Melbourne Bioinformatics
BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY

THE UNIVERSITY OF MELBOURNE

COMP90014

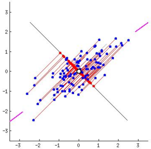
Algorithms for Bioinformatics

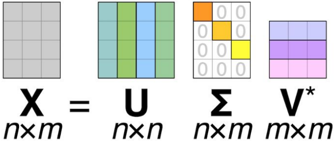
Week 9B: Dimensionality Reduction II

Dimensionality Reduction

1. Dimensionality Reduction
2. Multi-dimensional scaling (MDS)
3. ISOMAP
4. t-SNE
5. UMAP

Linear projections: PCA and SVD





PCA: Finds orthogonal vectors (components) to represent as much variance is as possible. Decomposing the covariance matrix, $C = W \Sigma W^T$. See this animation.

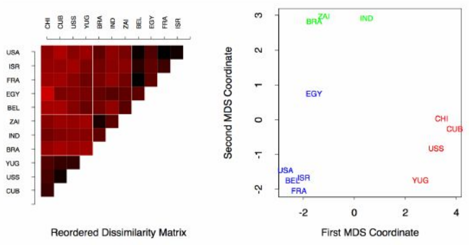
SVD: More efficient
Can interpret PCA as a SVD on the centered covariance matrix

- 🍇 StatQuest: Principal Component Analysis (20 minute video from Josh Starmer)

amoeba via stackexchange; Cmglee via Wikimedia Commons

Dimensionality Reduction

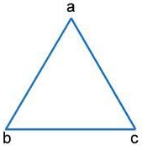
Multi-Dimensional Scaling (MDS)



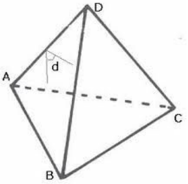
- feature extraction
- produces a lower-dimensional representation that ***preserves pairwise distances or dissimilarities***
- for visualising data
- sometimes called Principal Coordinates Analysis, but it's **not the same as PCA**!

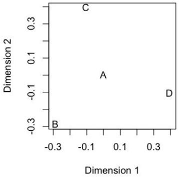
MDS

$$D(x_i, x_j) = \begin{array}{c} \\ a \\ b \\ c \end{array} \begin{array}{c} \begin{array}{ccc} a & b & c \end{array} \\ \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \end{array}$$



$$D(x_i, x_j) = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \end{array}$$





- start with a pairwise distance matrix or dissimilarity matrix
- we can represent three points that are equally-spaced in 3D **exactly in 2D**
- we can represent four points that are equally-spaced in 3D **exactly in 3D** ...
- **... but not in 2D**
- in general, we need $N - 1$ dimensions to exactly represent pairwise distances between N samples

Types of MDS

Metric (distance-based) MDS

- Minimise ***stress***
- Stress is the error in the distances (lack-of-fit measure)
- Try to preserve distance between each vector x_i, x_j

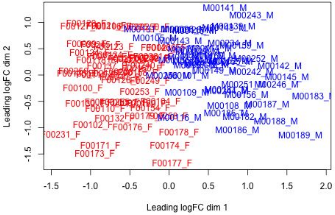
$$\text{Cost} = \sum_{i<j} \left(d_{ij} - \hat{d}_{ij} \right)^2$$

- actual cost/stress used may be more complex

Non-metric MDS

- Try to maintain original ranking of pairwise distances

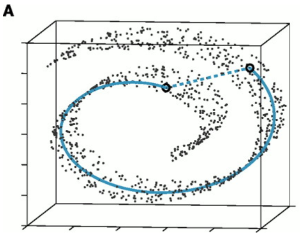
Interpreting low-dimensionality visualisation

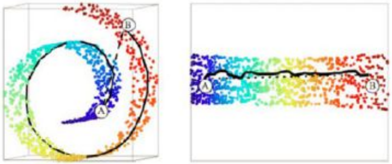


- clusters may represent real clusters
- outliers may represent real outliers
- the position of points is ***not useful*** in MDS
- resulting dimensions are not ordered by importance/variance
- can’t reconstruct original data

Dimensionality Reduction

ISOMAP (Isometric Feature Mapping)





ISOMAP

- nonlinear dimensionality reduction
- preserves the global, non-linear geometry of the data by preserving the ***geodesic*** distances

