Nonlinear Mapping Networks

Dimitris K. Agrafiotis* and Victor S. Lobanov

3-Dimensional Pharmaceuticals, Inc., 665 Stockton Drive, Exton, Pennsylvania 19341

Received April 11, 2000

Among the many dimensionality reduction techniques that have appeared in the statistical literature, multidimensional scaling and nonlinear mapping are unique for their conceptual simplicity and ability to reproduce the topology and structure of the data space in a faithful and unbiased manner. However, a major shortcoming of these methods is their quadratic dependence on the number of objects scaled, which imposes severe limitations on the size of data sets that can be effectively manipulated. Here we describe a novel approach that combines conventional nonlinear mapping techniques with feed-forward neural networks, and allows the processing of data sets orders of magnitude larger than those accessible with conventional methodologies. Rooted on the principle of probability sampling, the method employs a classical algorithm to project a small random sample, and then "learns" the underlying nonlinear transform using a multilayer neural network trained with the back-propagation algorithm. Once trained, the neural network can be used in a feed-forward manner to project the remaining members of the population as well as new, unseen samples with minimal distortion. Using examples from the fields of image processing and combinatorial chemistry, we demonstrate that this method can generate projections that are virtually indistinguishable from those derived by conventional approaches. The ability to encode the nonlinear transform in the form of a neural network makes nonlinear mapping applicable to a wide variety of data mining applications involving very large data sets that are otherwise computationally intractable.

I. INTRODUCTION

A. Dimensionality Reduction. Finding good data representations is a common and important objective in many data-mining applications. Of particular importance is the ability to understand the structure and topology of the data, and the interrelationships and associations between the objects of our study. Such relationships are frequently described by means of a similarity index derived either through direct observation or through the measurement of a set of characteristic features, which are subsequently combined in some form of dissimilarity or distance measure. Indeed, distance is an ubiquitous concept and represents one of the most reliable guiding principles for understanding our universe, one that we can comprehend, feel comfortable with, and navigate with ease and confidence.

This paper describes a novel approach for analyzing and visualizing the relationships between objects in high-dimensional data sets. High-dimensional spaces are sparse[1] and inherently difficult to understand, and they possess properties that challenge and often contradict the intuition that we have developed from our experience with two- or three-dimensional geometry.[2,3] This complexity has often been referred to as the "curse of dimensionality", a term that was originally introduced by Bellman to describe the complexity of combinatorial optimization in higher dimensions. Yet high-dimensional spaces are encountered in all disciplines of science, from chemistry and physics to social sciences and psychology. In modern science, phenomena are understood by measuring and analyzing a set of characteristic features, and by constructing models to explain their structure and causality. If the number of features is small, patterns in the data can be extracted using conventional graphical methods such as one- and two-dimenional histograms, scatter-plot diagrams, and/or kinematic techniques. In a higher-dimensional setting, however, such techniques are of limited value since they lead to a combinatorial explosion of possibilities and loss of potential higher-order relationships. High-dimensional representations pose a number of additional problems. The first, and perhaps most important, is the presence of substantial correlation between the variables. The importance of correlation is domain-dependent, but in general redundant variables tend to exert undue influence in data analysis. Moreover, if the features are to be used for regression or classification, overfitting can be a serious threat. The existence of a large number of variables can cause most regression and classification techniques to focus on the idiosyncrasies of the individual samples and lose sight of the broad picture that is essential for generalization beyond the training set. Finally, as the dimensionality of the space increases, the size of the computational effort needed to perform the analysis can be daunting even for today's most powerful computers. Fortunately, most multivariate data in R^d are almost never d-dimensional. That is, the underlying structure of the data is almost always of dimensionality lower than d. In the interest of parsimony, and to simplify the analysis and representation of the data, it is often desirable to reduce the dimensionality of the space by eliminating noisy and redundant features. Most methods reported to date attempt to do so by projecting the original space into one or more two- or three-dimensional representations.

* Corresponding author. Telephone: (610) 458-6045. FAX: (610) 458-8249. E-mail: dimitris@3dp.com.

10.1021/ci000033y CCC: $19.00 © 2000 American Chemical Society
Published on Web 08/12/2000

Perhaps the most common dimensionality reduction technique is principal component analysis (PCA).[4] PCA reduces a set of partially cross-correlated data into a smaller set of orthogonal variables with minimal loss in the contribution to variation. In effect, the method detects and combines features which behave in a similar way into a new set of variables that are orthogonal, i.e., noncorrelated. The main advantage of PCA is that it makes no assumptions about the probability distributions of the original variables. However, PCA (and other related techniques such as factor analysis[4] is sensitive to outliers, missing data, and poor correlations due to poorly distributed variables. More importantly, these techniques assume a linear constraint of the input space and perform poorly in high-dimensional, nonlinear spaces.

