Application of Andrew's Plots to Visualization of Multidimensional Data

Vadim Grinshpun[a]

[a]Institute of Mathematics and Computer Science, Academy of Sciences of Moldova, Chisinau, MOLDOVA.

ABSTRACT

Importance: The article raises a point of visual representation of big data, recently considered to be demanded for many scientific and real-life applications, and analyzes particulars for visualization of multi-dimensional data, giving examples of the visual analytics-related problems. Objectives: The purpose of this paper is to study application of Andrews plots to visualization of multidimensional data. Methods: Application of Andrews plots to multidimensional data visualization is investigated herein by means of analysis, logical generalization, scientific abstraction. Results: The direct interaction between the analyst and the visualization system projecting the multi-dimensional data into spaces with fewer dimensions, supporting formulation and testing of the hypotheses regarding the nature and the data structure have been researched. The article seems to be useful for working with multidimensional dataset to optimize the process.

Introduction

The demand for processing large volumes of data continues to grow. It can be statistical analysis, pattern discovery, social networking and other applications. One of the topics of big data processing is multi-criteria optimization (MCO-problem). The classical approach to solving MCO-problem is transforming it into a set of global single criteria optimization problems. Now a relatively new approach is emerging based on creation of a multi-dimensional Pareto approximation of the data set, using a fixed number of dimensions. The resulting Pareto front is presented to the decision maker, allowing to use informal methods for selecting solution – one of the points on the front.

CORRESPONDENCE Vadim Grinshpun ✉ Alex.Grin@hotmail.com

Humans are better at processing visual information, so it makes sense to use graphics to present the results of the approximation. If the number of criteria in the MCO is two or three than the visualization approach is easy, if not obvious. For a greater number of criteria the MCO visualization becomes increasingly problematic.

Modern problems related to big data analysis demand faster evolution of algorithms and their programmatic implementation, enabling the solutions to keep pace with the growing demand and complexity. Some answers comprise a relatively new interdisciplinary branch of computer science – the visual analysis, which is rapidly proliferating in all aspects of applied researches from medicine to social sciences.

We will propose a system purposed on visually analyzing multidimensional data, and will consider classical problems of multidimensional data analysis, such as cluster analysis, building object classification rule sets, building 2d projections of the multidimensional datasets into various coordinate systems. The system will enable the user: As part of the system being developed, the classical problems of multivariate data analysis were stipulated, such as: the construction of clusters and their shells in a multidimensional data cloud, building a system of decision rules to classify objects procedures, the implementation of multi-dimensional display of data volume in two-dimensional projections of all possible pairs of coordinates. The developed system allows users to:

— Directly view and manipulate data projections in two and three dimensional spaces;

— Interactively validate hypotheses regarding the presence and the nature of the clusters using the methods of geometric modeling;

— Display cluster boundaries closely approximating closely approximating the data in the selected sets of coordinates based on the main features;

— Make decisions regarding developing object classification rulesets;

— Conduct visual search of the clusters through multiple two-dimensional projections and weigh the importance of various coordinates from the dispersion point of view.

It's important to note that the proposed system of interactive visual analysis can also be used as a basis for further application of the methods of mathematical analysis of multidimensional data, using the geometric depictions as the initial hypothesis/approximation for precise calculations (Li-Xin & Wang, 2003; Mamdani, 2008; Wang & Mendel, 1992; Zadeh, 1994).

Methodological Framework

Typical Data Visualization Methods

There is a relatively large set of typical approaches to data visualization, each presents different advantages and challenges. To help optimization of selection of the most suitable visualization methods, we consider three following fundamental attributes of the methods (Belous et al., 2015):

— Attributes of data that has to be visualized with the specified method;

— Sample visualizations of various data using the specified visualization method;

— Ability to change the visualization method and to interact with any visualization produced by the selected method.

