

# On Pattern Classification with Sammon's Nonlinear Mapping- An Experimental Study

Boaz Lerner\*, Hugo Guterman, Mayer Aladjem, Its'hak Dinstein and Yitzhak Romem\*

Department of Electrical and Computer Engineering  
Ben-Gurion University of the Negev  
Beer-Sheva 84105, Israel

\*Genetics Institute, Soroka Medical Center  
Ben-Gurion University of the Negev  
Beer-Sheva 84101, Israel

## Abstract

Sammon's mapping is conventionally used for exploratory data projection, and as such is usually inapplicable for classification. In this paper we apply a neural network (NN) implementation of Sammon's mapping to classification by extracting an arbitrary number of projections. The projection map and classification accuracy of the mapping are compared with those of the auto-associative NN (AANN), multilayer perceptron (MLP) and principal component (PC) feature extractor for chromosome data. We demonstrate that chromosome classification based on Sammon's (unsupervised) mapping is superior to classification based on the AANN and PC feature extractor and highly comparable with that based on the (supervised) MLP.

Key Words: Chromosomes, Classification, Feature extraction, Multilayer perceptron, Neural networks, Sammon's mapping.

Published in *Pattern Recognition*, vol. 31, pp. 371-381, 1998.

\* Present address: University of Cambridge Computer Laboratory, New Museums Site, Cambridge CB2 3QG, UK (email: boaz.lerner@cl.cam.ac.uk).

This work was supported in part by the Paul Ivanier Center for Robotics and Production Management, Ben-Gurion University, Beer-Sheva, Israel.

## 1. Introduction

Feature extraction is the process of mapping original features (measurements) into fewer features, which preserve the main information of the data structure. A large variety of feature extraction paradigms appears in the literature,<sup>(1-4)</sup> some of them are based on NNs<sup>(5-10)</sup>. The NN based feature extraction paradigms provide adaptivity to a changing environment and the possibility of relatively easy hardware implementation. They can even overcome the drawbacks of classical algorithms<sup>(7, 10)</sup> or enhance the classification performance.<sup>(7, 11)</sup> In all the methods, a mapping  $f$  transforms a pattern  $y$  of a  $d$ -dimensional input space to a pattern  $x$  of an  $m$ -dimensional projected space,  $m < d$ , i.e.,

$$x = f(y), \quad (1)$$

such that a criterion  $J$  is optimized. The mapping  $f$  is determined from among all the transformations  $g$ , as one that satisfies<sup>(2)</sup>,

$$J\{f(y)\} = \max_g J\{g(y)\}. \quad (2)$$

The mappings differ by the functional forms of  $g$  and by the criteria they have to optimize.

Feature extraction methods can be grouped into four categories<sup>(7)</sup> based on *a priori* knowledge used for the computation of  $J$ : supervised *versus* unsupervised, and by the functional form of  $g$ : linear *versus* nonlinear. In cases where the target classes of the patterns are unknown, unsupervised methods are the only way to perform feature extraction, whereas in other cases, supervised paradigms are preferable. Linear methods are simpler and are often based on an analytical solution but they are inferior to nonlinear methods when the classification task requires complex separation hypersurfaces. Discriminant analysis is a well-known procedure for linearly projecting labeled data<sup>(3)</sup> in which the ratio of the determinants of the *between*-class scatter matrix ( $B$ ) and the *within*-class scatter matrix ( $W$ ) is maximized. Data is projected onto the space spanned by the eigenvectors corresponding to the largest (non-zero) eigenvalues of the matrix  $(W^{-1} \cdot B)$ . In a supervised nonlinear projection method, which has been suggested by Fukunaga<sup>(3)</sup> the projected data coordinates are a function of the distance to the  $k$ th nearest neighbor of each pattern. Popular unsupervised methods are principal component analysis (PCA)<sup>(2,3)</sup> (a linear mapping) and Sammon's (nonlinear) mapping.<sup>(4)</sup> The PCA attempts to preserve the

variance of the data, whereas Sammon's mapping tries to preserve the interpattern distances. Kohonen's self-organizing map (SOM)<sup>(6)</sup> is another example of an unsupervised nonlinear projection method based on an NN. A high-dimensional input space is projected by Kohonen's SOM onto a low-dimensional space such that the topology of the data is preserved. The MLP when acting as a feature extractor and the AANN provide, respectively, supervised and unsupervised nonlinear mappings of the input space into their hidden layers.

Feature extraction can be also grouped into exploratory data projection paradigms, which enable high-dimensional data visualization for better data structure understanding and cluster analysis, and paradigms for classification, in which it is desirable to extract reduced-dimensionality features to decrease the computational complexity and even improve the classification performance. There are strong connections between the two; For example, the selection of an appropriate classification paradigm or clustering algorithm is related to the information contained in an appropriate projection map. However, feature extraction criteria for exploratory data projection usually aim to minimize an error function, such as the mean square error or the interpattern distance difference whereas feature extraction criteria for classification aim to increase class separability as much as possible. Hence the optimum features (regarding a specific criterion) extracted for data projection are not necessarily the optimum features that enhance class separability and *vice versa*. Moreover, in exploratory data projection only one to three projections are extracted, whereas in classification more features are usually needed. Consequently, feature extraction paradigms for exploratory data projection are not generally used for classification, and *vice versa*.

In this study, the exploratory data projection method of Sammon<sup>(4)</sup> is used to extract an arbitrary number of projections and thereby to apply the mapping to classification. Sammon's algorithm is implemented here by an NN which provides a generalization capability to the original algorithm. The projection map and classification accuracy based on Sammon's algorithm are compared with those of feature extraction paradigms *per se*, such as the PC, AANN and MLP feature extractors for human chromosome data. In addition, we examine the benefits of using different initializations of Sammon's

mapping. Section 2 of the paper introduces Sammon's algorithm and an NN implementation of the mapping whereas the PC, AANN and MLP feature extractors are described in Section 3. Sections 4 and 5 present the experiments and their results, respectively, while Section 6 concludes the paper with a discussion.

## 2. Sammon's Nonlinear Mapping

Nonlinear mapping (NLM) algorithms employ nonlinear transformations, which attempt to preserve the inherent structure of the data when the patterns are projected from a higher-dimensional space onto a lower-dimensional space. The preservation of this inherent structure is achieved by preserving the distances between patterns under projection. Denote the interpattern distances between pattern  $\mu$  and pattern  $\nu$  in the input space and in the projected space as  $d^*(\mu, \nu)$  and  $d(\mu, \nu)$ , respectively. If  $d^*(\mu, \nu) = d(\mu, \nu)$  for all  $\mu$  and  $\nu$ , the structure of the data is *strictly preserved* by the NLM.<sup>(1)</sup> When only *approximate preservation* is reached an error term:

$$e(\mu, \nu) = d^*(\mu, \nu) - d(\mu, \nu) \quad (3)$$

is introduced for some or all values of  $\mu$  and  $\nu$ . To achieve *approximate preservation* with the lowest error, various NLM algorithms choose various distance measures and error functions.

### 2.1. Sammon's algorithm

In Sammon's nonlinear mapping,<sup>(4)</sup> the distance measure between two patterns is commonly the Euclidean metric and the error function (of  $n$  patterns) to minimize is Sammon's stress, defined as:

$$E = \frac{1}{\sum_{\mu=1}^{n-1} \sum_{\nu=\mu+1}^n d^*(\mu, \nu)} \sum_{\mu=1}^{n-1} \sum_{\nu=\mu+1}^n \frac{[d^*(\mu, \nu) - d(\mu, \nu)]^2}{d^*(\mu, \nu)}. \quad (4)$$

When employing the gradient-descent procedure to search for the minimum of Sammon's stress, a local minimum in the error surface could be reached. Therefore, a significant number of experiments with different random initializations may be necessary and the implementation becomes inappropriate.

This disadvantage is hard to cope with because there are no common rules to apply when determining the best initialization. Nevertheless, the initialization could be based on information which is obtained from the data, such as the first and second norms of the feature vectors<sup>(1)</sup> or the principal axes of the covariance matrix of the data.<sup>(1, 7)</sup>

The second disadvantage of Sammon's mapping is its computational load, which is  $O(n^2)$ . In each iteration  $n(n-1)/2$  distances, as well as the error derivatives, must be calculated. Therefore, as the number of patterns,  $n$ , increases, the computational requirements grow quadratically. To reduce these computational requirements, White<sup>(12)</sup> used Sammon's mapping with the Hamming metric as a distance measure between patterns, rather than the Euclidean metric. Although it has fewer computational requirements, White's approach has its drawbacks. For example, if the input space is the Euclidean space some distortion of the projected features will naturally occur resulting in increased Sammon's error.<sup>(1)</sup> Moreover, the interpretation of the projected data structure becomes more complex. Another drawback of Sammon's mapping is that it is data dependent; adding a new pattern requires a remapping of the "new" data set.

