1

Unsupervised learning of curved manifolds

Vin de Silva and Joshua B. Tenenbaum†*
Departments of Mathematics* and Psychology†, Stanford University
silva@math.stanford.edu, jbt@psych.stanford.edu

Abstract

We describe a variant of the Isomap manifold learning algorithm [1], called 'C-Isomap'. Isomap was designed to learn non-linear mappings which are *isometric* embeddings of a flat, convex data set. C-Isomap is designed to recover mappings in the larger class of *conformal* embeddings, provided that the original sampling density is reasonably uniform. We compare the performance of both versions of Isomap and other algorithms for manifold learning (MDS, LLE, GTM) on a range of synthetic data sets.

1.1 Introduction

We consider the problem of manifold learning: recovering meaningful low-dimensional structures hidden in high-dimensional data. An example might be a set of pixel images of an individual's face observed under different pose and lighting conditions; the task is to identify the underlying variables (angle of elevation, direction of light, etc.) given only the high-dimensional pixel image data [1].

Recently Tenenbaum et al. introduced the Isomap algorithm, which extends the classical techniques of principal components analysis (PCA) and multidimensional scaling (MDS) [2] to a class of nonlinear manifolds, those which are isometric to a convex domain of Euclidean space. This includes manifolds like the 'swiss roll' (Figure 1.1(b)), but not those like the 'fishbowl' (Figure 1.1(c)), which have intrinsic curvature. For nonlinear but intrinsically flat manifolds such as the swiss roll, Isomap efficiently finds a globally optimal low-dimensional representation that can be proven to converge asymptotically to the true structure of the data.

Here we extend the Isomap approach to a class of intrinsically curved data sets that are conformally equivalent to Euclidean space. This allows us to learn the structure of manifolds like the fishbowl, as well as other

more complex data manifolds where the conformal assumption may be approximately valid.

The paper is organised as follows. In Section 2 we introduce a hierarchy of generative models for manifold learning and describe archetypal data sets representing each of the models. Section 3 describes the conformal Isomap (C-Isomap) algorithm. In Section 4 we compare C-Isomap and several alternative algorithms on the test data sets.

1.2 Generative models for manifold learning

We can view the problem of manifold learning as trying to invert the following generative model for a set of observations [3].

Let Y be a d-dimensional domain contained in the Euclidean space $\mathbf{R}^d$, and let $f : Y \to \mathbf{R}^N$ be a smooth embedding, for some $N > d$. Data points $\{y_i\} \subset Y$ are generated by some random process, and are mapped by f to give the *observed data*, $\{x_i = f(y_i)\} \subset \mathbf{R}^N$. We refer to Y as the *latent space* and to $\{y_i\}$ as the *latent data*.

The task is to reconstruct f and $\{y_i\}$ from the observed data $\{x_i\}$ alone. The answer can take various forms. The approach taken by Isomap is to present *reconstructed data*, $\{z_i\} \subset \mathbf{R}^{d'}$, for some d'. In a successful solution, $d' = d$ and the configurations $\{z_i\} \subset \mathbf{R}^{d'}$ and $\{y_i\} \subset \mathbf{R}^d$ are congruent in a suitable sense. A mapping $f' : \mathbf{R}^{d'} \to \mathbf{R}^N$ can be constructed from the pairs (z_i, x_i) by any reasonable interpolation procedure (such as radial basis functions [4]); this amounts to recovering f.

Before the general problem stated above becomes meaningful, we require some constraints on the mapping f and on the random sampling process. Table 1.1 lists three such sets of constraints, together with a algorithm in each case that is provably correct: MDS exactly recovers the original configuration (see [8]), and the two Isomap algorithms converge to the right answer as the number of data points tends to infinity. Note that as we move from top to bottom in the table, assumptions about the mapping are relaxed but we require increasingly strong assumptions about the sampling density. In other words, there is no free lunch in manifold learning.

The simplest case is when f is a linear isometry $\mathbf{R}^d \to \mathbf{R}^N$; for example, a set of 2-dimensional data mapped into a plane in $\mathbf{R}^3$ (Figure 1.1(a)). The latent distribution may be arbitrary. PCA recovers the d significant dimensions of the observed data. Classical MDS produces the same results, but requires only the pairwise distance matrix as its input (not the actual embedding coordinates).