B. Nonlinear Mapping. Multidimensional scaling (MDS),[5] nonlinear mapping (NLM),[6] and Kohonen networks[7] represent alternative dimensionality reduction techniques that deal specifically with nonlinear spaces. The first two were designed to reproduce coordinates from a distance matrix, while the latter features data abstraction by means of prototyping, achieved though a self-organizing principle. In the first two techniques, the reduction is effected by reconstructing a low-dimensional coordinate set from a distance matrix computed from a higher-dimensional representation, while in the latter the original property vectors are mapped onto a two-dimensional cell array arranged in a way that preserves the topology and density of the original data set. These reduced representations can subsequently be used for a variety of pattern recognition and classification tasks.

Multidimensional scaling emerged from the need to visualize a set of objects described by means of a similarity or dissimilarity matrix. The technique originated in the field of psychology and can be traced back to the work of Torgerson[8] and Kruskal.[9] The problem is to construct a configuration of points in a low-dimensional space from information about the distances between these points. In particular, given a set of k data points in the input space $\{x_i, i = 1, 2, ... k\}$, a symmetric matrix d_{ij} of the observed dissimilarities between these points, and a set of images of x_i on a d-dimensional display plane $\{\xi_i, i = 1, 2, ..., k; \xi_i \in R^d\}$, the objective is to place ξ_i onto the plane in such a way that their Euclidean distances $\delta_{ij} = ||\xi_i - \xi_j||$ approximate as closely as possible the corresponding values d_{ij}. A sum-of-squares error function can be used to decide the quality of the embedding. The most commonly used criterion is *Kruskal's stress*:

$$S = \sqrt{\frac{\sum_{i<j}(\delta_{ij} - d_{ij})^2}{\sum_{i<j}\delta_{ij}^{\ 2}}} \tag{1}$$

The actual embedding is carried out in an iterative fashion. The process starts by (1) generating an initial set of coordinates ξ_i, (2) computing the distances δ_{ij}, (3) finding a new set of coordinates ξ_i using a steepest descent algorithm such as Kruskal's linear regression or Guttman's rank-image permutation, and (4) repeating steps 2 and 3 until the change in the stress function falls below some predefined threshold.

Nonlinear mapping is a closely related technique proposed by Sammon in 1969.[6] Just like MDS, NLM attempts to approximate local geometric relationships on a two- or three-dimensional plot. Although an exact projection is only possible when the distance matrix is positive definite, meaningful projections can be obtained even when this criterion is not satisfied. The embedding is carried out in an iterative fashion by minimizing an error function, E, which measures the difference between the distance matrixes of the original and projected vector sets:

$$E = \frac{\sum_{i<j}^{k}\frac{[d_{ij} - \delta_{ij}]^2}{d_{ij}}}{\sum_{i<j}^{k} d_{ij}} \tag{2}$$

E is minimized using a steepest-descent algorithm. The initial coordinates, ξ_i, are determined at random or by some other projection technique such as PCA, and are updated using:

$$\xi_{pq}(m+1) = \xi_{pq}(m) - \lambda\Delta_{pq}(m) \tag{3}$$

where m is the iteration number and λ is the learning rate parameter, and

$$\Delta_{pq}(m) = \frac{\partial E(m)}{\partial \xi_{pq}(m)} \bigg/ \left|\frac{\partial^2 E(m)}{\partial \xi_{pq}(m)^2}\right| \tag{4}$$

The advantage of nonlinear maps compared to Kohonen networks is that they provide much greater individual detail and are very well suited for interactive analysis and visual inspection. By preserving the distances of the original samples on the projected map, MDS and NLM are able to represent the topology and structural relationships in the data set in a unique and faithful manner. Although in most cases projection does lead to some loss of information, the amount of distortion induced by NLM and MDS is minimal compared to other dimensionality reduction techniques.

Unfortunately, Sammon's nonlinear mapping algorithm exhibits quadratic time complexity and scales adversely with the size of the data set. Several attempts have been made to alleviate this problem. Chang and Lee[10] proposed a heuristic relaxation approach in which a subject of the original objects (the frame) are scaled using a Sammon-like methodology, and the remaining objects are then added into the frame by adjusting their distance to the objects already in the frame. An alternative approach proposed by Pykett[11] is to use a clustering methodology and map only the prototypes which represent the centroids of the pattern vectors in each pattern class. In the resulting two-dimensional plots, the cluster prototypes are represented as circles whose radii are proportional to the spread in their respective classes. A very different technique is Lee's[12] triangulation method. Here, the points are processed in a sequential manner: each point is positioned on the plane so that its distances from the two nearest neighbors already mapped are preserved. An arbitrarily selected reference point may also be used to ensure that the resulting map is globally ordered. Biswas, Jain, and Dubes[13] later proposed a hybrid approach which combined the ability of Sammon's algorithm to preserve global

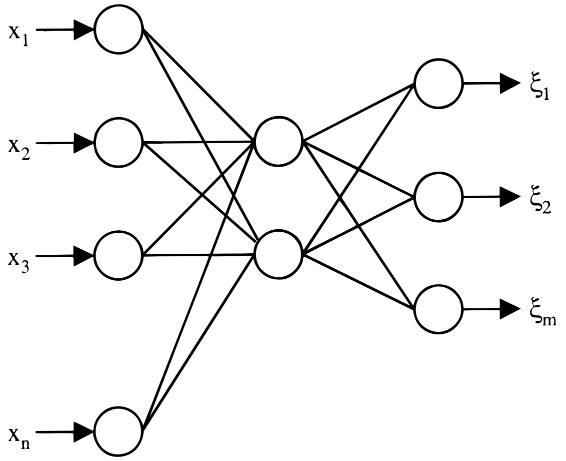

Figure 1. Three-layer feed-forward neural network for nonlinear mapping from R^n to R^m.

information with the efficiency of Lee's triangulation method. None of these methods, however, provides an explicit function that describes the mapping and can be used to project a large number of new patterns in an efficient way.

