Nonlinear dimensionality reduction of large datasets for data exploration

V. Tomenko & V. Popov
Wessex Institute of Technology, Southampton, UK

Abstract

Dimensionality reduction techniques are outlined; their strengths and limitations are discussed. The novel dimensionality reduction method is presented, which is a combination of input space approximation, nonlinear dimensionality reduction and function approximation techniques. The method is especially useful for large scale real-world datasets, where existing methods fail to succeed because of extreme computational expenses. The method can be used in exploratory data analysis and aims to create low dimensional data representation for better data structure understanding and for cluster analysis. The comparison of dimensionality reduction techniques is performed in order to justify the applicability of the proposed method.
Keywords: dimensionality reduction, self-organizing neural network, data exploration, function approximation.

1 Introduction

Advances in data collection and storage capabilities during the past decades have led to data overload in most sciences. In such diverse domains as medicine, biology, economics, astronomy and engineering, researchers have collected huge datasets with larger and larger number of observations and of high dimensionality. The dimension of the data is determined by the number of variables that are measured on each observation. Such datasets, in contrast with smaller, more traditional ones that have been studied extensively in the past, present new challenges in data analysis, yet they must be processed and understood in order to extend our knowledge in different domains.

Traditional statistical tools fail to meet the requirements of high-dimensional data processing because of the increase in the number of variables associated

WIT Transactions on Information and Communication Technologies, Vol 37, © 2006 WIT Press
www.witpress.com, ISSN 1743-3517 (on-line)
doi:10.2495/DATA060011

with each observation and also because of the increasing number of observations. High-dimensional datasets present many mathematical challenges and are bound to give rise to new theoretical developments. One of the problems with high-dimensional datasets is that, in many cases, not all the measured variables are "meaningful" for understanding the underlying phenomena of interest. Therefore it is of interest in many applications to reduce the dimensionality of the original data prior to any modeling.

In mathematical terms, the problem can be stated as follows: given the p-dimensional random variable $x = (x_1, \ldots, x_p)^T$, find a lower dimensional representation of it, $y = (y_1, \ldots, y_k)^T$ with $k \leq p$, that captures the content in the original data, according to some criterion. Different fields use different names for the multivariate vectors: the term "variable" is mostly used in statistics, while "feature" and "attribute" are alternatives commonly used in data mining and machine learning literature.

In this paper traditional dimensionality reduction techniques are outlined, their strengths and limitations are discussed. The nonlinear dimensionality reduction method is presented, which is a combination of data distribution modelling, function approximation and nonlinear projection technique. The method can be utilized in exploratory data analysis. The main advantage of the method is its applicability to real-world datasets with huge number of observations in contrast to existing techniques which fail because of extreme computational expenses. Finally the experimental results are presented comparing the performance of different methods.

2 Dimensionality reduction techniques

Two major types of dimensionality reduction techniques include linear and nonlinear techniques. Linear techniques result in each of the components of the new variable being a linear combination of the original variables. Among the most widely used methods are principal component analysis (PCA), projection pursuit and independent component analysis [1]. Nonlinear techniques maximize or minimize a function of a large number of variables iteratively, and although they are generally beneficial to linear techniques, yet computationally expensive. The commonly used nonlinear methods are principal curves, multidimensional scaling and Sammon reconstruction (SR) [2]. Additionally, particular neural network models [3] are regarded as implementation of both linear and nonlinear approaches. In the foregoing sections two representative methods are outlined, namely: PCA and SR.

