MOKSLAS – LIETUVOS ATEITIS
SCIENCE – FUTURE OF LITHUANIA
ISSN 2029-2341 / eISSN 2029-2252
http://www.mla.vgtu.lt

ELEKTRONIKA IR ELEKTROTECHNIKA
ELECTRONICS AND ELECTRICAL ENGINEERING

2015 7(3): 275–279

Sistemų inžinerija, kompiuterių technologija T 120
System Engineering, Computer Technologies T 120

http://dx.doi.org/10.3846/mla.2015.781

GENETIC ALGORITHMS FOR MULTIDIMENSIONAL SCALING

Agnė DZIDOLIKAITĖ

Vilnius University Institute of Informatics and Mathematics, Vilnius, Lithuania
E-mail agne.dzidolikaite@mii.vu.lt

Abstract. The paper analyzes global optimization problem. In order to solve this problem multidimensional scaling algorithm is combined with genetic algorithm. Using multidimensional scaling we search for multidimensional data projections in a lower-dimensional space and try to keep dissimilarities of the set that we analyze. Using genetic algorithms we can get more than one local solution, but the whole population of optimal points. Different optimal points give different images. Looking at several multidimensional data images an expert can notice some qualities of given multidimensional data. In the paper genetic algorithm is applied for multidimensional scaling and glass data is visualized, and certain qualities are noticed.

Keywords: multidimensional scaling, genetic algorithms, visualization.

Introduction

In various research fields human being has to deal with multidimensional data. Probably we would not find such research field where we could avoid multidimensional data. The amount of such data increases very quickly (Dzemyda *et al.* 2008).

The size of data grows in a very high speed, so it is necessary to solve certain problems: how to understand the data, comment it and gain information while refusing unimportant facts. In most cases there is a need to understand the structure of such data: groups of similar objects (clusters) and outliers, the similarity and dissimilarity of objects. It is difficult to understand multidimensional data because it means complicated phenomenon or object which has many parameters. These parameters can be numerical, logical, text or something else (Karbauskaitė 2005).

The higher is the dimension of data, the more difficult is to understand it. Multidimensional data should be given in a form that researcher could easily understand data structure, groups and connections in data, etc. Such a way could be visualization. Visualization is graphical view of multidimensional data. Graphical view is easier for human being to understand. From graphical view a researcher can see certain tendencies in data and make decisions.

Multidimensional scaling

Multidimensional scaling (MDS) is a statistical method which is meant to visualize dissimilarities (Groenen, Velden 2004). MDS is widely applied for multidimensional data analysis in many science fields, such as economics, psychology, etc. The aim of multidimensional scaling is to find multidimensional data projection in a lower dimension space (R^2 or R^3), so that similarities or dissimilarities of that data is kept. It is expected that after visualization similar objects are closer to each other and different objects are further from each other.

The primal data of multidimensional scaling method is quadratic symmetrical matrix which means the similarities (dissimilarities) of analyzed objects. In the simplest case it is Euclidean distances matrix. In general case they do not have to be mathematical distances.

Assume that every n dimensional vector $X_i \in R^n$, $i \in \{1,..,m\}$ corresponds to a lower dimensional vector $Y_i \in R^d$, $d < n$. Distance between vectors X_i and X_j, is called dissimilarity and is marked as δ_{ij}, and distance between vectors Y_i and $Y_i - d(Y_i, Y_j)$, $i, j = 1,\ldots,m.$ Distances between vectors are often Minkowski distances between points Y_i and Y_i (Žilinskas A., Žilinskas, J. 2006a):

$$d_p\left(Y_i, Y_j\right) = \left(\sum_{k=1}^{m} \left|Y_{ik} - Y_{jk}\right|^p\right). \quad (1)$$

When p = 2, we have Euclidean distances, when p = 1 we have city block distances.