The second case is where $f : Y \to \mathbf{R}^N$ is an isometric embedding, in the sense of Riemannian geometry. In other words, f preserves the length of (and angles between) infinitesimal vectors in Y. The image set $f(Y)$ is then an *intrinsically flat* submanifold of $\mathbf{R}^N$; our standard example is the Swiss

Table 1.1. Under different assumptions about the distribution of latent variables (first column) and the mapping from latent variables to observations (second column), different algorithms are appropriate.

DISTRIBUTION	MAPPING	ALGORITHM
arbitrary	linear isometry	classical MDS
convex, dense	isometry	Isomap
convex, uniformly dense	conformal embedding	conformal Isomap

roll (Figure 1.1(b)). When Y is a convex domain in $\mathbf{R}^d$, and provided the data points are sufficiently dense,[1] Isomap successfully recovers the approximate original structure of data sets generated in this way. The crux of the method is to estimate the global metric structure of the latent space using local geometric information.

The intrinsically flat model becomes unsuitable when the data manifold has any non-negligible Gaussian curvature; for example, data lying on the surface of a sphere. Curvature in the data manifold may occur in two places: it may be intrinsic to the latent space Y, or it may be introduced by the mapping f. The first situation represents a hard problem; it is not even clear what form the output from a manifold learning algorithm should be expected to take. The Euclidean assumption in our generative model deliberately excludes this possibility. The second situation, as we now show, can be handled for a certain class of functions.

Specifically, our third model allows f to be a *conformal* embedding. At each point y in Y, angles between infinitesimal vectors are preserved by f, but their lengths are scaled by a factor $\phi = \phi(y) > 0$, which varies smoothly over Y. Compensating for this extra degree of freedom, we require that the original data be *uniformly* dense in Y.

The simplest example of a conformal map is the stereographic projection in $\mathbf{R}^3$ from the plane $z = 0$ to the unit sphere.[2] A large, centered disk in the plane maps to a 'fishbowl' under this map. Note that data that is uniformly sampled in the disk bunches up non-uniformly near the rim of fishbowl.

In the next two sections we describe Isomap and explain how it can be modified to deal with uniformly sampled, conformally embedded data. The key idea is to use the density of the observed data to estimate $\phi(y)$. This strategy is similar to the motivation behind conformal approaches to nonlinear ICA [5], but differs in that our goal is to reduce dimensionality

[1] The data density required depends on certain geometric parameters of the embedding: the minimum radius of curvature, and the minimum branch separation (see [1]).

[2] The Mercator projection is another well-known conformal map.

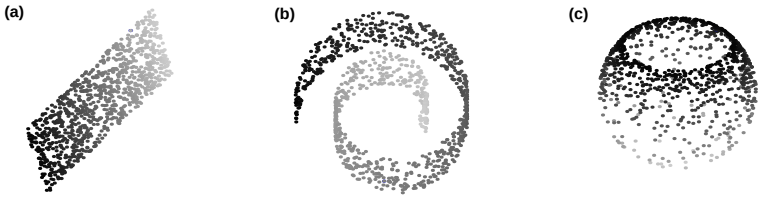

Figure 1.1. Archetypal datasets for the three generative models described in Table 1.1.

rather than to find a better basis for observations in a space of the same dimensionality.

It is worth contrasting the *stereographic fishbowl* described above with a *uniform fishbowl* dataset, where points are sampled uniformly from the fishbowl surface (Figure 1.1, column 3). Here there is no bunching effect at the rim, and hence no way to estimate $\phi(y)$. Our new methods do not extend to the uniform fishbowl; indeed it is not clear what the 'correct' answer should be.

1.3 The Isomap algorithm

We briefly describe the standard Isomap procedure [1].

