Learning low dimensional representations of visual objects with multiple use of prior knowledge

Learning low-dimensional representations

Nathan Intrator† § and Shimon Edelman‡

† Institute for Brain and Neural Systems
Brown University
Providence, RI 02192
nin@cns.brown.edu

‡ Center for Biological and Computational Learning
MIT E25-201
Cambridge, MA 02142
edelman@ai.mit.edu

Short title: Learning low dimensional representations of visual objects with multiple use of prior knowledge

February 14, 1997

§ On leave: School of Mathematical Sciences, Tel-Aviv University.

Abstract. Learning to recognize visual objects from examples requires the ability to find meaningful patterns in spaces of very high dimensionality. We present a method for dimensionality reduction which effectively biases the learning system by combining multiple constraints via the use of class labels. The use of multiple class labels steers the resulting low-dimensional representation to become invariant to those directions of variation in the input space that are irrelevant to classification; this is done merely by making class labels independent of these directions. We also show that prior knowledge of the proper dimensionality of the target representation can be imposed by training a multi-layer bottleneck network. Computational experiments involving non-trivial categorization of parameterized fractal images and of human faces indicate that the low-dimensional representation extracted by our method leads to improved generalization in the learned tasks and is likely to preserve the topology of the original space.

1. Introduction

Learning to recognize visual objects in images – represented as collections of pixel values – requires the ability to find meaningful patterns in spaces of tens or hundreds of thousands of dimensions. The resulting need for models with an extremely large number of parameters raises the problem of sparse data: the number of observed samples (training patterns) is smaller or similar to the number of model parameters that must be estimated. Even simple and widely used methods such as linear discriminant analysis (Fisher, 1936) should be adjusted to reflect the low ratio of training samples to the number of parameters (Buckheit and Donoho, 1995). Thus, it is clear that neural network learning methods require special care when applied to problems arising in vision.

1.1. General approaches to the facilitation of learning

A fundamental assumption frequently made in an attempt to alleviate the problem of sparse data, also known as *curse of dimensionality* (Bellman, 1961), is the existence of a low dimensional representation (LDR) of the problem space. Yet, postulating that an LDR exists does not provide an efficient way to find it. To do so, one may for instance, further assume that the data are clustered. If indeed data points belonging to the same class are clustered in the high dimensional space, a useful LDR can be found by looking for projections that emphasize the cluster structure (Intrator, 1993).

More generally, innovative use of the training data is needed. For example, methods for data reuse such as cross-validation (Stone, 1974) and bootstrap (Efron and Tibshirani, 1993) can help in obtaining confidence intervals (Baxt and White, 1995) and improved performance (Breiman, 1992; Breiman, 1994; LeBlanc and Tibshirani, 1994) of learning networks. Smooth bootstrap (Efron and Tibshirani, 1993) can also increase the independence among predictors for the purpose of ensemble averaging (Raviv and Intrator, 1996). Such methods lead to a reduction in the variance portion of the error, with little or no effect on the bias of the predictor.

One can also control the variance portion of the error by imposing global assumptions about the nature of the predictor that is to be learned. These include smoothness (Wahba, 1990; Poggio and Girosi, 1990), as well as assumptions about the distribution of the parameters, e.g., favoring small weights via a weight decay process, or favoring particular distributions such as mixtures of Gaussians (Nowlan and Hinton, 1992). A general framework for imposing such constraints is presented in (Intrator, 1993).

1.2. Specific assumptions and prior knowledge

Unlike data, class labels are not often reused to facilitate learning (see however, (Grossman and Lapedes, 1993)). In particular, few learning algorithms can accommodate multiple-class labels, which are likely to contain useful information regarding the structure of the data. Humans make natural and multiple use of the knowledge that objects may have several class associations (say, at different category levels). In contrast, in machine learning, it is not clear how one should proceed given multiple class labels, and whether such information can be used effectively or at all. We believe that through the use of multiple-class associations, learning can be constrained – biased – towards a better solution, and that innovative use of multiple-class labels may be a practical way to introduce prior knowledge into a high-capacity learning machine. We present a method for introducing such prior information during training, while avoiding the need to construct different low-level representations for different tasks defined on the same data. This approach naturally facilitates generalization across tasks, also known as transfer of skill — a hallmark of human cognitive prowess (see section 1.1 in (Intrator and Edelman, 1996) for review).

It has been observed in the past that training a classifier on multiple tasks (using the same data) may be an efficient way to introduce desirable bias into the solution (Caruana, 1993) and to improve generalization (Caruana, 1995). We further argue that forcing the learning system to use multiple class label information, and letting it ignore dimensions of data variation with respect to which invariance is required, leads to the extraction of an LDR that captures the dimensions relevant to the task, and is orthogonal to the dimensions along which the variance is irrelevant (e.g., the orientation of an object in a visual recognition task). Note that a complementary approach is to learn, instead of a variety of labeling schemes for a given data set, the *transformations* which leave its members invariant (Lando and Edelman, 1995), or the *invariances* of the individual data items (Simard et al., 1992; Thrun and Mitchell, 1995).

1.3. Topology-preserving dimensionality reduction

As we shall see, generating an LDR through multiple use of class labels generally results in the preservation of the topology of a low-dimensional space containing the examples (section 4). As topology-preserving mapping is the ultimate goal of a number of methods of dimensionality reduction, it is appropriate to mention here the typical approaches taken by these methods. The oldest among these is multi-dimensional scaling (MDS) (Young and Householder, 1938). It is discussed at some length in the appendix. The main problem with MDS, if it is considered as a method for massive dimensionality reduction rather than as a tool for exploration of experimental data in applied sciences (Shepard, 1980; Siedlecki et al., 1988), is its poor scaling with dimensionality.

In the context of learning, a number of methods for topology-preserving dimensionality reduction have been derived from the idea of a self-supervised auto-associative network (Elman and Zipser, 1988; DeMers and Cottrell, 1993; Kambhatla and Leen, 1994; Demartines and Hérault, 1996). Because these methods are unsupervised, they extract representations that are not necessarily orthogonal to the irrelevant dimensions of the input space. An interesting approach that combines supervised feature extraction with topology preservation was proposed in (Koontz and Fukunaga, 1972; Webb, 1995), whose dimensionality reduction algorithms explicitly optimize a joint measure of class separation and (input-space) distance preservation. This approach, which resembles MDS, suffers from the same poor scaling with the dimensionality. The performance of Radial Basis Function network as a topology preserving dimensionality reduction method has been studied by Lowe and colleagues (Lowe, 1993; Lowe and Tipping, 1996) and compared with other methods including MDS and Kohonene and Sammon mapping.

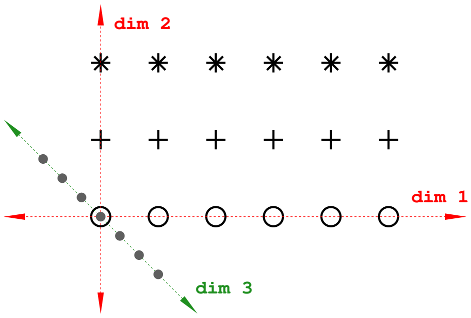
1.4. Overview of the paper

In the present paper, we use objects belonging to parametrically defined low-dimensional families to demonstrate that training with a combined objective of (1) discrimination among labeled categories known to reside within the same domain and (2) explicit collapse of dimensions over which discrimination is to be generalized, leads to a reliable recovery of the target low-dimensional manifold. Our method is effective even when the manifold to be extracted from the data is curved (i.e., when the problem is nonlinear), and is embedded in a measurement space of nearly a thousand dimensions. Furthermore, it allows the construction of classifiers of considerably lower complexity for other tasks involving the same objects, compared to what is possible under the usual approach of learning a separate representation for each task. As a control, we show that neither principal component analysis of the data, nor its nonlinear extension (implemented, respectively, by three-layer and by five-layer "bottleneck" autoencoder networks) can extract a meaningful LDR from our data. In addition we show that if class labels are not used to specify the invariance dimensions (i.e., directions that are orthogonal to the target LDR manifold), the extraction of the LDR also fails.

The paper is structured as follows. Section 2 introduces the fractal patterns and the face images, and explains the manner in which these data sets were generated to facilitate subsequent evaluation of the LDR extraction method. Section 3 explains in detail the variations on the basic LDR extraction method that we devised. Section 4 then presents the results obtained from a computational evaluation of the different methods. Finally, section 6 summarizes the results and relates them to some recent evidence of the relevance of dimensionality reduction to biological vision.

2. Data sets

To test the ability of a neural network to discover simple structure embedded in a high-dimensional measurement space, we created two data sets, in both of which the discovery of the LDR requires a highly nonlinear transformation on the measurement space. In the creation of each data set, we started with a two-dimensional parametric representation space, in which we placed 18 classes of objects on a regular 3×6 grid; an additional parametric dimension, orthogonal to the first two, modeled the within-class variation (see Figure 1). The first data set, Fractals, was computer-generated, while the second set, Faces, was derived from natural visual stimuli (3D laser-scanned human heads).



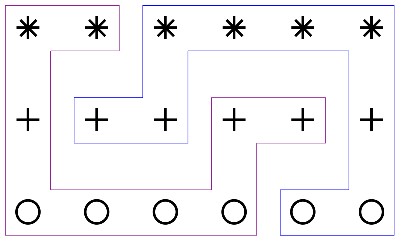


Figure 1. *Left:* The parametric representation from which the high dimensional data sets were created. Dim 3 is the within-class dimension of variation, used in training the LDR extractors, as explained in section 3 (see also Figure 6). *Right:* The dichotomy classification task, used in testing the LDR.

