Advanced Data Analytics
Lecture week 1: Overview

Ian T. Nabney

University of BRISTOL

Topics for this week

- The dimensionality reduction methods we shall study this week are all concerned with the topography of data: the similarities and dissimilarities between data points.
- The fundamental principle of the techniques we discuss is to make some or all of the dissimilarities between projected points match as closely as possible to the dissimilarities of the corresponding original points.
- We shall cover the following models:
  - Neuroscale (and the radial basis function neural network);
  - Stochastic neighbourhood embedding;
  - UMAP
- Along the way, we will discuss a few other techniques in passing.

University of BRISTOL

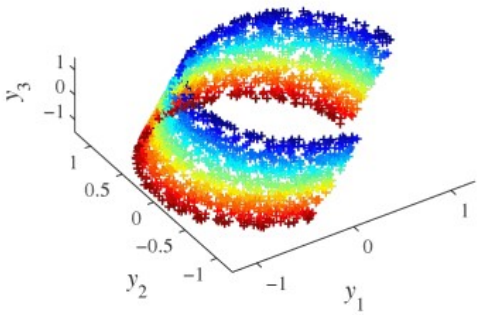
Reading

There is no single source for the topics covered this week, but the following books and papers are useful if you want more context.

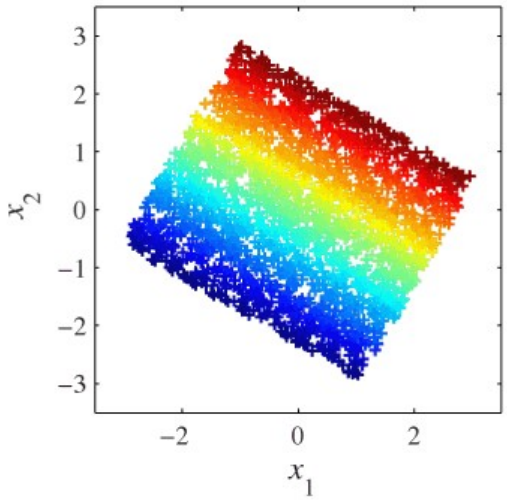
- John A Lee and Michel Verleysen. *Nonlinear Dimensionality Reduction. Information Science and Statistics.* Springer, 2007.
- Nabney, Ian T. *Netlab: Algorithms for Pattern Recognition.* Springer, 2002.
- Hinton, Geoffrey, and Sam T. Roweis. "Stochastic neighbor embedding." NIPS. Vol. 15. 2002.
- Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9.11, 2008.
- McInnes, Leland, John Healy, and James Melville. "Umap: Uniform manifold approximation and projection for dimension reduction." arXiv preprint arXiv:1802.03426, 2018.

University of BRISTOL

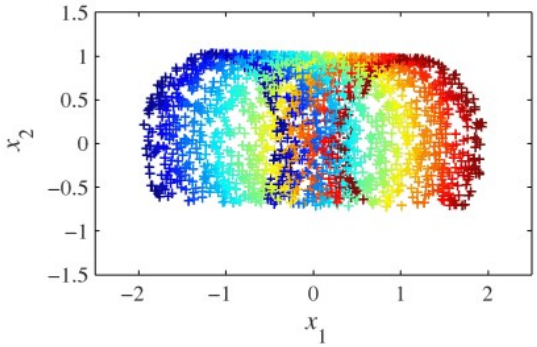
Manifolds



2D manifold embedded in 3D space.



Possible 2D embedding



Dimensionality reduction using PCA. The data does not fit the PCA model, and the initial rectangular distribution cannot be retrieved.

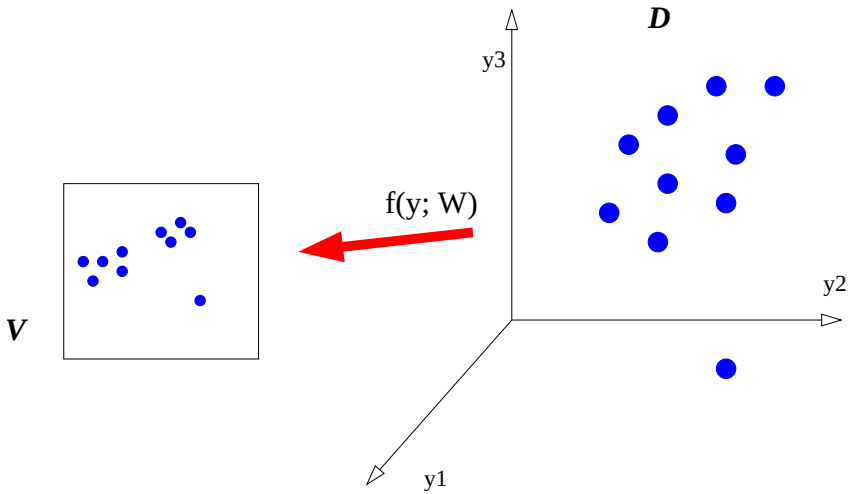
Categorising dimensionality reduction methods