— The following data types suit well for visualizations:

— One-dimensional data – data rows, time sequences, mathematical sequences and single value functions, etc.;

— Two-dimensional data – (x,y) coordinates, geographic coordinates, vector functions, etc.;

— Multidimensional data – financial results, results of experiments, metadata, etc.;

— Texts and hypertexts – news articles, web documents, books, periodicals, etc.;

— Hierarchical and connected data – organizational structure, social network, enterprise data flow;

— Physical, chemical and other processes, information streams – sensor data, stock data, electric currents, etc.

The visual concepts and implementation algorithms vary for each type of data enumerated above. Within the interactive visualization system, discussed herein, the following typical problems have been resolved:

Approaching of the cluster analysis problems through the 3D projection method enables the user to interactively work with projections of the original multi-dimensional in three dimensions, by the user's selection from the original sets of coordinates. The user is able to interactively build various clusters and cluster shells. To build the projection, a parameter d, corresponding to the furthest distance inside the cluster, is selected. If the distance between two points in the original cluster is less than d, that the two points are connected. The original points are represented as spheres and the connections between them as cylinders (Figure 1).

The optical model also includes the density of the color of the cylinder, the closer the points, the deeper shade of blue is.
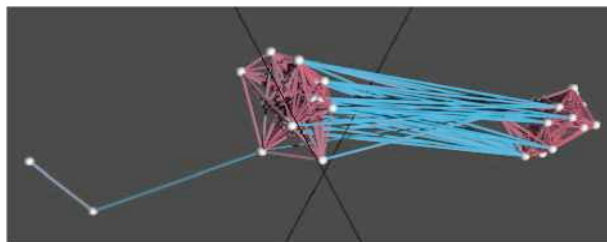

Figure 1. Multi-point 3D projection

The user can analyze a cluster depending on the selection of the parameter d. The analysis can be performed in two ways – sequential review of one model at the time for each d selected by the user, or selection of two values of d and auto generation of a series of transformation from one value to another (animation).

To analyze the shape of the cluster, its 3D shell is constructed by building a right-angled parallelepiped, by using an overlapping spheres method, or by mixing both approaches.

The right-angled parallelepiped is constructed around the axes, generated by the method of selecting the main components. This approach guarantees a close approximation of parallelepiped to the cluster (Figure 2).
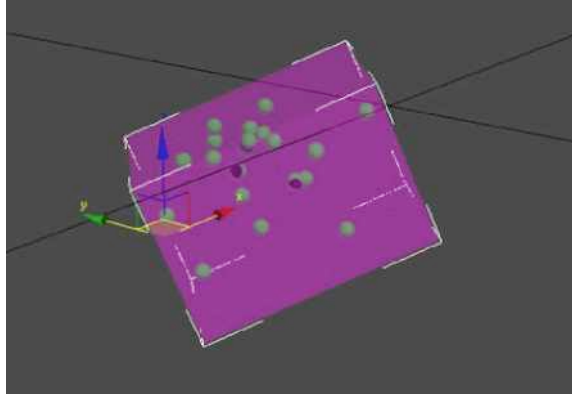

Figure 2. The right-angled parallelepiped cluster shell

The second method generates spheres with the center in each point of the cluster, and the radius equal to the largest distance from the center to any other point, which a priori is less than or equal to d. The surface of the area where spheres overlap presents the cluster shell that obviously includes all points in the cluster and provides a good approximation (Figure 3). This method is better than the previously described when the values of the correlation matrix inside the cluster are relatively close.

The mixed approach assumes building the shell using both methods, and then selecting the overlapping area.
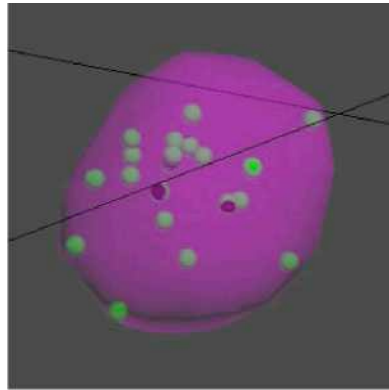

Figure 3. The overlapping spheres cluster shell

To solve the problem of a discriminant analysis using 2D and 3D projection methods, the user is able to create planar surfaces that separate different classes of points, when projecting multi-dimensional data sets into 2D and 3D original sets of coordinates.