2.1. Fractals

The patterns in the Fractals data set were generated using publicly available software (Xfractint 2.03). We chose the quaternion Julia set (entry quatjul in the Xfractint pattern menu), which is parameterized by six variables and therefore can be used for generating complicated patterns that depend on up to six parameters. The quatjul iteration formula is

$$\begin{aligned} q(0) &= (xpixel, ypixel, z_j, z_k) \\ q(n+1) &= q(n) * q(n) + c, \end{aligned}$$

where both q and $c = (c_1, c_i, c_j, c_k)$ are quaternions (for further details, see (Pickover, 1990), chapter 10). The three dimensions shown in Figure 2 correspond to the variation of parameters c_1, c_j, and c_k, respectively.

Note that the transformation from the 3D parameter space to the image space, implied by the above formula, is highly nonlinear. To quantify this characteristic of the data set, which bodes severe problems for linear projection methods for dimensionality reduction, we have carried out a Principal Component Analysis (PCA) of the data. We found that projection on the first 2, 5, and 13 eigenvectors accounted for 16.8%,

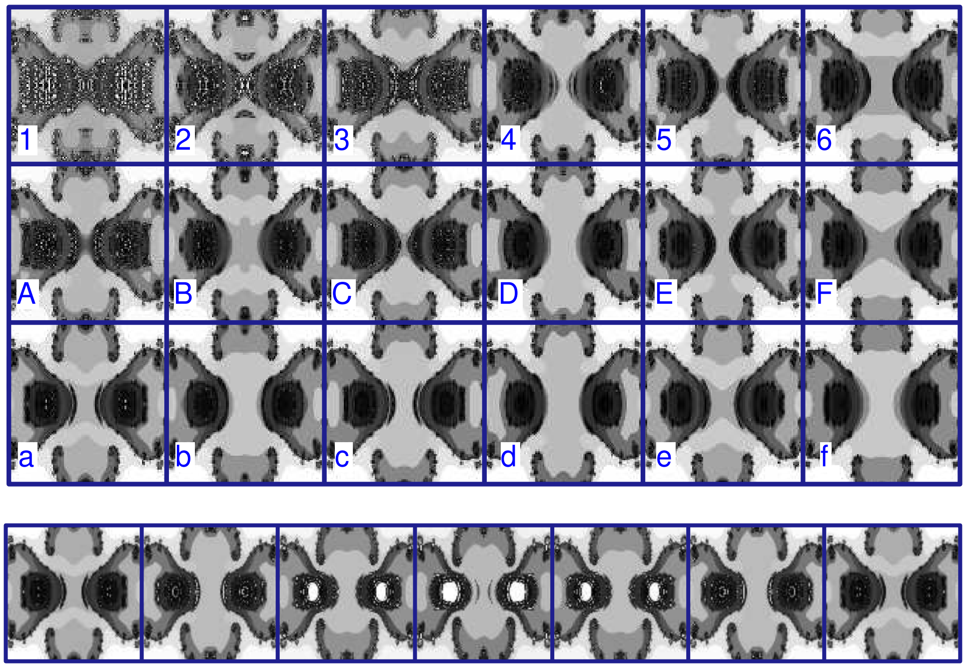


Figure 2. Some of the images from the Fractals data set (section 2.1). The 18 images in the upper part of the figure correspond to the 18 classes of fractal objects; only one member of each class, corresponding to one value along the third dimension of variation, is shown (cf. Figure 1). The seven images in the lower part of the figure are the seven exemplars from class a. Prior to classification, the images, originally of size 256×256, were reduced to $28 \times 28 = 784$ dimensions by correlation with a bank of filters (section 2.3).

31.8%, and 58.6% of the variance, respectively (see Figure 3, left; the dimensionalities specified above will figure in the reports of the performance of the different LDR extraction methods, listed in section 4).

To what extent could the PCA-derived LDR be regarded as a good replica of the original 2D parametric space in which the data were embedded? We tested the 13-dimensional PCA-derived projection of the data for planarity using multidimensional scaling (MDS) — a procedure designed to embed data points into a metric space of specified dimensionality (in this case, 2D), while preserving as much as possible the distances between the points (see the appendix for details and references on MDS). We found that the residual stress resulting from embedding the 13-dimensional PCA subspace into 2D was 0.25 (the stress vanishes for point configurations that can be embedded into the target space without distortion, and approaches 1 for poor embeddings). Furthermore, the 2D configuration of the data points derived by MDS from the inter-point distances in the 13 PCA dimensions (when these were forced into a 2D space) bore no resemblance to the 3×6 parametric grid pattern built into the stimuli. Thus, we conclude that the LDR extraction problem in the present case is indeed highly nonlinear.

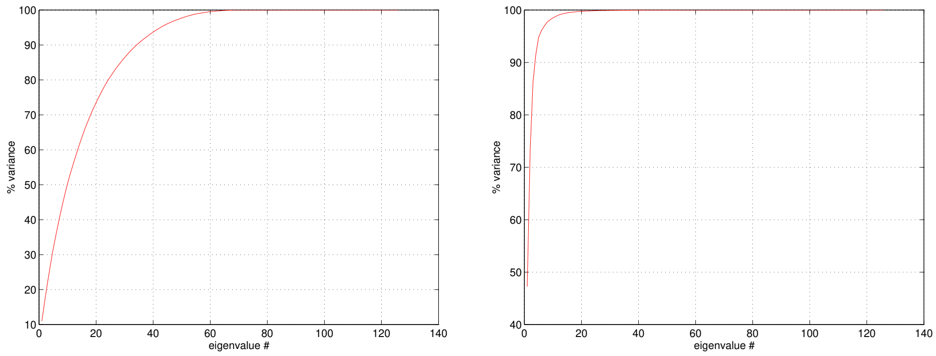


Figure 3. Cumulative percentage of variance plotted vs. the number of participating eigenvector projections, as obtained by principal component analysis (PCA) on the Fractals data set (left) and of the Faces data set (right).

2.2. Faces

We chose human faces as the basis for our second data set, mainly to facilitate intuitive understanding of the computational experiments and their expected results. Of all natural objects, faces are the category with which human observers are the most proficient; we surmised that the orthogonal manipulations involved in the generation of the two between-classes and one within-class dimensions of image variations would be most easily perceived if the images were those of faces. Because, unlike fractal patterns generated by a known algorithm, face images possess no inherent low-dimensional parameterization, we had to impose such a parameterization on the data. We chose to do it by starting with a set of nine 3D laser scans of human heads (see Figure 4), and by embedding the 3×6 grid in the 2D space spanned by the two leading "eigenheads" obtained from the data by PCA.†

Each of the 18 heads derived by PCA from the original scanned head data was piped through a graphics program, which rendered the head from seven viewpoints, obtained by stepping the (simulated) camera in $3°$ rotation steps around the midsagittal axis. The rendering program assumed a semi-glossy reflectance model for the head surface; because of that, and because of the trigonometric functions involved in the viewpoint transformation, we expected a nonlinear relationship between the planar 2D configuration formed by the 18 heads in the parameter space and the space spanned by the rendered images of these heads. As with the Fractals data set, we quantified the degree of nonlinearity by subjecting the image data to PCA and to MDS analysis. For the Faces data set, the nonlinearity was somewhat more moderate than for Fractals, but still considerable: projection onto the first two and the first 13 eigenfaces accounted for 73.1% and 99.2% of the variance, respectively; the subspace spanned by the first 13 eigenfaces was still significantly non-planar, as indicated by the residual stress of MDS, which was equal to 0.16.

† A similar approach to the generation of parametrically controlled head stimuli has been recently proposed in (Atick et al., 1996).

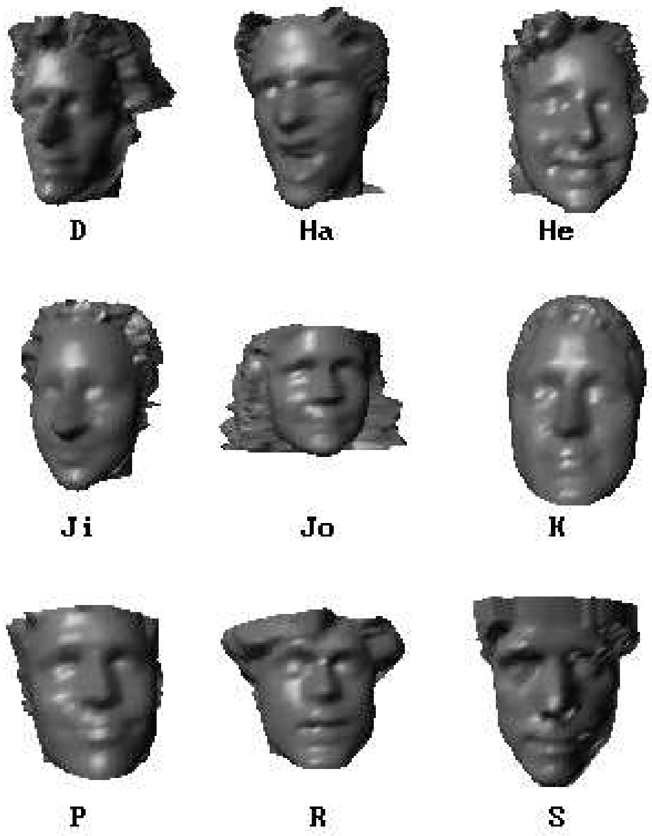


Figure 4. The nine 3D laser scans of human heads, used to generate the face images shown in Figure 5. Three of these are distributed with SGI systems, and the other six are available via anonymous `ftp`, courtesy of Cyberware Inc., as a part of their demonstration software.