Using MDS algorithm it is aimed to make distances $d(Y_i, Y_j)$ closer to dissimilarities δ_{ij}. Least squares function

(Stress) E_{DS} is used and is as follows:

$$E_{DS} = \sum_{i<j} w_{ij} \left(d\left(Y_i, Y_j\right) - \delta_{ij} \right)^2. \qquad (2)$$

Global optimization algorithms for multidimensional scaling

One of the most famous minimization algorithms for multidimensional scaling is SMACOF (Leeuw 1977). This algorithm is based on the objective function majorization. Here the minimization of least squares function (*Stress*) is changed to simpler minimization of supportive function. It is proved that majorization method is globally convergent. In most cases convergence is linear.

If sequential estimation method is applied (Miyano, Inukai 1982), the view is renewed when new objects are inserted into the data set. Here the insertion of a new object requires less calculations then the search of a view gained from the whole data set by multidimensional scaling.

It also exist tunneling algorithm (Groenen, Heiser 1996), which is applied to use for multidimensional scaling with Minkowski distances. Here local minimization is combined with tunneling step. It is aimed to find another image with the same objective function value as a previous local minimum. Iteratively applying local search and tunneling better solutions are found and the last solution may be global solution.

In multidimensional scaling local minimization can be combined with evolutionary search so that new points could be generated (Mathar, Žilinskas 1993). Such a hybrid algorithm where global evolutionary search is applied with local search is very effective, but needs a lot of calculations.

Heuristic simulated annealing algorithm is begun by dividing each coordinate axis into discrete points. The algorithm searches for the minimum of the objective function in the grid made of these points (Brusco 2001).

Distance smoothing method applied for multidimensional scaling is offered in paper (Hubert *et al.* 1992). This method allows avoiding local minima.

Two levels method for multidimensional scaling is offered in paper (Žilinskas A., Žilinskas, J. 2008). Here global optimization problem is changed into two levels minimization problem. In upper level combinatorial optimization problem is solved. In lower level quadratic programming problem is solved with a positively defined objective function and linear restrictions. Lower level problem is solved by applying standard quadratic programming algorithm. Upper level problem can be solved by using evolutionary search if datasets are bigger. Parallel two level optimization algorithm with evolutionary search and quadratic programming is offered in article (Žilinskas A., Žilinskas, J. 2006b).

Genetic algorithms for multidimensional scaling

Genetic algorithm mimics nature (Alba, Dorronsoro 2008). Here the most important part is recombination. Nevertheless, algorithm starts its work from initialization when a random list of individuals is created where each element means certain solution of a problem. In most cases initial population is made of hundreds or thousands individuals that have poor adaptation.

Later the generated population is evaluated according to each member's adaptation. There are many ways to calculate individual's adaptation. Assume that the given individual codes equation's solution. Therefore, the closer is the individual's value to the equation's solution, the higher is its adaptation.

After evaluation selection is carried out and parents are selected for recombination. The most commonly selection methods are:

- Roulette wheel selection;
- Tournament selection.

By applying roulette wheel selection, the adaptation of each individual is evaluated. Then it is calculated which part of the whole adaptation has certain individual's adaptation. The greater is the individual's adaptation, the higher is probability to be selected. Roulette wheel selection is carried out randomly, and the probability for a certain individual to be selected as a parent is:

$$P = \frac{f_i}{\sum_{j=1}^{n} f_j}, \qquad (3)$$

where: f_i – adaptation of a certain individual, $\sum_j f_j$ – adaptation of all the members of the population.

For tournament selection *k* individuals are randomly selected. The best individual is chosen with a probability *p*. The second best individual has a probability $p(p-1)$. The third best individual has a probability $p(p-1)(p-1)$, and etc.

When selection is carried out, offsprings are made from parents. This process is called reproduction and consists of recombination and mutation. There are many recombination variants, but most often used are these:

- recombination, when parents' chromosomes are cut into two parts;
- recombination, when parents' chromosomes are cut into more than two parts.

