ISSN 1392-8619 print/ISSN 1822-3613 online
ŪKIO TECHNOLOGINIS IR EKONOMINIS VYSTYMAS
TECHNOLOGICAL AND ECONOMIC DEVELOPMENT OF ECONOMY
http://www.tede.vgtu.lt
2006, Vol XII, No 1, 44–49

LARGE-SCALE DATA VISUALIZATION WITH MISSING VALUES

Sergiy Popov

National University of Radio Electronics, Lenin av. 14, 61166 Kharkiv, Ukraine
E-mail: Serge.Popov@ieee.org

Received 3 October 2005; accepted 4 January 2006

Abstract. Visualization of large-scale data inherently requires dimensionality reduction to 1D, 2D, or 3D space. Autoassociative neural networks with a bottleneck layer are commonly used as a nonlinear dimensionality reduction technique. However, many real-world problems suffer from incomplete data sets, i.e. some values can be missing. Common methods dealing with missing data include the deletion of all cases with missing values from the data set or replacement with mean or "normal" values for specific variables. Such methods are appropriate when just a few values are missing. But in the case when a substantial portion of data is missing, these methods can significantly bias the results of modeling. To overcome this difficulty, we propose a modified learning procedure for the autoassociative neural network that directly takes the missing values into account. The outputs of the trained network may be used for substitution of the missing values in the original data set.

Keywords: data visualization, dimensionality reduction, autoassociative neural network, network training, missing values, incomplete data set.

1. Introduction

When a scientist or an engineer faces a new problem, the first steps towards its solution are to understand what is given and which aspects are the most important. Since humans perceive most of the information in the course of their life in a visual form, it is preferable to present this new problem also in some kind of a visual form: directly or through some analogy, i.e. to visualize it. It is quite easy to visualize structures or logical relationships by means of flow charts and block diagrams. But when we come to data sets describing quantitative characteristics of objects or their relationships, the problems of dealing with high dimensionality arises.

People inherently are able to think only in 1D, 2D, and 3D spaces. On the other hand, most real-world scientific and engineering problems deal with tens to thousands of dimensions. Thus, presenting (visualizing) high-dimensional data in low-dimensional space requires dimensionality reduction. It is a technique intended to cut the number of dimensions while preserving maximum useful information in the data set.

Some of the well-known dimensionality reduction methods are the following:

- principal component analysis (PCA) [1, 2];
- principal curves [3, 4];
- multidimensional scaling [5, 6];
- autoassociative (bottleneck) artificial neural networks (AANN) [7, 8].

These and many other methods work seamlessly on complete data sets, when all numerical values are present, but most of them cannot be applied to data sets with missing values. The essence of the problem lies in mathematics: for the formulas to be computed all included variables must take some exact numerical values. When the value is missing, the formula cannot be computed at all, or it must be modified to omit this value. When missing values are positioned randomly in the data set, formulas cannot be modified to handle all possible situations, thus a way is sought to fill the missing values with some numerical values.

There are several simple methods to fill missing values:

1. When the number of samples with missing measurements is very small, discard these whole samples.

2. Replace missing values with mean or some "normal" (tolerable) value for this parameter.

3. If it is appropriate for the problem at hand, interpolate the value from the neighboring cells.

These methods have common drawbacks:

1. Missing data replacement (imputation) leads to biased estimates.

2. When the "restored" data set is presented to a dimensionality reduction algorithm, it does not "know" which values are true and which are replaced, and thus they have the same ranking in terms of their information load. On the other hand, dimensionality reduction implies information loss, so it is preferable to keep as much as possible information from the true data and completely ignore all missing data.

Thus, it is desirable to develop an algorithm that would explicitly handle missing data, eliminating the abovementioned drawbacks. For PCA such an algorithm exists, it is a well-known expectation maximization (EM) algorithm [9]. However, PCA provides only linear projection, and more efficient results can be obtained by employing nonlinear dimensionality reduction techniques.

Autoassociative (bottleneck) neural networks can be seen as the generalization of PCA to the nonlinear case. It is proven [10, 11] that if only linear activation functions are used and the network is optimally trained, it performs exactly the same projection as PCA. In this paper we propose modifications to standard learning procedures for AANN, which allow direct handling of the missing data, extract most information from the present data and estimate the missing values from the low-dimensional representation of the data set.