2.3. Preprocessing

The images generated by the two methods described above were imported into Matlab^{TM}, and were preprocessed prior to LDR extraction. The Fractals data were subjected to histogram equalization, then convolved with a bank of $28 \times 28 = 784$ receptive fields (Matlab Image Processing toolbox; Laplacian of Gaussian, kernel size 9, $\sigma = 0.6$). Images in the Faces data set, originally of size 400×400, were reduced to 49×16 dimensions by cropping the background and by correlation with a bank of filters (Matlab Image Processing toolbox; Laplacian of Gaussian, kernel size 11, $\sigma = 0.9$). In both cases, the preprocessing reduced the dimensionality of the data from 65536 to 784, and served as a crude approximation of the transformations that a stimulus undergoes on its way to the primary cortical visual area in the mammalian brain.

3. Methods

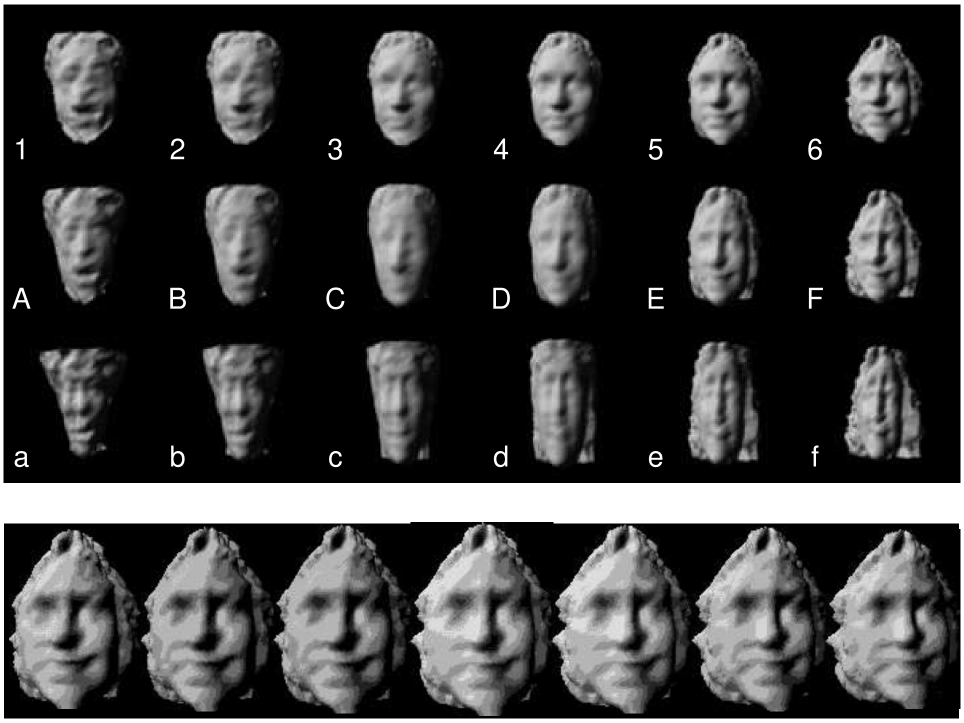


Figure 5. Some of the images from the Faces data set (see section 2.2). *Top:* the 18 heads obtained by placing a 3×6 grid in the space of the two leading principal components of the original nine heads. *Bottom:* the 7 views of the rightmost head in the top row above; the views differ by $3°$ steps of rotation in depth, summing up to a total difference of $18°$. It should be noted that orientation differences of up to $20°$ go unnoticed by human viewers (Busey et al., 1990), while presenting nontrivial problems for neural networks, which must be trained explicitly to compensate for or to tolerate the misorientation. Prior to classification, the images, originally of size 400×400, were reduced to 784 dimensions by cropping the background and by correlation with a 49×16 bank of Gaussian-profile filters (section 2.3).

All our computational experiments consisted of two phases: training the LDR extractor, and evaluating its performance (see Figure 6). We now describe these two procedures in details (the results of the experiments are reported in the next section).

3.1. Derivation of the LDR

As we argued in the introduction, a faithful low-dimensional representation of the data space should serve as a good basis for efficient classification of the data; the aim of the experiments we describe here was to determine whether a classifier, faced solely with the task of learning to label a selection of data points, is likely to do so by deriving an LDR that would help it in the classification process. Note that the extraction of an LDR was made possible, at least in principle, by the design of the data sets, into which we built a low-dimensional structure. At the same time, the embedding of this

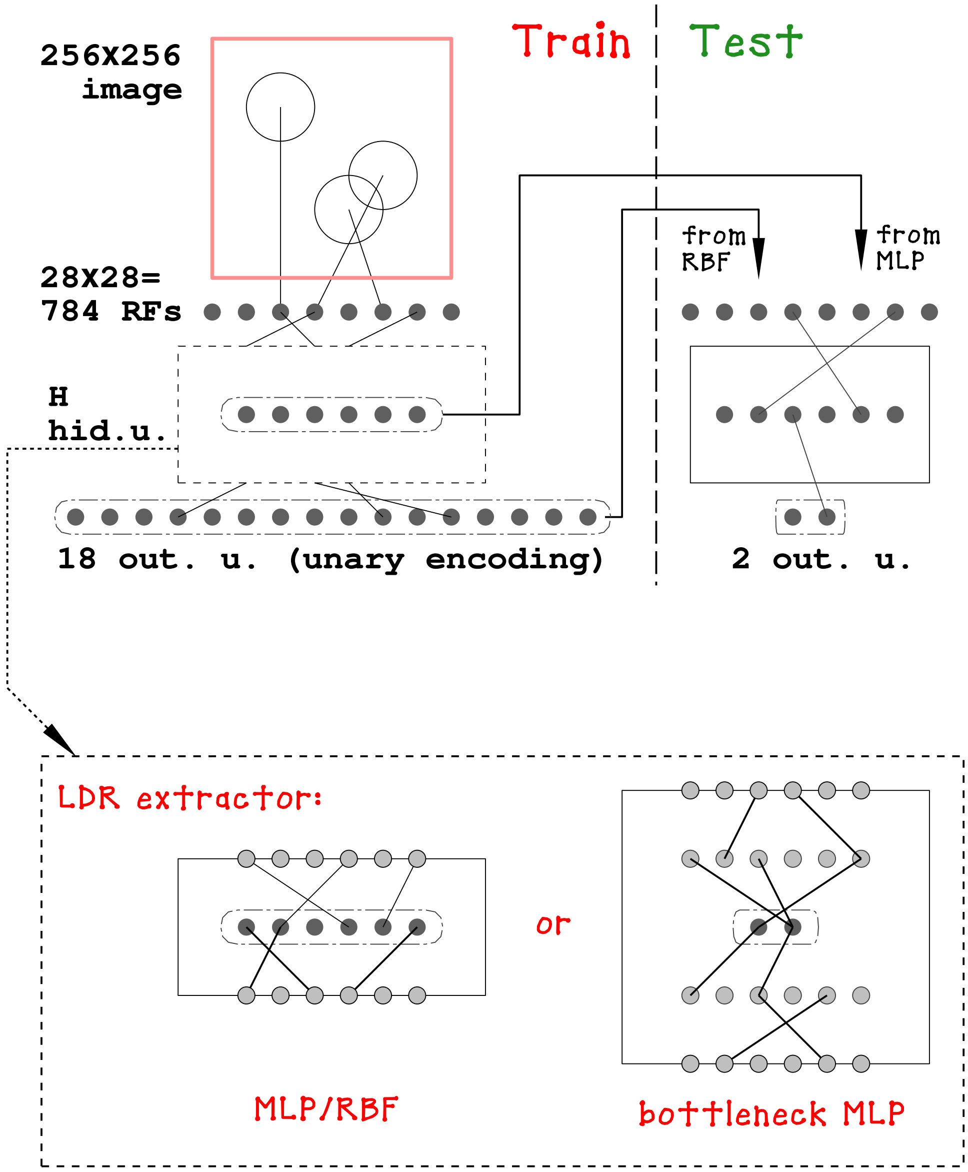


Figure 6. The low-dimensional representation (LDR) extraction scheme (see section 3). The LDR extraction network appears on the left, under the label Train. Following preprocessing (see section 2.3), a learning module — a multi-layer (here, 3-layer) perceptron (MLP), a 5-layer "bottleneck" MLP or a radial basis function interpolator (RBF) — is trained to produce a unary (1 out of 18) encoding of the class labels associated with the input image. The testing procedure (illustrated on the right) is described in detail in the text.

structure in a high-dimensional image space, which, moreover, was nonlinearly related to the original parametric representation of the data, made the extraction of LDR a highly nontrivial task.