In multidimensional scaling case we assume that genes are the points that represent objects. During recombination some points are taken from one individual (see parent 1 in Fig. 1) and some points-from the other individual (see parent 2 in Fig. 2). Resulting offspring is shown in Figure 3.

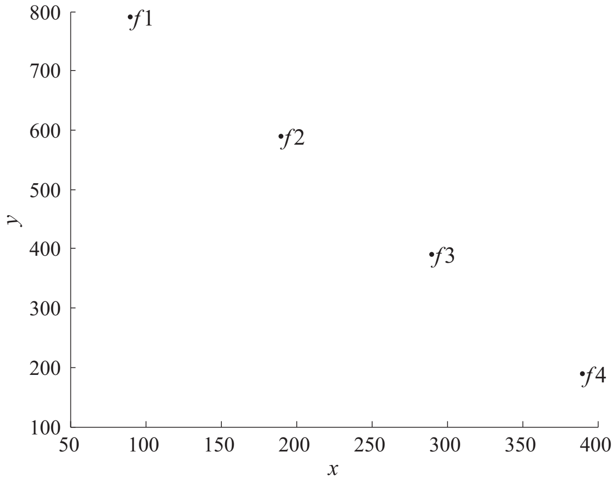

Fig. 1. Parent 1 – the first chromosome from parents' population used for recombination (dimensions *x* and *y*)

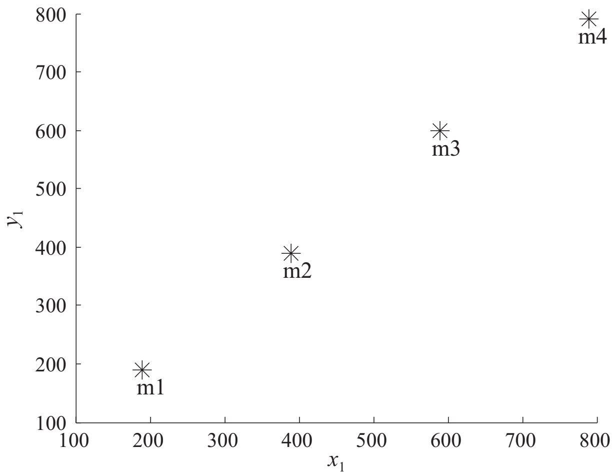

Fig. 2. Parent 2 – the second chromosome from parents' population used for recombination (dimensions x_1 and y_1)

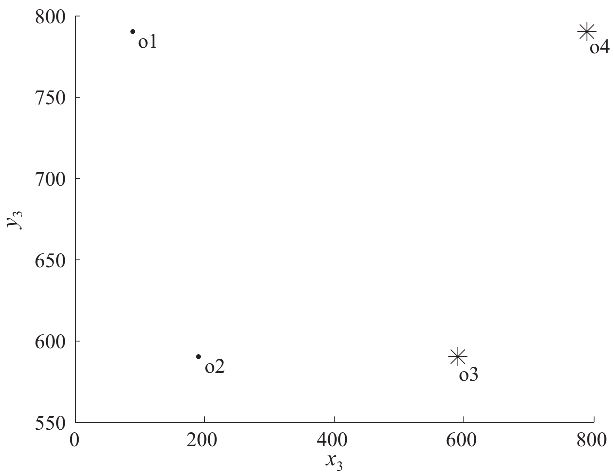

Fig. 3. Offspring – the chromosome of parent 1 and 2 recombination: points "o1" and "o2" are taken from parent 1, while points "o3" and "o4" – from parent 2

During mutation one or more components of the gene are changed and new trait of the individual is gained. If the components of genes are nulls and ones then ones are changed to nulls and nulls-to ones.

After mutation replacement is carried out. It means that the population of offsprings changes some part or the whole population of parents. Elitist replacement is popular as it saves the fittest individuals. In most cases 10% poorest parent individuals are selected and they are replaced by individuals from the population of offsprings.