- generative vs. non-generative model
- linear vs. non-linear
- continuous vs. discrete
- type of criterion to be optimised
- implicit vs. explicit mapping
- integrated vs. external estimation of dimensionality
- batch vs. online algorithm

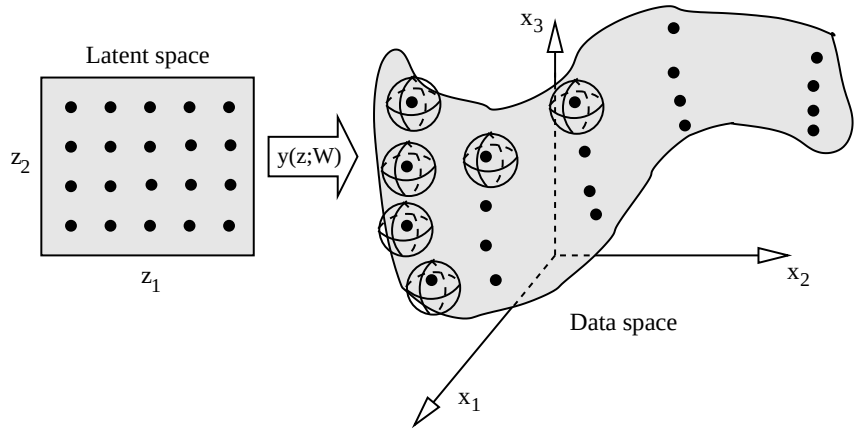
University of BRISTOL

Generative vs. non-generative models

- The model connects the latent variables with the observed variables
- This connection can go in either direction:
  - A generative approach models the observed variables as a function of the unknown latent variables. Also need an algorithm to invert the model.
  - A non-generative approach is simply a mapping from the observed space to the latent space.



Non-generative projection



Generative model

Distance Preservation

- Distance preservation is a non-generative way to perform dimensionality reduction.
- This criterion does not need an explicit model: we can simply identify points in the projection space.
- The intuition is that any manifold can be described by pairwise distances.
- It is clear that if close points are kept close, and if far points remain far, then the initial manifold and its low-dimensional embedding share the same shape.
- Several ways to define distance:

$$\|\mathbf{a}\|_p = \sqrt[p]{\sum_{k=1}^{D} |a_k|^p}$$

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_p = \|\mathbf{b} - \mathbf{a}\|_p$$

- Maximum distance ($p = \infty$): $\|\mathbf{a} - \mathbf{b}\|_\infty = \max_{1 \leq k \leq D} |a_k - b_k|$
- City-block distance ($p = 1$): $\|\mathbf{a} - \mathbf{b}\|_1 = \sum_{k=1}^{D} |a_k - b_k|$
- Euclidean distance ($p = 2$): $\|\mathbf{a} - \mathbf{b}\|_2 = \sqrt{\sum_{k=1}^{D} (a_k - b_k)^2}$

University of BRISTOL

Distance-preserving algorithms

- It is not essential that the same distance measure is used in both spaces. But we do see with 'Euclidean eyes'.
- If we use Euclidean distance in both original space and projection space, and use a linear projection, the optimal mapping is PCA (again!)
- So, what happens if we allow the mapping to be non-linear?
- Given a dissimilarity matrix d_{ij}, we want to map data points $\mathbf{x}_i \in \mathbb{R}^D$ to points $\mathbf{y}_i \in \mathbb{R}^Q$ in a feature space such that their dissimilarities in feature space, $\tilde{d}_{ij}$, are as close as possible to the d_{ij}. We say that the map preserves similarities.
- The stress measure is used as objective function

$$E = \frac{1}{\sum_{ij} d_{ij}} \sum_{i<j} \frac{\left(d_{ij} - \tilde{d}_{ij}\right)^2}{d_{ij}}$$

Incorporating class information

- One useful change can be made to the distance measure if some additional dissimilarity information is known about the data.
- If each data point $\mathbf{x}_i$ belongs to a known class $\mathcal{C}_i$, a dissimilarity measure s_{ij} can be defined by

$$s_{ij} = \begin{cases} 0 & \text{if } i = j \\ 1 & \text{if } i \neq j. \end{cases}$$

  This can then be incorporated into the distance measure:

$$\delta_{ij} = (1 - \alpha) d^*_{ij} + \alpha s_{ij},$$

  where the parameter $\alpha \in [0, 1]$ controls the degree of supervisory information in the mapping.
- When $\alpha = 0$, this is the original definition, while if $\alpha = 1$, there is no distance information and points will be projected using only the class information.
- Intermediate values of α allow the points to be projected so as to retain the distance structure with extra separation from the classes. This can help to expose inconsistencies such as incorrectly labelled data.