Manifold: a nonlinear low-dimensional surface

- data often lies on or near manifolds

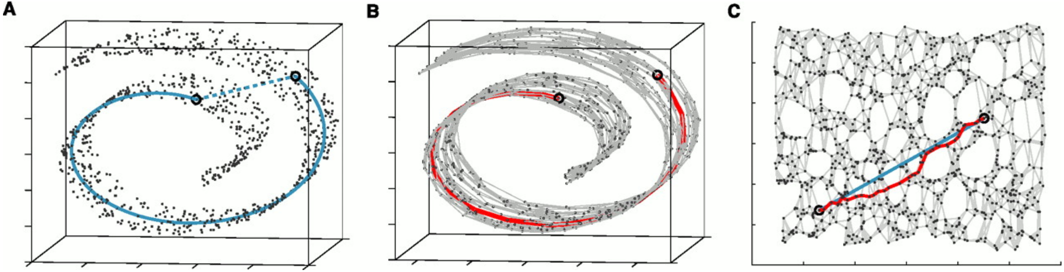
Geodesic: shortest route between two points on the surface of the manifold

- (shortest path on a graph)

Tenenbaum *et al.* 2000. [10.1126/science.290.5500.2319]; J. Jeffry Howbert, [University of Washington].

ISOMAP

- Goal: use geodesic distance between points (with respect to manifold)
- Estimate manifold using graph. Distance between points given by distance of shortest path
- Embed onto 2D plane so that Euclidean distance approximates graph distance

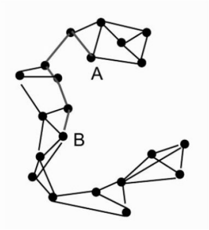


Tenenbaum *et al.* 2000. 10.1126/science.290.5500.2319.

ISOMAP

Two steps:

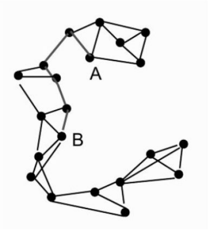
1. Calculate the geodesic distance between every pair of points
   - *e.g.* use Euclidean distance for points that are ‘close enough’
   - For points that are far apart, calculate geodesic distance by summing up local Euclidean distances
2. **Find Euclidean mapping of the data that preserves the geodesic distance**



ISOMAP

Two steps:

1. Calculate the geodesic distance between every pair of points
   - *e.g.* use Euclidean distance for points that are 'close enough'
   - For points that are far apart, calculate geodesic distance by summing up local Euclidean distances
2. **Find Euclidean mapping of the data that preserves the geodesic distance**



1. construct a graph

if $d(i,j) < \varepsilon$ `// ε-isomap`

  or

  i is one of j's k nearest neighbours `// k-isomap`

then

  Connect *i* and *j*

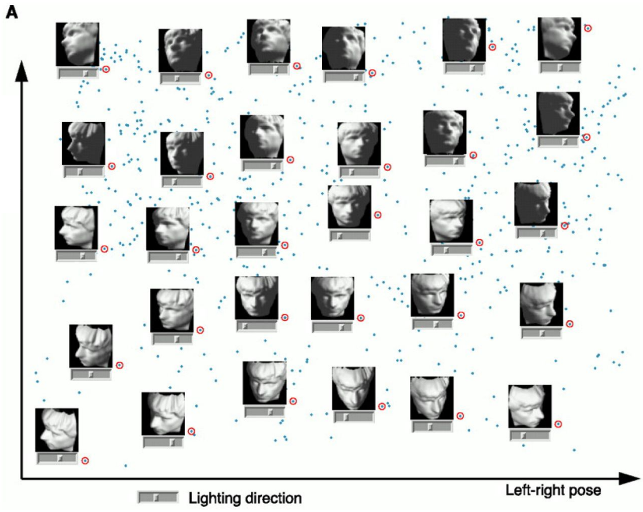
  Set the edge weight as $d(i,j)$ (Euclidean distance)

Compute the geodesic distance between all pairs of points (shortest paths)

Using ISOMAP

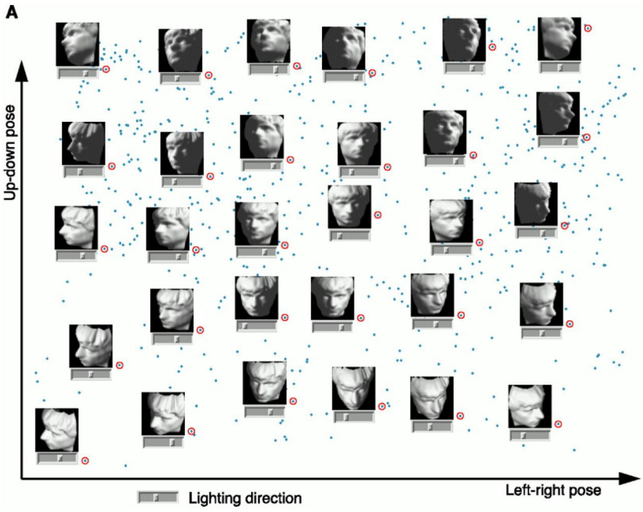
Tenenbaum et al. 2000. https://doi.org/10.1126/science.290.5500.2319