Finally, an important question relevant to the implementation of Sammon's mapping is how to select the error function which measures the projection distortion. Different error functions emphasize different characteristics of the data structure. The error function in Equation 4 is one of several functions emphasizing the local structure of the data. Other error functions emphasizing the global structure of the data, the global and the local structure of the data or the continuity of the data, also exist.<sup>(1)</sup>

## 2.2. An NN implementation of Sammon's mapping

Figure 1 shows a two-layer perceptron network that has been suggested by Mao and Jain<sup>(7)</sup> to implement Sammon's mapping. The number of network input units is set to be the input space dimension,  $d$ , the number of output units is specified as the extracted feature space dimension,  $m$ , but no rule for determining the number of hidden layers and the number of hidden units in each hidden layer

is suggested. They derived a weight updating rule for a multilayer network that minimizes Sammon's stress based on the gradient descent method. The general updating rule for all the hidden layers,  $l=1, \dots, L-1$  and for the output layer ( $l=L$ ) is:

$$\Delta \boldsymbol{\omega}_{jk}^{(l)} = -\eta \frac{\partial E}{\partial \boldsymbol{\omega}_{jk}^{(l)}} = -\eta (\Delta_{jk}^{(l)}(\boldsymbol{\mu}) y_j^{(l-1)}(\boldsymbol{\mu}) - \Delta_{jk}^{(l)}(\boldsymbol{\nu}) y_j^{(l-1)}(\boldsymbol{\nu})) \quad (5)$$

where  $\boldsymbol{\omega}_{jk}^{(l)}$  is the weight between unit  $j$  in layer  $l-1$  and unit  $k$  in layer  $l$ ,  $\eta$  is the learning rate,  $y_j^{(l-1)}$  is the output of the  $j$ th unit in layer  $(l-1)$  and  $\boldsymbol{\mu}$  and  $\boldsymbol{\nu}$  are two patterns. The  $\Delta_{jk}^{(l)}$  are the errors accumulated in each layer and backpropagated to a preceding layer, similar to the standard backpropagation (BP), but unlike the BP algorithm these errors are functions of the interpattern distances. A momentum constant is frequently added, as in the BP algorithm.

In Mao and Jain's implementation the network is able to project new patterns after training, a property Sammon's algorithm does not have. Similar to Chien,<sup>(1)</sup> Mao and Jain have suggested the use of data projections along the PCs as an initialization of Sammon's mapping. They employed a two-stage training phase using the standard BP algorithm for the first stage and their modified unsupervised BP algorithm for a refinement in the second stage. We use a similar but simpler implementation, in which only one training stage using Mao and Jain's unsupervised BP algorithm (their second stage) is employed. Moreover, we employ and compare random and PC-based initializations. When the PC-based initialization of Sammon's mapping is tested, the eigenvectors of the sample covariance matrix estimated from the training data set are exploited to establish the columns of the initial input-hidden weight matrix,  $\boldsymbol{\omega}$ , i.e.,

$$\boldsymbol{\omega} = [\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \dots, \boldsymbol{\varphi}_m] \quad (6)$$

where  $\boldsymbol{\varphi}_i$ ,  $i=1, \dots, m$  are the eigenvectors corresponding to the  $m$  largest eigenvalues. Finally, we have extracted an arbitrary number of projections and thereby applied Sammon's mapping to classification.

### 3. The Alternative Feature Extraction Paradigms Considered

The feature extraction paradigms which are alternatives to Sammon's mapping are described in this section whereas the methodology of the experiments to compare the paradigms is given in Section 4.

#### 3.1. The PC feature extractor

Among the unsupervised linear projection methods the PCA is probably the most widely used.<sup>(2, 3)</sup> The PCA, also known as the Karhunen-Loe`ve expansion, attempts to reduce the dimensionality of the feature space by creating new features that are linear combinations of the original features. The procedure begins with a rotation of the original data space followed by ranking the transformed features and picking out a few features. This procedure finds the subspace in which the original patterns may be approximated with the least mean-square error for a given dimensionality.

Recently several implementations of NN based PCA have been suggested.<sup>(5, 8, 9)</sup> Usually, the connections between input and output units and between output units themselves are updated using the Hebbian and anti-Hebbian rule, respectively.<sup>(8, 9)</sup> Alternatively, an MLP NN when working in auto-associative mode with a linear activation function for all the units can implement a PCA.<sup>(5)</sup> Extending a linear PCA network to a nonlinear is achieved by using a sigmoidal activation function for the output units. Higher-order statistics of the data can be exploited by the nonlinear network but at the cost of losing the eigenvector orthogonality. Implementation of an NN based PCA has a number of advantages over standard eigen-decomposition techniques. The network can be allowed to adapt to slowly varying changes in the input and can be more computationally efficient when  $m \ll d$ .<sup>(13)</sup> However, the network converges very slowly when  $d$  is high and all the  $d$  eigenvectors need to be computed, especially when some eigenvalues are very small.<sup>(7)</sup> Moreover, MLP based PCA is obtained iteratively and may well miss the optimum since it relies on a gradient technique and, thus, can become trapped in local minima. Conversely, the standard eigen-decomposition technique is obtained explicitly in terms of the training data. For these reasons, the standard eigen-decomposition technique is preferred in this study.

Let  $\mathbf{X}=f(\mathbf{Y})$  be a linear mapping of a random vector  $\mathbf{Y}$ ,  $\mathbf{Y} \in R^d$ ,  $\mathbf{X} \in R^m$  and  $m < d$ . The approximation

$$\hat{\mathbf{Y}} = \sum_{j=1}^m \mathbf{X}_j \mathbf{u}_j \quad (7)$$

with the minimum mean-square error,

$$\varepsilon = E \left\{ (\mathbf{Y} - \hat{\mathbf{Y}})^t (\mathbf{Y} - \hat{\mathbf{Y}}) \right\} \quad (8)$$

is obtained when  $\mathbf{u}_j$  ( $\forall j=1,m$ ) are the eigenvectors associated with the  $m$  largest eigenvalues  $\lambda_j$  of the covariance matrix of the mixture density ( $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m \geq \dots \geq \lambda_d$ ). The expansion coefficient  $\mathbf{X}_j$  associated with  $\mathbf{u}_j$  is the  $j$ th PCA feature of  $\mathbf{X}$ ,

$$\mathbf{X}_j = \mathbf{u}_j^t \mathbf{Y}. \quad (9)$$

The magnitude  $|\mathbf{u}_{ji}|$  of the  $i$ th component of the  $j$ th eigenvector indicates its relative contribution to the  $j$ th eigenfeature. Considering these magnitudes and before the PCA implementation, some of the measurements can be rejected. This ‘‘incomplete’’ eigenfeature approach enables a designer to trade-off between pre-selection of measurements before the projection and the extraction of a minimal number of features.<sup>(11)</sup> However, in this study we applied the PCA to all the measurements.

### 3.2. The AANN

It is sometimes helpful to view an AANN as an encoder-decoder mechanism, in which the implemented network is forced to perform an identity mapping through a deliberately small hidden layer. Forcing the mapping to proceed through a small hidden layer ensures efficient encoding. Hence, an AANN has  $d$  units in the input layer, as well as in the output layer and  $m < d$  hidden units in the hidden layer (Fig. 1). Depending upon a random initial state, the coding behavior of the hidden layer is unpredictable; however, the use of an AANN is very common, especially in the field of image compression. For example, Cottrell et al.<sup>(14)</sup> have tried to compress sub-regions of an image through a feedforward AANN trained by the BP algorithm. As Bourland and Kamp<sup>(5)</sup> claimed and Cottrell et al.

experimentally validated, the nonlinearity of the hidden units is useless. Whether or not nonlinearity is contained in the hidden layer, the AANN performs in the same way. Another conclusion was that a linear hidden layer with  $m$  units projects the input space onto the subspace spanned by the first  $m$  principal components of the input.<sup>(5)</sup> When trying to closely approximate the inputs, the outputs of the AANN should be linear. Therefore, linear output and hidden units are employed here.