3.1.1. 1-HL MLP classifier Researchers have explored in the past the ability of multilayer perceptrons, trained as autoencoders, to form low-dimensional representations of the data in the hidden layer (DeMers and Cottrell, 1993). In the autoencoder approach, the data are forced to pass through a low-dimensional "bottleneck" — the hidden layer of an MLP. The network is taught to reconstruct the input patterns from the LDR formed at the hidden layer. A major disadvantage of this approach is that it is inherently oblivious to any regularities that may be present in the data.† For example, face images depend not only on the identity of the person, but also (sometimes in an overwhelming fashion) on the viewing conditions; a resourceful LDR extractor attempting to recover a low-dimensional "face space" from a batch of face images should take this into account and discount the "irrelevant" dimensions (in this example, variations due to viewpoint and illumination), while preserving the relevant ones.

One way to bias the LDR extractor to do that is by training it as a classifier, rather than as an autoencoder, and, in particular, to introduce dimensions to which the LDR should be orthogonal. Specifically, we propose to look for the LDR in the hidden layer of an MLP trained to classify the input images into a relatively large number of classes (striving for adequate coverage of the postulated low-dimensional pattern space), while ignoring within-class variation due to extraneous factors. In the present case, the data sets were broken into 18 distinct classes, placed on a grid in two parametric dimensions; the exemplars within each class differed along a third dimension, which the MLP had to learn to ignore (in the Faces data set, this was the face orientation dimension).

3.1.2. 3-HL bottleneck MLP classifier Because the regular three-layer (one hidden layer, 1-HL) MLP, if trained as an autoencoder, is known to carry out an approximate PCA of the data, (Cottrell et al., 1987; Baldi and Hornik, 1989; Oja, 1989), we could not expect it to perform too well on our nonlinear data sets. A natural modification of the 1-HL MLP architecture, which may be better suited to nonlinear LDR extraction, is the three hidden layer (3-HL) MLP (Leen and Kambhatla, 1994). We have, therefore, chosen to explore this approach, but, following the considerations stated above, the 3-HL MLP was to be trained as a classifier, and not as an autoencoder.

The results obtained from the 3-hidden layer experiments further indicate that it is the specific training task (which included a specific generalization subtask) that led to a useful LDR, while a similar and even more powerful architecture without the generalization task was unable to recover a useful LDR.

† Autoencoder training is also especially costly, because of the large size of the required network in image processing applications.

3.1.3. RBF classifiers The third LDR extraction method that we have chosen to examine is projection into a space spanned by the responses of a number of "prototype detectors" (Edelman, 1995b; Edelman et al., 1996). Each detector, which can be realized as a radial basis function (RBF) classifier, is trained to output a constant value for a set of instances of a given class (e.g., images of a given face taken from a series of viewpoints). If the RBF module successfully generalizes to other instances of the same class by maintaining a relatively unperturbed response level for those inputs, and if its response drops off monotonically with increasing distance between the input and the optimal stimulus (the prototype for that module), a collection of modules tuned to different classes forms a distributed representation of the input that is likely to capture the low-dimensional structure of the input space (Edelman and Duvdevani-Bar, 1997a).

For the present purpose, we trained a single RBF network to output a unary representation of the class membership, as we did with the MLP classifiers described above (this is equivalent, of course, to training 18 separate RBFs, sharing the same "hidden" or basis function layer). Training was confined to the computation of the optimal hidden to output weights (the basis functions being centered on a subset of the input examples), and could be carried out, therefore, by simple pseudoinverse (Poggio and Girosi, 1990). As a result, this LDR extraction method was much faster than the two methods involving MLPs (which were trained by back-propagation). As we shall see, this rapid LDR extraction produced representations nearly as good as those obtained by MLP.

3.2. Evaluation of the LDR

An LDR could only be passed on to the evaluation stage if the LDR extractor exhibited good generalization to previously unseen members of each of the 18 classes. Recall that the data sets included seven instances of each class; of these, four were used in training, and the remaining three were set aside for assessing generalization.†

If the generalization in a given trial was satisfactory (as measured by the error rate, which was required not to exceed 0.1 or 0.2, depending on the difficulty of the task), the LDR was evaluated using three different methods, as described below. Note that the data at this point consisted of 54 vectors (18 classes times 3 exemplars), of dimensionality that depended on the LDR extraction method, and varied between 2 (for the 3-HL bottleneck MLP classifier) and 13 (for the RBF classifier).

3.2.1. RBF performance in the dichotomy task The first method was designed to assess the utility of the LDR for supporting a representative classification task not directly related to the one-out-of-18 classification used to train the LDR extractor. For that purpose, the recovered LDR was used to train an RBF classifier on a nonlinear two-class (dichotomy) problem (see Figure 1, right). The generalization performance of this classifier was then compared with that of an identical classifier

† In some experiments, the four training inputs were situated contiguously along the third dimension of the parameter space; in others, the training and the testing inputs were intercalated.

trained on the raw 784-dimensional filter-space representation of the image set, on the same dichotomy problem.

3.2.2. MDS The second method was designed to allow visualization of the configuration formed by the 18 classes in the LDR space. Because in general this space had more than two dimensions (except in the case of the 3-HL bottleneck MLP), the points had to be embedded in 2D for easy plotting and visualization. This embedding was carried out by multidimensional scaling (MDS), which accepted a 54×54 table of mean inter-point distances and produced an embedding of the 54 points (each corresponding to a single test instance) into two dimensions. The mean-distance table was computed by element-wise averaging of a number of tables arising from repeated trials; the LDR extraction experiment was run repeatedly to reduce the dependence of the results on the randomized initial conditions, to which MLP training is known to be sensitive.

3.2.3. 3-HL bottleneck MLP As mentioned above, the LDR derived by the 3-HL bottleneck MLP with two units in the middle hidden layer is two-dimensional, and thus can be visualized directly, without the mediation of MDS. Note that in this case the results of repeated runs cannot be simply averaged (unless they are first converted into the distance-table format, which would necessitate subsequent application of MDS). To combine the results of a number of runs we used Procrustes transformations (Borg and Lingoes, 1987) to normalize (scale, rotate, and translate) all the LDR configurations to the configuration obtained in the first run.† As a direct control of the outcome of MDS-based visualization, we also ran the 3-HL bottleneck MLP on the LDRs obtained by other methods, as a kind of post-processing. In the next session, the LDR configurations derived by the different methods are plotted alongside each other, where appropriate.

4. Results

We now proceed to describe the results of the computational experiments in dimensionality reduction that we performed on the Fractals and the Faces data sets. The same experiments were carried out in both cases; the reported results include

(i) Evaluation of the three methods of LDR extraction listed in section 3.1;
(ii) A control comparison between the LDR obtained with 18-class training with that obtained with 2-class training;
(iii) A plot of the 2D configuration derived from the raw 784-dimensional data by MDS.

We also performed some additional experiments with the Fractals data set, which explored the effect of imposing categorical structure on the LDR, and are reported

† Procrustes, or similarity, transformations, by definition, leave the shape of a set of points unchanged — a property that is useful for comparing a number of configurations that have similar shapes, but may differ in size, orientation, and location relative to a coordinate system.

in the next section. Information regarding other experiments with the Fractals data (namely, a study of the effect of the number of bases on the quality of the LDR extracted by an RBF network, and a comparison of LDRs derived from several versions of Fractals data of varying difficulty) can be found in (Intrator and Edelman, 1996).

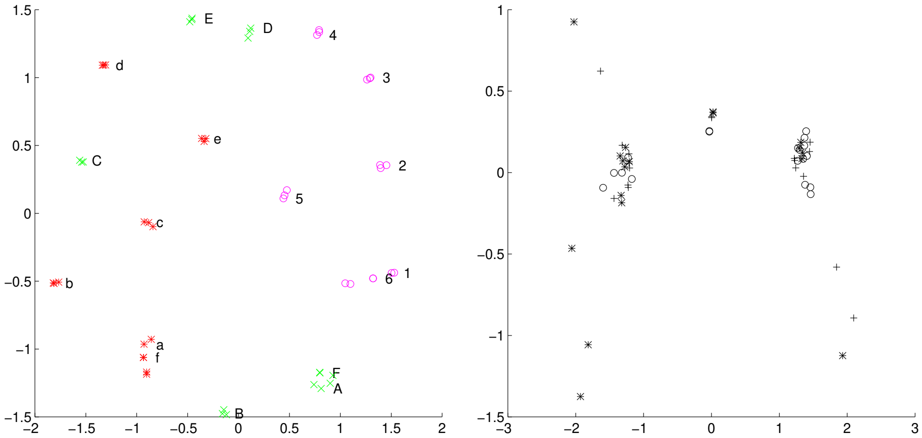


Figure 7. Fractals data, LDR by 1-HL MLP with 5 hidden units (section 3.1.1), MDS visualization (section 3.2.2). *Left:* MLP trained on 18 classes; here and in the subsequent plots, the three symbol kinds ($\circ$, $+$, and $*$) correspond to points belonging to the different rows of Figure 1. Note the good separation of the 54 points into 18 classes; the 3 points in each class (corresponding to the 3 test images per class) are usually clustered together. *Right:* MLP trained on a dichotomy; 2 rather than 18 clusters are apparent. The test dichotomy classification error (section 3.2.1) was typically about 0.05 in both cases, compared to about 0.3 on the raw data.