2.1 Principal component analysis

If no category information about the patterns (category labels) is available, the principal component analysis (eigenvector projection) is used. A linear projection that replaces features in the raw data by uncorrelated ones is defined by the eigenvectors of the covariance matrix $\Re$:

$$\Re = (1/n)\mathrm{A}^{\mathrm{T}}\mathrm{A} \tag{1}$$

where A is an $n \times p$ matrix of raw data patterns. Eigenvalues of $\Re$ are real, because $\Re$ is a $n \times n$ positive definite matrix. They can be labeled so that:

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0 \tag{2}$$

The set of eigenvectors, also called *principal components*, c_1, c_2, c_n, is labeled according to corresponding eigenvalues. The $k \times p$ transformation matrix H_k is defined from the eigenvectors of the covariance matrix as follows:

$$\mathrm{H}_k = \left[c_1^T, c_2^T, \ldots, c_k^T\right]^T \tag{3}$$

The projected patterns can be written as follows:

$$B_m = \left[y_1^T, y_2^T, \ldots, y_n^T\right]^T = \left[x_1^T, x_2^T, \ldots, x_n^T\right]^T = \mathrm{H}_k^T = \mathrm{AH}_k^T \tag{4}$$

The covariance matrix in the new k-dimensional space becomes a diagonal matrix. This implies that the new k features obtained as result of linear transformation defined by H_k are uncorrelated.

2.2 Sammon reconstruction

In SR [4] the cost function being minimized is determined by the difference in distances in original p-dimensional space and new k-dimensional space, normalized by the distances in the original pattern space. So, because of normalization, the preservation of small distances is emphasized.

Assume $d_{ij}^* = \left(x_i, x_j\right)$ are the distances between patterns in the original space and $d_{ij} = \left(y_i, y_j\right)$ – in k-dimensional space. To define distance Euclidean measure can be utilized. The goal of SR is to find values for y_i features that minimize cost function E:

$$E(D, A) = (1/c)\sum_{i<j}^{n}\left[d_{ij}^* - d_{ij}\right]^2 / d_{ij}^* \tag{5}$$

where

$$c = \sum_{i<j}^{n} d_{ij}^* \tag{6}$$

$$d_{ij} = \sqrt{\sum_{z=1}^{k}\left[y_{iz} - y_{jz}\right]^2} \tag{7}$$

and y_{ij} defines j-th feature of y_i.

To minimize E simplified Newton method can be applied:

$$y_{pk}(k+1) = y_{pk} - \eta\Delta_{pk}(k) \tag{8}$$

where

$$\Delta_{pk}(k) = \left(\frac{\partial E}{\partial y_{pq}}\right) \Big/ \left(\left|\frac{\partial^2 E}{\partial y_{pq}^2}\right|\right) \tag{9}$$

In equation (9), $\partial E / \partial y_{pq}$ is the respective component of gradient of E and $\left|\partial^2 E / \partial y_{pq}^2\right|$ is the diagonal element of Gaussian matrix, defined on k-th iteration. η is a learning rate, chosen from interval [0,3; 0,4]. If error function is defined as equation (5), respective components of gradient and Gaussian matrix are defined as follows:

$$\frac{\partial E}{\partial y_{pq}} = -\frac{2}{c} \sum_{j=1, j\neq p}^{n} \left[\frac{d_{pj}^* - d_{pj}}{d_{pj}^* d_{pj}} \right] \left[y_{pj} - y_{jq} \right] \tag{10}$$

$$\frac{\partial^2 E}{\partial y_{pq}^2} = -\frac{2}{c} \sum_{j=1, j\neq p}^{n} \frac{1}{d_{pj}^* d_{pj}} \left[\left(d_{pj}^* - d_{pj} \right) - \frac{\left(y_{pq} - y_{jq} \right)^2}{d_{pj}} \left(1 + \frac{d_{pj}^* - d_{pj}}{d_{pj}} \right) \right] \tag{11}$$

2.3 Survey of dimensionality reduction techniques

The dimensionality reduction techniques differ by the criteria they have to optimize. Dimensionality reduction for exploratory data analysis enables high-dimensional data visualization for data structure understanding. In feature extraction for classification, it is desirable to extract high discriminative reduced-dimensionality features which reduce the classification computational requirements.

However, dimensionality reduction criteria for exploratory data analysis regularly are minimization of error functions, such as interpattern distance difference whereas feature extraction criteria for classification aim to increase class separability as much as possible. Hence, the optimum extracted features (regarding a specific criterion) calculated for exploratory data analysis are not necessarily the optimum features regarding class separability and vice versa. Consequently, dimensionality reduction techniques for exploratory data projection are not generally used for classification and vice versa.