### 3.3. The MLP feature extractor

When acting as a classifier, the MLP hidden units can be used as an implementation of a nonlinear projection of the patterns.<sup>(15)</sup> The projections of the patterns are more easily separated by the network output layer. Furthermore, visualization of the last hidden internal representations may supply an insight into the data structure, and hence, act as a mean of data projection. Using this approach, the classifier acts ideally as feature extractor and as an exploratory data projector. However practical considerations can often force the separation of the feature extraction stage from the classification stage. Moreover, for complex classification problems the combined feature extractor-classifier architecture can often dictate huge training periods and/or the use of large training sets. Nevertheless, it is also likely that together with the separation and simplification of the architecture there will be a deterioration in the classification performance. Thus it is both interesting and practical to consider the MLP NN both as a combined feature extractor-classifier, as well as considering two MLP NNs- one for feature extraction and the other for classification. Indeed, the second option better coincides with our study aim, which is a comparison of feature extraction paradigms. Therefore, we employed a two-layer perceptron NN both as one of the feature extraction paradigms and the classifier (Section 4F) of *all* the feature extraction paradigms. Although not acting as a classifier, the MLP feature extractor training is based on class label information, and hence, it is supervised. The number of input units is specified to be the input space dimension and the number of output units to be the number of pattern classes (Fig. 1). The hidden layer dimension is set according to the task, exploratory data projection or a classification.

## 4. The Experiments

### 4.1. The methodology

Chromosome analysis is used here as an application to compare Sammon's mapping with the PC, AANN and MLP feature extractors. This application is motivated by earlier successful attempts to classify human chromosomes.<sup>(16, 17)</sup> In the present study, the experiments were performed with 300 patterns of three classes (chromosome types "13", "19" and "x"), using 100 patterns of each type. The chromosome patterns were represented by 64 density profile (d.p.) features, which are integral intensities along sections perpendicular to the medial axis of the chromosome.<sup>(16, 18)</sup>

As Fig. 2 indicates, the paradigms extract features of the 64-dimensional chromosome patterns. The outputs of the four feature extraction paradigms are used to project the patterns onto two-dimensional maps and to train and test an MLP classifier. The two-dimensional projection maps are visually analyzed and compared with a two-dimensional scatter plot of two of the original features. The MLP probability of correct classification is evaluated for various numbers of extracted features and compared with the same probability based on the first 10 original features. These first 10 original d.p. features, which are extracted from the upper tip of the chromosome, provide the cytotechnician with an enhanced discriminative capability and, in addition, they are ranked among the "best" 16 d.p. features.<sup>(16)</sup>

Twenty-one training and test sets were derived from the entire chromosome data set for the classification experiments. Each training set contained 90% of the data set randomly selected, while the remaining patterns were reserved for the test (the holdout method<sup>(3)</sup>). Each feature extraction paradigm was applied to these data sets and the classification results were averaged over the twenty-one data sets and ten classifier initializations (Section 4F).

### 4.2. Sammon's mapping

The configuration and mapping parameters, which based on experience, yielded the highest classification performance of the NN implementation of Sammon's mapping were used in this study.

That experience suggested an NN trained for 40 epochs using a learning rate of 0.9, a momentum constant of 0.5 and 64 hidden units. When the PC based initialization was investigated, eigenvectors corresponding to the largest eigenvalues replaced the random initial input-hidden weight vectors. However, the initial hidden-output weight matrix was randomly selected.

#### *4.3. The PC feature extractor*

The eigenvectors corresponding to the first four to ten and the first two eigenvalues were respectively used in the classification and the exploratory data projection experiments.

#### *4.4. The AANN*

A two-layer perceptron trained by the BP algorithm with the same input and output was employed as an AANN. The input (output) was the 64-dimensional d.p. feature vector whereas the hidden layer dimension was set by the experiment (2 and 4 to 10 in the exploratory data projection and classification experiments, respectively). The AANN parameters, which were previously found to yield satisfactory results were: a learning rate of 0.1, a momentum constant of 0.95 and a training period of 50 epochs. All initial weight matrices were randomly selected.

#### *4.5. The MLP feature extractor*

A two-layer perceptron trained by the BP algorithm was used as a feature extractor. The input layer was 64-dimensional and the number of hidden layer units was set at 2 in the exploratory data projection experiment and it was changed from 4 to 10 during the classification experiments. The two initial weight matrices were randomly selected.

#### *4.6. The classifier*

Higher complex architectures than the two-layer perceptron are not considered here as candidates for the classifier because only a comparative study of feature extraction paradigms is concerned. The

number  $m$ , of input units is set by the projected space dimension and the number of output units is determined by the number of classes (three in this case). The classifier parameters are<sup>(17)</sup>: learning rate of 0.1, momentum constant of 0.95, 2 hidden units and a training period of 500 epochs. Each experiment with the classifier is repeated ten times with different randomly chosen initial weight matrices and the results are averaged. The same ten classifier initializations are used to investigate all the feature extraction paradigms.

## 5. Experimental Results

An evaluation of Sammon's mapping as a feature extraction paradigm for both exploratory data projection and classification, as compared with the PC, AANN and MLP feature extractors is made here.

### 5.1. Projection maps

Figure 3 presents the projection of the patterns onto the plane defined by the 1<sup>st</sup> and 2<sup>nd</sup> original d.p. features. Figure 3 reveals a great deal of overlap among the three chromosome clusters, which may result in relatively poor classification performance using these features. The two-dimensional projection maps of the PC feature extractor, Sammon's mapping, MLP and AANN are shown, respectively in Figs. 4(a)-(d). The randomly initialized configuration of Sammon's mapping is preferred here because the PC based initial configuration is found to yield very similar maps to those of the PC feature extractor.<sup>(7)</sup> The maps of Fig. 4 were obtained using 50 test patterns *per* class. Producing the same maps for the case that was tested in the classification experiment (90% of the data set used for training) is of less interest because only ten test patterns *per* class were available for the experiment. The evaluation of the projection maps is based on visual judgment which is, in our opinion, the best qualitative way to evaluate these maps, except for complex psychophysical experiments. A quantitative evaluation of the projections appears to be inherently biased toward one of the paradigms. For example, Mao and Jain<sup>(7)</sup> when using Sammon's stress for a quantitative evaluation of projection methods,

ranked Sammon's mapping as the best projection method. Visually analyzed, the maps of the PC and the MLP are clearer than the others and the pattern spread is more evident. Moreover, the ratio of the *between*-class scatter to the *within*-class scatter of these two maps is larger. It should not be forgotten however, that projecting along the axes with the largest data variances, as the PCA does, is the easiest way to interpret projection maps. Considering discriminative power, the maps of the MLP are superior, the maps of the PCA and Sammon's mapping are second best with a slight advantage to the PCA and the AANN maps are the least discriminative among all maps. It is important to mention, however, that the MLP is a supervised feature extraction paradigm while the other are unsupervised. Another interesting point to observe is the way the MLP shrinks each class pattern to almost one point (or line), a quality which eases the classification process. These shrunken clusters are (almost) concentrated in three of the four map corners corresponding to the ultimate values of the hidden unit activation function (sigmoid).

## 5.2. Classification

We have used the MLP probability of correct classification of the test set as the criterion to evaluate the classification performance using the four feature extraction paradigms for 4 to 10 extracted features (Fig. 5). For comparison, the same probability using the first 10 original d.p. features is indicated by an asterisk (\*) in Fig. 5. As is shown in the figure, the MLP feature extractor and Sammon's mapping yield the highest performance, which is superior to that of the PC and AANN feature extractors. In addition, all the paradigms achieve better performance than the first 10 original d.p. features for almost any number of extracted features.

To compare the two distinct initializations of Sammon's mapping we used an NN configuration based on 20 hidden units with either random or PC based initial weights. The probability of correct classification for various mapping dimensions (network outputs) in the [1,15] range was examined. This probability was averaged over 5 randomly chosen data sets and 10 randomly initialized classifiers (total of 50 experiments). Fig. 6 shows the superiority of the PC based initialization over the random

one for almost every number of projections. This superiority of the PC initialization contributes to the remarkable classification performance which Sammon's mapping yields compared with that of the other feature extraction paradigms (Fig. 5).

Finally, the inferiority of the AANN compared with the other paradigms encourages another experiment. If this inferiority is due to insufficiency of discriminative competence, increasing the extraction ability of the hidden layer may be helpful. Fig. 7 shows the probability of correct classification for an extended range of extracted features. The AANN can gradually improve its generalization ability to achieve almost comparable performance with the other paradigms before a decline in performance occurs, probably due to the "curse of dimensionality" effect.