Using ISOMAP



Tenenbaum et al. 2000. https://doi.org/10.1126/science.290.5500.2319

Using ISOMAP

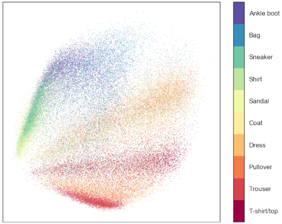


Tenenbaum et al. 2000. https://doi.org/10.1126/science.290.5500.2319

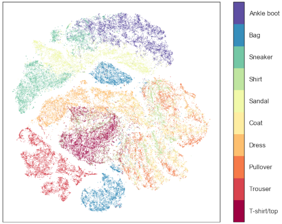
Dimensionality Reduction

Fashion MNIST data set

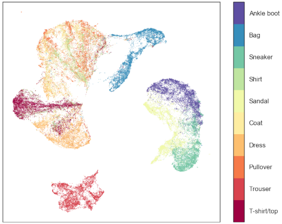
PCA



t-SNE



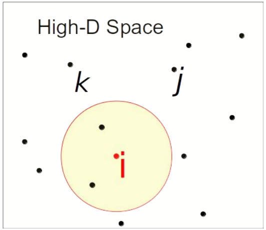
UMAP



- Based on images of fashion items.
- Each item is categorized
- Dataset: [zalandoresearch/fashion-mnist]

Leland McInnes, SciPy 2018.

t-Distributed Stochastic Neighbour Embedding



$$p_{j|i} = \frac{\exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(\frac{-\|x_i - k_k\|^2}{2\sigma_i^2}\right)}$$

PCA: tries to preserve **global** structure

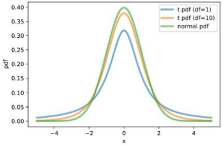
t-SNE: tries to preserve **local** structure

- probabilistic version of MDS
- each point in high-dimensionality has a conditional probability of picking each other point as its neighbor
  - probability of picking *j* given you start at *i*

t-Distributed Stochastic Neighbour Embedding

$$p_{j|i} = \frac{\exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(\frac{-\|x_i - k_k\|^2}{2\sigma_i^2}\right)}$$

$$q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|y_k - y_l\|^2\right)^{-1}}$$

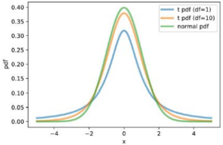


- calculate the Gaussian probabilities ($p_{j|i}$) in high-dimensional space
- compare to a probability in lower-dimensional space

t-Distributed Stochastic Neighbour Embedding

$$p_{j|i} = \frac{\exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k\neq i}\exp\left(\frac{-\|x_i - k_k\|^2}{2\sigma_i^2}\right)}$$

$$q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k\neq l}\left(1 + \|y_k - y_l\|^2\right)^{-1}}$$



- calculate the Gaussian probabilities ($p_{j|i}$) in high-dimensional space
- compare to a probability in lower-dimensional space

Evaluate a mapping:

- in lower-dimensional space, calculate probabilities using Student's t-distribution (q)
- probability goes to zero more slowly than a Gaussian
- this relaxes distances in lower-dimensional space

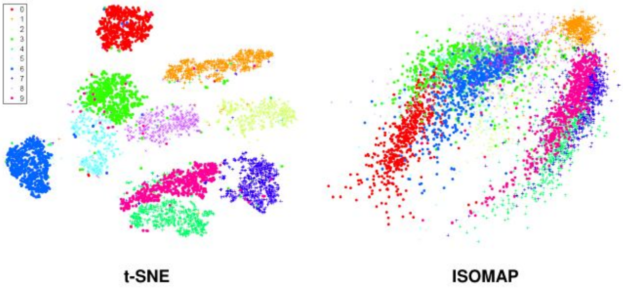
t-SNE cost function

Kullback-Liebler divergence: $C = \mathrm{KL}\left(P \mid\mid Q\right) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$

