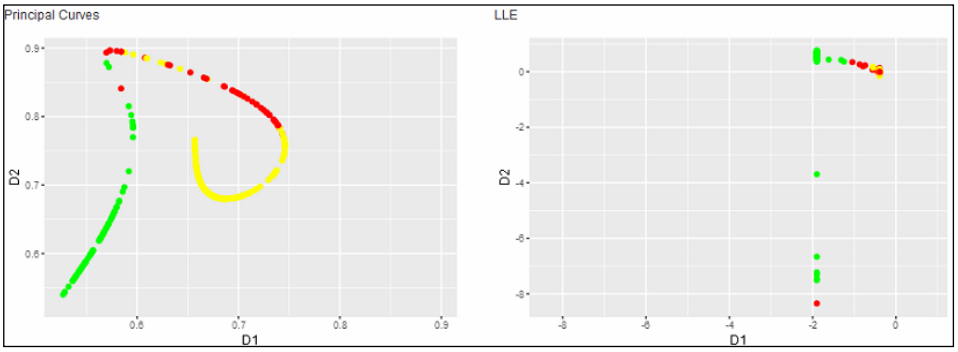

Fig. 14. Data visualization by using the Principal Curves and LLE methods

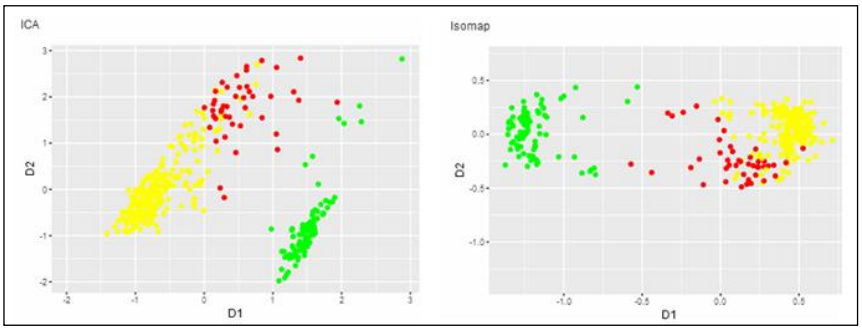

Fig. 15. Data visualization by using ICA and Isomap methods

Accuracy measures:

	MDS	PCA	ICA	Principal_Curves	LLE	Isomap
Stress	0.03	0.18	6.50	0.61	5.67	0.74
Spearman	0.94	0.94	0.83	0.78	0.79	0.90
Entropy	0.10	0.18	0.08	0.10	0.91	0.22

Fig. 16. Accuracy measures of the dimensionality reductions methods

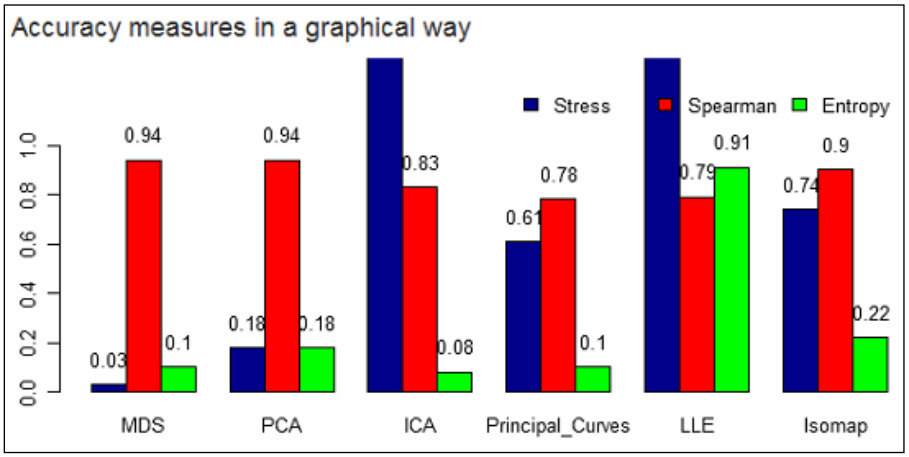

Fig. 17. Accuracy measures

In all the plots, there is also a possibility to see the IDs of each item. This enables to see which particular cluster each item belongs to (Fig. 18).

When data are presented in the 2D graph, it allows to graphically seeing the relationships between items and their dependency to the particular cluster. However, in such a graph, we do not see the exact parameters (values of initial dimensions) of each item. Therefore, the list of selected items and their features are presented in the table. This enables to find by what characteristics each cluster distinguishes.

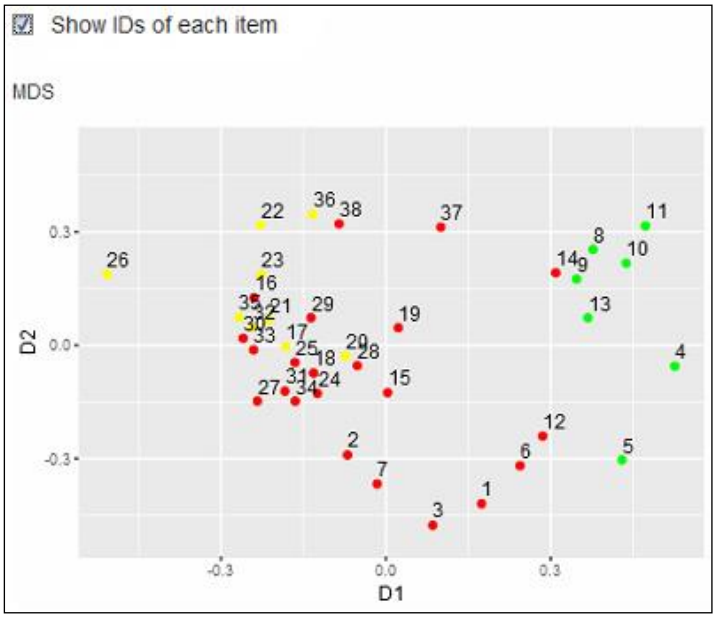

Fig. 18. Showing IDs of each item

4. Conclusions

In this paper, we have proposed the methodology that enables multi-level massive data visualization in interactive way. We have developed the special tool, which implements the proposed methodology. The principles of the methodology and the features of the tool are presented by describing the specific use case.