## 6. Discussion

An NN implementation of Sammon's mapping is compared in this investigation with the PC, AANN and MLP feature extractors for exploratory data projection and classification. The implementation of Sammon's mapping, as well as those of the AANN and MLP feature extractors, employ adaptive learning algorithms, which supply flexibility to changing environments and enhanced generalization capability. In these three paradigms, the information about the importance of an extracted feature is gained iteratively during the training of the network and is accommodated in the network weights. The PC feature extractor is an unsupervised linear approach in which its eigenvalues implicitly accommodate the information contained in the data. For these paradigms and chromosome analysis, we have found a strong relationship between a highly visual exploratory projection map and high discriminatory power. Superior paradigms for exploratory data projection are very often found to be superior paradigms for classification and *vice versa*.

Although originally designed and used for exploratory data projection, Sammon's mapping is found here to have impressive classification capability of chromosome data. When the eigenvectors of the sample covariance matrix replace the random initialization of Sammon's mapping, even *one* experiment yields superior classification results.

In addition, it has been concluded that a combination of a nonlinear feature extraction paradigm and class information improves discriminative capability. The MLP feature extractor, which is a supervised nonlinear paradigm, is found to be a preferred feature extraction paradigm for both chromosome pattern classification and data projection. A similar conclusion about this MLP superiority was drawn in other applications<sup>(7)</sup> as well. The AANN, on the other hand, discloses a trade-off between an acceptable generalization capability and a beneficial compression ratio.

Finally, this study presents the advantage of applying feature extraction paradigms to chromosome analysis, either to understand the chromosome data structure through projection maps or to improve the classification of the chromosomes into their types.

## References

1. Y. Chien, *Interactive Pattern Recognition*. NY: Marcel Dekker, Inc. (1978).
2. P. A. Devijver and J. Kittler, *Pattern Recognition- A Statistical Approach*. NJ :Prentice Hall (1982).
3. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd edn.. New York: Academic Press (1990).
4. J. W. Sammon Jr., A nonlinear mapping for data structure analysis, *IEEE Trans. Comput.* **18**, 401-409 (1969).
5. H. Bourland and Y. Kamp, Auto-association by multilayer perceptrons and singular value decomposition, *Biol. Cybern.* **59**, 291-294 (1988).
6. T. Kohonen, The self organizing map, *Proc. IEEE* **78**, 1464-1480 (1990).
7. J. Mao and A. K. Jain, Artificial neural networks for feature extraction and multivariate data projection, *IEEE Trans. Neural Networks* **6**, 296-317 (1995).
8. J. Rubner and K. Schulten, Development of feature detectors by self-organization, *Biol. Cybern.* **62**, 193-199 (1990).

9. T. D. Sanger, Optimal unsupervised learning in a single-layer linear feedforward neural network, *Neural Networks* **2**, 459-473 (1989).
10. E. Saund, Dimensionality-reduction using connectionist networks, *Trans. Pattern Anal. Machine Intell.* **11**, 304-314 (1989).
11. B. Lerner, H. Guterman, M. Aladjem and I. Dinstein, Unsupervised feature extraction for nonlinear supervised classification with application to chromosome analysis, in *Proc. Int. Conf. Neural Information Process. (ICONIP'95)* **1**, Beijing, 279-284 (1995).
12. I. White, Comment on 'A nonlinear mapping for data structure analysis', *IEEE Trans. Comput.* **21**, 220-221 (1972).
13. R. D. Dony and S. Haykin, Neural network approaches to image compression, *Proc. IEEE* **83**, 288-303 (1995).
14. G. W. Cottrell, P. Munro and D. Zipser, Learning internal representations from gray-scale images: An example of extensional programming, in *9th Ann. Conf. Cognitive Sci. Soc.*, Seattle. Hillsdale: Erlbaum, 462-473 (1987).
15. R. P. Gorman and T. J. Sejnowski, Analysis of hidden units in a layered network trained to classify sonar targets, *Neural Networks* **1**, 75-89 (1988).
16. B. Lerner, H. Guterman, I. Dinstein and Y. Romem, Medial axis transform based features and a neural network for human chromosome classification, *Patt. Rec.* **28**, 1673-1683 (1995).
17. B. Lerner, H. Guterman, I. Dinstein and Y. Romem, Human chromosome classification using multilayer perceptron neural network, *Int. J. Neural Syst.* **6**, 359-370 (1995).
18. A. Carothers and J. Piper, Computer-aided classification of human chromosomes: a review, *Statistics and Computing* **4**, 161-171 (1994).

## Figure caption

Fig. 1. A two-layer perceptron used for the implementation of the NN based feature extraction paradigms.

Fig. 2. The experimental layout.

Fig. 3. Two-dimensional scatter map of two d.p. features (the 1st and 2nd) (“o”, “\*” and “x” for chromosome types “13”, “19” and “x”, respectively).

Fig. 4. Two-dimensional projection maps using the four paradigms: (a) the PC, (b) Sammon’s mapping, (c) the MLP and (d) the AANN (“o”, “\*” and “x” for chromosome types “13”, “19” and “x”, respectively).

Fig. 5. The probability of correct classification of the test using the four paradigms for an increasing number of extracted features (PCA (\*\*\*), Sammon’s mapping (PC based initialization) (-\*-), MLP (—), AANN (---) and the first 10 d.p. features (\*)).

Fig. 6. The probability of correct classification of the test using a different number of Sammon’s projections and random (—) and PC (---) based initializations.

Fig. 7. The probability of correct classification of the test using the AANN for a various number of hidden units (features).

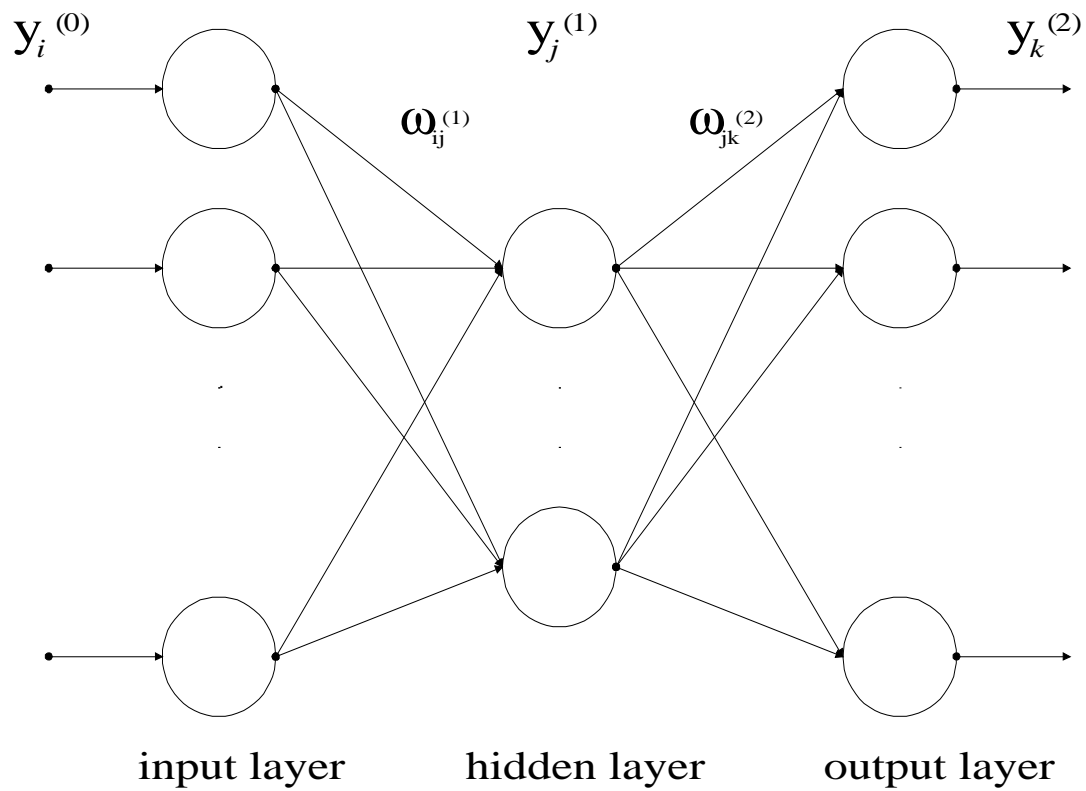


Fig. 1.