Genetic algorithm stops when termination condition is met. It means that quite good solution is reached or maximum number of generations is made.

Experimental investigation

Data for experimental investigation is selected from "UCI Machine Learning Repository" (UCI Machine Learning... 2014) database. Data is created by B. German in 1987. Data file consists of 214 lines that correspond to various glasses. Each line is made of eleven attributes that have ID. There is also a class of the glass that is at the end of every line. ID has values from 1 to 214, and class attribute: 1, 2, 3, 5, 6, 7. Attributes 2–10 are as follow: refractive index, Sodium, Magnesium, Aluminum, Silicon, Potassium, Calcium, Barium, Iron oxides percentage in glass. Glass classes are as follow: 1 – building windows float processed, 2 – building windows non float processed, 3 – vehicle windows float processed, 5 – containers, 6 – tableware, 7 – headlamps.

Glass data is chosen for experimental investigation, because it is numerical and there is no missing data, so it is easier to visualize such data. Furthermore, it is wanted to compare glass data visualization done by multidimensional scaling using genetic algorithms with other authors' work.

In glass data chromosome consists from nine attributes (2–10), where each attribute is a gene. Every chromosome is a line from glass file without ID and class number. Glass data file consists of 214 lines which correspond to 214 chromosomes. Each chromosome is made from nine genes and it means that glass data is nine-dimensional.

Glass data is visualized by applying Matlab (Fig. 4). Genetic multidimensional scaling algorithm is applied and two dimensional points (vectors) are gained from nine-dimensional data. Then two dimensional points are visualized in two-dimensional space. It is noticed that two dimensional points usually form two clusters. The first cluster is from first, second and third glass classes and the second cluster is from seventh glass class. Other glasses do not form clusters and are scattered in two-dimensional space.

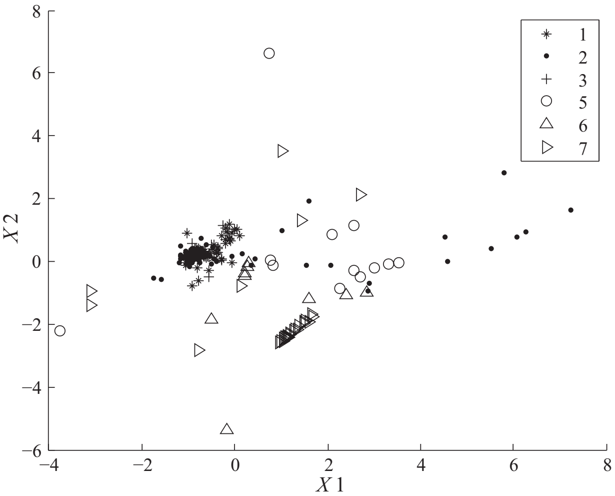

Fig. 4. Glass data visualized by genetic algorithm for multidimensional scaling

Other authors also carried out experiments with glass data. In paper (Stefanovič, Kurasova 2011) glass data is analyzed using artificial neural networks and observed how glasses are distributed into classes while teaching neural networks. K. Ząbkiewicz (2013) offers to combine principal component analysis and evolutional methods. The image which is gained by combining these two methods consists of two clusters. The first cluster is from first, second and third glass classes and the second is from seventh glass class. In paper (Ding *et al.* 2011) artificial neural network is combined with genetic algorithm. Neural network easily "stucks" in a local minimum, and the effectiveness of a new hybrid algorithm is greater than of these two methods separately. This hybrid algorithm with 94.44% probability separates building windows from other glass classes. In paper (Runkler, Bezdek 2013) three algorithms are compared: Sammon, principal component analysis (PCA) and MTP-MSO algorithm, which is a modification of particle swarm algorithm. Three images are gained by these methods. Every image has two clusters. One cluster is of 1, 2, 3 glass classes, second cluster is of 7 glass classes. In article (Benabdeslem, Lebbah 2007) glass data is visualized using HI-SOM method. There is one cluster made of 1, 2, 3 glass classes and other glass classes are scattered in two dimensional space. In paper (Carriosa, Pliastra 2010) there are two clusters. One cluster is from 1, 2, 3 glass classes and the other is from 7 glass class. Here glass data is visualized by Principal Separation and Shrinkage Analysis method.