University of BRISTOL

Sammon's non-linear mapping

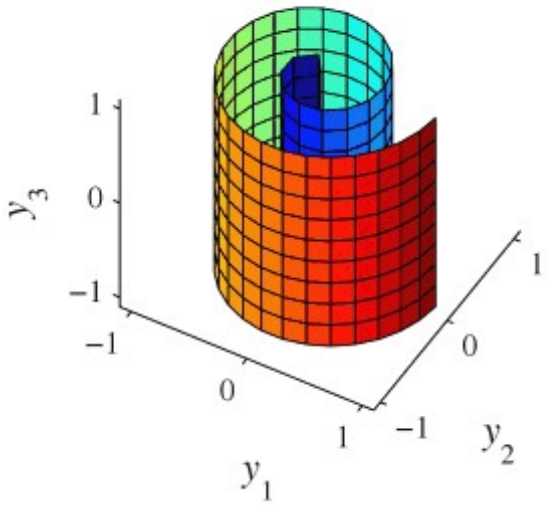
- Sammon's method does not determine a continuous mapping between two spaces: instead it finds the images of a finite set of data points.
- This means that it does not 'generalise'to new data.
- The stress measure is non-negative and is zero iff $d_{ij} = \tilde{d}_{ij}$ for all i, j.
- Unlike PCA, there is no closed form or general solution for this: have to apply general non-linear optimisation algorithm to the points $\mathbf{y}_i$.
- We will cover such algorithms (briefly) later in the unit. Sammon's original paper used the quasi-Newton algorithm. This can be applied because it is possible to compute

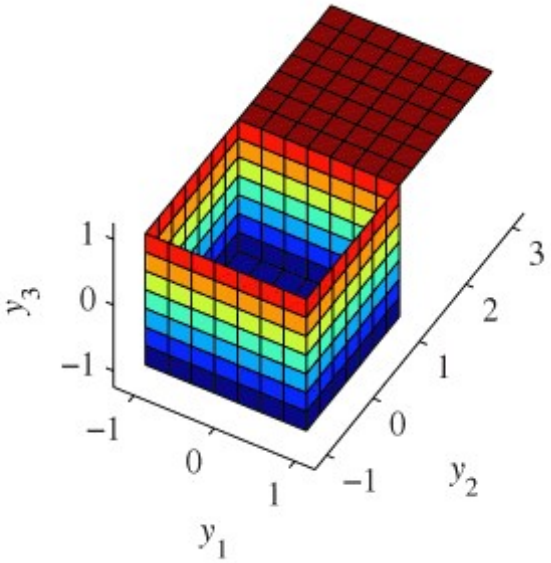
$$\frac{\partial E}{\partial \mathbf{y}_i}$$

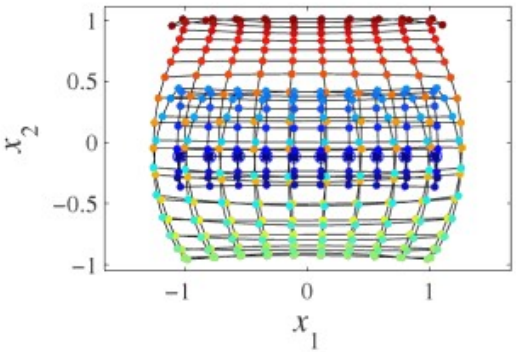
- Note that as we are optimising the location of the points $\mathbf{y}_i$ the number of parameters to optimise is NQ. So the method scales very poorly with larger datasets. Our normal solution to this challenge would be to train a model on a selection of data and then project the rest of the data using the model: but we can't do this for Sammon mapping!

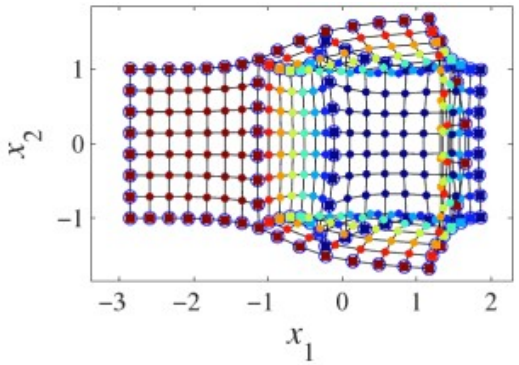
University of BRISTOL

Example mappings









University of BRISTOL

Advantages and disadvantages

- Sammon's mapping can handle non-linear manifolds if they are not too heavily folded.
- It lacks the ability to generalise to new points.
- Computing the complete matrix of distances requires $O(N^2)$ entries, which is a significant memory resource.
- The optimisation process may be slow for large datasets (remember its scaling).
- The stress function is not guaranteed to be concave, so the optimisation can get stuck in a local minimum.