The basic supposition of the discriminant analysis is the assumption that there are two or more groups that in some parameter differ from the other groups, and these parameters can be measured either by using interval or

relationship scale. The discriminant analysis helps to highlight the differences between the groups and allows to classify data points based on the maximum similarity.

The main method of solving the discriminant analysis problem is the method of defining the R.A. Fisher's (1936) hyperplane. As a result of the research, the method, ultimately selected to build the Fisher's hyperplane? Is to generate a sequence of projections. The idea is that is if we can build the separating hyperplane in a given space, this plane will continue to separate classes when a dimension is added. The algorithm presumes sequential review of 2D and 3D projections in order to find the separating plane or a system of planes (Figure 4).
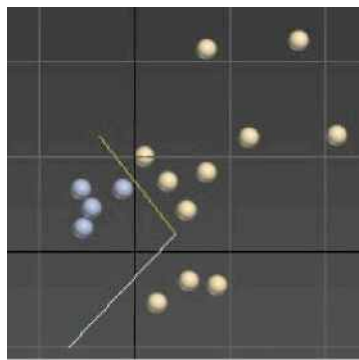
Figure 4. Separating lines for two groupings of points.

After the generation of the separating system, the user is able to verify the result by creating a system of formal rules. Upon completion of verification, these rules can be applied to the classification problem of the new points added to the original multi-dimensional set (Figure 5).
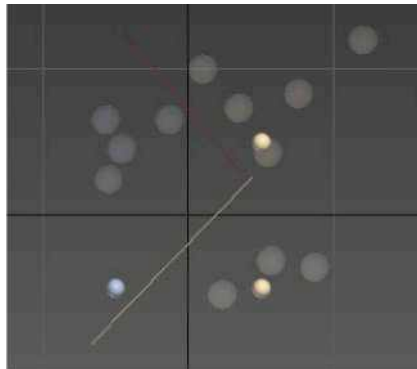
Figure 5. Classification of new points

This method helps to make decisions regarding the rules for classification of the new objects.

Solving the problem of cluster identification using a 2D projection, and implementing an interactive system allow the users to work simultaneously with all projections of the original multi-dimensional data set into varying 2D subspaces generated from the original set of coordinates. Given that points close to each other in all 2D projections will be close in the original space as well, the

user of the interactive system can remove, add and color close points in the 2D projection. All user actions are simultaneously reflected in all projections.

The solution algorithm:

Step 1. The points from the original multi-dimensional data set are project into all 2D subspace (planes) generate by the pairs of the original coordinates. This process results in the projection matrix (Figure 6).
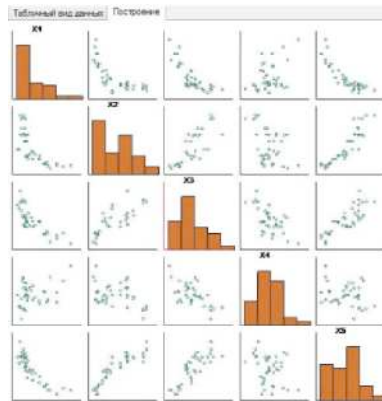

Figure 6. Projection matrix

Step 2. Candidates are selected on one of the projections cluster.

Step 3. The rest of the projections is reviewed and the points far removed from the main body are excluded from further consideration.

Step 4. The remained points are marked as a cluster, and are excluded from further consideration. If there are no obvious cluster candidates (single random points) then proceed to step 5, otherwise return to step 2.