For example in PCA, if the patterns are projected onto an k-dimensional space by equation (4), their scatter is maximized in the new space with respect to all other orthogonal k-dimensional projections because equation (4) uses eigenvectors corresponding to k largest eigenvalues of $\Re$. Therefore, PCA is widely used for feature extraction in classification problems.

Nonlinear techniques have become popular because of inability of linear methods to preserve so called "*complex data structures*". These are patterns that lie on a curved surface. If patterns lie along a helix in three dimensions, PCA will not be able to obtain good two-dimensional representation of such data set. On the other hand, Sammon reconstruction attempts to spread data as widely as possible and consequently is mainly applied for visualization purposes.

Most linear and nonlinear techniques, including PCA and SR suffer from large number of data vectors in the dataset. Additionally, nonlinear techniques, which iteratively minimize predefined cost function, are extremely computationally expensive in case of large datasets.

3 The novel projection method

As it was stated above, dimensionality reduction techniques are heavily dependent on the number of vectors in the dataset. PCA requires $n \times n$ covariance matrix, SR creates $n \times n$ distance matrix and $n \times k$ projection matrix, where n is the number of observation vectors. Furthermore, both PCA and SR require the whole dataset to be processed in order to create "meaningful" data representation or feature extraction. Consequently, they are not applied to datasets exceeding certain size and neural network based approaches are used instead, e.g. so called *distance image* [3], derived from trained Kohonen's self-organizing map (SOM) [2, 3, 5, 6]. Although SOM was found to be useful for high-dimensional data representation, it is usually difficult to draw a conclusion about cluster shapes, their separability and mutual arrangement by utilizing distance image alone.

Generally, real-world datasets are extremely large and high-dimensional. Moreover, such datasets are usually noisy and differ in actual and intrinsic dimensionalities, e.g. data points lie in subspaces of the given space of dimensionality p and these subspaces may be nonlinear. Consequently, linear dimensionality reduction techniques fail to represent data structures adequately. Therefore, it is desirable to utilize nonlinear dimensionality reduction techniques for data representation and to extend their applicability to large scale datasets.

The proposed method combines three approaches, which are applied in a stepwise manner (Fig. 1):

Approximation of input space: reducing the number of data patterns by representing similar in some sense patterns by a single prototype vector;

Nonlinear dimensionality reduction: projection of the set of prototype vectors onto a space of lower dimensionality with the help of nonlinear dimensionality reduction technique;

Function approximation: approximating the function, which maps patterns from original p-dimensional space onto k-dimensional space $(k \le p)$ by utilizing the resulting input-output pairs of dimensionality reduction.

After function approximation step, either whole dataset or any subset of desired size may be projected onto two- or three-dimensional space. In the foregoing sections the actual implementation will be described in detail.

3.1 Approximation of input space

Input space can be approximated with the help of competitive learning neural networks. In the process of competitive learning, different neurons become more sensitive to different categories of input patterns. In other words, neurons are trained to "represent" different data categories. This specialization is the result of competition between neurons: when input pattern x is presented, the winner is the neuron that represents the category, to which given pattern belongs, better than others. Then, neuron's weights are adapted for better representation of given category. Later on, the probability for this neuron to become winner for another pattern from current category becomes higher.

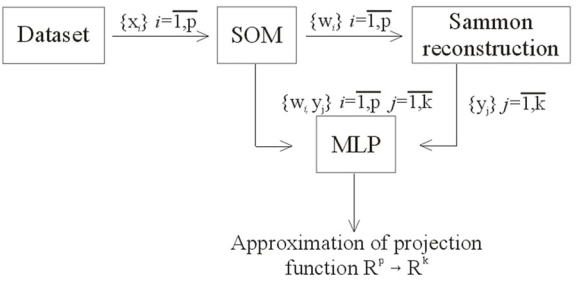

Figure 1: Scheme of the proposed method.