4.1. The Fractals *data set*

The LDR configuration obtained on the Fractals data by the 1-HL MLP method is shown in Figure 7, left. The first striking feature of this configuration is its mandala-like general structure: the MLP network did a good job of spreading the 18 classes as far apart as possible from each other, while preserving the grouping of the triplets of points corresponding to different exemplars of the same class. The second notable feature emerges from a scrutiny of the relative locations of the classes belonging to the same 6-class row (see Figure 1, left). Each of the three rows is marked by a different symbol ($\circ$, $+$, and $*$); the six clusters in a row are labeled consecutively (`1` – `6`, `A` – `F`, `a` – `f`). Note that each row curves upon itself; e.g., clusters 2, 3, and 4 are progressively farther away from cluster 1, while clusters 5 and 6 are progressively closer to cluster 1. An intuitive explanation of this pattern may lie in the nonmonotonic (cyclical) dependence of the appearance of fractal images produced by the `quatjul` procedure under progressively larger parametric change on the parameter-space distance to some reference image.

Figure 7, right, shows the configuration derived by a 1-HL MLP trained on a dichotomy. Even though the resulting LDR is about as good for supporting this

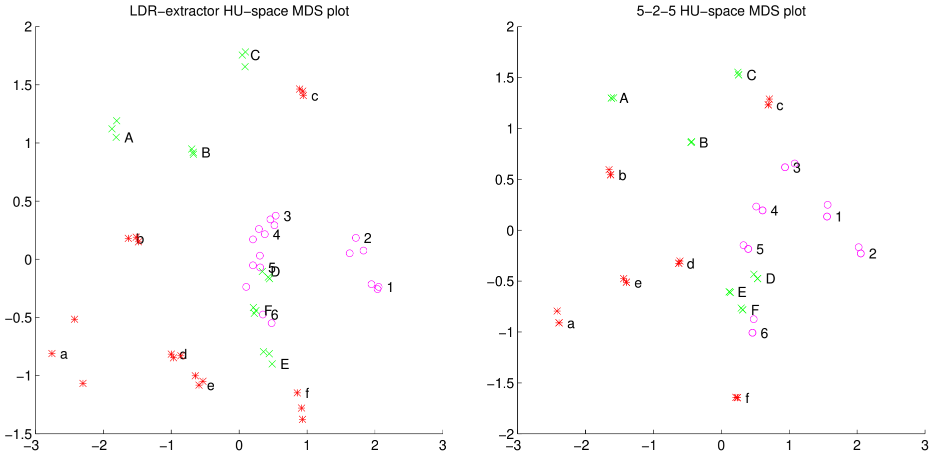


Figure 8. Fractals data, LDR by 1-HL MLP with 5 hidden units. *Left:* MDS visualization. *Right:* 3-HL bottleneck MLP visualization. The two visualization methods yield similar configurations, although they rely on entirely different embedding algorithms. This adds credibility to the use of distance (or distance rank) preserving methods for embedding data in points in 2D for the purpose of visualization (cf. (Sammon, 1969; Siedlecki et al., 1988)).

dichotomy as the one obtained following 18-class training, it is much worse as far as the faithful representation of the original parameter space is concerned: the 18-cluster structure is lost in this LDR.

The performance of the 3-HL bottleneck MLP in LDR recovery is illustrated in Figure 9. Note that in this case the LDR can be read off the middle hidden layer of the MLP (which contained two hidden units) and plotted directly, without post-processing by multidimensional scaling.

The recovery of LDR by an RBF classifier (the third method we explored) is illustrated in Figure 10. Note that the mandala-like structure in this plot prevails over the preservation of the within-row order of clusters (compare with Figure 7). Thus, the representations derived by MLP-based methods are more faithful than those obtained by RBFs to the true low-dimensional parametric variation built into the data.

We next illustrate the ability of the MLP-based LDR extraction method to assimilate hierarchical category knowledge in a natural manner. In the first experiment that examined this issue, three higher-level class labels were added to the set of 18 labels normally used in the training stage. For each data point, the higher-level label indicated the row to which it belonged (see Figure 1). In the resulting configuration, the 18 clusters were separated, on a coarser level, into three groups, corresponding to the three higher-level class labels (see Figure 11, left). In the second experiment, the LDR extractor was taught three labels corresponding to the rows and six labels corresponding to the columns of the parameter-space configuration. The resulting configuration depended to a significant degree on the relative weights given to the row and column labels. Under nearly equal weights, the points were separated into three clusters by the row label (see Figure 11, middle); when the column

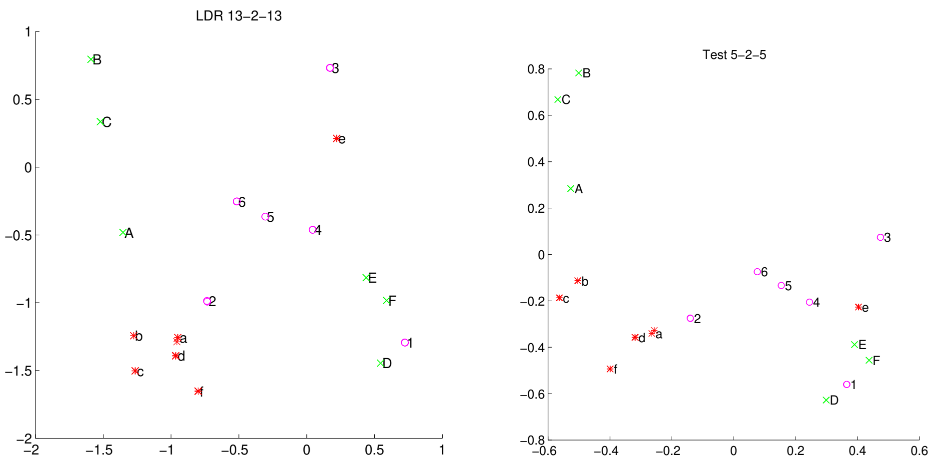


Figure 9. Fractals data, LDR by 3-HL bottleneck MLP (section 3.1.2), Procrustes visualization (section 3.2.3). *Left:* results for 3-HL bottleneck MLP with 2 hidden units, trained with class labels on the filter data. The test dichotomy error rate was 0.03, compared to 0.41 on the raw data. *Right:* results for 3-HL bottleneck MLP, trained as an autoencoder on the LDR derived from the middle HL of the previous 3-HL MLP.

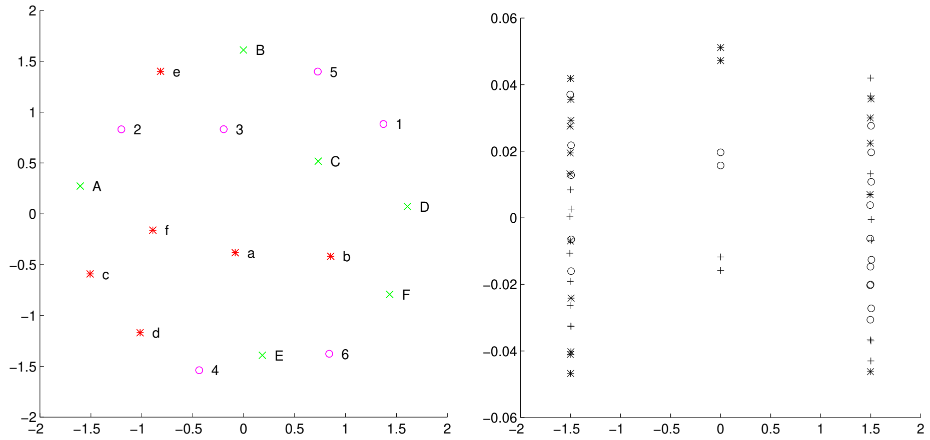


Figure 10. Fractals data, LDR by RBF (section 3.1.3), MDS visualization. *Left:* results for a 72-center RBF; test dichotomy error 0.0, compared to 0.24 on the raw data; *Right:* control results obtained by a 72-center RBF trained on a dichotomy; test dichotomy errors as above. A parametric study of the performance of this method on the number of centers of the RBF network can be found in (Intrator and Edelman, 1996).

weight predominated, the separation was into six clusters (i.e., by column), with some additional structure within each cluster (see Figure 11, right).