The paper is organized as follows: section 2 gives a short overview of standard AANN architecture and learning algorithms; in section 3 the proposed modifications are presented; section 4 supports theoretical findings with experimental evidence; and finally, conclusions are made on the basis of the obtained results.

The following notation is adopted:

X – $N \times D$ matrix containing the data set, where
N – number of samples,
D – original (high) dimensionality of the data;
$x(k)$ – k-th row of X, i.e. one sample;
$y(k)$ – low-dimensional representation of $x(k)$;
$\hat{x}(k)$ – reconstruction of $x(k)$, obtained from $y(k)$.

2. Autoassociative neural networks

AANN is a kind of feedforward neural network with multiple hidden layers. Depending on the architecture and activation functions used, AANN can perform linear or nonlinear mapping.

General AANN architecture [7, 8] is presented in Fig 1. It consists of input and output layers (with linear activation functions) with the number of neurons equal to the original dimensionality of the data. The first and the third hidden layers (with nonlinear activation functions) contain equal number of neurons which is chosen according to the problem at hand. The second hidden layer (with linear activation functions) is the "bottleneck" layer which number of neurons is equal to the target low dimensionality. The outputs of the network are extracted from this layer. Such a network can be considered as two parts: input and the first two hidden layers form a multilayer perceptron (MLP1) with one hidden layer that performs nonlinear mapping $x(k) \Rightarrow y(k)$; the second and the third hidden layers with the output layer form the second multilayer perceptron (MLP2) that solves the reverse problem of reconstructing the original data $y(k) \Rightarrow \hat{x}(k)$. The idea is to "squeeze" high-dimensional data through a low-dimensional "bottleneck" (the second hidden layer) so that the reconstruction $\hat{x}(k)$ is as close to the original data $x(k)$ as possible, i.e. maximum of information is retained. Thus the network inputs are also used as learning targets.

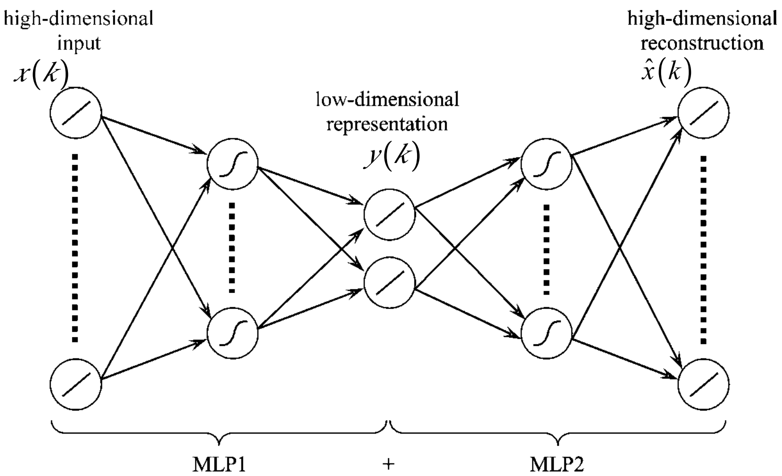
high-dimensional input
$x(k)$
low-dimensional representation
$y(k)$
high-dimensional reconstruction
$\hat{x}(k)$
MLP1
+
MLP2

Fig 1. General AANN architecture

To achieve this goal, the network is trained in the supervised mode with respect to the following criterion

$$E = \sum_{k=1}^{N} \left\| x(k) - \hat{x}(k) \right\|^2 . \tag{1}$$

If only linear mapping is required, the first and the third hidden layers (with nonlinear activation functions) are unnecessary and the general architecture can be simplified (Fig 2).

AANN can be trained with any learning algorithm suitable for feedforward neural network. Most of them are based on the backpropagation procedure to calculate error gradients for hidden layers. Since for our further consideration, the choice of a particular learning algorithm does not matter, we will not focus on this issue.

3. Modified learning procedure

Now consider the case of missing values. The goal is to eliminate their influence on the network output and weights update.

The network outputs are formed by feeding forward the inputs through the network layers. The only layer that directly receives the inputs $x(k)$ is the first hidden layer. The weighted inputs are accumulated to form the neurons activations as follows:

$$a_j = \sum_{i=1}^{D} w_{ji} x_i , \tag{2}$$

where a_i – activation of *j*-th neuron, x_i – *i*-th input, w_{ji} – the corresponding synaptic weight.

If the input is missing, it is natural to exclude the corresponding term from summation, which is equivalent to setting the corresponding x_i to zero. In this way, missing value does not influence the sum (the neuron activation), hence, the neuron output, hence, the network output.