1. Determine a *neighbourhood graph* G of the observed data $\{x_i\}$ in a suitable way. For example, G might contain $x_i x_j$ iff x_j is one of the k nearest neighbours of x_i (and vice versa). Alternatively, G might contain the edge $x_i x_j$ iff $|x_i - x_j| < \epsilon$, for some ϵ.
2. Compute shortest paths in the graph for all pairs of data points. Each edge $x_i x_j$ in the graph is weighted by its Euclidean length $|x_i - x_j|$, or by some other useful metric.
3. Apply MDS to the resulting shortest-path distance matrix D, to find the reconstructed data points $\{z_i\}$ in $\mathbf{R}^{d'}$.

The premise is that *local* metric information (in this case, lengths of edges $x_i x_j$ in the neighbourhood graph) is regarded as a trustworthy guide to the local metric structure in the original (latent) space. The shortest-paths computation then gives an estimate of the global metric structure, which can be fed into MDS to produce the required embedding.[3]

As an example, Isomap is effective in recovering the original rectangular structure of a Swiss roll data set with 2000 points, as illustrated in the

[3]Note that in the extreme case where G is a complete graph, meaning that all distances are regarded as trustworthy, the computation reduces to Step 3 alone, the MDS calculation. Thus Isomap may be viewed as a generalisation of MDS.

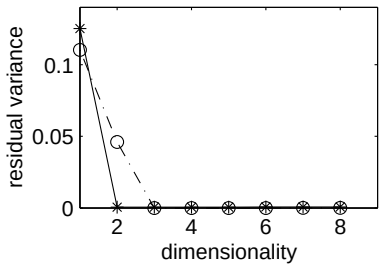

Figure 1.2. The residual variance plot shows that Isomap [unbroken line] detects the true 2-dimensional structure of the Swiss roll data, whereas MDS [dots-and-dashes] identifies the data as being 3-dimensional.

first column of Figure 1.6. We used the k-nearest neighbours method with $k = 10$. Compare the Isomap output (row 3) with the output from MDS by itself (row 2). We will return to this example later.

The embedding dimension ($d' = 2$ in this case) is chosen at the end, by consulting a suitable 'residual variance' function, which measures how badly the MDS embedding preserves the distance matrix obtained during Step 2. Figure 1.2 shows the residual variance for Isomap and MDS embeddings in dimensions $d' = 1, 2, \ldots 8$. At $d' = 2$, the residual variance of the Isomap embedding has fallen virtually to zero, so there is little gain in choosing a higher value of d'. In a similar way, it is clear from the graph that MDS prefers a three-dimensional representation of the data. In general, one looks for an 'elbow' in the graph to indicate the preferred dimensionality [1].

When does Isomap succeed? The *recovered* shortest-paths distance matrix D needs to be a good approximation to the *original* Euclidean distance matrix for the points Y. The following convergence theorem is proved in [7]:

Theorem. *Let Y be sampled from a bounded convex region in $\mathbf{R}^d$, with respect to a density function $\alpha = \alpha(y)$. Let f be a C^2-smooth isometric embedding of that region in $\mathbf{R}^N$. Given $\lambda, \mu > 0$, for a suitable choice of neighborhood size parameter ϵ or k, we have*

$$1 - \lambda \leq \frac{\text{recovered distance}}{\text{original distance}} \leq 1 + \lambda$$

with probability at least $1 - \mu$, provided that the sample size is sufficiently large. [The formula is taken to hold for all pairs of points simultaneously.]

Explicit bounds for "sufficiently large" are given in [7] which depend on certain geometric parameters measuring the nonlinearity of the embedding. Specifically, if the embedding is highly curved, or if widely separated points in the domain are brought close to each other by f, then the task is more difficult and the required sample size increases.

The proof takes the following line. Since f is an isometric embedding, by definition it preserves the *infinitesimal* metric structure of Y perfectly.

When the neighbourhood size is small, the *local* metric structure of the data is preserved approximately. The shortest-paths calculation effectively computes an approximation to geodesic distance in the original domain. For convex domains, geodesic distance is equal to Euclidean distance, which finishes the proof. Details are in [7].

1.4 The C-Isomap algorithm

C-Isomap is a simple variation on standard Isomap. Specifically, we use the k-neighbours method in Step 1, and replace Step 2 with the following:

2a. Compute shortest paths in the graph for all pairs of data points. Each edge $x_i x_j$ in the graph is weighted by $|x_i - x_j|/\sqrt{M(i)M(j)}$. Here $M(i)$ is the mean distance of x_i to its k nearest neighbours.

