# An Introduction to Nonlinear Dimensionality Reduction by Maximum Variance Unfolding

**Kilian Q. Weinberger** and **Lawrence K. Saul**
Department of Computer and Information Science, University of Pennsylvania
Levine Hall, 3330 Walnut Street, Philadelphia, PA 19104-6389
{kilianw,lsaul}@cis.upenn.edu

## Abstract

Many problems in AI are simplified by clever representations of sensory or symbolic input. How to discover such representations automatically, from large amounts of unlabeled data, remains a fundamental challenge. The goal of statistical methods for dimensionality reduction is to detect and discover low dimensional structure in high dimensional data. In this paper, we review a recently proposed algorithm—*maximum variance unfolding*—for learning faithful low dimensional representations of high dimensional data. The algorithm relies on modern tools in convex optimization that are proving increasingly useful in many areas of machine learning.

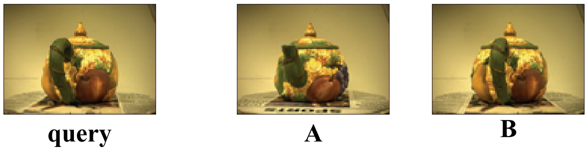

Figure 1: Images of teapots: pixel distances versus perceptual distances. As measured by the mean-squared-difference of pixel intensities, image A is closer to the query image than image B, despite the fact that the view in image A involves a full 180 degrees of rotation.

## Introduction

A fundamental challenge of AI is to develop useful internal representations of the external world. The human brain excels at extracting small numbers of relevant features from large amounts of sensory data. Consider, for example, how we perceive a familiar face. A friendly smile or a menacing glare can be discerned in an instant and described by a few well chosen words. On the other hand, the digital representations of these images may consist of hundreds or thousands of pixels. Clearly, there are much more compact representations of images, sounds, and text than their native digital formats. With such representations in mind, we have spent the last few years studying the problem of dimensionality reduction—how to detect and discover low dimensional structure in high dimensional data.

For higher-level decision-making in AI, the right representation makes all the difference. We mean this quite literally, in the sense that proper judgments of similiarity and difference depend crucially on our internal representations of the external world. Consider, for example, the images of teapots in Fig. 1. Each image shows the same teapot from a different angle. Compared on a pixel-by-pixel basis, the query image and image A are the most similar pair of images; that is, their pixel intensities have the smallest mean-squared-difference. The viewing angle in image B, however, is much closer to the viewing angle in the query image—evidence that distances in pixel space do not support crucial judgments of similarity and difference. (Consider the embarrassment when your robotic butler grabs the teapot by its spout rather than its handle, not to mention the liability when it subsequently attempts to refill your guest's cup.) A more useful representation of these images would index them by the teapot's angle of rotation, thus locating image B closer to the query image than image A.

Objects may be similar or different in many ways. In the teapot example of Fig. 1, there is only one degree of freedom: the angle of rotation. More generally, there may be many criteria that are relevant to judgments of similarity and difference, each associated with its own degree of freedom. These degrees of freedom are manifested over time by variabilities in appearance or presentation.

The most important modes of variability can often be distilled by automatic procedures that have access to large numbers of observations. In essence, this is the goal of statistical methods for dimensionality reduction (Burges 2005; Saul *et al.* 2006). The observations, initially represented as high dimensional vectors, are mapped into a lower dimensional space. If this mapping is done faithfully, then the axes of the lower dimensional space relate to the data's intrinsic degrees of freedom.

The linear method of principal components analysis (PCA) performs this mapping by projecting high dimensional data into low dimensional subspaces. The principal subspaces of PCA have the property that they maximize the variance of the projected data. PCA works well if the most important modes of variability are approximately linear. In this case, the high dimensional observations can be very well

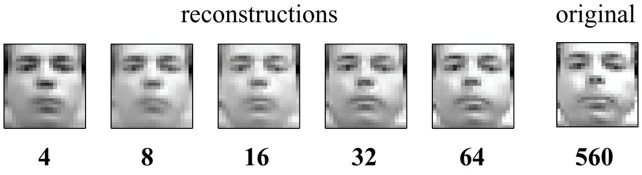

Figure 2: Results of PCA applied to a data set of face images. The figure shows a grayscale face image (right) and its linear reconstructions from different numbers of principal components. The number of principal components required for accurate reconstruction greatly exceeds the small number of characteristic poses and expressions in the data set.

reconstructed from their low dimensional linear projections.