The network learning is basically an optimization procedure performed with respect to criterion (1). Ideally, the absolute value of *E* can drop to 0, if the network perfectly reconstructs its inputs. In reality it is always above 0 and the goal of learning is to minimize it by adjusting the synaptic weights of the network. Obviously, higher errors at particular network outputs lead to bigger adjustment of weights. On the contrary, zero errors lead to no adjustment. Thus, to eliminate the influence of missing values on the network learning, it is necessary to zero out errors at those outputs, where the target values are missing.

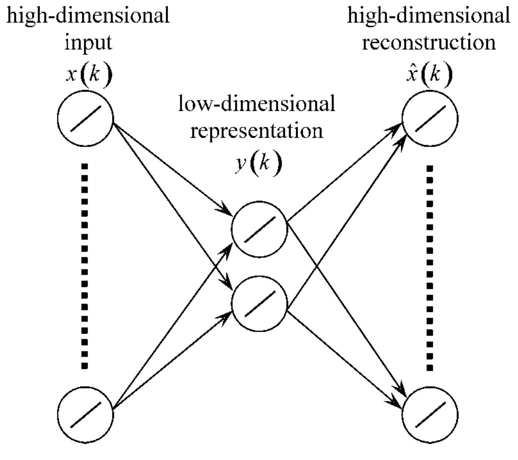
high-dimensional input
$x(k)$
low-dimensional representation
$y(k)$
high-dimensional reconstruction
$\hat{x}(k)$

Fig 2. Simplified AANN architecture for linear mapping

Such a modification to weight update scheme has a very important advantage: it is equivalent to weighting output errors with 1 (when the target is present) and 0 (when the target is missing), thus shifting the learning “attention” only to real data and completely discarding missing values. Hence, maximum retention of useful information from the data set is achieved in the course of dimensionality reduction.

When the learning procedure converges, the outputs of the network may be used to replace the corresponding missing values in the data set. If the task of missing value restoration is the primary one, the outputs of the network, where the target values are missing, may be fed back to the corresponding inputs. This will lead to an iterative process of missing value reconstruction.

Thus, the proposed modifications can be summarized in the following two rules:

1. Replace missing inputs with zeros.
2. Replace learning errors with zeros where targets are missing.

4. Experimental results

The proposed approach was applied to a real-world problem of biomedical data visualization. The data set contains blood tests (35 parameters) for 26 patients taken before and after treatment (total of 52 samples). Dimensionality reduction is performed from 35D to 2D. To compare different algorithms, 20% of data is randomly discarded, so we have both full and incomplete data sets.

The reference visualization is obtained by applying standard PCA to the full data set (Fig 3). Each line represents one patient, the end with a solid circle corresponds to the blood test taken before treatment, and the end with an empty circle corresponds to the blood test taken after treatment. The incomplete data set is visualized using three different approaches: Fig 4 – missing data are replaced with mean values for the corresponding parameter and then standard PCA is applied; Fig 5 – EM algorithm is applied directly to the incomplete data set; Fig 6 – the proposed modified AANN method is applied directly to the incomplete data set. Linear AANN (35-2-35 architecture) was used because we are comparing to linear PCA methods.

Visual analysis of the obtained visualizations reveals the following:

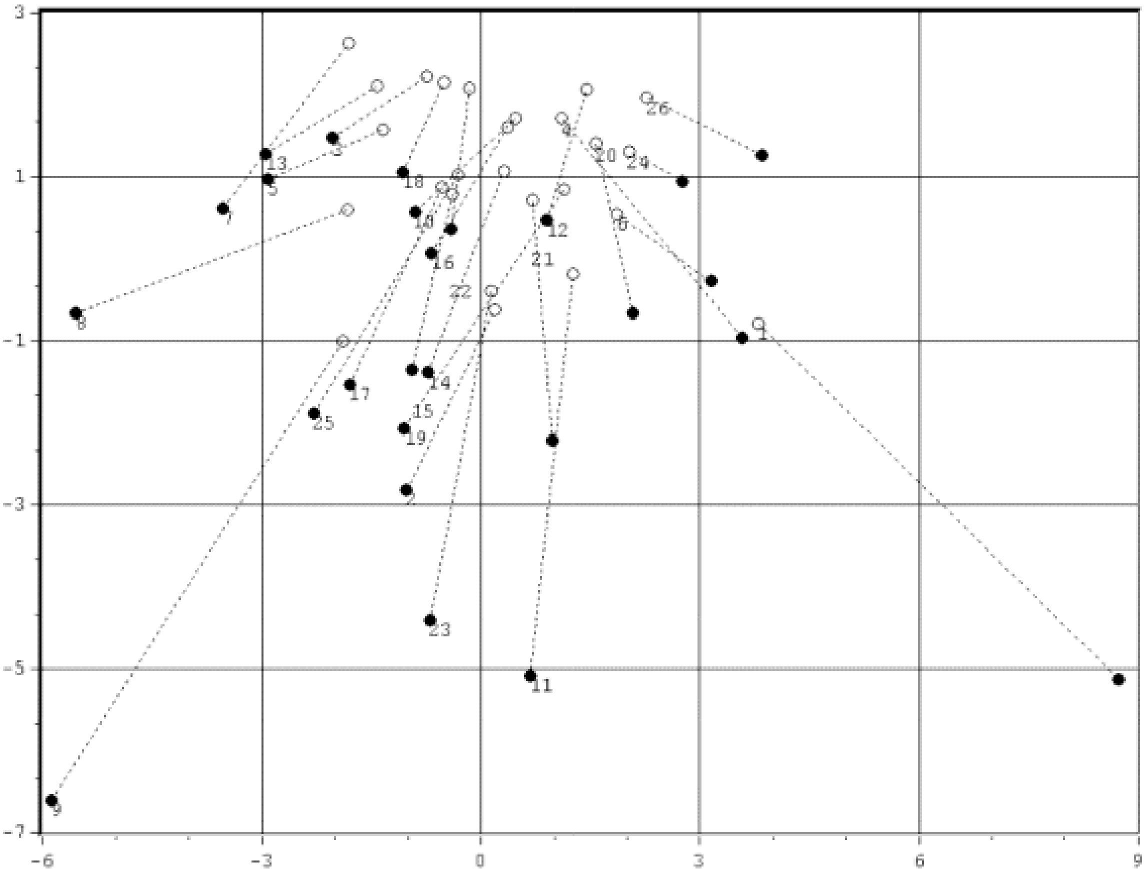

Fig 3. Visualization of complete data set

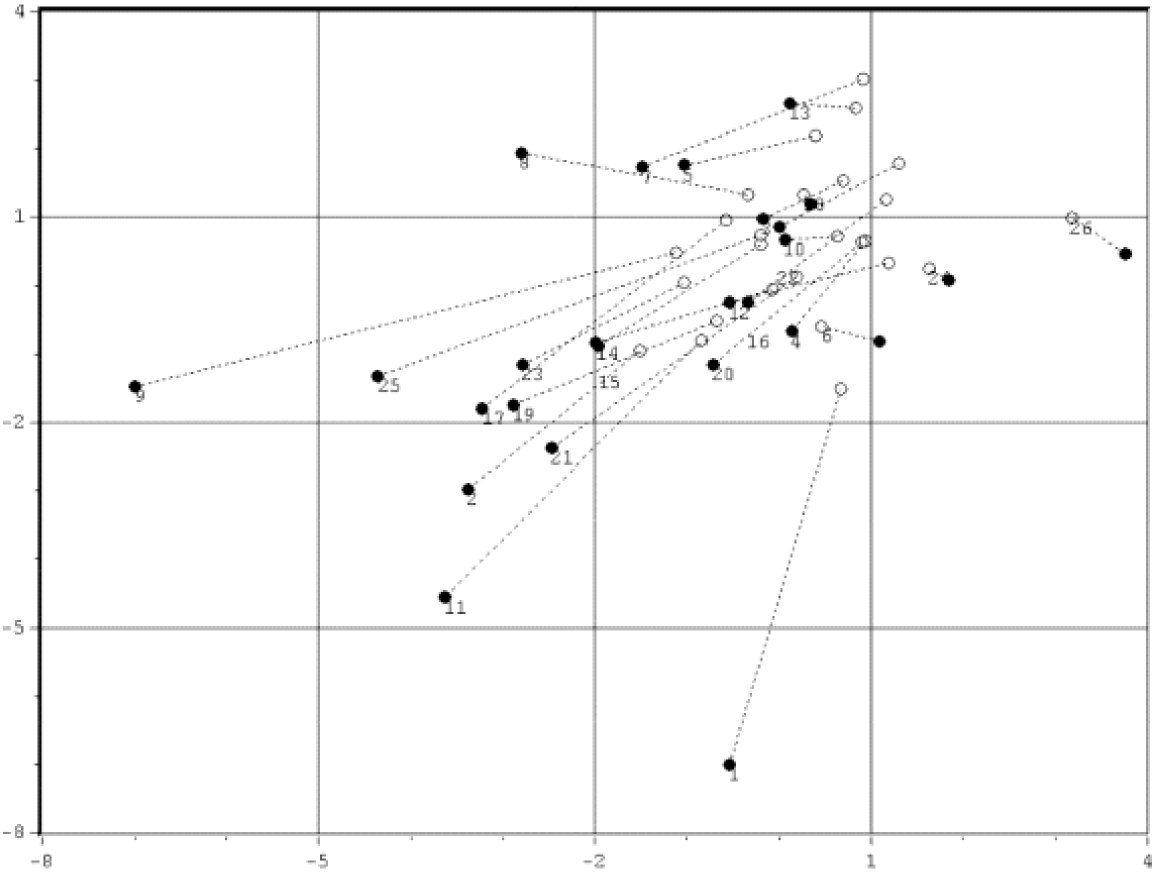

Fig 4. Visualization of incomplete data set with PCA (missing values are replaced by mean)