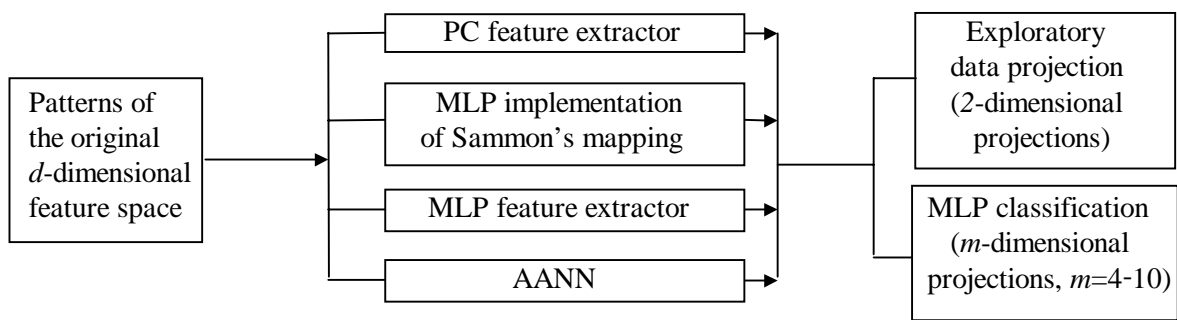


Fig. 2.

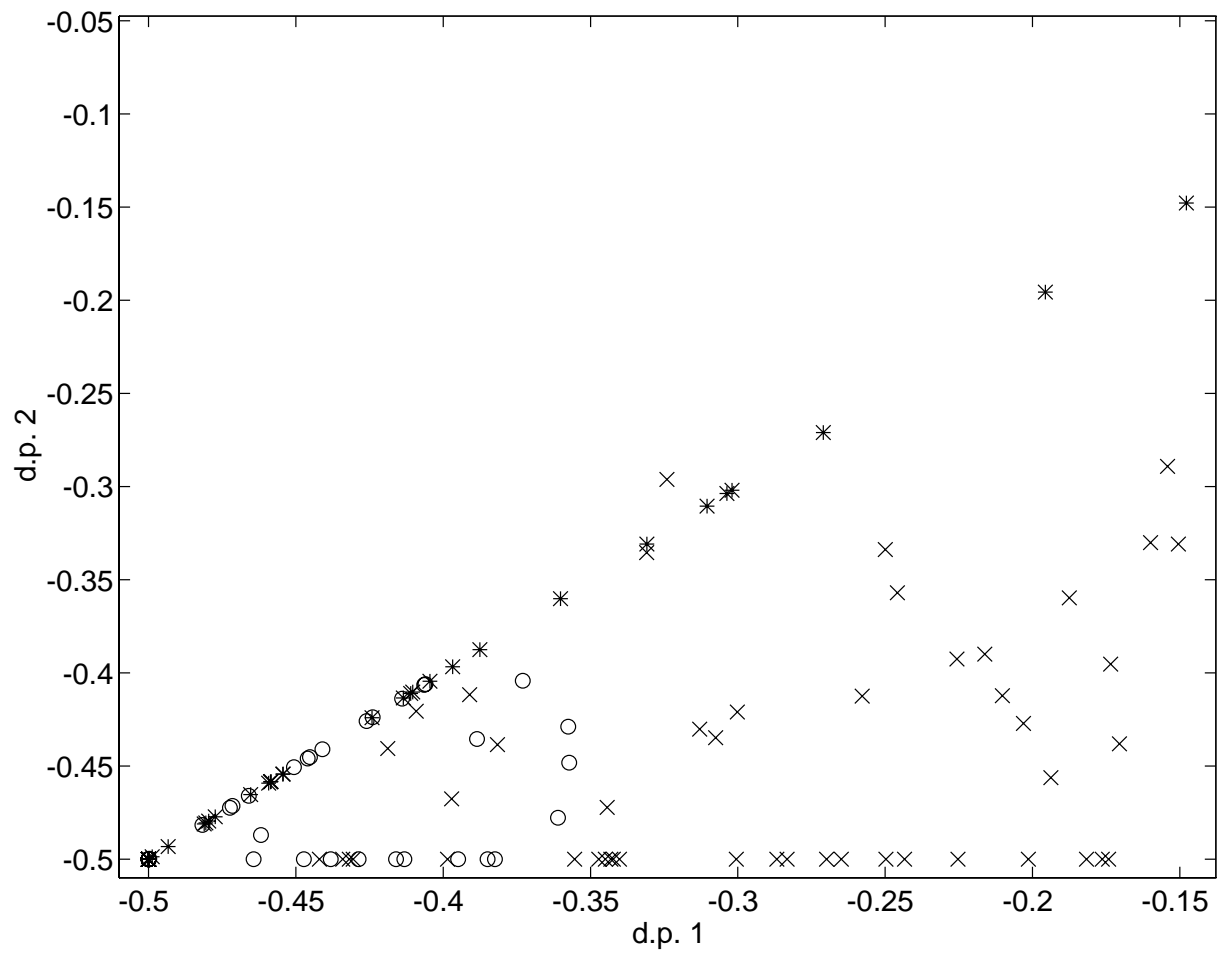


Fig. 3.

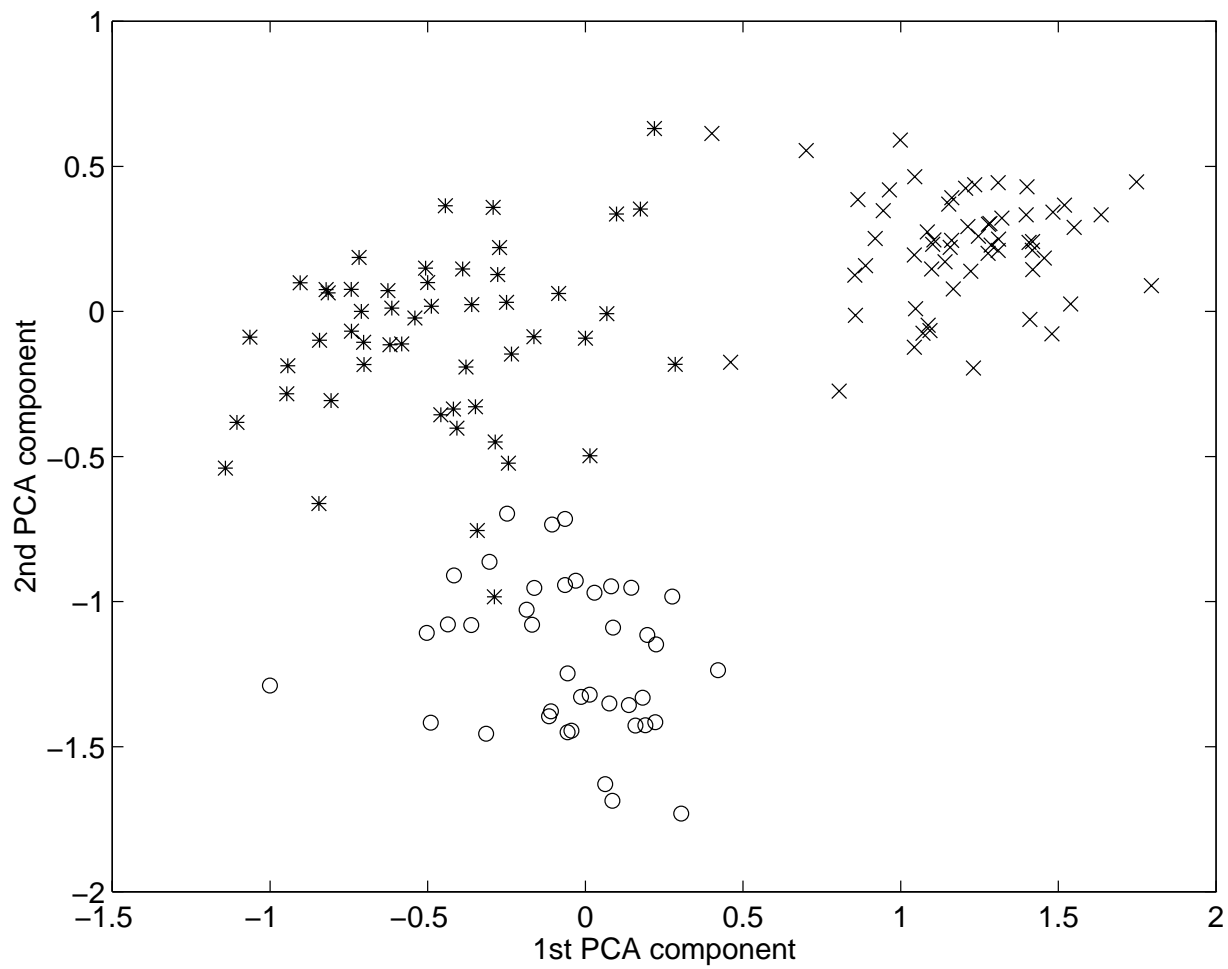


Fig. 4a.