PCA works poorly if the most important modes of variability are nonlinear. To illustrate the effects of nonlinearity, we applied PCA to a data set of 28 × 20 grayscale images. Each image in the data set depicted a different pose or expression of the same person's face. The variability of faces is not expressed linearly in the pixel space of grayscale images. Fig. 2 shows the linear reconstructions of a particular image from different numbers of principal components (that is, from principal subspaces of different dimensionality). The reconstructions are not accurate even when the number of principal components greatly exceeds the small number of characteristic poses and expressions in this data set.

In this paper, we review a recently proposed algorithm for *nonlinear dimensionality reduction*. The algorithm, known as "maximum variance unfolding" (Sun *et al.* 2006; Saul *et al.* 2006), discovers faithful low dimensional representations of high dimensional data, such as images, sounds, and text. It also illustrates many ideas in convex optimization that are proving increasingly useful in the broader field of machine learning.

Our work builds on earlier frameworks for analyzing high dimensional data that lies on or near a low dimensional *manifold* (Tenenbaum, de Silva, & Langford 2000; Roweis & Saul 2000). Manifolds are spaces that are locally linear, but unlike Euclidean subspaces, they can be globally nonlinear. Curves and surfaces are familiar examples of one and two dimensional manifolds. Compared to earlier frameworks for manifold learning, maximum variance unfolding has many interesting properties, which we describe in the following sections.

## Maximum Variance Unfolding

Algorithms for nonlinear dimensionality reduction map high dimensional inputs $\{\vec{x}_i\}_{i=1}^n$ to low dimensional outputs $\{\vec{y}_i\}_{i=1}^n$, where $\vec{x}_i \in \Re^d$, $\vec{y}_i \in \Re^r$, and $r \ll d$. The reduced dimensionality $r$ is chosen to be as small as possible, yet sufficiently large to guarantee that the outputs $\vec{y}_i \in \Re^r$ provide a faithful representation of the inputs $\vec{x}_i \in \Re^d$.

What constitutes a "faithful" representation? Suppose that the high dimensional inputs lie on a low dimensional manifold. For a faithful representation, we ask that the distances between nearby inputs match the distances between nearby outputs. Such locally distance-preserving representations are exactly the kind constructed by maximum variance unfolding.

The algorithm for maximum variance unfolding is based on a simple intuition. Imagine the inputs $\vec{x}_i$ as connected to their $k$ nearest neighbors by rigid rods. (The value of $k$ is the algorithm's one free parameter.) The algorithm attempts to pull the inputs apart, maximizing the sum total of their pairwise distances without breaking (or stretching) the rigid rods that connect nearest neighbors. The outputs are obtained from the final state of this transformation.

The effect of this transformation is easy to visualize for inputs that lie on low dimensional manifolds, such as curves or surfaces. For example, imagine the inputs as beads on a necklace that is coiled up in three dimensions. By pulling the necklace taut, the beads are arranged in a line, a nonlinear dimensionality reduction from $\Re^3$ to $\Re^1$. Alternatively, imagine the inputs as the lattice of sites in a crumpled fishing net. By pulling on the ends of the net, the inputs are arranged in a plane, a nonlinear dimensionality reduction from $\Re^3$ to $\Re^2$. As we shall see, this intuition for maximum variance unfolding also generalizes to higher dimensions.

The "unfolding" transformation described above can be formulated as a quadratic program. Let $\eta_{ij} \in \{0,1\}$ denote whether inputs $\vec{x}_i$ and $\vec{x}_j$ are $k$-nearest neighbors. The outputs $\vec{y}_i$ from maximum variance unfolding, as described above, are those that solve the following optimization:

**Maximize** $\sum_{ij} \|\vec{y}_i - \vec{y}_j\|^2$ **subject to:**
**(1)** $\|\vec{y}_i - \vec{y}_j\|^2 = \|\vec{x}_i - \vec{x}_j\|^2$ **for all** $(i,j)$ **with** $\eta_{ij}=1$**.**
**(2)** $\sum_i \vec{y}_i = 0$

Here, the first constraint enforces that distances between nearby inputs match distances between nearby outputs, while the second constraint yields a unique solution (up to rotation) by centering the outputs on the origin.