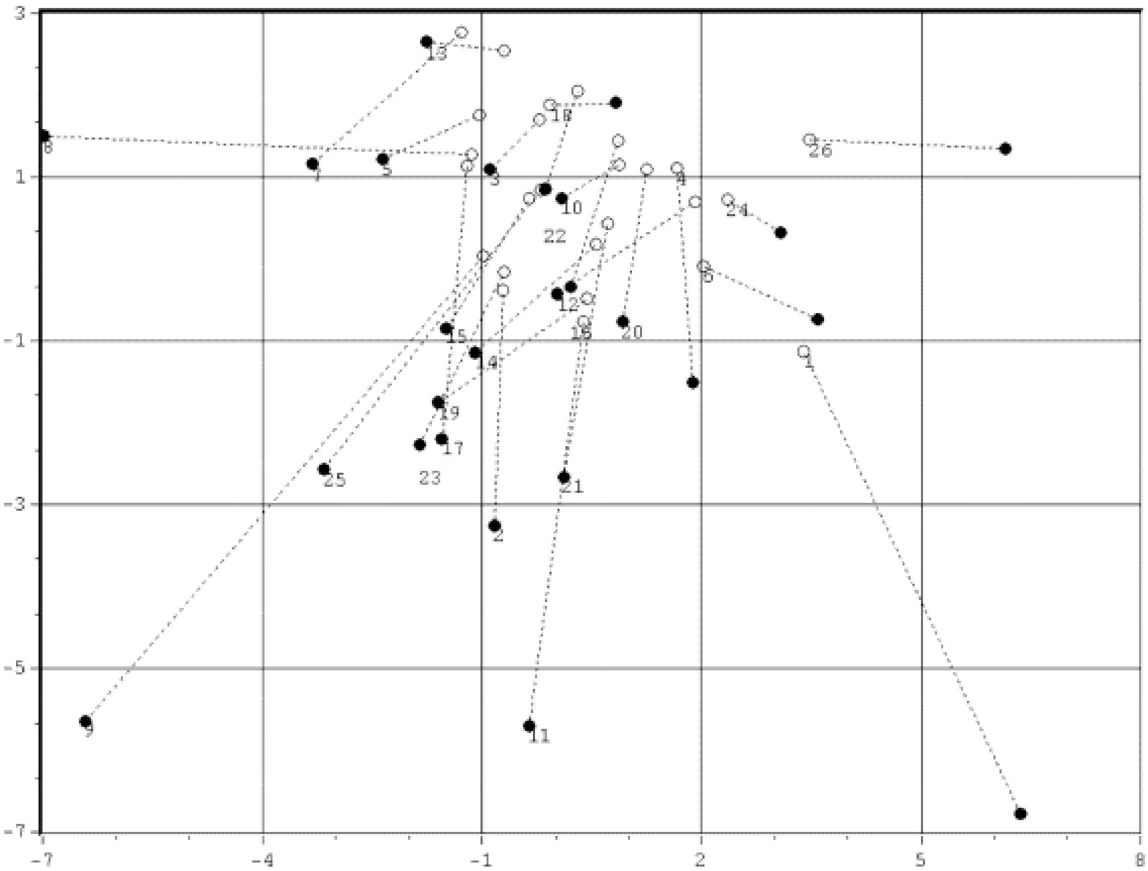

Fig 5. Visualization of incomplete data set with PCA using EM algorithm

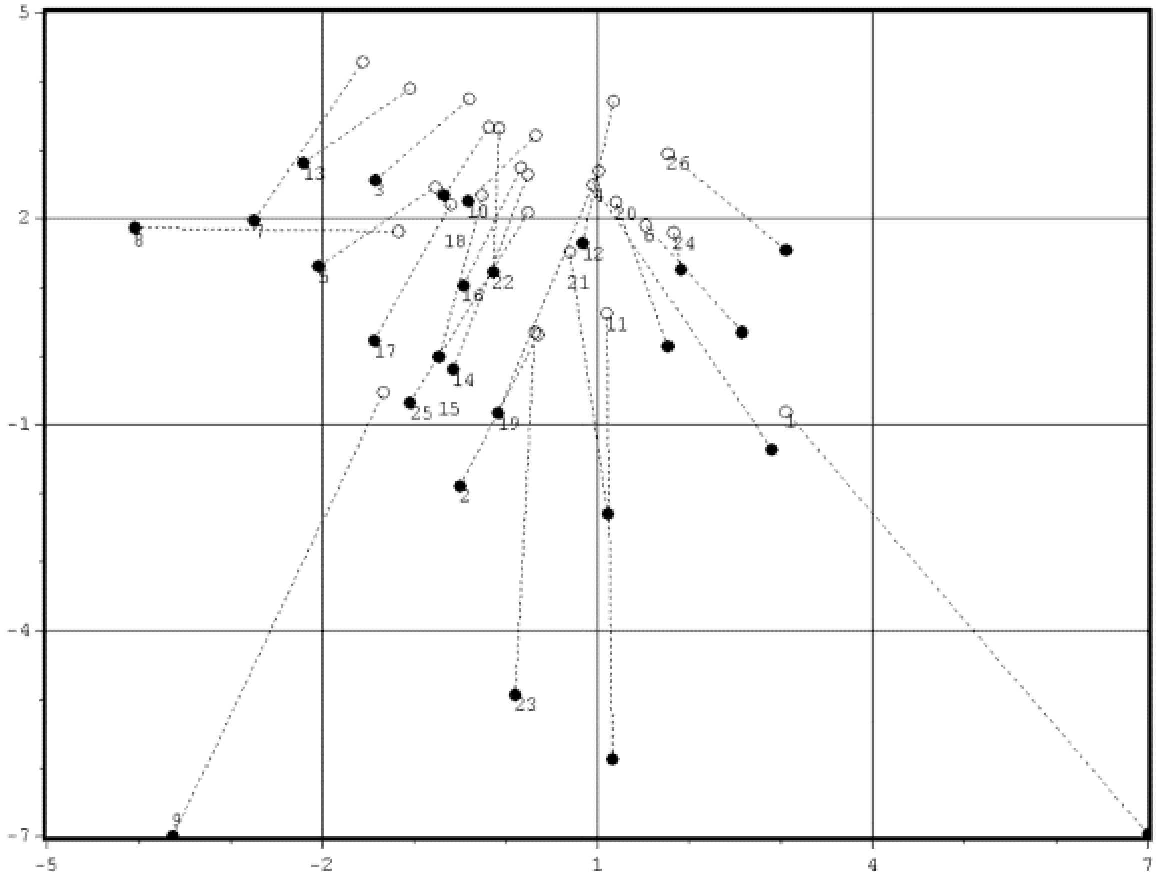

Fig 6. Visualization of incomplete data set with the proposed approach

1. Replacing of missing data by mean led to a severe distortion of the data cloud shape and interrelations between data points. This could lead to wrong conclusions, if this visualization was used for decision making or express diagnostics.

2. EM algorithm performed much better than the previous method. The overall shape is only slightly distorted, but interrelations between data points are still wrong in many cases and are closer to the result of the previous method than to the reference.

3. The proposed approach yielded the best visualization in terms of its closeness to the reference PCA visualization of the complete data set. The shape is preserved almost precisely and data points interrelations are only slightly distorted.