We can motivate Step 2a as follows. Assume that we have a very large number of uniformly sampled data points. In the latent space, the k nearest neighbours of a given point y_i occupy a d-dimensional disk of approximately a certain radius r that depends on d and on the sampling density, but not on y_i. The map f carries this approximately to a d-dimensional disk of radius $r\phi(y_i)$ in $\mathbf{R}^N$. The expected average distance of these k points from x_i is therefore proportional to $\phi(y_i)$. The computed quantity $M(i)$ is a fair approximation, so Step 2a has the effect of correcting for ϕ.

Using this kind of reasoning, one can prove a convergence theorem for C-Isomap:

Theorem. *Let Y be sampled uniformly from a bounded convex region in $\mathbf{R}^d$. Let f be a C^2-smooth conformal embedding of that region in $\mathbf{R}^N$. Given $\lambda, \mu > 0$, for a suitable choice of neighborhood size parameter k, we have*

$$1 - \lambda \leq \frac{\text{recovered distance}}{\text{original distance}} \leq 1 + \lambda$$

with probability at least $1 - \mu$, provided that the sample size is sufficiently large.

Explicit lower bounds for the sample size are much more difficult to formulate here; certainly we expect to require a larger sample than in regular Isomap to obtain good approximations. In situations where both Isomap and C-Isomap are applicable, it may be preferable to use Isomap, being less susceptible to local fluctuations in the sample density.

In general, the crude effect of C-Isomap is to magnify regions of the data where the point density is high, and to shrink regions where the point density is low. Whether or not the conformal model is valid in a given case, this tendency towards uniformity may still be useful for representing the

large-scale structure of a data set. The 'wavelet' data set discussed later is a good illustration of this.

1.5 Comparative performance of C-Isomap

We present the results of tests using four different data sets: the Swiss roll, stereographic fishbowl, a 'wavelet' data set, and the uniform fishbowl. As well as C-Isomap, we tested four other algorithms on the same data sets: MDS, Isomap, Locally Linear Embedding (LLE) [6] and the Generative Topographic Mapping (GTM) [3]. LLE, like Isomap, computes a global low-dimensional embedding of the data from only local metric information, but differs in that the embedding aims to preserve only that local metric structure, rather than an estimate of the manifold's global metric structure, as in Isomap or C-Isomap. Respecting only local constraints allows LLE to handle more curvature than Isomap, at the cost of sometimes producing less globally stable results. GTM uses the EM algorithm to fit a version of the generative model in Section 2, under the assumptions of a uniform square density in latent space and a soft smoothness prior on the nonlinear mapping. Assuming only a sufficiently smooth mapping is more general than our conformal assumption, but the strong constraint of a uniform square density and the use of a greedy optimization procedure prone to local minima problems frequently lead GTM to distorted pictures of a data manifold.

For each data set, each algorithm was used to obtain a 2-dimensional embedding of the points. Figure 2 summarises the results. Each column shows the results of applying the five algorithms on a particular data set. The data set itself is shown at the top, in a 3-dimensional representation. The Swiss roll data points are shaded to indicate one of the original rectangular coordinates. The shading on the other three data sets indicates the original z-varaiable.

The images shown are intended to be reasonably typical of the results a careful user might obtain; specifically the parameter k in Isomap, C-Isomap and LLE was tuned for good performance. For GTM we used $20^2 = 400$ sample points on a grid; the same number of basis functions; relative width $\sigma = 2$; and weight regularisation factor $l = 1$. In all cases the training process converged by the 50th iteration (often much sooner); we show the posterior mean coordinates after convergence.

Results: Swiss roll, two fishbowls, and a wavelet

The Swiss roll (column 1) was constructed by sampling 2000 points uniformly from a rectangle and mapping into $\mathbf{R}^3$ using an isometric spiral embedding. The 2-dimensional projection given by MDS fails to resolve

the true non-linear structure of the data. The three techniques Isomap, C-Isomap and LLE perform well. Isomap recovers the original rectangle, including its aspect ratio, with very little distortion. C-Isomap gives similar results but is noticeably slightly worse that Isomap, as we would expect. The sides of the rectangle curve even more in the LLE coordinates. Note that LLE is unable to recover the aspect ratio of the rectangle, because it is constrained to give output with equal covariance in all directions. Finally, GTM entirely fails to detect the 2-dimensional structure of the data. Instead it converges to a representation of the data by its (1-dimensional) principal axis.