Step 5. The result is the initial breakdown of the multi-dimensional dataset in the clusters. To further improve the result, the k-means algorithm is applied. For better illustration of the result of the clustering process, the system implements the profile diagram (Figure 7) as a means of 2d representation of the cluster entities.
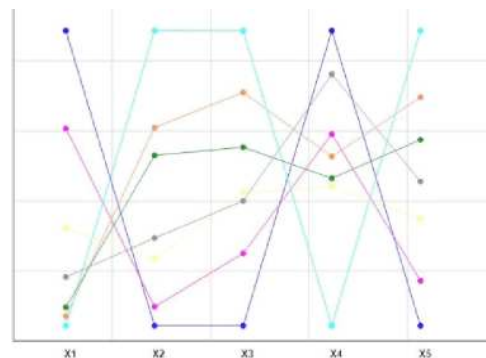

Figure 7. The profile diagram

As a result, the user can also identify the points that can potentially belong to more than one cluster, and determine the rules to resolve this problem, if necessary.

Geometric Transformations Methods

The following visualizations are using methods involving geometric transformations:

— Scatter graphs

— Parallel coordinates

— Pixel-oriented method

— Patterns of recursion

— Cyclical segments

— Hierarchical visualizations

— Dimensional over imposition

Data visualizations produced using the specified methods aren't always sufficient. The user has to have an ability to manipulate these visualizations, to view them from various angles, scale them up and down, add relative markings such as min, max, average, trend, etc. This requires implementation of the supporting functionality:

— Dynamic projections;

— Interactive filtering;

— Scaling and zooming;

— Interactive distortion;

— Interactive combination.

The main idea behind the dynamic projection is to dynamically project objects onto a 2D or 3D space when researching a multi-dimensional dataset. As an example, project all interesting objects onto a 2D plane as a scatter plot. It is important to note that the number of possible projections grows exponentially with the increase in the number of dimensions, and therefore having all projections displayed at once will be difficult to grasp.

When researching large data sets (Big Data), it is important to have the capability to separate the data sets, and identify interesting subsets, and it is important that selecting those subsets and filtering the data is performed in real time or near-real time. The selection of the subset can be done directly from the data list, or by defining filtering attributes for the selection criteria.

The main idea behind combination with the visualization method is to emphasize advantages and to minimize shortcomings of each individual visualization method. For instance, combining a scatter plot with a color map or a heat map can provide a more information-rich visualization taking exactly the same screen space and a comparable amount of time to produce.

Any visualization tool can be classified by all three key parameters – by the type of data it visualizes, by the type of visual representation it produces and by the type of interaction with the visualizations it enables. Obviously a single tool or product can support multiple datatypes, different visualization methods, and types of interactions.

Pixel-Oriented Methods

The main idea behind the pixel-oriented methods is projection of a value from each dimension into a colored pixel, and the grouping of the pixel by their respective dimensions. Since one pixel reflects a single value, this method allows to directly reflect over a million data points on a single standard monitor and over 8 million data points on a UHD monitor (8.3 megapixels).

One of the primary interaction techniques for pixel-oriented methods is zooming. An example of zooming is a "Magnifying Glass" (AKA the "Lens"), its primary function is to provide custom filtering for visualization. The data shown under the "Magnifying Glass" are filtered and enlarged, and so they are differentiated from the basic data. The Lens shows the modified image of the selected subset of date (data region) in detail, whereas the remaining data points are not being detailed.

Zooming is a well-known approach to interaction, used in almost all applications. When working with a large data set, it has the advantage of presenting the overall picture, and at the same time allows for reflection of each specific subset in detail. Zooming may consist not only of a simple enlargement of objects, but for changing the levels of data representation. For instance, on a lower level the object, represented by a pixel, can be shown as an image, and at even lower level as a text string.

The method based on interactive distortion supports the data research process by dynamically changing the zooming scale and presenting more detailed information. The idea is to simultaneously display all the data at a low level of detail, while simultaneously showing parts of the data in higher detail. Most common methods producing this effect are considered to be hyperbolic and spherical distortions.