This paper describes a novel approach that combines conventional nonlinear mapping techniques with feed-forward neural networks and allows the processing of data sets orders of magnitude larger than those accessible with conventional methodologies. The key algorithmic details are described, and the advantages of this approach are demonstrated using examples from the fields of computer graphics, character recognition, and combinatorial chemistry. This approach differs significantly from that of Mao and Jain,[14] who employed a similar neural network architecture trained with a special back-propagation rule that relied on errors that were functions of the interpattern distances. However, because only a single distance is examined during each iteration, these networks require a very large number of iterations and converge extremely slowly.

II. PROPOSED ALGORITHM

The method described herein is rooted in the principle of probability sampling, i.e., the notion that a small number of randomly chosen members of a given population will tend to have the same characteristics, and in the same proportion, as the population as a whole. Our approach is to employ a classical algorithm to multidimensionally scale a small random sample which reflects the overall structure of the data, and then "learn" the underlying nonlinear transform using a multilayer neural network trained with the back-propagation algorithm.[15] Once trained, the neural network can be used in a feed-forward manner to project the remaining members of the population as well as new, unseen samples with minimal distortion. For a nonlinear projection from n to m dimensions, a standard three-layer neural network with n input and m output units with logistic transfer functions is employed (Figure 1). Each n-dimensional pattern is presented to the input layer, and its coordinates on the nonlinear map are obtained by the respective units in the output layer. The number of hidden neurons is determined empirically based on the dimensionality and structure of the input space and the size of the training set.

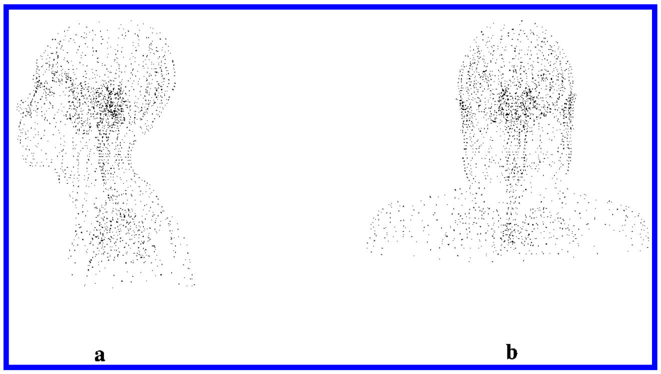

Figure 2. Two orthogonal views of the face data set. (a) Lateral view; (b) frontal view.

III. RESULTS AND DISCUSSION

Our analysis was based on three data sets extracted from the computer graphics, character recognition, and combinatorial chemistry fields. They differ greatly in size, origin, and dimensionality and are used to illustrate various aspects of our nonlinear mapping algorithm, as well as its broad applicability across many different application domains.

A. Face Data Set. The first data set was taken from the computer graphics literature and represents a three-dimensional image of a man's face comprised of 2630 data points. The task was to project that image onto a plane and visually inspect the results. The object can be easily recognized in both its original and projected forms, and helps to illustrate the subtle differences between linear and nonlinear dimensionality reduction techniques. The original data are shown in two orthogonal views in Figure 2, and the two-dimensional PCA and NLM projections in parts a and b, respectively, of Figure 3. For both data sets used in this study, the PCA projection was derived from the first two principal components that accounted for most of the variance in the data, while the nonlinear map was obtained with a variant of Sammon's original algorithm developed by our group.[16] In general, the two projections are very similar, but they differ in one important aspect: in the principal component projection, one dimension is completely suppressed and all characteristics of the man's profile are virtually lost (see Figure 3a). In contrast, the nonlinear map represents a hybrid view that combines important, distinctive features of the entire object. While the general shape is still dominated by the head-on view, one can clearly recognize key elements of the facial profile such as the nose, the lips, and the chin, as well as a detectable protrusion in the occipital area of the skull (Figure 3b). In terms of distortion, NLM does a much better job in preserving the distance matrix than PCA, as manifested by a Kruskal stress of 0.152 and 0.218 for the NLM and PCA projections, respectively.