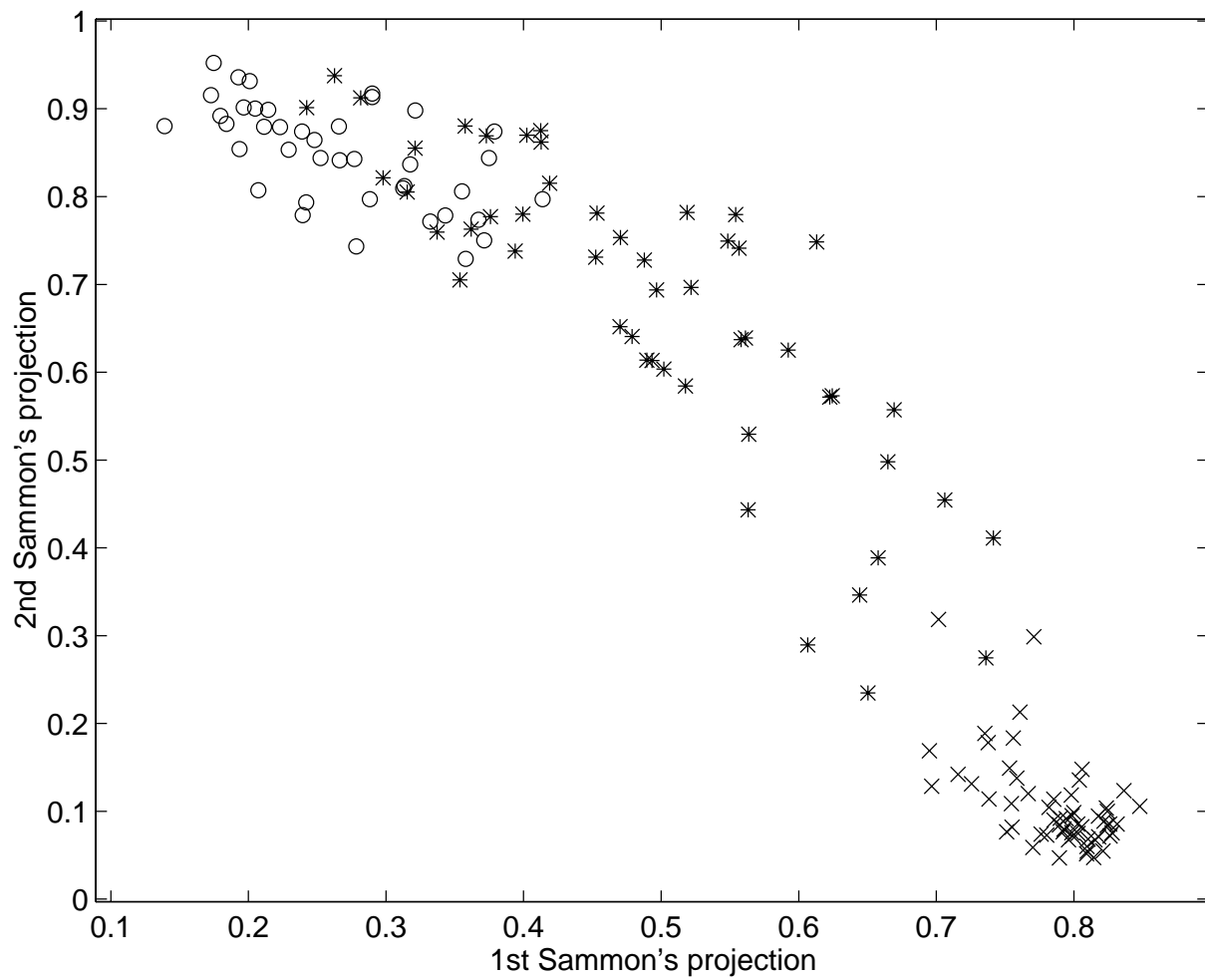


Fig. 4b.

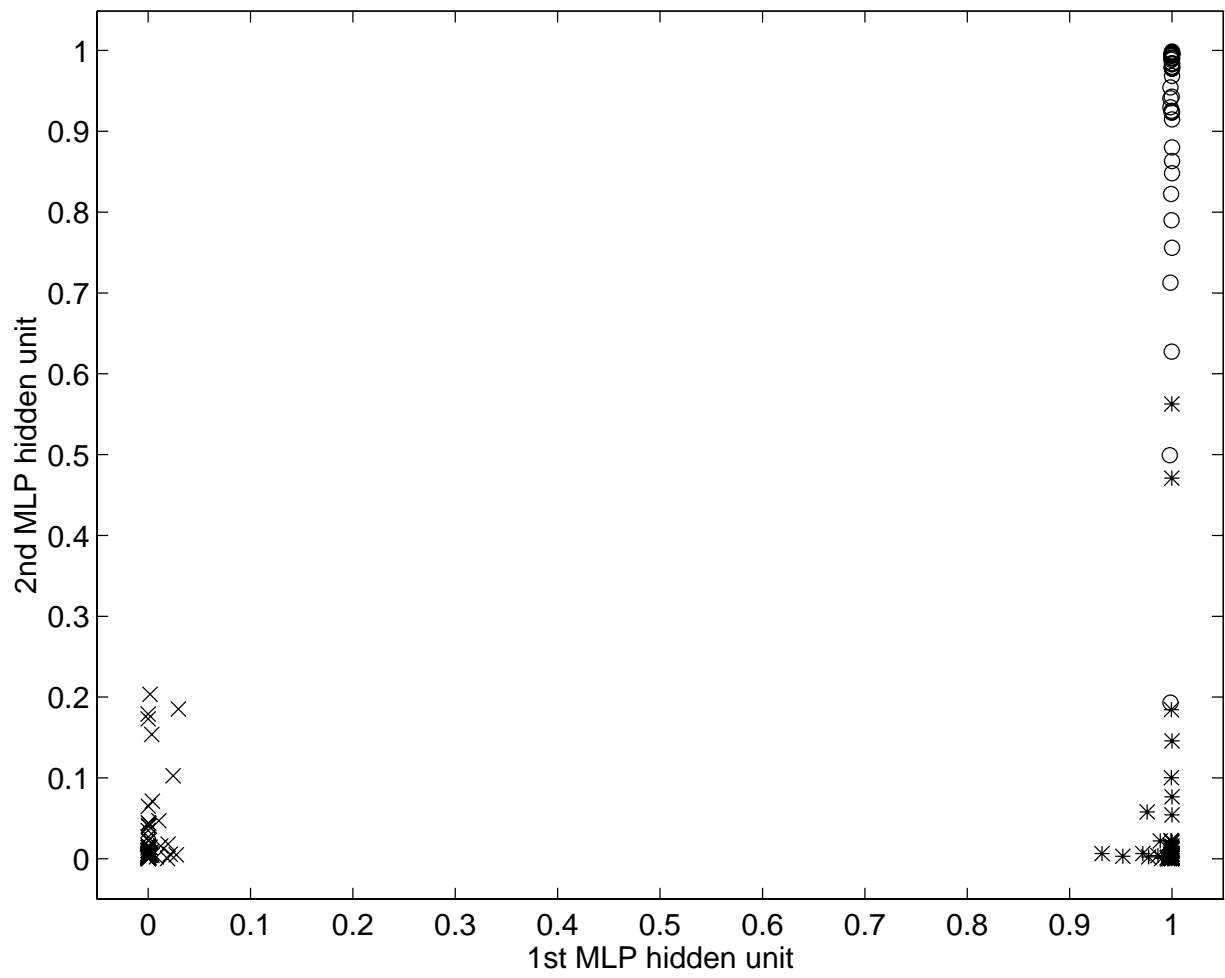


Fig. 4c.