Hierarchical Images

Hierarchical images are used to reflect hierarchical and other types of relationships in the data. The following methods a typically used to build the hierarchies:

— hierarchical axes;

— dimensional superimposition;

— trees;

Hierarchical axes are being reflecting data attributes, with the first axis being of the attribute with the most variation. This method can reflect up two 20 attributes on a single screen, for a larger number of attributes it is possible to use the dimensional superimposition and generate a tree-like structure.

The idea behind the dimensional superimposition is to insert one system of coordinates inside another. For instance, one pair or triplet of attributes creates one system of coordinates, and another set creates a different coordinate system, that is inserted into the first one, thus the first set of attributes creating the external system, while the second set creating the inserted system. The process can be repeated multiple times to accommodate large number of attributes.

The scientific quality of this visualization method is heavily depends on the selection of the external coordinate system, therefore selection the attributes,

used for creating the external system, should start with the most important dimension.

For visualization of tree structures, two primary methods are typically used:

— tree maps;

— canonical trees;

Tree maps hierarchy divide the screen by filling out the visible space, using the demarcation for the visualization of the trees. Color is used to represent the content of the node. Varied brightness, textures and shading can be utilized to provide a more robust visualization.

A canonical tree is a "tree-like" interactive structure that can be rotated, with the branches that can be expanded, showing more data, or collapsed by the user.

Results

In his work D.F. Andrews (1972) proposed a simple and convenient method for reflecting big data set on a surface. If the data have the fixed number of m, then every point x = (x1,..., xm), where xi, (i = 1, ..., m) – are variables, can be represented by the following Fourier function:

$$f_x(t) = x_1 \cdot 2^{-\frac{1}{2}} + x_2 \cdot \sin t + x_3 \cdot \cos t + x_4 \cdot \sin 2t + x_5 \cdot \cos 2t + ...$$

which appears on the graphical interval $-\pi < t < \pi$. So for each point in the data set there a corresponding line in this interval.

For this function if $x_i = x_{1i}, ..., x_{mi}, (i = 1, ..., n)$ – where n is the *n* of points, the for the mean vector *x* the following equation is true:

$$f_{\bar{x}}(t) = \frac{1}{n} \sum_{i=1}^{n} f_{xi}(t)$$

$$\left\| f_{xj} - f_{xi} \right\|_L^2 = \int_{-\pi}^{\pi} \left\{ f_{xj}(t) - f_{xi}(t) \right\}^2 dt = \pi \sum_{k=1}^{m} \left(x_{kj} - x_{ki} \right)^2$$

Moreover, if $x_{ki}, (k = 1, ..., m; i = 1,..., n)$ – are un-correlated random numbers with the dispersion σ^2, than the following equation is also true:

$$D[f_x(t)] = \sigma^2 \left(2^{-1} + \sin^2 t + \cos^2 t + \sin^2 2t + \cos^2 2t + ... \right) = \begin{cases} 2^{-1}\sigma^2 m, \text{ if m is odd-numbered} \\ 2^{-1}\sigma^2 \left\{ m - 1 + 2\sin^2\left(\frac{mt}{2}\right) \right\}, \text{if m is even} \end{cases}$$

As a result we have

$$2^{-1}\sigma^2 (m-1) \leq D[f_x(t)] \leq 2^{-1}\sigma^2 (m-1)(-\pi \prec t \prec \pi)$$

Thus D.F. Andrews (1972) plots preserve the information about mean values, distance and dispersion and produce a large number of one-dimensional projections onto the vectors (2-1/2, sin t, cos t, …)(– п < t < п). Since the distances between D.F. Andrews (1972) plots are a linear reflection of the distances between data points, the two plots that a closer to each other correspond to the two points that are closer as well. This property proves very useful when representing big datasets.

To illustrate the application of Andrews plots, let's consider R.A. Fisher's (1936) Irises as the original multi-dimensional data (Fisher, 1936; Embrechts, 1991). Fisher's Irises is a set of data for a classification problem, used in 1936 by Ronald Fisher (1936) to demonstrate the workings of his discriminant analysis method. This data set has become a classic and is frequently used for illustration of various data-related algorithms.