- measures the difference between the high-dimensional probability distributions and the low-dimensional probability distributions

- t-SNE algorithm performs gradient descent on the cost function

t-SNE vs. ISOMAP



Example on MNIST dataset

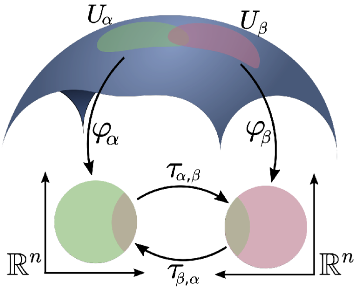
- a classic machine learning dataset of images of handwritten digits

- MNIST dataset: yann.lecun.com/exdb/mnist
- t-SNE publication and code: lvdmaaten.github.io/tsne

Dimensionality Reduction

1. Dimensionality Reduction
2. Multi-dimensional scaling (MDS)
3. ISOMAP
4. t-SNE
5. UMAP

UMAP

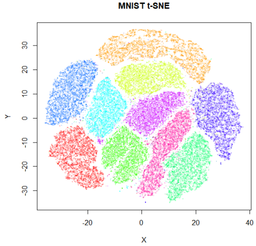


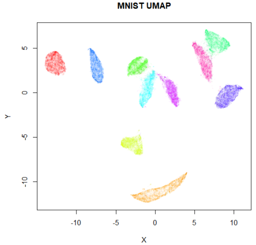
Uniform Manifold Approximation and Projection (UMAP):

- inspired by t-SNE, based on manifolds and topology rather than probability distributions.
- publications and code at lmcinnes/umap
- UMAP talk at SciPy: youtube.com
- UMAP talk at PyData: youtube.com

Leland McInnes, SciPy 2018.

UMAP





James Melville, jlmelville/uwot.

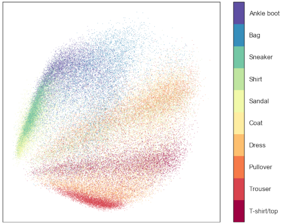
- In the first phase a weighted k-neighbour graph is constructed.
- In the second phase a low dimensional layout of this graph is computed

Force directed graph layout

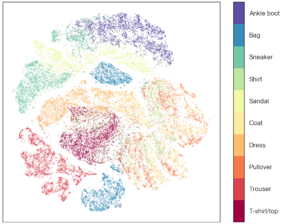
- Edge weights mean that nearby points attract each other
- Weight normalization acts as a global force pushing points apart
- Algorithm: iteratively apply attractive and repulsive forces at each edge or node

Fashion MNIST data set

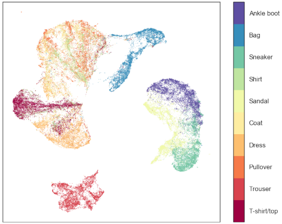
PCA



t-SNE



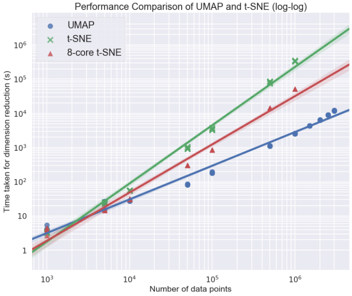
UMAP

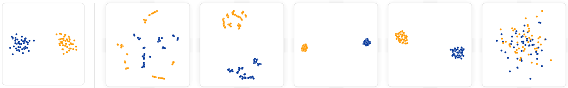


- Based on images of fashion items.
- Each item is categorized
- Dataset: zalandoresearch/fashion-mnist

Leland McInnes, SciPy 2018.

UMAP





Original | Perplexity: 2 Step: 5,000 | Perplexity: 5 Step: 5,000 | Perplexity: 30 Step: 5,000 | Perplexity: 50 Step: 5,000 | Perplexity: 100 Step: 5,000

Advantages

- an order of magnitude faster than t-SNE
- deterministic (t-SNE and MDS are stochastic)

Disadvantages

- distances in the plot are not real distances
- dimensions have no interpretable meaning
- hyperparameters need to be tuned carefully and have a huge impact on the result

McInnes & Healy, 2018. 10.48550/arXiv.1802.03426; Wattenberg *et al.* 2016. via Distill.

Dimensionality Reduction

1. Dimensionality Reduction
2. Multi-dimensional scaling (MDS)
3. ISOMAP
4. t-SNE
5. UMAP

Thank you!

Today: Dimensionality Reduction II

Next time: Unsupervised Learning