A natural extrapolation of this strategy would be to teach the network many

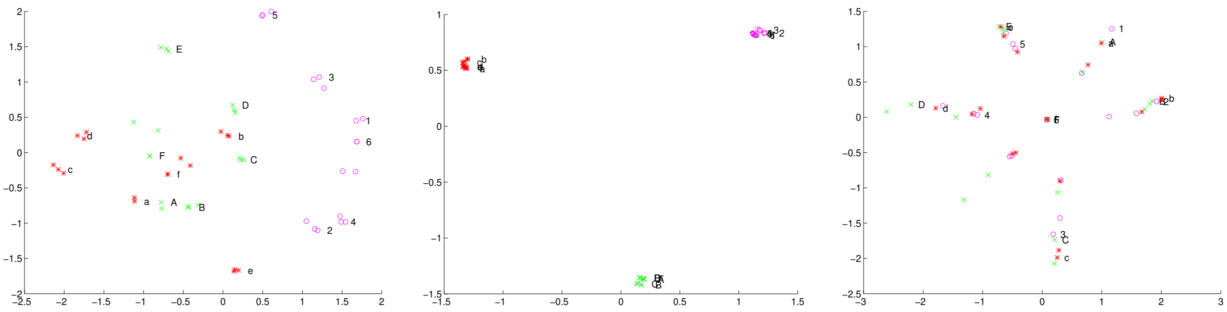


Figure 11. Fractals data, LDR by 1-HL MLP, MDS visualization. This figure illustrates the incorporation of prior hierarchical category knowledge into the LDR extraction process. *Left:* LDR derived by a 1-HL MLP with 5 hidden units, trained to produce a unary encoding of the 18-way label set, to which a coarser set of 3-way class labels, corresponding to the row number in Figure 1, has been appended (also in a unary format). The row variables were given a weight of $w_c = 0.1$ relative to the identity variables. The test dichotomy error rate on the resulting LDR was 0.056. *Middle:* results for an MLP trained to produce a unary encoding of the 3 row and the 6 column numbers of the stimulus. The test dichotomy error rate on the resulting LDR was 0.056. The relative weights of the row and the column variables were $w_r = 1.0$, $w_c = 0.75$. *Right:* same 3×6 class structure as before, but the relative weights of the row and the column variables were $w_r = 0.05$, $w_c = 1.0$. The test dichotomy error rate on the resulting LDR was 0.20. The test dichotomy error rate on the raw data in all three cases was 0.28.

possible dichotomies, in the hope that the structure of the underlying LDR can be recovered from the multiple two-way classifications (Price et al., 1995). The advantage of operating at the level of 18 classes (or of three classes, with six subclasses each) is in the much shorter training procedure. On the other hand, training on multiple dichotomies may have the advantage of forcing the LDR extractor to consider multiple, hopefully disjoint, sets of features relevant to the collection of tasks, and not letting it zero in on distinctive features specific for each one-vs-all discrimination. We leave for future research the quest for an optimal compromise between these considerations.

4.2. The Faces *data set*

LDR extraction from the Faces data set was easier than from the Fractals, as could be expected from the comparison of the degree of nonlinearity of the two sets, described in section 2. This expectation was supported by another comparison — between the configurations derived by MDS from the raw 784-dimensional Fractals and Faces data (see Figure 12) — and was confirmed by the results of the experiments involving the Faces data set, which we describe next.

Good results were obtained on the Faces data using all three methods for LDR extraction: 1-HL MLP, 3-HL bottleneck MLP, and RBF. The performance of the MLP-based methods in recovering the topology of the row/column parametric structure of

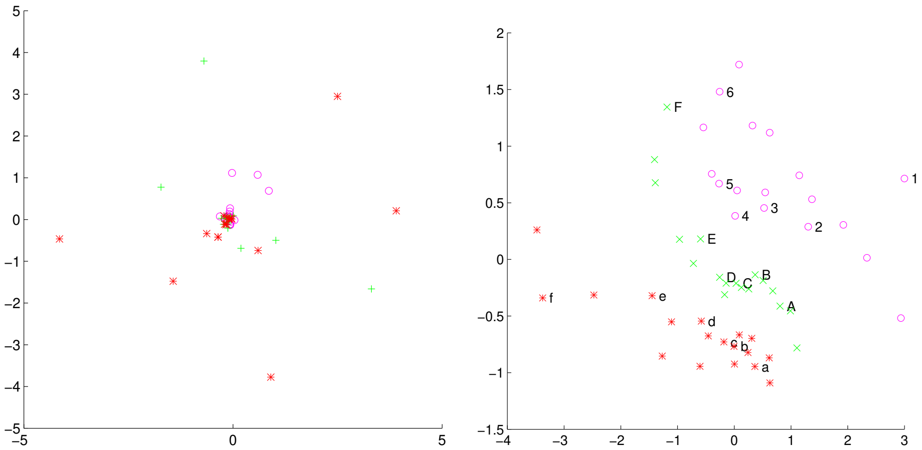


Figure 12. *Left:* Fractals data; the pattern recovered by MDS from the raw 784-dimensional data set. No separation of the 18 classes is apparent; most of the points are concentrated in the middle of the plot (compare with Figure 7, left). *Right:* Faces data; the pattern recovered by MDS from the raw 784-dimensional data set. Although the general layout of the 18 classes relative to one another is preserved, the points belonging to different classes are not separated (compare with Figure 4, left). Indeed, MDS by itself had no reason to separate the classes, unlike the MLP, which has been explicitly trained to do so (but not necessarily to preserve the 2D metric layout of the classes relative to each other, which it did, just like MDS!).

the 18 classes seems to be especially amazing (compare, e.g., Figure 13, left, with the labels in Figure 5, top). Importantly, this recovery was possible even when the network was trained on half of the 18 classes, then tested on the full data set (see Figure 16). The implications of this and the other results are discussed in section 6.

5. Control experiments

The difficulty of LDR extraction in the present case is demonstrated by a comparison to the results obtained by more conventional neural network methods for dimensionality reduction. The best-known such methods employ self-supervised bottleneck autoencoder training. While the imposition of a low-dimensional bottleneck is common to these methods and to our approach, there is a crucial difference: an autoencoder is trained to reproduce the data, while our networks are trained to assign the data a certain category structure. To characterize the importance of this feature of our approach to the extraction of useful LDRs, we conducted several experiments.

First, we asked whether a self-supervised 3-layer MLP autoencoder, which aims at the best reconstruction of the inputs, can reveal the correct low-dimensional structure in our data. Although in the linear case such networks do quite well, essentially by extracting the principal components of the data (Elman and Zipser, 1988), the performance on the Faces data was poor. Specifically, the autoencoder network consistently converged to the mean of the data, presumably due to the nonlinearity

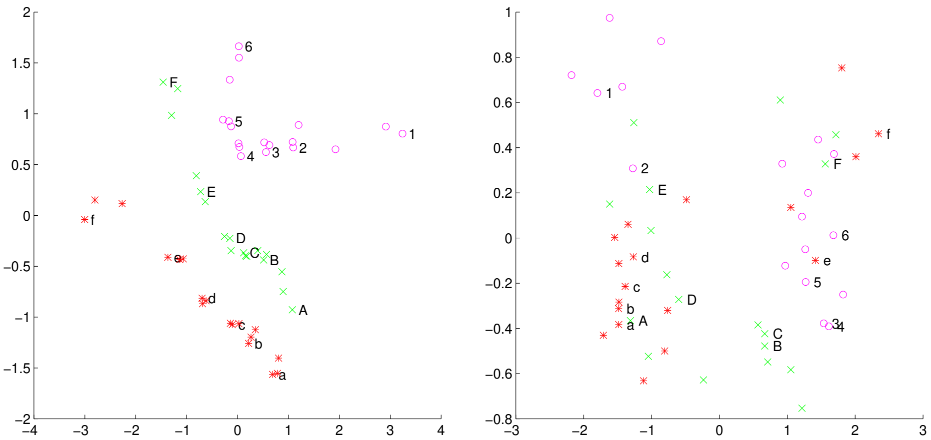


Figure 13. Faces data, MDS visualization (section 3.2.2). *Left:* LDR by 1-HL MLP with 13 hidden units (section 3.1.1), trained for 20,000 epochs on the 18-class task. The test dichotomy error rate was 0.02, compared to 0.07 on the raw data. *Right:* control; results for a 1-HL MLP with 13 hidden units, trained for 20,000 epochs on a dichotomy task. The test dichotomy error rate was 0.04, compared to 0.07 on the raw data.

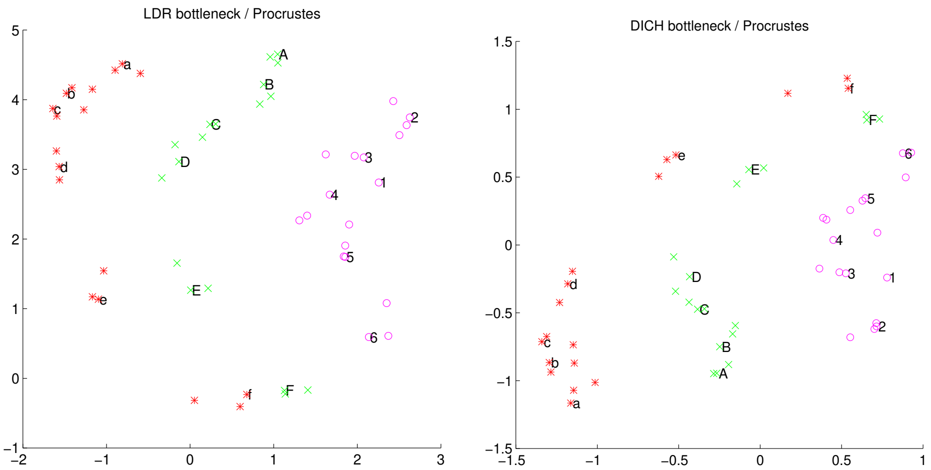


Figure 14. Faces data, LDR by 3-HL bottleneck MLP, MDS visualization. *Left:* results for 3-HL bottleneck MLP with 2 hidden units, trained on the 18-class task. The test dichotomy error rate was 0.1, compared to 0.29 on the raw data. *Right:* results for 3-HL bottleneck MLP, trained as an autoencoder on the LDR derived from the middle HL of the previous 3-HL MLP.

introduced by the imaging step.