R.A. Fisher's (1936) Irises set consists of the data about 150 iris flowers – 50 of each of the three species – Iris setosa, Iris virginica and Iris versicolor. For each flower four characteristics were recorded: x1 – sepal length, x2 – sepal width, x3 – petal length, x4 – petal width.

There is a number of points that stand out in class setosa. Classification of the Fisher's Irises is a four-dimensional problem, and its visualization is relatively simple. The problems begin with a large number of dimensions. This data set is shown as an Andrews plot in Figure 8.

Please note that the lines corresponding to similar values also have a similar shape, and the number of dimensions is irrelevant – each point will always have a single corresponding line in the plot. It also clearly shows how class setosa stands apart from the other two classes. For instance all lines are very closely located around the t0 = -2,5. It means that in the direction perpendicular to the vector

$$\left(\frac{1}{\sqrt{2}}, \sin t_0, \cos t_0, \sin 2t_0 \right)$$

the data cloud is mostly flat, so it makes sense to reduce the number of dimensions from four to three.
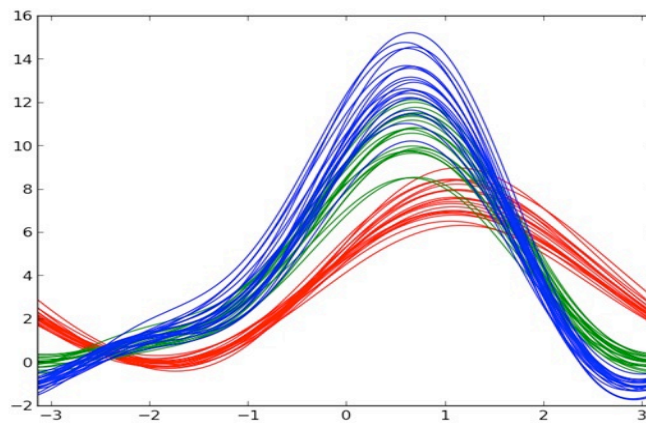

Figure 8. D.F. Andrews (1972) Plots of R.A. Fisher's (1936) Irises

Classes virginica and versicolor are expected more difficult to be distinguished. As expected, it is difficult to distinguish class virginica and versicolor class, although on some intervals of t the difference is very clear. For instance on the interval [-2.5; -1.5], both classes demonstrate different curves. An interactively created plot would allow to highlight this difference from vividly. Also please note some potentially remote lines in the versicolor class.

The main advantage of the Andrews diagram in this case is that we produced a clear and easily readable presentation of the data.

For better readability of the data various orthogonal projections into flat coordinate system, or projections into certain subspaces. In this case we will review only the case of inclusion or of the x-coordinate. The equation for the plot after its projection contains xj = 0, (j = 1, ..., m). For instance if xj = 0 it's difficult to differentiate the plots, which indicates a weak dispersion of the data along that coordinate, and vice versa. This technique is directly applicable to the data research (Embrechts, 1991).

Range of one of the coordinates can exceed the ranges of other coordinates by so much, that it will obfuscate the influence of all other coordinates on the plot. To avoid that coordinate scaling is applied as follows. Let's say we have a set of n points xi = (x1i, ..., xmi) (i = 1, ..., n) in a multi-dimensional space. Let's assume we also have a function

$$\overline{x_j} = \frac{1}{n}\sum_{i=1}^{m} x_{ji}, s_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(x_{ji} - \overline{x_j}\right)^2$$

where $\overline{x_j}$, s_j^2 – are the mean average and dispersion of the j-variable. Let's represent the points after scaling as

$$y_i = \left(y_{1i}, ..., y_{mi}\right)\left(i = 1, ..., n\right)$$

where

$$y_{ji} = \frac{x_{ji} - x_j}{s_j}\left(j = 1, ..., m\right)$$

The square of the Euclidian distance between points x_j and x_i equals

$$\left\|x_j - x_i\right\|^2 = \sum_{k=1}^{m}\left(x_{kj} - x_{ki}\right)^2$$

and between the corresponding points after the scaling

$$\left\|y_j - y_i\right\|^2 = \sum_{k=1}^{m}\frac{\left(x_{kj} - x_{ki}\right)^2}{s_k^2}$$