To determine how well the mapping produced by the steepest descent algorithm can be captured by a neural network, we carried out an extensive set of simulations using several sample sizes ranging from 100 to 1600 points. The experiment consisted of the following steps. For each sample size, n, 100 different random subsets of n points were extracted from the original three-dimensional object, and were independently mapped using our gradient-based nonlinear mapping algorithm. The three-dimensional input and two-dimensional output coordinates obtained from the NLM were then used to train 100 separate neural networks with 3

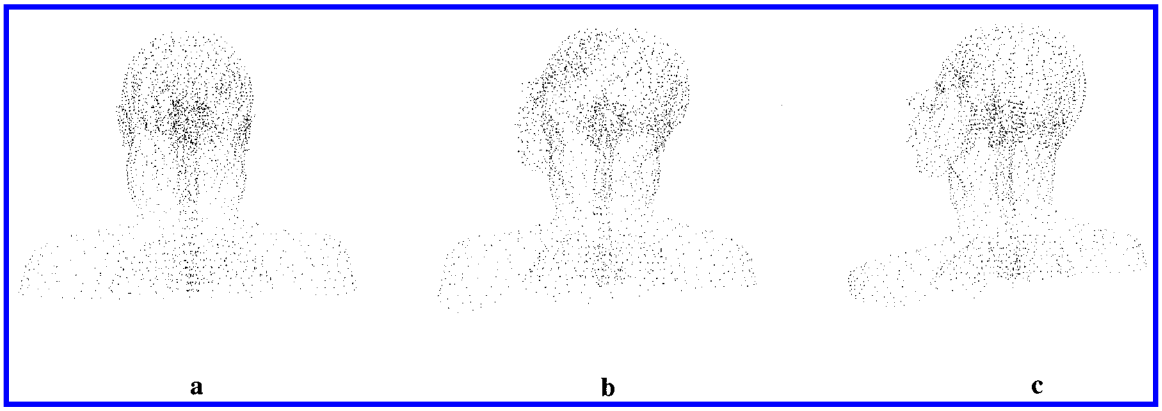

Figure 3. Two-dimensional projections of the face data set. (a) PC a projection; (b) NLM projection; (c) neural network projection (400-point training set).

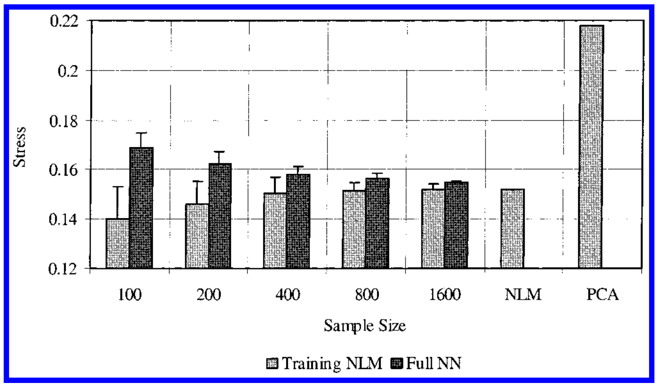

Figure 4. Stress as a function of sample size for the face data set. The two sets of columns and their respective error bars represent the mean and standard deviation of the stress of the NLM projection for the training set (light gray) and the neural network projection for the entire data set (dark gray) over 100 runs for five different sample sizes. Each run represents a different set of points comprising the training set. The last two columns represent the stress of the NLM and the PCA projections of the entire data set, respectively.

input, 10 hidden, and 2 output neurons having logistic activation functions. The networks were trained for 16 000, 8000, 4000, 2000, and 1000 epochs for samples containing 100, 200, 400, 800, and 1600 points, respectively, and with a linearly decreasing learning rate from 0.3 to 0.001 and a momentum of 0.8. To ensure that the results reflect true differences in the underlying distributions of the training sets, the number of epochs was selected so that the training time was constant and independent of the amount of data in each training set. Once the networks were trained, the entire data set of 2630 points was presented to each one, and 100 new sets of two-dimensional coordinates were obtained. As stated above, this procedure was repeated for five different sample sizes containing 100, 200, 400, 800, and 1600 points, respectively. To simplify the notation, we will refer to each of these 100 subsets and all of its associated data as a separate "run".

The results of this experiment are summarized in Figure 4. The two columns on the right represent the stress of the full NLM and PCA projections, respectively, and serve as reference points to assess the quality of the neural approximation. The five pairs of columns on the left and their respective error bars represent the mean and standard deviation of the stress of the NLM projection for the training set (in light gray) and the neural projection for the entire data set (in dark gray) for each particular sample size. These results are fully consistent with our intuition and expectations. First, the average stress of the nonlinear map increases with the number of points in an asymptotic manner. The average stress of 400 points for example (15% of the entire data set) is within 0.001 unit of the stress of the full NLM, with a standard deviation of only 0.006. More importantly, the stress of the corresponding neural projection exhibits a trend similar to but opposite of the same asymptotic character. As one would expect, the more information is used to train the neural network, the more predictive it becomes and the better it approximates the true underlying function governing the relationship of the objects on the nonlinear map. Interestingly enough, even with a mere 100 points, every single set that we tried led to a projection that was far better than that obtained with PCA. The most significant observation, however, is that the standard deviation of the neural stress is very small and lies well within the limits of acceptable error.

To get a better appreciation of what the stress values really mean in the context of structure, Figure 3c shows the nonlinear map obtained from a neural network trained with an average 400-point sample and having a stress of 0.158. The map is virtually identical to that obtained by NLM, revealing the same characteristic mix of features from the frontal and lateral views, albeit in a more regular form. Indeed, for the purposes of exploratory data analysis, the two images are virtually indistinguishable.