Second, we experimented with a 5-layer nonlinear bottleneck autoencoder (Leen and Kambhatla, 1994). This training scheme, likewise, performed poorly on our data set. The outcome of this experiment showed that self-supervised dimensionality reduction

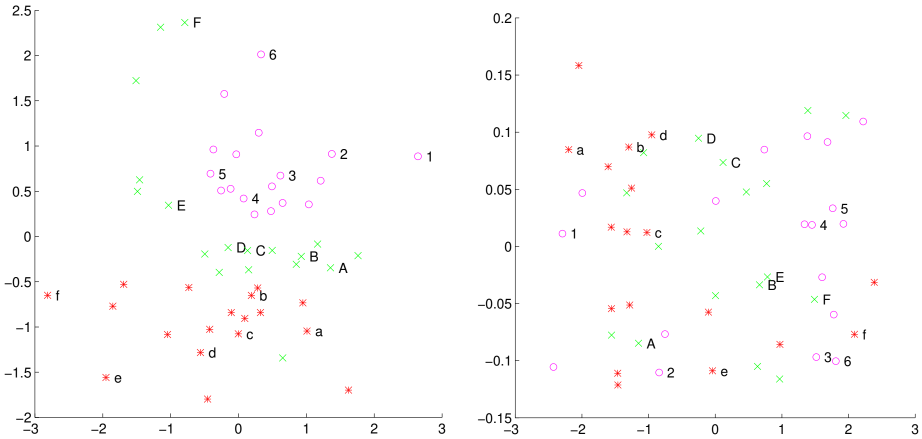


Figure 15. Faces data, MDS visualization. *Left:* an 18-center RBF (section 3.1.3), trained on the 18-class task. The test dichotomy error rate was 0.04, compared to 0.07 on the raw data. Recall that each point corresponds to one test view; the three views belonging to each of the 18 test faces are usually grouped together. The labels should be compared with those in Figure 4. *Right:* an 18-center RBF, trained on the dichotomy. The test dichotomy error rate was 0.11, worse than the error rate of 0.07 on the raw data.

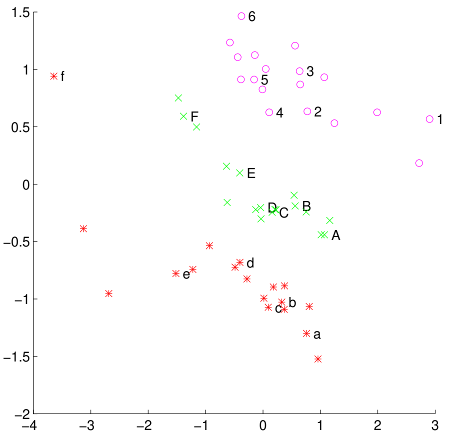


Figure 16. Faces data set, LDR by a 1-HL multilayer perceptron (MLP), trained on every second class; MDS visualization. These results were obtained with a 1-HL MLP network with nine hidden units, trained on half of the 18 classes that comprise the problem space (the nine classes used for training and the nine omitted classes formed a checkerboard pattern). Note that all 18 classes – both familiar ones and those not seen by the system – are in a topology-preserving formation. The test dichotomy error rate was 0.14, compared to 0.28 on the raw data.

cannot recover a good LDR in the present case, illustrating the importance of guidance provided by the class labels.

A bottleneck autoencoder employed as a dimensionality reducing device is required to map a high-dimensional space (in our case, a space of 784 dimensions) to itself, whereas the output space in our training scheme was 18-dimensional. One may ask here whether a certain reduction in the dimensionality of the output space, combined with an imposition of certain category structure on that space, would enable the network to learn the proper LDR. To address this question, we tested a modified version of our method, in which the classifier was not trained to ignore the direction orthogonal to the target manifold (cf. Figure 17; this was done by training on the 72 face-view labels, instead of the 18 face identity labels). Thus, the network's output space was 72-dimensional, with each dimension corresponding to a conjunction of face identity and face orientation labels. This manipulation, however, did not help: the LDR extracted by the modified autoencoder was poor, underscoring the importance of guidance provided by an explicit specification of the dimension to be collapsed (in this case, the dimension of the viewpoint-related variation).

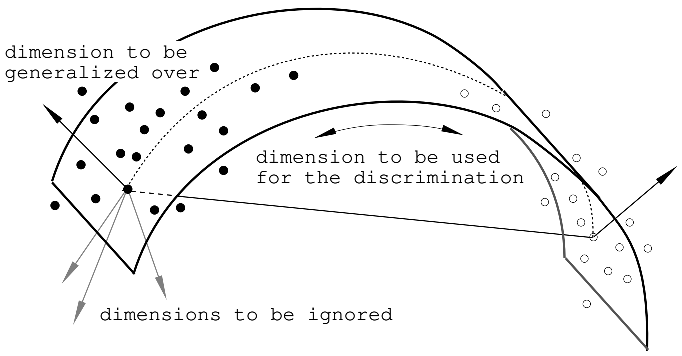


Figure 17. A schematic illustration of a problem space whose efficient representation requires nonlinear dimensionality reduction. The instances of the two classes cling to a low-dimensional manifold, embedded in a measurement space, whose dimensionality may run in the tens of thousands. See section 6 for a discussion of this example.

6. Conclusions

We have shown that combining multiple constraints via a multiple use of class labels is an effective way to impose bias on a learning system whose goal is to find a good LDR. In particular, the use of multiple class labels steers the system to become *invariant* to those directions of variation in the input space that play no role in the classification tasks. This is done merely by using class labels that are independent of these directions: in the Fractal images, the image labels are invariant to the third dimension of variation (Figure 1, left); similarly, the identity labels of the Faces are independent of their orientation (Figure 5, bottom panel). We have also shown that prior knowledge of the "proper" dimensionality of the target LDR can be imposed by training a multi-layer bottleneck network (Figure 6, bottom right). Both these features of our approach lead to improved generalization in the learned tasks.

A useful intuition concerning the effectiveness of our method can be developed by considering an analogy with discriminant analysis, a well-known technique for projecting data onto dimensions important for a particular classification task (see Figure 17). Assume that some dimensions of a set of measurements performed on the world are crucial for distinguishing between the categories, while other dimensions must be downplayed, or collapsed; in the context of object recognition, the former may be the dimensions of object identity, and the latter – of object orientation. Whereas standard discriminant analysis methods in multidimensional spaces are plagued by the presence of irrelevant dimensions, in this paper we showed that training with a combined objective of (1) discrimination among labeled categories known to reside within the manifold, and (2) explicit collapse of dimensions over which discrimination is to be generalized, leads to an improved performance in other discrimination tasks involving the same objects, and to a reliable recovery of a low-dimensional manifold containing the objects, even when it is curved (i.e., when the problem is nonlinear), and is embedded in a measurement space of nearly a thousand dimensions.

An important feature of the LDR computed by our method is the preservation of the topology of the parameter space underlying the data. The clusters corresponding to the labels we imposed during training were both separated from each other (as dictated by the training procedure), **and** arranged in the resulting low-dimensional space in a pattern that reflected their arrangement in the parameter space used to create the data. This latter property of the representation was obtained even when the network had been trained only on half of the objects over which topology preservation was evaluated.

Topology preservation is useful because it allows the representational system to categorize novel instances of familiar object classes, as well as make sense of novel classes (Edelman and Duvdevani-Bar, 1997a). Specifically, if proximities in the representation space reflect similarities among objects "out there" in the world, the following two operations can be carried out safely: (1) a new instance of a class can be categorized by finding, in the representation space, the cluster to which the current stimulus is the closest; (2) a new class can be defined by its representation-space distances to the familiar classes (Edelman and Duvdevani-Bar, 1997b).†

We remark that topology preservation appear to be true of representation formed by human subjects in a variety of perceptual tasks. Studies in experimental psychology indicate that a low-dimensional pattern built into complex 2D shapes (by arranging these shapes in a conspicuous configuration in an underlying parameter space) is recovered by the visual system of subjects required to judge similarities between the shapes (Shepard and Cermak, 1973; Cortese and Dyre, 1996). Recently, similar findings have been achieved in experiments that involved 3D objects, arranged in a variety of planar configurations in a parameter space of several dozen dimensions (Edelman, 1995a; Cutzu and Edelman, 1996). The upshot of these findings is that the human visual system is capable of recovering the proper low-dimensional representation

† In the context of Figure 16, this operation corresponds to the definition of a novel face (say, face `B`, which was not included in the training set) in terms of its similarities to familiar faces (e.g., defining `B` as the stimulus that is halfway between `A` and `C`, as well as between `2` and `b`).

of the stimuli from a million-dimensional measurement space (dictated by the number of axons leading from the retina to the brain), while preserving the topology of the original space (and in many cases the exact relative placement of the stimuli in that space). The conditions on the LDR extraction process that makes such recovery possible, and the wider philosophical implications of this phenomenon, are discussed in (Edelman, 1997).

Acknowledgments

We thank P. Dayan and J. Tenenbaum for useful suggestions.