Let's assume y is a point that corresponds to x after the scaling, then the relative distances along the axis will remain unchanged, moreover for an uncorrelated set of data with normal distribution the following equation is correct:

$$D\left[f_y(t)\right] = \begin{cases} \dfrac{(n-1)^2 m}{2n(n-3)}, & \text{if m is odd-numbered} \\ \dfrac{(n-1)^2}{2n(n-3)}\left\{m - 1 + 2\sin^2\left(\dfrac{mt}{2}\right)\right\}, & \text{if m is even} \end{cases}$$

As a basic rule we can assume that correlated variables should be grouped, and the significantly discriminating variables should be assigned different frequencies. To illustrate the proposed method of projection, using trigonometric functions, to project the data into the subspace $x_k = 0$, (k = 1, 2, 3, 4).

So we have researched the application of the mathematical apparatus to construct Andrews plots. As a base set of data we used Fisher's Irises, the classic set of data for the classification problem. We've shown that building Andrews plots using Fourier functions and polynomial functions produces similar results, which in turn suggest that Andrews plots can be used to analyze a wide variety of multi-dimensional data. We've also shown the advantages and disadvantages of application of these types of functions.

Discussions and Conclusion

In practical terms, visual analytics can be viewed as interactive problem-solving via a visual interface, in other words visual analytics is a way to organize computer-human interface to amplify individual's analytical skills (Big Data Visualization, 2013; Dillon, 1984; Keim, Qu & Ma, 2013; Maletic, Marcus & Collard, 2002; Manakov, Mukhachev & Shinkevich, 2003; North, 2006; Shneiderman, 2014; Shneiderman, 1996).

Basic approaches and algorithms of visual analytics are described in the works (Keim et al., 2008a; Keim et al., 2008b; Keim et al., 2010; Kielman & Thomas, 2009; Thomas & Cook, 2005). Same works demonstrate a series of applications of contemporary visual analytics in various aspects of human activity, as well as descriptions of a number of software products, developed for visual analytics.

Careful review of the literature containing specific applications of visual analytics reveals that in reality a lot less attention is paid to the systems focusing on multi-dimensional data, when compared to the systems reflecting the results of the application of the modern Data Analysis methods (Assuncao et al., 2014; Baker & Wickens, 1995; Dasgupta, Chen & Kosara, 2012; Fout & Ma, 2012).

We have presented problems related to the interactive system for visualization of multidimensional data. The main purpose of this system is to utilize interactive capabilities in working with 2D and 3D projections of the original multidimensional dataset to create and test initial hypotheses or approximations regarding the structure, the nature and the relative positioning of the data points inside the studied dataset.

The materials of the article are of practical value to the visual analysis of data in different size spaces.

Disclosure statement

No potential conflict of interest was reported by the authors.

Notes on contributors

Vadim Grinshpun is Doctorant of the Institute of Mathematics and Computer Science, Academy of Sciences of Moldova, Chisinau, Moldova.

References

Andrews, D.F. (1972). Plots of high-dimensional data. *Biometrics, 28,* 69-97.

Assuncao, M.D., Calheiros, R.N., Bianchi, S., Netto, M.A.S., Buyya, R. (2014) *Big Data Computing and Clouds: Challenges, Solutions, and Future Directions. Journal of Parallel and DistributedComputing, 79-80,* 3-15.

Baker, M.P. & Wickens, C.D. (1995) *Human Factors in Virtual Environments for the Visual Analysis of Scientific Data.* Technical Report. – NCSA. draft. http://monet.ncsa.uiuc.edu/~baker/PNL/paper.html

Belous, V.V., Groshev, S.V., Karpenko, A.P. & Ostroushko, V.A. (2015). Imaging methods for Pareto front in the problem of multi-criteria optimization. Overview. *20 Baikal'skaya Vserossiiskaya konferentsiya "Informatsionnye i matematicheskie tekhnologii v nauke i upravlenii". Proc. of the 20th Baikal Conference on Information and Mathematical Technologies in Science and Management.* Irkutsk, 22-29.