The apparent intractability of this quadratic program can be finessed by a simple change of variables. Note that as written above, the optimization over the outputs $\vec{y}_i$ is not convex, meaning that it potentially suffers from spurious local minima. Defining the inner product matrix $K_{ij} = \vec{y}_i \cdot \vec{y}_j$, we can reformulate the optimization as a *semidefinite program* (SDP) (Vandenberghe & Boyd 1996) over the matrix $K$. The resulting optimization is simply a linear program over the matrix elements $K_{ij}$, with the additional constraint that the matrix $K$ has only nonnegative eigenvalues, a property that holds for all inner product matrices. In earlier work (Weinberger & Saul 2004; Weinberger, Sha, & Saul 2004), we showed that the SDP over $K$ can be written as:

**Maximize trace**$(K)$ **subject to:**
**(1)** $K_{ii} - 2K_{ij} + K_{jj} = \|\vec{x}_i - \vec{x}_j\|^2$ **for all** $(i,j)$
**with** $\eta_{ij}=1$**.**
**(2)** $\Sigma_{ij} K_{ij} = 0$**.**
**(3)** $K \succeq 0$**.**

The last (additional) constraint $K \succeq 0$ requires the matrix $K$ to be positive semidefinite. Unlike the original quadratic program for maximum variance unfolding, this SDP is convex. In particular, it can be solved efficiently with polynomial-time guarantees, and many off-the-shelf solvers are available in the public domain.

From the solution of the SDP in the matrix $K$, we can derive outputs $\vec{y}_i \in \Re^n$ satisfying $K_{ij} = \vec{y}_i \cdot \vec{y}_j$ by singular value decomposition. An $r$-dimensional representation that approximately satisfies $K_{ij} \approx \vec{y}_i \cdot \vec{y}_j$ can be obtained from the top $r$ eigenvalues and eigenvectors of $K$. Roughly speaking, the number of dominant eigenvalues of $K$ indicates the number of dimensions needed to preserve local distances while maximizing variance. In particular, if the top $r$ eigenvalues of $K$ account for (say) $95\%$ of its trace, this indicates that an $r$-dimensional representation can capture $95\%$ of the *unfolded* data's variance.

## Experimental Results

We have used maximum variance unfolding (MVU) to analyze many high dimensional data sets of interest. Here we show some solutions (Weinberger & Saul 2004; Blitzer *et al.* 2005) that are particularly easy to visualize.

Fig. 3 shows a two dimensional representation of teapot images discovered by MVU. The data set consisted of $n = 400$ high resolution color images showing a porcelain teapot viewed from different angles in the plane. The teapot was viewed under a full 360 degrees of rotation. Each image contained $76 \times 101$ RGB pixels, so that the pixel space had dimensionality $d = 23028$. The two dimensional representation discovered by MVU is easily visualized by superimposing represenative images on top of their corresponding outputs in the plane. The outputs are arranged in a circle, reflecting the cyclic degree of freedom in the data set. Note also how this representation supports judgments of similarity and difference that are not evident in the original pixel space, as discussed in Fig. 1.

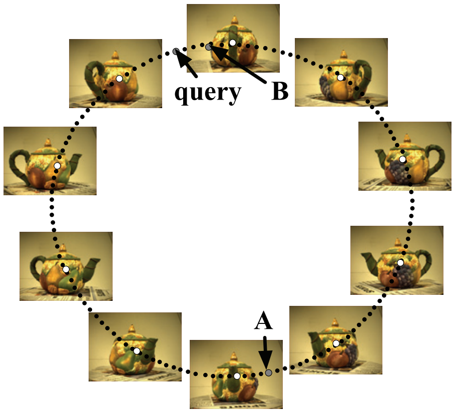

Figure 3: Two dimensional representation from MVU of $n = 400$ images of a teapot, viewed from different angles in the plane. The circular arrangement reflects the underlying rotational degree of freedom. In this representation, image B is closer to the query image than image A, unlike in Fig. 1.