Appendix: Multidimensional scaling (MDS)

MDS has been originally developed in psychometrics, as a method for the recovery of the coordinates of a set of points from measurements of the pairwise distances between those points (Young and Householder, 1938). In a typical application, the experimenter would attempt to characterize a subject's performance by placing a point corresponding to each stimulus perceived by the subject in a coordinate space, derived from subjective similarity ratings of pairs of stimuli. The power of MDS as a tool for the study of internal representations (of human subjects) was revealed when Shepard discovered in 1962 that fixing the *relative* distances of a set of points effectively determines their coordinates (Shepard, 1966). This discovery led to the development of the nonmetric MDS algorithm (Kruskal, 1964), which employs gradient descent to seek a monotonic transformation between measured distances and distances computed from the hypothesized point configuration, which would minimize stress (defined as the discrepancy between the ranks of the measured and the computed distances). In the present work, we used a modern implementation of nonmetric MDS, available in version 6 of the SAS statistical analysis software (Sas, 1989), to allow the visualization of high-dimensional data sets. The points in a given set were embedded into a 2D metric space reflecting as closely as possible the pattern of inter-point distances, then plotted and subjected to inspection.

References

Atick, J. J., Griffin, P. A., and Redlich, A. N. (1996). The vocabulary of shape: principal shapes for probing perception and neural response. *Network*, 7:1–5.

Baldi, P. and Hornik, K. (1989). Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2:53–58.

Baxt, W. G. and White, H. (1995). Bootstrapping confidence intervals for clinical input variable effects in network trained to identify the presence of acute myocardial infraction. *Neural Computation*, 7(3):624–638.

Bellman, R. E. (1961). *Adaptive Control Processes.* Princeton University Press, Princeton, NJ.

Borg, I. and Lingoes, J. (1987). *Multidimensional Similarity Structure Analysis.* Springer, Berlin.

Breiman, L. (1992). Stacked regression. Technical Report TR-367, Department of Statistics, University of California, Berkeley.

Breiman, L. (1994). Bagging predictors. Technical Report TR-421, Department of Statistics, University of California, Berkeley.

Buckheit, J. and Donoho, D. L. (1995). Improved linear discrimination using time-frequency dictionaries. Stanford university technical report.

Busey, T. A., Brady, N. P., and Cutting, J. E. (1990). Compensation is unnecessary for the perception of faces in slanted pictures. *Perception and Psychophysics*, 48:1–11.

Caruana, R. (1993). Multitask connectionist learning. In *Proceedings of the 1993 Connectionist Models Summer School*, pages 372–379, San Mateo, CA.

Caruana, R. (1995). Learning many related tasks at the same time with backpropagation. In Tesauro, G., Touretzky, D., and Leen, T., editors, *Advances in Neural Information Processing Systems*, volume 7, pages 657–664. Morgan Kaufmann, San Mateo, CA.

Cortese, J. M. and Dyre, B. P. (1996). Perceptual similarity of shapes generated from Fourier Descriptors. *Journal of Experimental Psychology: Human Perception and Performance*, 22:133–143.

Cottrell, G. W., Munro, P., and Zipser, D. (1987). Learning internal representations from gray-scale images: An example of extensional programming. In *Ninth Annual Conference of the Cognitive Science Society*, pages 462–473, Hillsdale. Erlbaum.

Cutzu, F. and Edelman, S. (1996). Faithful representation of similarities among three-dimensional shapes in human vision. *Proceedings of the National Academy of Science*, 93:12046–12050.

Demartines, P. and Hérault, J. (1996). Curvilinear component analysis: a self-organizing neural network for non linear mapping of data sets. Submitted to IEEE Transaction on Neural Networks.

DeMers, D. and Cottrell, G. (1993). Nonlinear dimensionality reduction. In Hanson, S. J., Cowan, J. D., and Giles, C. L., editors, *Advances in Neural Information Processing Systems 5*, pages 580–587. Morgan Kaufmann.

Edelman, S. (1995a). Representation of similarity in 3D object discrimination. *Neural Computation*, 7:407–422.

Edelman, S. (1995b). Representation, Similarity, and the Chorus of Prototypes. *Minds and Machines*, 5:45–68.

Edelman, S. (1997). Representation is representation of similarity. Behavioral and Brain Sciences, to appear.

Edelman, S., Cutzu, F., and Duvdevani-Bar, S. (1996). Similarity to reference shapes as a basis for shape representation. In Cottrell, G. W., editor, *Proceedings of 18th Annual Conf. of the Cognitive Science Society*, pages 260–265, San Diego, CA.

Edelman, S. and Duvdevani-Bar, S. (1997a). Similarity, connectionism, and the problem of representation in vision. *Neural Computation*, 9:–. in press.

Edelman, S. and Duvdevani-Bar, S. (1997b). A unified framework for recognition and categorization of visual objects. *Phil. Trans. R. Soc. Lond. (B)*, 352:–. to appear.

Efron, B. and Tibshirani, R. (1993). *An introduction to the bootstrap.* Chapman and Hall, London.

Elman, J. L. and Zipser, D. (1988). Learning the hidden structure of speech. *Journal of the Acoustical Society of America*, 4(83):1615–1626.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188.

Grossman, T. and Lapedes, A. (1993). Use of bad training data for better prediction. In Cowan, J. D., Tesauro, G., and Alspector, J., editors, *Advances in Neural Information Processing Systems*, volume 6, pages 342–350. Morgan Kaufmann.

Intrator, N. (1993). Combining exploratory projection pursuit and projection pursuit regression with application to neural networks. *Neural Computation*, 5(3):443–455.

Intrator, N. and Edelman, S. (1996). How to make a low-dimensional representation suitable for diverse tasks. *Connection Science*, pages –.

Kambhatla, N. and Leen, T. K. (1994). Fast non-linear dimension reduction. In Cowan, J. D., Tesauro, G., and Alspector, J., editors, *Advances in Neural Information Processing Systems*, volume 6. Morgan Kaufmann, San Mateo, CA.

Koontz, W. L. G. and Fukunaga, K. (1972). A nonlinear feature extraction algorithm using distance information. *IEEE Trans. Comput.*, 21:56–63.

Kruskal, J. B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27.

Lando, M. and Edelman, S. (1995). Receptive field spaces and class-based generalization from a single view in face recognition. *Network*, 6:551–576.

LeBlanc, M. and Tibshirani, R. (1994). Combining estimates in regression and classification. Preprint.

Leen, T. K. and Kambhatla, N. (1994). Fast non-linear dimension reduction. In Cowan, J. D., Tesauro, G., and Alspector, J., editors, *Advances in Neural Information Processing Systems*, volume 6. Morgan Kauffman, San Francisco, CA.

Lowe, D. (1993). Novel 'topographic' nonlinear feature extraction using radial basis functions for concentration codeing in the 'artificial nose'. In *Proceedings of the 3rd IEE International Conference on Artificial Neural Networks.* IEE Conference Publications.

Lowe, D. and Tipping, M. E. (1996). Feed-forward neural networks and topographic mappings for exploratory data analysis. *Neural Computing and Applications*, 4:83–95.

Nowlan, S. J. and Hinton, G. E. (1992). Simplifying neural networks by soft weight-sharing. *Neural Computation*, 4:473–493.

Oja, E. (1989). Neural networks, principal components, and subspaces. *International Journal of Neural Systems*, 1:61–68.

Pickover, C. (1990). *Computers, Pattern, Chaos, and Beauty.* St. Martin's Press.

Poggio, T. and Girosi, F. (1990). Regularization algorithms for learning that are equivalent to multilayer networks. *Science*, 247:978–982.

Price, D., Knerr, S., Personnaz, L., and Dreyfus, G. (1995). Pairwise neural network classifiers with probabilistic outputs. In G. Tesauro, D. S. T. and Leen, T. K., editors, *Advances in Neural Information Processing 7*, pages 1109–1116. MIT Press.

Raviv, Y. and Intrator, N. (1996). Bootstrapping with noise: An effective regularization technique. *Connection Science, Special issue on Combining Estimators*, 8:356–372.

Sammon, J. W. (1969). A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.*, 18:401–409.

Sas (1989). *SAS/STAT User's Guide, Version 6.* SAS Institute Inc., Cary, NC.

Shepard, R. N. (1966). Metric structures in ordinal data. *J. Math. Psychology*, 3:287–315.

Shepard, R. N. (1980). Multidimensional scaling, tree-fitting, and clustering. *Science*, 210:390–397.

Shepard, R. N. and Cermak, G. W. (1973). Perceptual-cognitive explorations of a toroidal set of free-form stimuli. *Cognitive Psychology*, 4:351–377.

Siedlecki, W., Siedlecka, K., and Sklansky, J. (1988). An overview of mapping techniques for exploratory pattern analysis. *Pattern Recognition*, 21:411–429.

Simard, P., Victorri, B., LeCun, Y., and Denker, J. (1992). Tangent prop – a formalism for specifying selected invariances in an adaptive network. In Moody, J., Lippman, R., and Hanson, S. J., editors, *Neural Information Processing Systems*, volume 4, pages 895–903. Morgan Kaufmann, San Mateo, CA.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions (with discussion). *J. Royal Statistics Society B*, 36:111–147.

Thrun, S. and Mitchell, T. (1995). Learning one more thing. In Mellish, C., editor, *Proc. 14th IJCAI*, volume 2, pages 1217–1223, San Mateo, CA. Morgan Kaufmann.

Wahba, G. (1990). *Splines Models for Observational Data.* Series in Applied Mathematics, Vol. 59, SIAM, Philadelphia.

Webb, A. R. (1995). Multidimensional-scaling by iterative majorization using radial basis functions. *Pattern Recognition*, 28:753–759.

Young, G. and Householder, A. S. (1938). Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3:19–22.