Competitive learning models are distinguished by the presence or absence of topological structure (links between neurons in output space) and by the dimensionality of this structure. Typical applications of models without topological structure, e.g. *winner-takes-all* (WTA), *neural gas* (NG) and *growing neural gas* (GNG) include *vector quantization* and *clustering* [7, 8]. If neurons are ordered, i.e. form some topological structure, competitive learning can be generalized: if not only weights of the winner, but also the weights of the neighboring neurons are adapted, these neighboring neurons will specialize in representing close in terms of input space categories. Therefore, the mapping of input space will be ordered. This property extends the application of competitive learning to creation of data abstractions and data visualization. The representative models of this type include SOM, growing cell structures (GCS), growing grid (GG) [7] and CGG [8].

Since it is important to compare different data visualization techniques the competitive learning model chosen for input space approximation was SOM. Therefore, this allows comparing distance image approach with the proposed method.

SOM learning process consists of ordering phase and fine-tuning phase. At the ordering phase, the neighborhood radius and learning-rate factor (step) are chosen big enough to allow all neurons change their reference vectors (prototype vectors) when different input patterns are presented. The neighborhood radius significantly decreases with iterations. At the end of the ordering phase the ordering of neurons in input and output spaces resembles each other.

During fine-tuning phase the neighborhood radius continues to decrease monotonically, but much slower, than during first phase. Together with the small values of learning-rate factor these changes allow adjusting reference vectors more precisely.

SOM algorithm is comprised of two steps [6]. When input pattern x is presented, neuron with the nearest reference vector is defined. Then, weights of the nearest neuron and its topological neighbors are updated according to:

$$w_i(t) = w_i(t-1) + \eta(t)G_i(t)(x(t) - w_i(t-1)) \quad (12)$$

where $\eta(t)$ is the learning-rate factor on iteration t, $G_i(t)$ is the neighboring function value. These two steps are repeated $t_{\max}$ iterations.

The result of input space approximation is the set of prototype vectors, which approximate input space and reduce the amount of data.

3.2 Nonlinear dimensionality reduction

In order to compile learning dataset for function approximation step (input-output pairs), the prototype vectors estimated at the previous step should be projected onto a lower dimensional space. The choice of dimensionality reduction technique is dependent on the desired result. If feature extraction is of interest, PCA should be used to maximize the scatter and minimize squared error. However, since visualization is the main goal of this study, some nonlinear technique should be utilized to deal with complex data structures. Therefore, Sammon reconstruction (Section 2.2) was chosen to reduce the dimensionality of prototype vectors. The output of this step is the set of input-output vector pairs $\langle x, d \rangle$, where x is the original prototype vector of dimensionality p, and d is its projection onto k-dimensional space $(k \le p)$.

3.3 Function approximation

The goal of this step is to approximate the unknown function, which maps patterns from original p-dimensional space onto a space of lower dimensionality k. Combination of input space approximation and function approximation allows approximating this function not only for prototype vectors, but for the whole dataset.

Two widely used in practice models for function approximation include Multilayer Perceptron (MLP) and Radial-Basis Function networks (RBF) [9]. Both have inherent advantages and disadvantages (MLP is slower but RBF requires much more units in the hidden layer with the increase in dimensionality of input space). In the current study MLP was chosen to perform function approximation [9].

4 Experimental results

Three datasets were chosen for evaluation purposes. The first dataset is comprised of seven non-overlapping clusters generated by normal distribution (Fig. 2(a)). The PCA and SR transformation are shown in Fig. 2(b) and (c) respectively. Distance image for SOM is shown in Fig. 2(d) and the result of the method described (MLP projection) is presented in Fig. 2(e). PCA fails to create understandable 2D visualization, whereas results obtained by SR and MLP projection are satisfactory and resemble each other.