Having compared glass data visualized by genetic multidimensional scaling algorithm to other authors' methods it has been noticed that data images usually consist of two clusters. One cluster is from 1, 2, 3 glass classes and the other is from 7 glass class. 7 glass class is headlamps. Because of the fact that headlamps (7 glass class) form a separate cluster, it means that the glass of headlamps is recognized and can be used in criminology as a proof to investigate crimes.

Conclusions

When multidimensional data is visualized, data that is in a higher-dimensional space is projected in a lower-dimensional space. It is expected that certain structures such as clusters and outliers are kept. By using multidimensional scaling it is expected to understand more than three dimensional data which human being does not perceive. In this article multidimensional scaling algorithm is analyzed and is showed that it can be combined with genetic algorithm to visualize data into a lower-dimensional space. The new hybrid algorithm is suitable to visualize multidimensional datasets.

References

Alba, E.; Dorronsoro, B. 2008. *Cellular genetic algorithms*. Springer. 248 p.

Benabdeslem, K.; Lebbah, M. 2007. Feature selection for self organizing map, in *29th International Conference on Information Technology Interfaces,* 25–28 June 2007, Cavtat, 45–50. http://dx.doi.org/10.1109/iti.2007.4283742

Brusco, M. J. 2001. A simulated annealing heuristics for unidimensional and multidimensional (city block) scaling of symmetric proximity matrices, *Journal of Classification* 18(1): 3–33. http://dx.doi.org/10.1007/s00357-0003-4

Carriosa, E.; Pliastra, F. 2010. *Principal seperation and shrinkage analysis* [online], [cited 25 October 2014]. Available from Internet: http://research.vub.ac.be/sites/default/files/uploads/BUTO/Working-Papers/mosi_working_paper_46_-_carrizosa_e._2010_principal_separation_and_shrinkage_analysis.pdf .

Ding, S.; Su, C.; Yu, J. 2011. An optimizing BP neural network algorithm based on genetic algorithm, *Artificial Intelligence Review* 36(2): 153–162. http://dx.doi.org/10.1007/s10462-011-9208-z

Dzemyda, G.; Kurasova, O.; Žilinskas, J. 2008. *Daugiamačių duomenų vizualizavimo metodai.* Vilnius: Mokslo aidai.

Groenen, P. J. F.; Heiser, W. J. 1996. The tunneling method for global optimization in multidimensional scaling, *Psychometrika* 61: 529–550. http://dx.doi.org/10.1007/BF02294553

Groenen, P.; Velden, M. 2004. *Multidimensional Scaling*. Econometric Institute Report EI 2004-15.

Hubert, L.; Arabie, P.; Hesson-Mcinnis, M. 1992. Multidimensional scaling in the city-block metric: a combinatorial approach, *Journal of Classification* 9(2): 211–236. http://dx.doi.org/10.1007/BF02621407

Karbauskaitė, R. 2005. *Daugiamačių duomenų vizualizavimo metodų, išlaikančių lokalią struktūrą, analizė*: Daktaro disertacija. Vytauto Didžiojo universitetas, Matematikos ir informatikos institutas.

Leeuw, J. 1977. *Applications of convex analysis to multidimensional scaling*, *Recent Developments in Statistics*. Amsterdam: North Holland Publishing company, 133–146.

Li, Te-S. 2006. Feature selection for classification by using a GA-based neural network approach, *Journal of the Chinese Institute of Industrial Engineers* 23(1): 55–64.