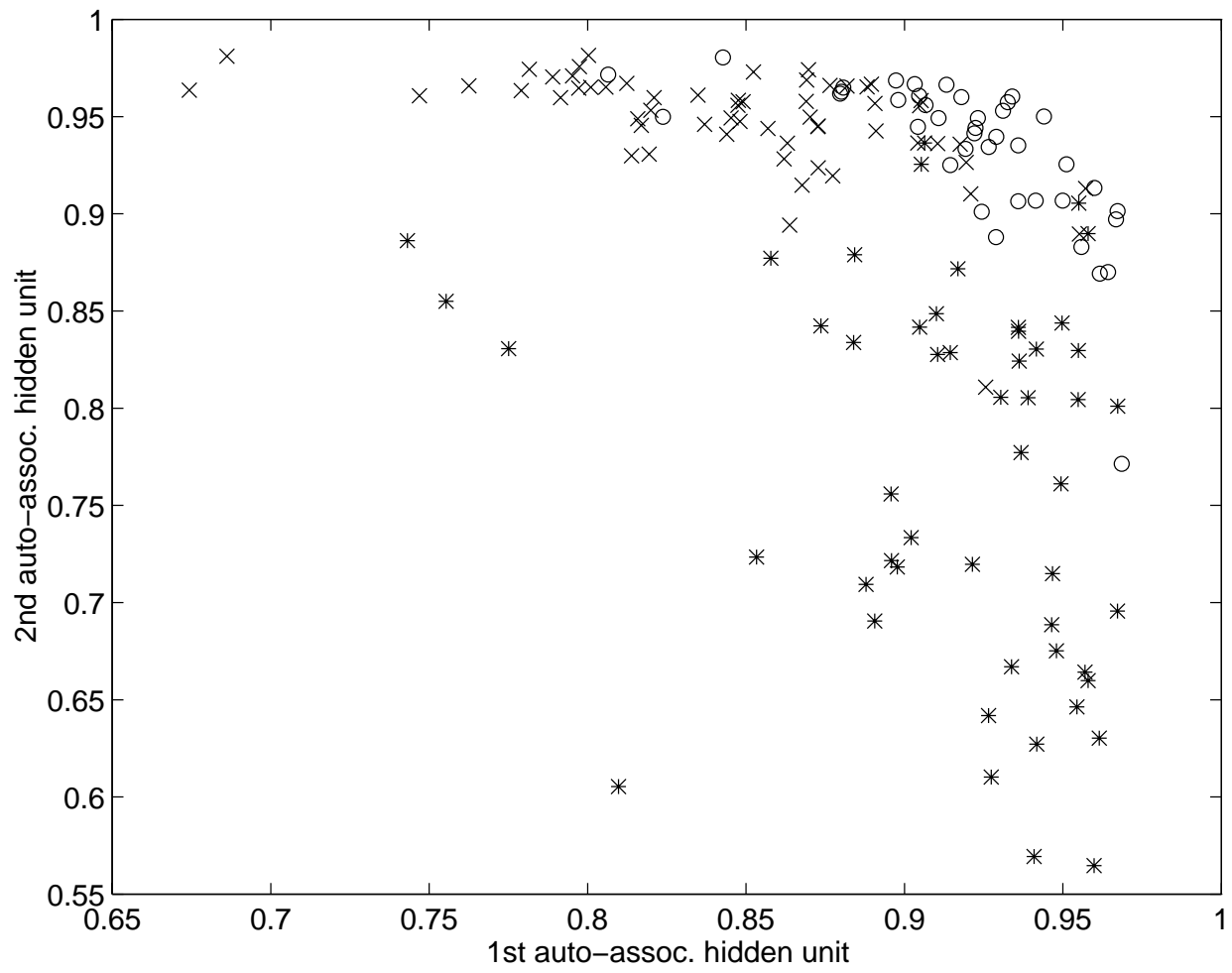


Fig. 4d.

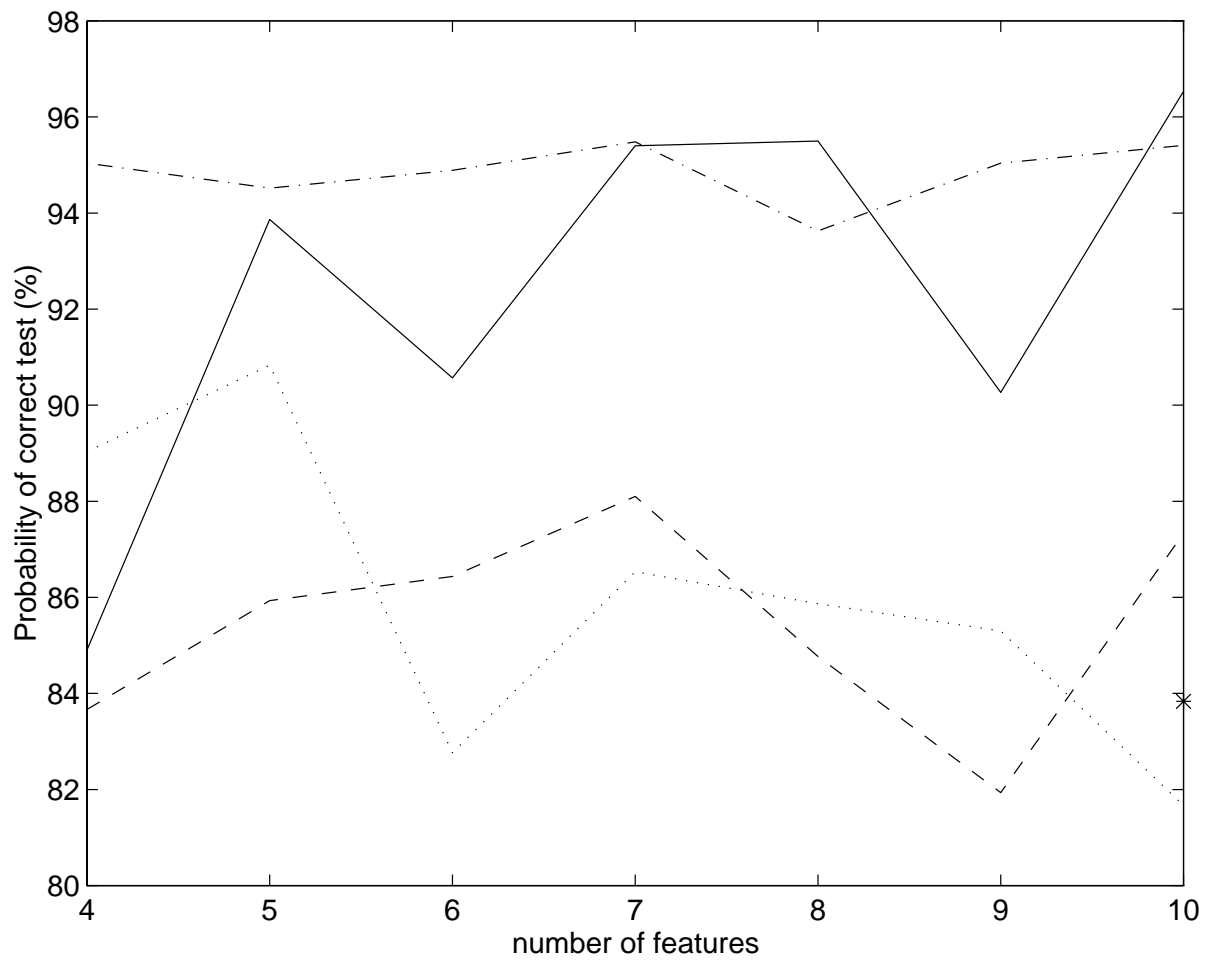


Fig. 5.