5. Conclusions

In this paper we proposed modifications to AANN learning procedures that allow direct handling of data sets with missing values. One of its most important advantages for dimensionality reduction is the ability to preserve most of information from the present data while completely ignoring missing values that is very useful when a substantial portion of a data set is missing. This approach can be used for both linear and nonlinear dimensionality reduction and does not depend on the network architecture and learning algorithm, i.e. any supervised learning can be applied, any type of neural network can be used that performs weighted summation of inputs. The proposed approach can be easily generalized to other types of feedforward neural networks that are trained by supervised learning algorithms.

The comparison of experimental results has shown the superiority of the proposed method over other approaches to missing data handling, in particular, the replacement of missing values by mean values and the expectation maximization algorithm.

References

1. Jolliffe, I. T. Principal component analysis. Springer Series in Statistics. New-York: Spinger-Verlag, 1986.
2. Pearson, K. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Sciences*, No 6, 1901, p. 559–572.
3. Hastie, T.; Stuetzle, W. Principal curves. *Journal of the American Statistical Association*, Vol 84, 1989, p. 502–516.
4. Kegl, B.; Krzyzak, A.; Linder, T.; Zeger, K. Learning and design of principal curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 22, 2000, p. 281–297.
5. Kruskal, J. B.; Wish, M. Multidimensional Scaling. Newbury Park: Sage Publications, 1978.
6. Torgerson, W. S. Multidimensional scaling: I. Theory and method. *Psychometrika*, Vol 17, 1952, p. 401–419.
7. Kramer, M. A. Nonlinear principal component analysis using autoassociative neural networks. *Journal of the American Institute of Chemical Engineers,* Vol 37, 1991, p. 233–243.
8. Oja, E. Data compression, feature extraction, and autoassociation in feedforward neural networks. In: Proceedings of the International Conference on Artificial Neural Networks. Edited by T. Kohonen, M. Makisara, O. Simula and J. Kangas, Vol 1, 1991, p. 737–745.
9. Roweis, S. EM algorithm for PCA and SPCA. *Neural Information Processing Systems*, Vol 10, 1997, p. 626–632.
10. Baldi, P.; Hornik, K. Neural networks and principal component analysis: learning from examples without local minima. *Neural Networks*, Vol 2, 1989, p. 53–58.
11. Bourlard, H.; Kamp, Y. Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, Vol 59, 1988, p. 291–294.

DAUGIAMATĖS ERDVĖS DUOMENŲ GRAFINIS VAIZDAVIMAS, KAI TRŪKSTA REIKŠMIŲ

S. Popov

Santrauka

Vaizduojant daugiamatę informaciją, paprastai reikia ją transformuoti į vienmatę, dvimatę arba trimatę erdvę. Nelinijinei daugiamatės erdvės transformacijai paprastai naudojami autoasociatyvieji neuroniniai tinklai. Tačiau, dažnai sprendžiant realias problemas, dalis informacijos dingsta. Taikant tradicinius metodus, elgiamasi dvejopai: trūkstama informacija ignoruojama; trūkstamos reikšmės pakeičiamos vidutinėmis arba tam tikromis konkrečiam kintamajam būdingomis reikšmėmis. Šie metodai tinka tada, kai trūksta tik kelių reikšmių. Kai trūksta daugelio duomenų, minėtieji metodai gali labai iškraipyti modeliavimo rezultatus. Šiai problemai išspręsti autoriai pasiūlė procedūrą, kuri, naudojant autoasociatyvųjį neuroninį tinklą duomenų transformacijai, įvertina trūkstamas reikšmes. Skaičiavimo rezultatai, tobulesniu neuroniniu tinklu gali būti naudojami trūkstamoms reikšmėms pirminėje duomenų aibėje pakeisti.

Pagrindiniai žodžiai: duomenų vaizdavimas, daugiamatės erdvės transformacija, autoasociatyvusis neuroninis tinklas, tinklo tobulinimas, neišsamių duomenų imtis.

Sergiy POPOV. Doctor of Science, Senior Researcher. Control Systems Research Laboratory National University of Radio Electronics of Kharkiv. Degree in computer aided design (1998), Ph.D. in control systems and processes (2001). Member of IEEE since 2000 (Computational Intelligence Society). Author of more than 50 scientific papers published in Germany, Greece, Japan, Lithuania, Russia, Turkey, Ukraine, and United Kingdom.

Research interests: the use of computational intelligence technologies in data processing and prediction problems under the conditions of structural and parametric uncertainty.