B. Letter Data Set. Although the face data set provides a useful test case, it is miniscule in comparison to the data sets for which this method was intended, in terms of both size and dimensionality. The second data set that we studied was taken from the UCI Machine Learning Repository[17] and represents raster scan images of 20 000 letters of the English alphabet. It was originally developed by Frey and Slate[18] and was used to compare the classification accuracy of several genetic rule-based classifiers. The data set is comprised of 20 000 unique black-and-white rectangular letter images, generated by randomly distorting pixel images of the 26 uppercase letters from 20 different commercial fonts. The parent fonts represented a full range of character types including script, italic, serif, and Gothic. Each letter within these 20 fonts was randomly distorted to produce a file of 20 000 unique stimuli which were subsequently compressed into 16 primitive numerical attributes (statistical moments and edge counts) and scaled to fit into a range of integer values from 0 to 15.

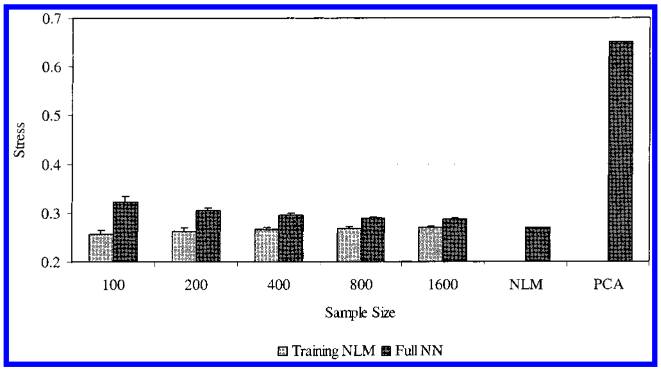

Figure 5. Stress as a function of sample size for the letter data set. The two sets of columns and their respective error bars represent the mean and standard deviation of the stress of the NLM projection for the training set (light gray) and the NN projection for the entire data set (dark gray) over 100 runs for five different sample sizes. Each run represents a different set of points comprising the training set. The last two columns represent the stress of the NLM and the PCA projections of the entire data set, respectively.

This data set was subjected to the same type of analysis that was used in the previous example. In particular, 100 independent runs were carried out at five different sample sizes (100, 200, 400, 800, and 1600 points), each involving a different set of random points comprising the training set. The networks consisted of 16 input, 10 hidden, and 2 output units, having a total of 180 freely adjustable synaptic weights. Similar training parameters were used, i.e., 16 000, 8000, 4000, 2000, and 1000 epochs for samples containing 100, 200, 400, 800, and 1600 points, respectively. The mean and standard deviation of the stress distributions for each sample size are shown in Figure 5, along with the stress of the full NLM and PCA projections. The plot reveals trends similar to those established in the analysis of the face data. The average stress of the random sample increases with sample size, and the composition of the sample itself becomes less significant as it becomes increasingly more representative of the entire population. Again, in all cases, a small fraction of the entire data set (100 points or 0.5%) led to network projections that were far superior to that derived from PCA, as manifested by a mean stress of 0.324 ± 0.011 compared to 0.651 for PCA and 0.271 for full NLM, respectively. As the number of training patterns increases, the mean stress drops to 0.305 ± 0.006 for 200 points, 0.296 ± 0.004 for 400 points, and 0.289 ± 0.003 for 800 points, and finally levels off at 0.289 ± 0.002 for 1600 points. These samples represent 1%, 2%, 4%, and 8% of the entire collection, respectively. It appears that this value (~0.29) represents the learning capacity of a network with 10 hidden units for this particular data set.

The PCA, NLM, and NN maps of the letter data are shown in Figure 6. To assist in the interpretation, we use color coding to highlight two distinct clusters within that set, representing the letters A (in dark gray) and W (in light gray), respectively. The neural map in Figure 6c was derived from a network trained with 400 patterns and had a stress of 0.292. Even though this training set is relatively small and better approximations can be obtained from larger samples, it is clear that the network is able to reproduce the structure of the map produced by the iterative algorithm quite well. As with the face data, there are notable differences between the linear (Figure 6a) and nonlinear (Figure 6b,c) projections which are not limited to scale (the PCA projection represents a truncation of dimensionality and, therefore, distances are underestimated). Although these differences are not qualitative in nature, the three elongated clusters which are evident in all three of the maps appear to be more compact in the PCA projection. This "squashing" effect was to some extent also observed in the face data set, and is even more pronounced in the diamine data set described below.