The stereographic fishbowl (column 2) was constructed by sampling 2000 points uniformly from a disk in the plane, and projecting the result stereographically onto a sphere. The MDS coordinates are just the 2-dimensional vertical projection of the data. There is an 'annulus of ambiguity' where the projection map is 2-to-1. As expected, regular Isomap does no better than MDS; however C-Isomap and LLE are both successful in flattening the fishbowl and recovering the original disk shape. GTM appears to be confused by the heavy point density around the rim; the output incorrectly indicates a ring-shaped structure.

The uniform fishbowl (column 4) was constructed by sampling 2000 points uniformly from the fishbowl surface itself. Again the first two dimensions of MDS give the vertical projection. Since the sampling density is uniform, Isomap and C-Isomap behave alike, giving a version of the MDS projection slightly widened at the rim. LLE is more successful at opening out the rim, but it remains partly turned in. GTM gives the best performace, flattening out the fishbowl. What is missing (and what GTM does not seek to provide) is any clear indication of the round shape of data set.

The wavelet (column 3) was chosen as an example of a non-conformally generated data set on which C-Isomap performs well. 2000 data points were sampled uniformly in a disk and then translated in the z-direction by a function with one positive peak and one negative peak (specifically, we used the x-derivative of a Gaussian, scaled to exaggerate the peaks). The projection given by MDS is tilted; we see two dark 'horns' corresponding to the noise peaks, where the 2-dimensional representation is ambiguous. Regular Isomap and LLE both show the same phenomenon, with Isomap doing worst. C-Isomap flattens the wavelet almost perfectly, with a slight stretching in the x-direction. GTM also flattens the data, but the results are comparatively distorted.

1.6 Conclusions

We have presented a simple variant of the Isomap manifold learning algorithm which addresses a recurrent difficulty in nonlinear data analysis: the

problem of scale. Naturally occuring data manifolds are often generated by the action of some group of transformations (for example, rotations, translations and enlargements of an image). A given transformation may have a small effect in one region of the data, but a very large effect somewhere else. Any attempt to discover the global structure of this kind of data set must in some way deal with this issue.

C-Isomap uses an elementary mechanism to defuse the problem. As a variant of Isomap, C-Isomap inherits its theoretical advantages: it is a non-iterative, computationally efficient solution to a global optimisation problem, and can be proved to converge asymptotically to the right answer for a large class of manifolds. In practice it performs well on a range of synthetic data sets, including some outside the strict parameters of the conformal generative model.

In forthcoming work, we will investigate why it makes sense to apply a conformal model in the case of certain naturally occuring data sets. One good example is a manifold of images of a face or other object seen (in various orientations) at a continuous range of distances. Changes in orientation at a long distance have a smaller effect on local pixel distances than the corresponding changes at a shorter distance. The results of these investigations will appear in a future paper.

Acknowledgement. We gratefully acknowledge the support of the DARPA Human ID project, the Office of Naval Research, and the National Science Foundation. We thank Sam Roweis for stimulating discussions, and Larry Saul for suggesting the conformal fishbowl example as a test case.

References

[1] J. B. Tenenbaum, V. de Silva & J. C. Langford. *Science* **290**, 2319 (2000).

[2] K. V. Mardia, J. T. Kent & J. M. Bibby. *Multivariate Analysis*, (Academic Press, London, 1979).

[3] C. M. Bishop, M. Svensén, C. K. I. Williams. *Neural Computation* **10**, 215 (1998).

[4] D. Beymer, T. Poggio, *Science* **272**, 1905 (1996).

[5] A. Hyvärinen & P. Pajunen (1998). Nonlinear Independent Component Analysis: Existence and Uniqueness Results. Submitted to *Neural Networks*.

[6] S. Roweis & L. Saul. *Science* **290**, 2323 (2000).