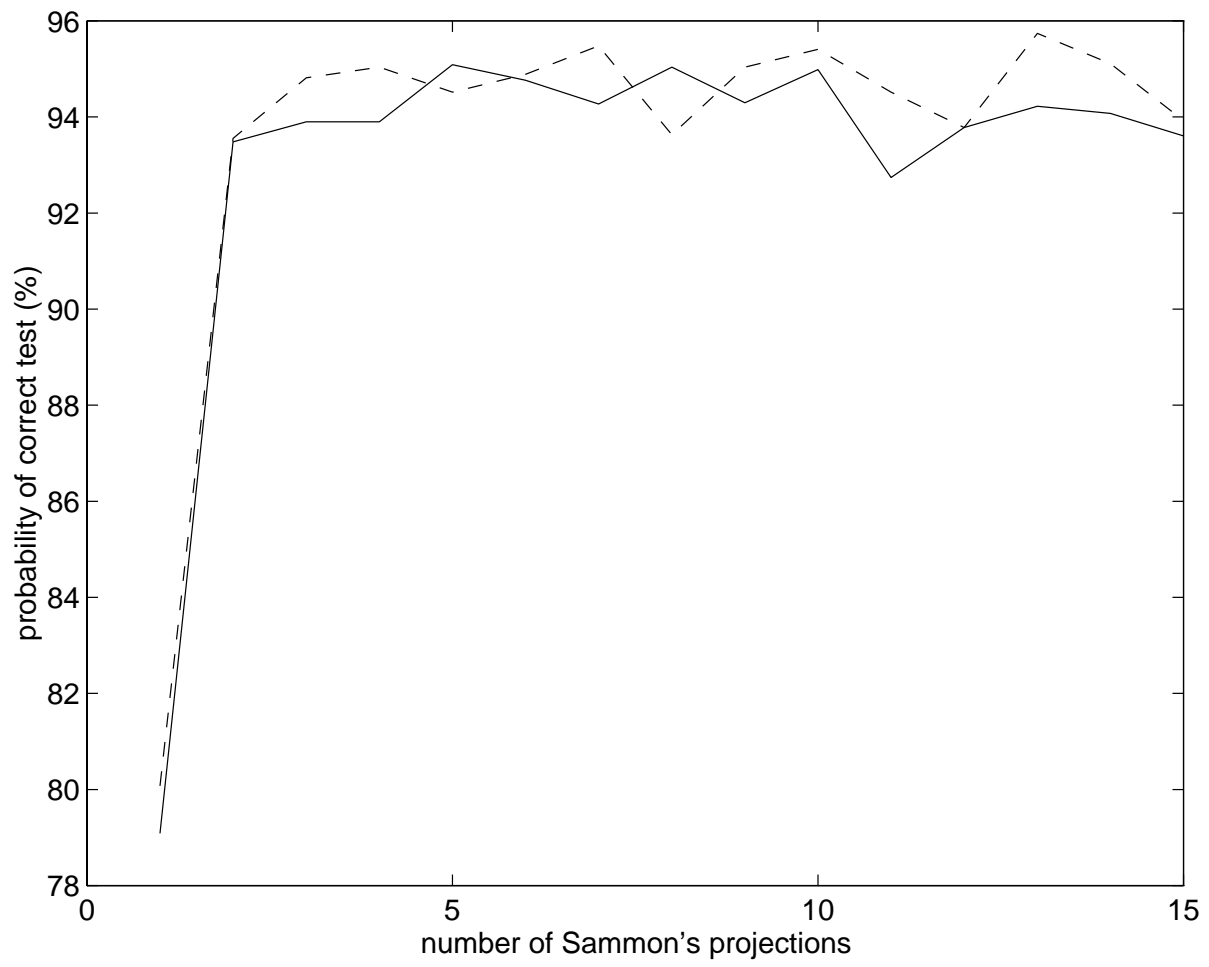


Fig. 6.

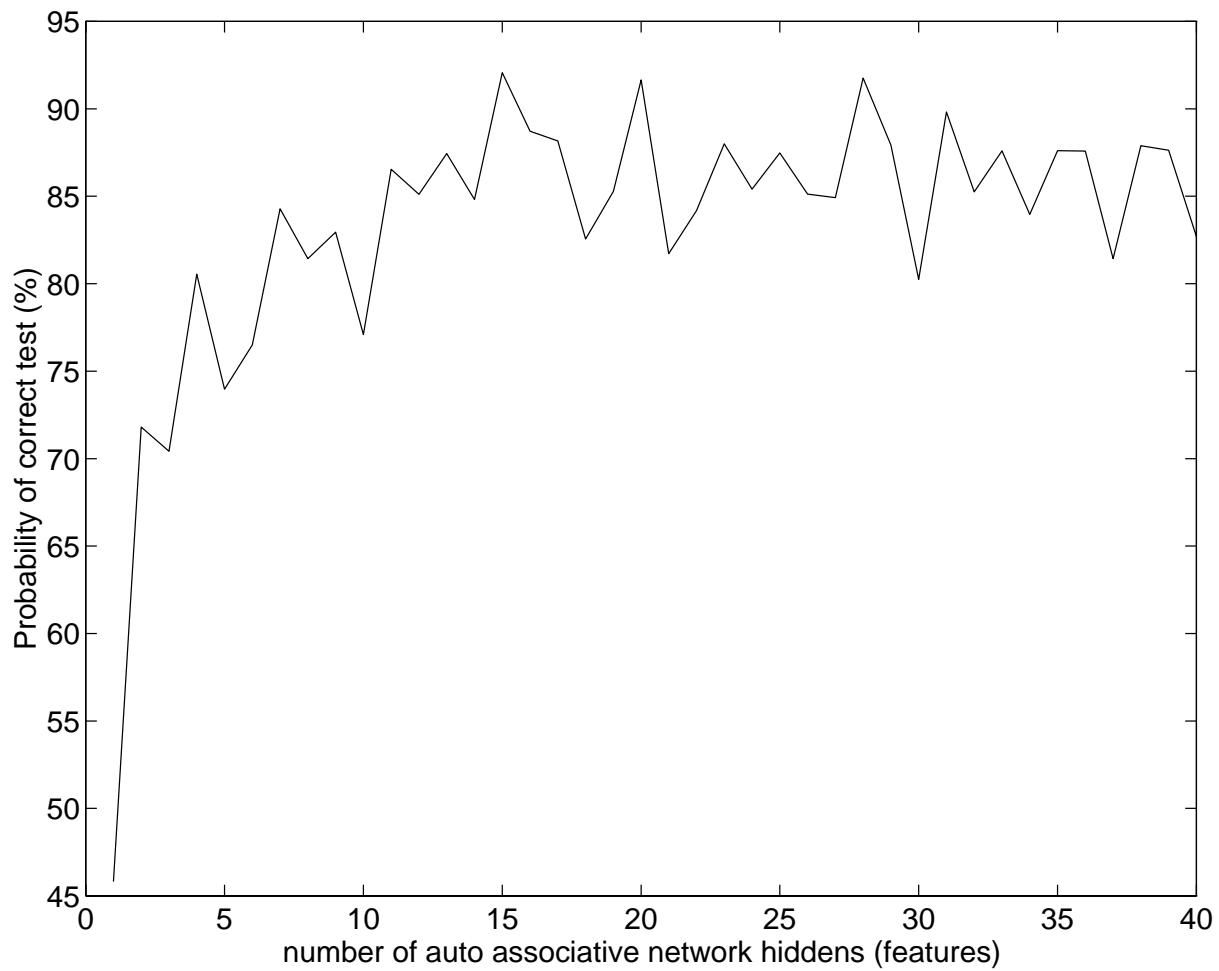


Fig. 7.

Boaz Lerner is currently a Research Fellow with the University of Cambridge Computer Laboratory, Cambridge, UK. He received his B.A from the Hebrew University, Israel, in 1982, M.Sc. from Tel-Aviv University, Israel, in 1988 and a Ph.D. from Ben-Gurion University, Israel, in 1996. His research interests include neural networks, pattern recognition and image processing.

Hugo Guterman received a Ph.D. degree in electrical and computer engineering from the Ben-Gurion University of the Negev, Israel in 1988. During the years 1988-90 he was a visiting researcher at the Center of Biotechnology at the MIT. He is currently a senior lecturer at the Department of Electrical and Computer Engineering at the Ben Gurion University of the Negev. His research interests include control theory, signal and image processing, biotechnology, neural networks and fuzzy logic.

Mayer Aladjem received the MSc and PhD degrees in electrical engineering from the Technical University, Sofia, Bulgaria in 1975 and 1980, respectively. He was an Associate Professor of Biomedical Cybernetics in the Central Laboratory on Bioinstrumentation and Automation of the Bulgarian Academy of Sciences (1988-1990). He is currently a senior lecturer in the Department of Electrical and Computer Engineering at Ben-Gurion University of the Negev, Israel. His research interest is in feature extraction and selection methods, interactive pattern recognition, systems for data analysis and their applications in biology and medicine.

Itzhak Dinstein received his Ph.D. in electrical engineering from the University of Kansas, Lawrence Kansas, in 1974. He was with COMSAT Laboratories, Gaithersburg Maryland, from 1974 till 1977, when he returned to Israel and joined Ben-Gurion University, in Beer Sheva. He is now a full professor at the Electrical and Computer Engineering Department. During 1982-1984 he was a visiting scientist at IBM Research Laboratory, San Jose, California. During 1988-1990 he was a visiting associate professor at Polytechnic University, Brooklyn, New York. His research interests include image processing and computer vision.

Yitzhak Romem received his M.D. degree from the Hadassa School, Hebrew University, Jerusalem, Israel. He specialized in Obstetrics and Gynecology. From 1982 to 1985 he was a fellow in Maternal-Fetal Medicine at Women's Country Hospital, Los Angeles, University of Southern California. He was

the director of the Genetics Institute at Soroka Medical Center, Ben-Gurion University until 1996. He is interested mainly in Prenatal Diagnosis of Congenital Disorders and established the Prenatal Screening for Down's Syndrome for the population of southern Israel. His interest includes the use of computers in Medicine.