C. Diamine Data Set. The last data set is taken from the field of combinatorial chemistry. This is an area of particular interest to us, and one in which large data sets are commonplace. In recent years, the pharmaceutical and chemical industries have embraced a new set of technologies that allow the simultaneous synthesis and biological evaluation of large chemical libraries containing hundreds to hundreds of thousands or even millions of molecules.[19] A combinatorial library is a collection of chemical compounds derived from the systematic application of a synthetic principle to a prescribed set of building blocks. The design and analysis of combinatorial libraries is becoming an integral part of modern drug design and involves data sets of truly staggering size.[20,21] Nonlinear mapping was shown to be particularly useful in this respect because of its ability

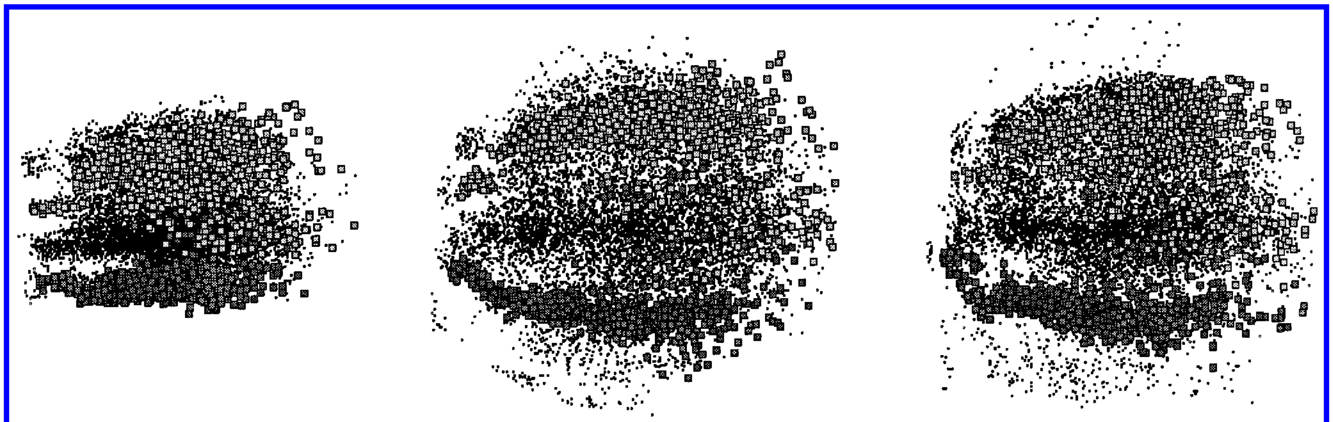

Figure 6. Letter data set. (a, left) PCA projection; (b, middle) NLM projection; (c, right) neural network projection (400-point training set).

$$\text{HN-X-NH} \xrightarrow{\text{Protection}} \text{HN-X-N[P]} \xrightarrow{R_1X} R_1\text{N-X-N[P]} \xrightarrow{\text{Deprotection}} R_1\text{N-X-NH} \xrightarrow{R_2X} R_1\text{N-X-N}R_2$$

Figure 7. Synthetic sequence for the generation of the diamine library.

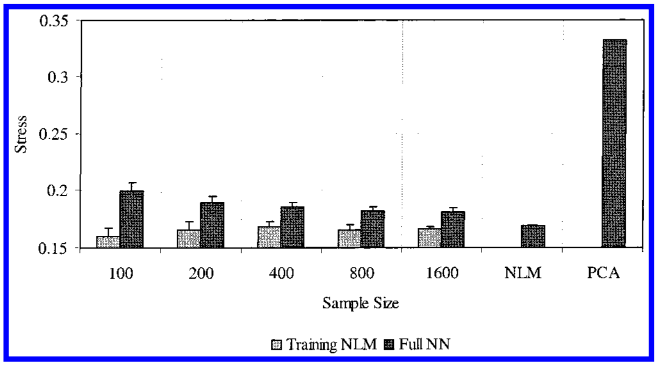

Figure 8. Stress as a function of sample size for the diamine data set. The two sets of columns and their respective error bars represent the mean and standard deviation of the stress of the NLM projection for the training set (light gray) and the NN projection for the entire data set (dark gray) over 100 runs for five different sample sizes. Each run represents a different set of points comprising the training set. The last two columns represent the stress of the NLM and the PCA projections of the entire data set, respectively.

to convey complex relationships in an intuitive manner without loss of individual detail.[22,23]

The example that we chose is a three-component combinatorial library taken from the work of Cramer et al.[24] A diamine molecule containing two primary or secondary amines served as the central scaffold, and was derivatized on both sides using an acylating agent, reactive halide, or carbonyl group susceptible to reductive amination. The synthetic sequence required to generate this library involves selective protection of one of the amines and introduction of the first side chain, followed by deprotection and introduction of the second side chain (Figure 7). As the original authors noted, use of commercially available reagents alone (the 1996 Available Chemical Directory[25] contained 1750 reagents of type HNXNH and 26 700 reagents of type RX) would yield over 10^{12} potential products, 50 000 times more than the world's cumulative chemical literature. Since the objective of this work was to validate the nonlinear mapping algorithm, we generated a small library comprised of 57 498 compounds using 42 commercially available diamines and 37 acid chlorides and alkylating agents. Each compound was described by 117 topological indices designed to capture the essential features of the molecular graph, which were subsequently decorrelated to 16 principal components, which accounted for 99% of the total variance in the data. These principal components were used as input to the nonlinear dimensionality reduction techniques described above. The PCA preprocessing step was necessary in order to eliminate duplication and redundancy in the data, which is typical of graph-theoretic descriptors.