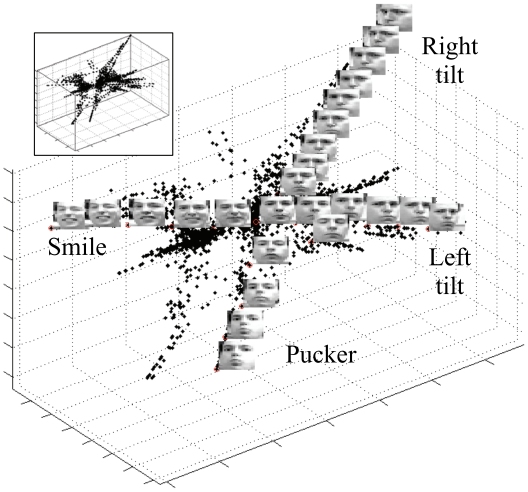

Figure 4: Three dimensional representation from MVU of $n = 1960$ grayscale images of faces. The superimposed images reveal a small number of characteristic actions as the underlying degrees of freedom.

Fig. 4 shows a three dimensional representation of face images discovered by MVU. The data set consisted of $n = 1960$ grayscale images of the same person's face in different poses and expressions. Each image had $28 \times 20$ pixels, so that the pixel space had dimensionality $d = 560$. In contrast to the results from PCA in Fig. 2, the solution obtained by MVU reveals a small number of characteristic actions (e.g., left and right tilt, smile, pucker) as the underlying degrees of freedom in the data set.

Fig. 5 shows a two dimensional representation of words discovered by MVU. The inputs to MVU were derived from the co-occurrence statistics of the $n = 2000$ most frequently occuring words in a large corpus of text. Each word was represented by a sparse $d = 60000$ dimensional vector of normalized counts, as typically collected for bigram language modeling. The figure shows that many semantic relationships between words are preserved despite the drastic reduction in dimensionality from $d = 60000$ to two dimensions (for visualization in the plane).

Table 1 compares the estimated dimensionalities of the data sets in Figs. 3–5 from the results of linear versus nonlinear dimensionality reduction. The estimates from PCA were computed from the minimum dimensionality subspace required to contain $95\%$ of the original data's variance. The estimates from MVU were computed from the minimum dimensionality subspace required to contain $95\%$ of the "unfolded" data's variance. For all these data sets, MVU discovers much more compact representations than PCA.

## Discussion

In this paper we have described the use of *maximum variance unfolding* for nonlinear dimensionality reduction. Large-scale applications of maximum variance unfolding require one additional insight. As originally formulated, the size of the SDP scales linearly with the number of examples, $n$. In previous work (Weinberger, Packer, & Saul 2005; Sha & Saul 2005), we showed that the SDP can be tremen-

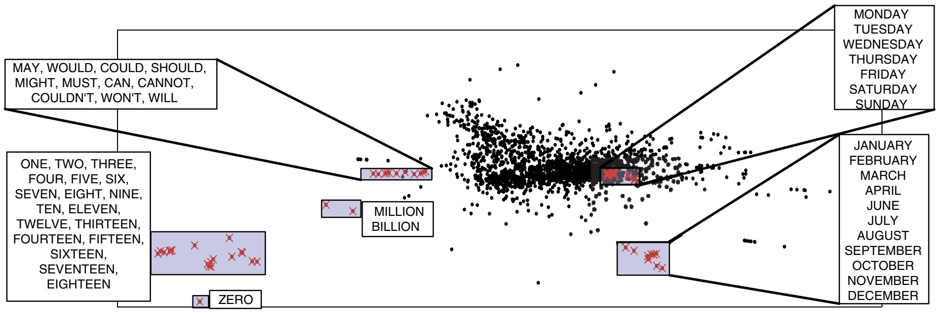

Figure 5: Two dimensional representation from MVU of the 2000 most frequently occuring words in the NAB corpus. The representation preserves clusters of words with similar meanings.