Mathar, R.; Žilinskas, A. 1993. On global optimization in two-dimensional scaling, *Acta Applicandae Mathematicae* 33: 109–118. http://dx.doi.org/10.1007/BF00995497

Miyano, H.; Inukai, Y. 1982. Sequential estimation in multidimensional scaling, *Psychometrika* 47: 321–361. http://dx.doi.org/10.1007/BF02294163

Runkler, T. A.; Bezdek, J. C. 2013. Topology preserving feature extraction with multiswarm optimization, *IEEE International Conference on Systems, Man, and Cybernetics*, 13–16 October 2013, Manchester, UK, 2997–3002. http://dx.doi.org/10.1109/SMC.2013.511

Stefanovič, P.; Kurasova, O. 2011. Influence of learning rates and neighboring functions an self organizing maps, *Lecture Notes in Computer Science* 6731: 141–150. http://dx.doi.org/10.1007/978-3-642-21566-7_14

UCI Machine Learning Repository [online], [cited 25 October 2014]. Available from Internet: http://www.ics.uci.edu/~mlearn/

Ząbkiewicz, K. 2013. Evolutionary nonlinear data transformation for visualization and classification task, in *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*, 8–11 September 2013, Krakow, Poland, 683–685.

Žilinskas, A.; Žilinskas, J. 2006a. On multidimensional scaling with Euclidean and city block metrics, *Ūkio ir technologijos vystymas* 7(1): 69–75.

Žilinskas, A.; Žilinskas, J. 2006b. Parallel hybrid algorithm for global optimization of problems occurring in MDS-based visualization, *Computers and Mathematics with Applications*, 52(1–2): 211–224. http://dx.doi.org/10.1016/j.camwa.2006.08.016

Žilinskas A.; Žilinskas, J. 2008. A hybrid method for multidimensional scaling using city-block distances, *Mathematical Methods of Operations Research* 68(3): 429–443. http://dx.doi.org/10.1007/s00186-008-0238-5

GENETINIŲ ALGORITMŲ TAIKYMAS DAUGIAMATĖMS SKALĖMS

A. Dzidolikaitė

Santrauka

Analizuojamas globaliojo optimizavimo uždavinys. Jis apibrėžiamas kaip netiesinės tolydžiųjų kintamųjų tikslo funkcijos optimizavimas leistinojoje srityje. Optimizuojant taikomi įvairūs algoritmai. Paprastai taikant tikslius algoritmus randamas tikslus sprendinys, tačiau tai gali trukti labai ilgai. Dažnai norima gauti gerą sprendinį per priimtiną laiko tarpą. Tokiu atveju galimi kiti – euristiniai, algoritmai, kitaip dar vadinami euristikomis. Viena iš euristikų yra genetiniai algoritmai, kopijuojantys gyvojoje gamtoje vykstančią evoliuciją. Sudarant algoritmus naudojami evoliuciniai operatoriai: paveldimumas, mutacija, selekcija ir rekombinacija. Taikant genetinius algoritmus galima rasti pakankamai gerus sprendinius tų uždavinių, kuriems nėra tikslių algoritmų. Genetiniai algoritmai taip pat taikytini vizualizuojant duomenis daugiamačių skalių metodu. Taikant daugiamates skales ieškoma daugiamačių duomenų projekcijų mažesnio skaičiaus matmenų erdvėje siekiant išsaugoti analizuojamos aibės panašumus arba skirtingumus. Taikant genetinius algoritmus gaunamas ne vienas lokalusis sprendinys, o visa optimumų populiacija. Skirtingi optimumai atitinka skirtingus vaizdus. Matydamas kelis daugiamačių duomenų variantus, ekspertas gali įžvelgti daugiau daugiamačių duomenų savybių. Straipsnyje genetinis algoritmas pritaikytas daugiamatėms skalėms. Parodoma, kad daugiamačių skalių algoritmą galima kombinuoti su genetiniu algoritmu ir panaudoti daugiamačiams duomenims vizualizuoti.

Reikšminiai žodžiai: daugiamatės skalės, genetiniai algoritmai, vizualizavimas.