The results are summarized in Figure 8. The network had 16 input, 10 hidden, and 2 output neurons (i.e., a total of 180 synaptic weights) and was trained for 16 000, 8000, 4000, 2000, and 1000 epochs for samples containing 100, 200, 400, 800, and 1600 points, respectively. As in the two previous cases, the network does not overfit even with training sets of 100 points (less than 0.2% of the entire library) where there are nearly two synapses per training case, and every single network that we trained outperformed PCA by a wide margin. As for the overall trends, they are no different than those observed in the two previous examples: increase in sample size leads to better approximations and less variability across different samples. A random sample of 400 points (0.7% of the entire library) leads to a nonlinear map with an average stress of 0.185 ± 0.004, while a sample of 800 points brings the stress down to 0.182 ± 0.004. With 1600 points (2.8% of the entire library) the mean stress improves slightly to 0.181 ± 0.003, close to the actual NLM stress of 0.169. The resulting maps (Figure 9) confirm the close agreement between the conventional and neural nonlinear mapping algorithms, and the substantial differences between them and PCA, which had a stress of 0.332. A look at the variances of the principal components reveals why PCA is such a poor method in this case. The first two PCAs account for only 69% of the total variance in the data, 14% less than that of the respective components in the face data set. This unaccounted "residual" variance leads to significant distortion in the principal component map, which in the case of the diamine library provides only a hint of the true structure of the data, evidenced only by the presence of the two disproportionately populated clusters.

IV. CONCLUDING REMARKS

The aim of this paper was to present a promising new nonlinear mapping algorithm for reducing the dimensionality of very large data sets. The main focus of the paper is sampling. To enable meaningful comparisons, the complexity of the network did not change as a function of sample size for the three data sets used in our study. The training sets differed greatly in size, and poor convergence may have been a contributing factor for some of the results, particularly those obtained from smaller samples. Although 100 runs were carried out for each particular sample size, the network was

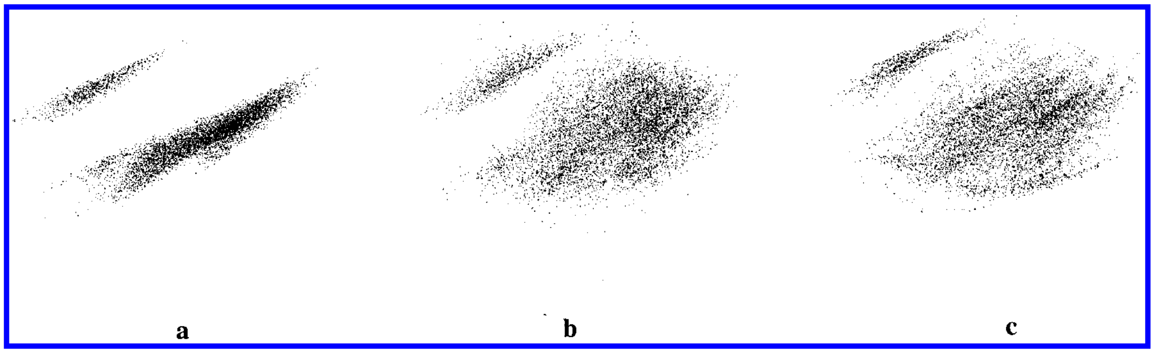

Figure 9. Diamine data set. (a) PCA projection; (b) NLM projection; (c) neural network projection (400-point training set).

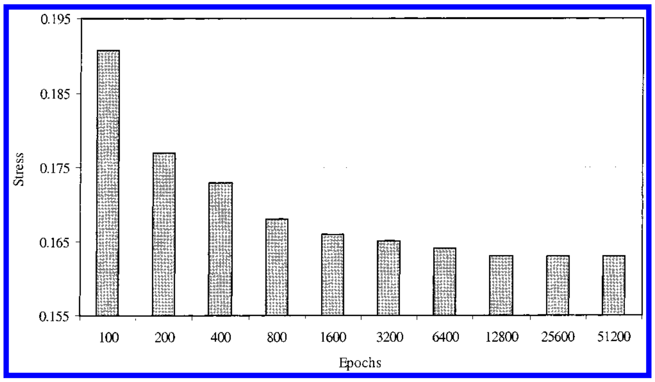

Figure 10. Stress as a function of training epochs for the face data set using a 100-point training sample and a three-layer perceptron with 10 hidden nodes.

trained only once for each particular training set. As a consequence, part of the observed variance in stress may be due to poorly optimized networks stuck in local minima in synaptic weight space. Preliminary calculations have confirmed that there is indeed some variability from one optimization run to another, and the results become better if we perform several optimizations and select the best network for each particular training set. Further improvements can also be achieved by increasing the number of training epochs or by fine-tuning other training parameters such as the learning schedule and momentum, as well as by increasing the number of hidden neurons, particularly for larger samples. A systematic analysis of the effect of these parameters on the quality of the nonlinear map is beyond the scope of this paper, and will be discussed elsewhere.[26,27] We should point out, however, that nonlinear mapping networks are remarkably resistant to overfitting. This is illustrated in Figure 10, which shows the stress of the neural map for the face data set as a function of training time. The maps were based on a training sample of 100 randomly chosen points, and were derived with a standard three-layer perceptron containing 10 hidden nodes. It is clear that increasing the number of training epochs improves the stress in an asymptotic manner, and there is no evidence of overtraining whatsoever. Our experience has shown that this is always the case, regardless of the nature, size, and dimensionality of the data set.