| | teapots | faces | words |
|---|---|---|---|
| initial | 23028 | 560 | 60000 |
| linear | 59 | 80 | 23 |
| nonlinear | 2 | 4 | 6 |

Table 1: Dimensionalities of different data sets, as estimated from the results of linear versus nonlinear dimensionality reduction. The top row shows the dimensionality of the data's original representation.

dously simplified by factoring the $n \times n$ target matrix as $K \approx QLQ^\top$, where $L \in \mathcal{R}^{m\times m}$ and $Q \in \mathcal{R}^{n\times m}$ with $m \ll n$. The matrix $Q$ in this factorization can be precomputed from the results of faster but less robust methods for nonlinear dimensionality reduction. The factorization transforms the original SDP over the matrix $K$ into a much smaller SDP over the matrix $L$. This approach works well in practice, enabling maximum variance unfolding to analyze much larger data sets than we originally imagined.

One advantage of maximum variance unfolding is its flexibility to be adapted to particular applications. For example, the distance-preserving constraints in the SDP can be relaxed to handle noisy data or to yield more aggressive results in dimensionality reduction (Sha & Saul 2005). Alternatively, additional constraints can be enforced to incorporate prior knowledge. Along these lines, a rather novel extension of maximum variance unfolding has been developed for visual robot navigation and mapping (Bowling, Ghodsi, & Wilkinson 2005). The authors use a semidefinite program to construct a map of a simulated robot's virtual environment. They adapt our framework to learn from the actions of the robot as well as the images of its environment. The algorithm has also been applied to statistical language modeling (Blitzer *et al.* 2005), where low dimensional representations of words were derived from bigram counts and used to improve on traditional models. We are hopeful that applications will continue to emerge in many areas of AI.

## Acknowledgments

This work was supported by NSF Award 0238323.

## References

Blitzer, J.; Weinberger, K. Q.; Saul, L. K.; and Pereira, F. C. N. 2005. Hierarchical distributed representations for statistical language modeling. In *Advances in Neural and Information Processing Systems*, volume 17. Cambridge, MA: MIT Press.

Bowling, M.; Ghodsi, A.; and Wilkinson, D. 2005. Action respecting embedding. In *Proceedings of the Twenty-second International Conference on Machine Learning (ICML 2005)*.

Burges, C. J. C. 2005. Geometric methods for feature extraction and dimensional reduction. In Rokach, L., and Maimon, O., eds., *Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*. Kluwer Academic Publishers.

Roweis, S. T., and Saul, L. K. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290:2323–2326.

Saul, L. K.; Weinberger, K. Q.; Ham, J. H.; Sha, F.; and Lee, D. D. 2006. Spectral methods for dimensionality reduction. In B. Schoelkopf, O. C., and Zien, A., eds., *Semisupervised Learning*. MIT Press.

Sha, F., and Saul, L. K. 2005. Analysis and extension of spectral methods for nonlinear dimensionality reduction. In *Proceedings of the Twenty-second International Conference on Machine Learning (ICML 2005)*.

Sun, J.; Boyd, S.; Xiao, L.; and Diaconis, P. 2006. The fastest mixing Markov process on a graph and a connection to a maximum variance unfolding problem. To appear in *SIAM Review*.

Tenenbaum, J. B.; de Silva, V.; and Langford, J. C. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science* 290:2319–2323.

Vandenberghe, L., and Boyd, S. P. 1996. Semidefinite programming. *SIAM Review* 38(1):49–95.

Weinberger, K. Q., and Saul, L. K. 2004. Unsupervised learning of image manifolds by semidefinite programming. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-04)*, volume 2, 988–995.

Weinberger, K. Q.; Packer, B. D.; and Saul, L. K. 2005. Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. In Ghahramani, Z., and Cowell, R., eds., *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*.

Weinberger, K. Q.; Sha, F.; and Saul, L. K. 2004. Learning a kernel matrix for nonlinear dimensionality reduction. In *Proceedings of the Twenty First International Conference on Machine Learning (ICML-04)*, 839–846.