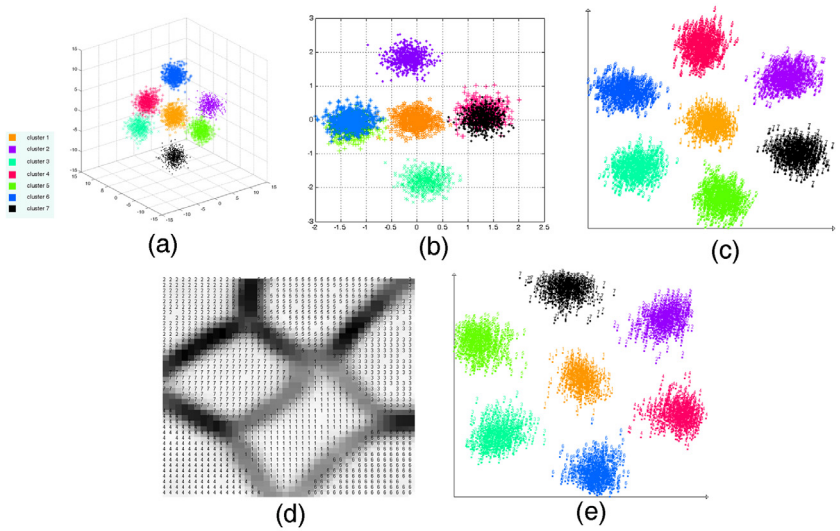

Figure 2: Clustered data.

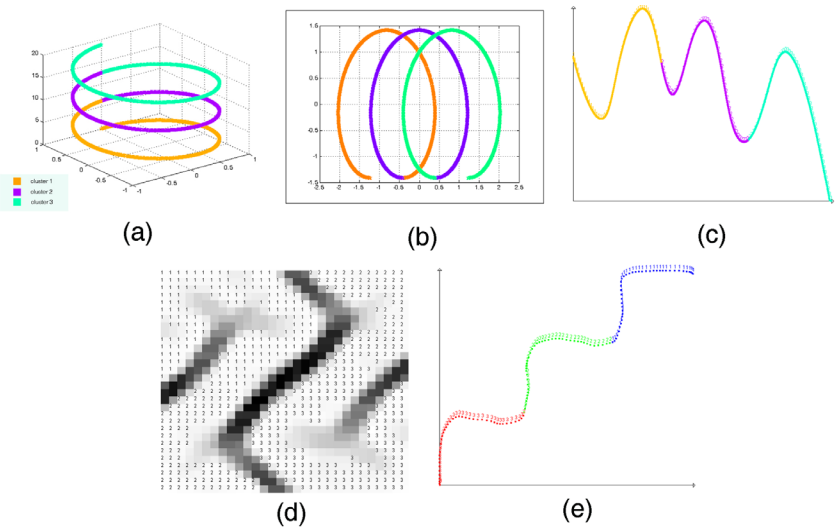

Figure 3: Spiral.

The second dataset is the three-dimensional spiral (Fig. 3(a)). Again, SR (Fig. 3(c)) and MLP projection (Fig. 3(e)) results resemble each other, whereas PCA performance is comparatively poor.

Finally, the real-world dataset of forest types was utilized [11]. The results of MLP projection, which are shown in Fig. 4 (the dataset is too big to be projected either by PCA or by SR) explain poor experimental classification results for this

dataset [12] (classes are inseparable). The 2D representation obtained by MLP projection resembles PCA projections of randomly chosen subsets of given dataset (11000 patterns in each subset).