A final word on timing. Even though our direct NLM[16] is dramatically faster than Sammon's original algorithm[6] (for example, the projection of all 2360 points comprising the face data set required only 2 CPU s as compared to 322 CPU s for Sammon's algorithm on an 800 MHz Pentium III processor), it is still iterative in nature, and thus unsuitable for massive data sets ($>10^7$ items). Conversely, the training of the neural network required 28 CPU s on the same machine, whereas the feed-forward projection was virtually instantaneous. These results and our extensive experience with the use of this algorithm suggest that the method is able to process data sets of any size, origin, and dimensionality, and can do so in nearly interactive time scales. In addition, although all our examples were limited to two-dimensional projections for presentation purposes, the proposed architecture is general and can be used to extract constraint surfaces of any desired dimensionality.

ACKNOWLEDGMENT

The authors are grateful to Dr. Raymond F. Salemme of 3-Dimensional Pharmaceuticals, Inc., for his insightful comments and support of this work.

REFERENCES AND NOTES

(1) Bellman, R. E. *Adaptive Control Processes*; Princeton University Press: Princeton, 1961.
(2) Wegman, E. *J. Ann. Stat.* **1970**, *41*, 457−471.
(3) Scott, D. W. *Multivariate Density Estimation: Theory, Practice and Visualization*; Wiley: New York, 1992.
(4) Cooley, W.; Lohnes, P. *Multivariate Data Analysis*; Wiley: New York, 1971.
(5) Borg, I.; Groenen, P. *Modern Multidimensional Scaling*; Springer-Verlag: New York, 1997.
(6) Sammon, J. W. *IEEE Trans. Comput.* **1969**, *C-18*, 401−409.
(7) Kohonen, T. *Self-Organizing Maps*; Springer-Verlag: Heidelberg, 1996.
(8) Torgeson, W. S. *Psychometrika* **1952**, *17*, 401−419.
(9) Kruskal, J. B. *Phychometrika* **1964**, *29*, 115−129.
(10) Chang, C. L.; Lee, R. C. T. *IEEE Trans. Syst., Man, Cybern.* **1973**, *SMC-3*, 197−200.
(11) Pykett, C. E. *Electron. Lett.* **1978**, *14*, 799−800.
(12) Lee, R. C. Y.; Slagle, J. R.; Blum, H. *IEEE Trans. Comput.* **1977**, *C-27*, 288−292.
(13) Biswas, G.; Jain, A. K.; Dubes, R. C. *IEEE Trans. Pattern Anal. Machine Intell.* **1981**, *PAMI-3* (6), 701−708.
(14) Mao, J.; Jain, A. K. *IEEE Trans. Neural Networks* **1995**, *6* (2), 296−317.
(15) Haykin, S. *Neural Networks: A Comprehensive Foundation*; Prentice Hall, 1998.
(16) Agrafiotis, D. K.; Lobanov, V. S.; Salemme, F. R. Patents pending.
(17) Blake, C. L.; Merz, C. J. UCI Repository of machine learning databases, Irvine, CA: University of California, Department of Information and Computer Science (http://www.ics.uci.edu/∼mlearn/MLRepository.html); 1998.
(18) Frey, P. W.; Slate, D. J. *Machine Learning* **1991**, *6* (2), 491.
(19) Thompson, L. A.; Ellman, J. A. *Chem. Rev.* **1996**, *96*, 555−600.
(20) Agrafiotis, D. K. In *The Encyclopedia of Computational Chemistry*; Schleyer, P. v. R., Allinger, N. L., Clark, T., Gasteiger, J., Kollman, P. A., Schaefer III, H. F., Schreiner, P. R., Eds.; John Wiley & Sons: Chichester, 1998; pp 742−761.
(21) Agrafiotis, D. K.; Myslik, J. C.; Salemme, F. R. *Mol. Diversity* **1999**, *4* (1), 1−22. Agrafiotis, D. K.; Myslik, J. C.; Salemme, F. R. In *Annual Reports in Combinatorial Chemistry and Molecular Diversity*; Pavia, M., Moos, W., Eds.; Kluwer: Norwell, MA, 1999; Vol 2, pp 71−92.
(22) Agrafiotis, D. K. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 841−851.
(23) Agrafiotis, D. K. *Protein Sci.* **1997**, *6*, 287−293.
(24) Cramer, R. D.; Patterson, D. E.; Clark, R. D.; Soltanshahi, F.; Lawless, M. S. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 1010−1023.
(25) Available Chemicals Directory is marketed by MDL Information Systems, Inc., 140 Catalina Street, San Leandro, CA 94577.
(26) Rassokhin, D. N.; Lobanov, V. S.; Agrafiotis, D. K. *J. Comp Chem.*, in press.
(27) Agrafiotis, D. K.; Rassokhin, D. N.; Lobanov, V. S. *J. Comp. Chem.*, in press.

CI000033Y