[7] M. Bernstein, V. de Silva, J. C. Langford, J. B. Tenenbaum. Preprint (December 2000) available at

`http://isomap.stanford.edu/BdSLT.pdf`

[8] T. F. Cox & M. A. A. Cox, *Multidimensional Scaling*, (Chapman & Hall, London, 1994).

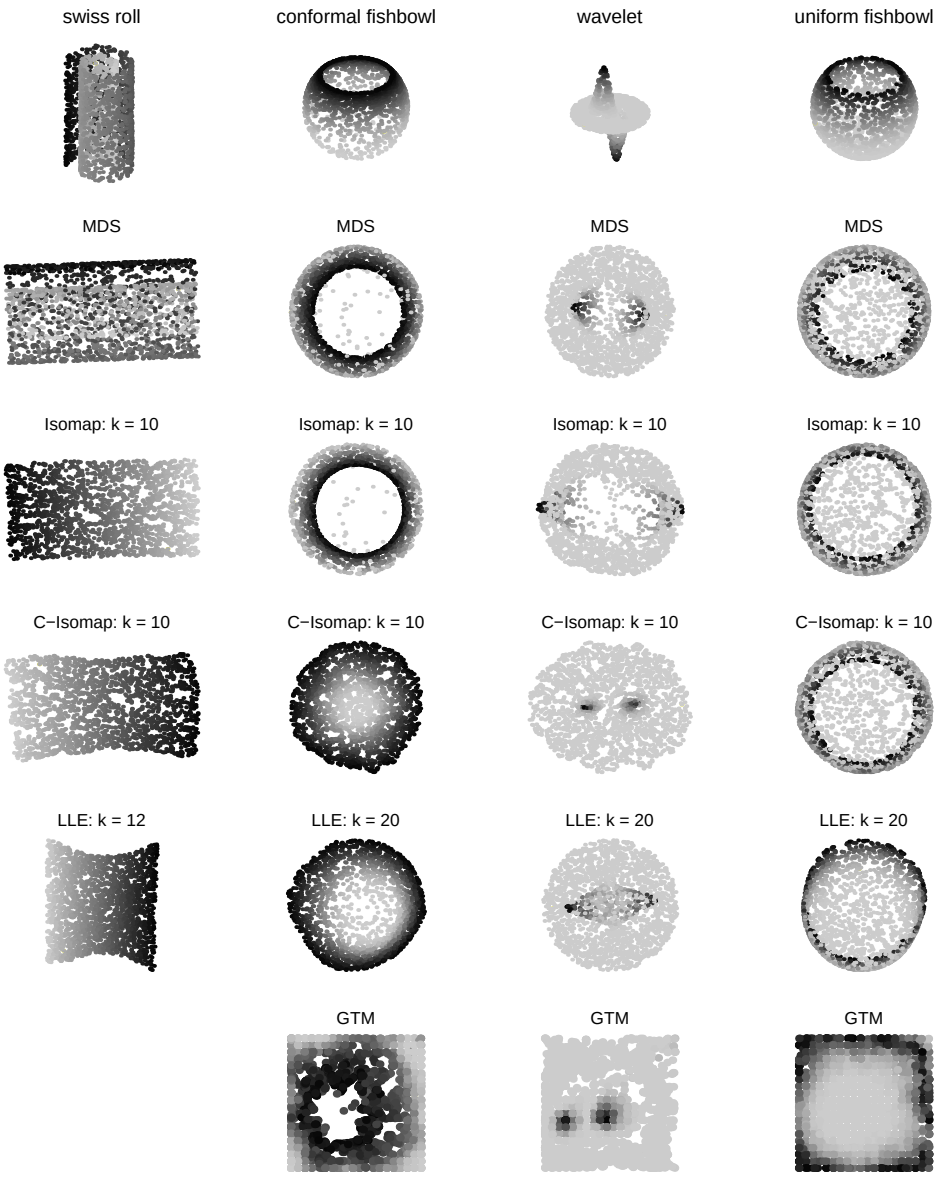

Figure 1.3. Applying MDS, Isomap, C-Isomap, LLE and GTM to four different data sets.