Big Data Visualization. (2013). *Turning Big Data into Big Insights. The Rise of Visualization-based Data Discovery Tools.* http://www.intel.ru/content/dam/www/public/us/en/documents/white-papers/big-data-visualization-turning-big-data-into-big-insights.pdf

Dasgupta, A., Chen, M., Kosara, R. (2012*).* Conceptualizing Visual Uncertainty in Parallel Coordinates. *Comput. Graph. Forum, 31(3),* 1015-1024.

Dillon W.R., Goldstein M.(1984). *Multivariate Analysis: Methods and Applications.* New York: Wiley.

Embrechts, P. & Herzberg, A.M. (1991). Variations of Andrews' Plots. *International Statistical Review, 59(2),* 175-194.

Fisher, R.A. (1936). The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics, 7,* 179-188.

Fout, N. & Ma, K.L. (2012) *Reliable Visualization: Verification of Visualization based on Uncertainty Analysis.* Los Angeles : Tech. rep., University of California. – Davis.

Keim, D.A., Mansmann, F., Schneidewind, J., Thomas, J. & Ziegler, H. (2008a) *Visual analytics: Scope and challenges.* San Francisco : Visual Data Mining.

Keim, D. Qu, H. & Ma, K.L. (2013) *Big-Data Visualization.* IEEE Computer Graphics and Applications, July/August.

Keim, D., Andrienko, G., Fekete, J.-D., Gorg, C, Kohlhammer, J., and Melancon, G. (2008b) *Visual Analytics: Definition, Process, and Challenges. Information Visualisation, Springer-Verlag, Berlin Heidelberg, 4950,* 154-175.

Keim, D., Kohlhammer, J., Ellis, G. & Mansmann, F. (2010). *Mastering the Information Age – Solving Problems with Visual Analytics.* http://www.vismaster.eu/wp-content/uploads/2010/11/VisMaster-book-lowres.pdf

Kielman, J. & Thomas, J. (2009). Special Issue: Foundations and Frontiers of Visual Analytics, *Information Visualization, 8(4),* 239-314.

Li-Xin, & Wang. (2003). The WM method completed: a flexible fuzzy system approach to data mining. *IEEE Trans. Fuzzy Systems, 11(6),* 768-782.

Maletic, J.I., Marcus, A. & Collard, M.L. (2002). A task oriented view of software visualization. *International Workshop on Visualizing Software for Understanding and Analysis*, 32-40.

Mamdani, E.H. (2008). Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis. *IEEE Trans. Computers, 12,* 1182-1191.

Manakov, D., Mukhachev, A. & Shinkevich, A. (2003). Visualization of the Distributed Data of Huge Volume. Assembly, Filtration, Sorting. *Proceedings of the 13-th International Conference on Computer Graphics and Vision Graphicon-2003,* 198-201.

North, Ch. (2006). Toward Measuring Visualization Insight. *IEEE Computer Graphics and Applications, 26(3)*, 20-23.

Shneiderman, B. (1996). The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. *Proceedings of the IEEE Conference on Visual Languages*, 336-343.

Shneiderman, B. (2014). *The big picture for big data.* San Francisco : Science.

Thomas, J. & Cook, K. (2005) *Illuminating the Path: Research and Development Agenda for Visual Analytics* San Francisco : IEEE-Press.

Wang, L.X. & Mendel, J.M. (1992). Generating fuzzy rules by learning from examples. *IEEE Trans. Syst., Man, Cybern, 22(6)*, 1414-1427.

Zadeh, L.A. (1994). Fuzzy logic, neural networks, and soft computing. *Communications of the ACM, 3,* 77-84.