The proposed methodology improves visual data analysis process and brings new possibilities. It allows zooming and analysing the selected parts of data. Different dimensionality reduction methods can be applied in each particular case according to data type and size. 2D plots are supplemented by statistical characteristics. This enables the investigation of the same data from different points of view.

References

Diamond, M., Mattia, A. (2017), Data Visualization: An Exploratory Study into the Software Tools Used by Businesses. *Journal of Instructional Pedagogies*, Vol. 18, 2017, available at: https://eric.ed.gov/?id=EJ1151731 (accessed March 18, 2018).

Domeniconi, Dr. (2004), Comparison of Principal Component Analysis and Random Projection in Text Mining. INFS, 795.

Fodor, I. K. (2002), A survey of dimension reduction techniques. *Center for Applied Scientific Computing*, Lawrence Livermore National Laboratory.

Hassan, A, Elragal, A. (2017), Big Data Visualization Tool: a Best-Practice Selection Model. *Institute of Electrical and Electronics Engineers (IEEE)*, 2017. pp. 59-68, available at: http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1072292&dswid=6232 (accessed March 18, 2018).

Hauke, J., Kossowski, J. (2011), Comparison of values of Pearson's and Spearman's correlation coefficients on the same sets of data. *Quaestiones geographicae*, 30(2), 87-93.

Hausser, J., Strimmer, K. (2009), Entropy inference and the James-Stein estimator, with application to nonlinear gene association networks. *Journal of Machine Learning Research*, 10, 1469-1484.

Khomtchouk BB, Hennessy JR, Wahlestedt C. (2017), Shinyheatmap: Ultra fast low memory heatmap web interface for big data genomics. *PLoS ONE* 2017, 12(5): e0176334, available at: https://doi.org/10.1371/journal.pone.0176334 (accessed March 18, 2018).

Menon, A. K. (2007), Random projections and applications to dimensionality reduction. School of Information Technologies, The University of Sydney, available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.164.640&rep=rep1&type=pdf (accessed November 2, 2017).

Mizuta, M. (2007), Dimension Reduction Methods. *Center for Applied Statistics and Economics (CASE)*, Humboldt-Universität Berlin, 15.

Rosaria, R. S., Adae, I., Hart, A., Berthold, M. (2014), Seven Techniques for Dimensionality Reduction. Knime, available at: https://www.knime.com/blog/seven-techniques-for-data-dimensionality-reduction (accessed November 2, 2017).

Santoyo, S. (2017), A Brief Overview of Outlier Detection Techniques, available at: https://towardsdatascience.com/a-brief-overview-of-outlier-detection-techniques-1e0b2c19e561 (accessed March 18, 2018).

Sorzano, C. O. S., Vargas, J., Montano, A. P. (2014), A survey of dimensionality reduction techniques, available at: https://arxiv.org/abs/1403.2877 (accessed November 2, 2017).

Xuedi Qin, Yuyu Luo, Nan Tang, Guoliang Li. (2018), DEEPEYE: An Automatic Big Data Visualization Framework. *Big data mining and analytics*, Vol. 1, 75– 82, available at: http://ieeexplore.ieee.org/document/8268737/#full-text-section (accessed March 18, 2018).

Yongjie Li, Zheng Wang, Yang Hao. (2018), A Hierarchical Visualization Analysis Model of Power Big Data. *IOP Conference Series: Earth and Environmental Science*, Vol. 108, available at: http://iopscience.iop.org/article/10.1088/1755-1315/108/5/052064/meta (accessed March 18, 2018).

Zubova, J., Kurasova, O., Liutvinavicius, M. (2016), Parallel computing for dimensionality reduction. *Information and Software Technologies*, Springer-Verlag. ISBN 978-3-319-46254-7, 230-241.

Zubova, J., Kurasova, O., Liutvinavicius, M. (2018), Dimensionality Reduction Methods: The Comparison Of Speed And Accuracy. *Information Technology And Control*, Vol 47, No 1, 151-160.

WEB (a). Machine Learning Repository. Anuran Calls Data Set Description, available at: https://archive.ics.uci.edu/ml/datasets/Anuran+Calls+%28MFCCs%29 (accessed March 18, 2018).

WEB (b). *R package 'entropy'* - Estimation of Entropy, Mutual Information and Related Quantities. https://cran.r-project.org/web/packages/entropy/entropy.pdf (accessed November 2, 2017).

WEB (c). *R package 'smacof'* - Multidimensional Scaling. 2017. https://cran.r-project.org/web/packages/smacof/smacof.pdf (accessed November 2, 2017).

WEB (d). R Shiny, available at: https://shiny.rstudio.com/ (accessed March 18, 2018).

Received April 3, 2018, revised September 23, 2018, accepted October 5, 2018