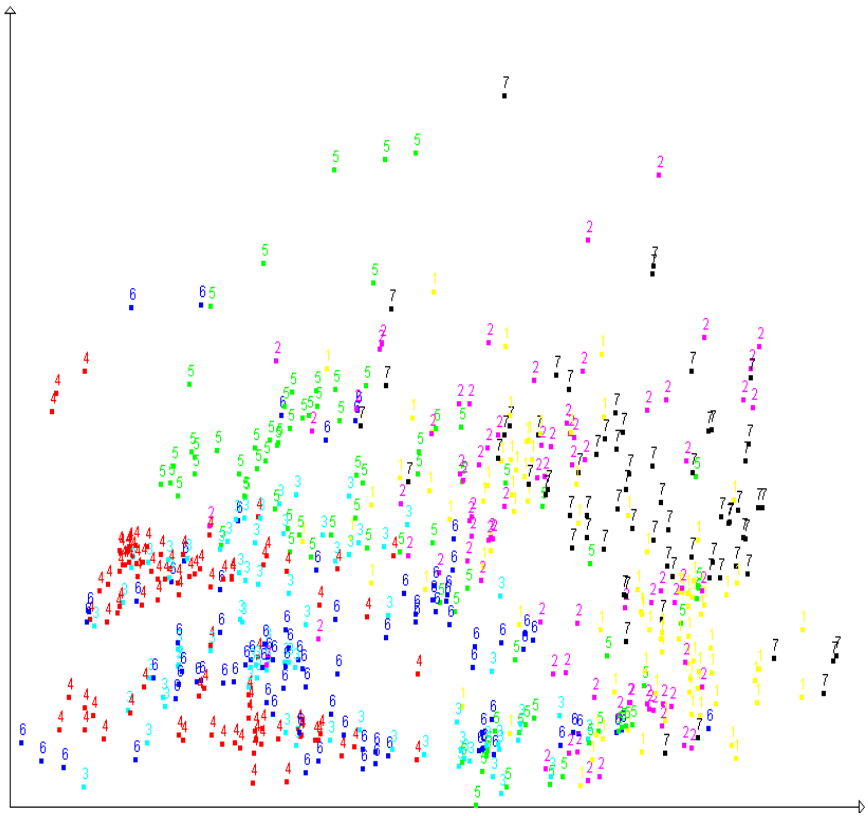

Figure 4: Forest cover type dataset projection.

For quantitative comparison, equation (5) was used. Results for four representative datasets are summarized in Table 1. The experiments on a range of datasets [13] suggest that MLP projection results are acceptably close to those of SR.

Table 1: Quantitative comparison.

	SR	**MLP projection**
Clustered data	0.0432	0.0455
Spiral	0.0015	0.0018
Iris	0.0054	0.0060
Wine	0.0588	0.0609

5 Conclusions

The method proposed in the study allows reducing the dimensionality of large scale high-dimensional datasets. The goal of this reduction may vary with respect to task, which can be either data visualization or feature extraction for classification. Experimental results prove the applicability and efficiency of the method.

References

[1] Jain, A. K., Dubes, R. C., *Algorithms for Clustering Data*, Prentice Hall: Englewood Cliffs, New Jersey, 1988.

[2] Fodor, I.K**.,** A Survey of Dimension Reduction Techniques, *LLNL technical report*, 2002. UCRL-ID-148494.

[3] J. Mao, and A. K. Jain, Artificial neural networks for feature extraction and multivariate data projection, *IEEE Trans. Neural Networks*, vol. 6, pp. 296-317, 1995.

[4] J. W. Sammon Jr., A nonlinear mapping for data structure analysis, *IEEE Trans. Comput*., vol. 18, pp. 401-409, 1969.

[5] Kaski, S., Data exploration using self-organizing maps. Acta Polytechnica Scandinavica, *Math., Computing and Management in Engineering* Series No. 82, Espoo 1997.

[6] Kohonen, T*., Self-Organizing Maps*, Springer: Berlin, pp. 77-130, 1995

[7] B. Fritzke, Some competitive learning methods: Inst. Neural Comput., Ruhr-Univ. Bochum, Germany, *Tech. Rep.*, 1997

[8] Tomenko, V., and Popov, V. Cooling Growing Grid: incremental self-organizing neural network for data exploration. *Data Mining VI*, 2005, 247-256.

[9] S. Haykin,. *Neural Networks: a Comprehensive Foundation*, 2nd Ed. Upper Saddle River, NJ: Prentice Hall, 1999.

[10] Widrow B., Hoff M. E. Adaptive switching circuits. *In 1960 IRE WESCON Convention Record*, 1960, pp. 96-104.

[11] KDD datasets: http://kdd.ics.uci.edu/

[12] Blackard, Jock A. Comparison of Neural Networks and Discriminant Analysis in Predicting Forest Cover Types. *Ph.D. dissertation.* Department of Forest Sciences. Colorado State University. Fort Collins, Colorado, 1998.

[13] UCI ML Repository: http://www.ics.uci.edu/~mlearn/MLRepository.html