Applied Machine Learning
Lecture 9: Dimensionality reduction

Selpi (selpi@chalmers.se)

The slides are further development of Richard Johansson's slides

February 21, 2020

Overview

Why use dimensionality reduction?

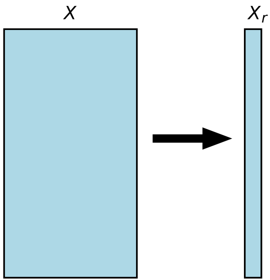
Principal Component Analysis (PCA)

Dealing with non-linear structure

Review of today's lecture

Dimensionality reduction

- reducing a high-dimensional dataset to a low-dimensional one



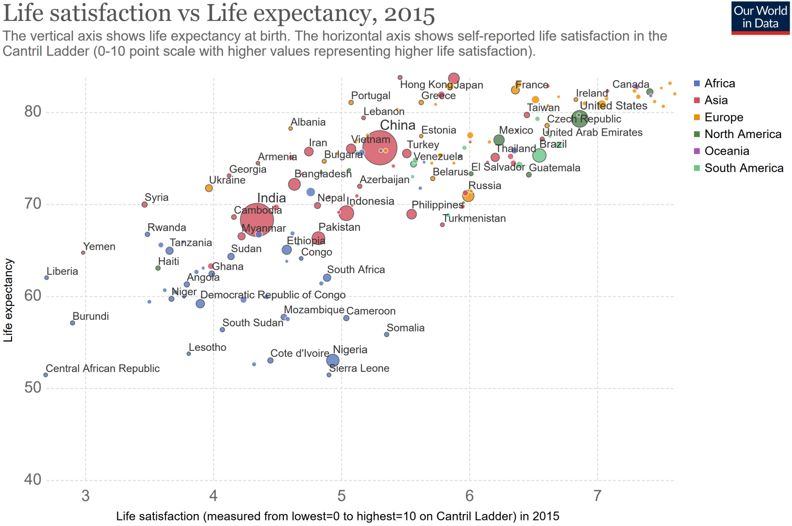
- **why?**
  - to compress data –> reduce storage and memory, make learning easier, make faster computation
  - to enable visualisation –> increase understanding of data
  - to find features for supervised learning

Approaches to dimensionality reduction

- Use feature selection (i.e., use only important features)
- Map existing features to smaller number of **new** features
  - By projecting or using linear combination
  - By using non-linear combination

Life satisfaction vs Life expectancy, 2015

The vertical axis shows life expectancy at birth. The horizontal axis shows self-reported life satisfaction in the Cantril Ladder (0-10 point scale with higher values representing higher life satisfaction).



Source: UN Population Division (2017 Revision), World Happiness Report (2019)

Overview

Why use dimensionality reduction?

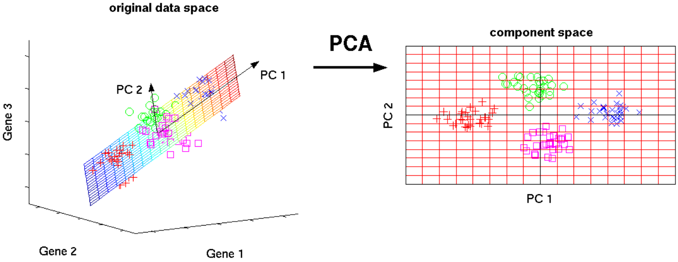
Principal Component Analysis (PCA)

Dealing with non-linear structure

Review of today's lecture

Principal Component Analysis

- Reduce data from n-dimension to k-dimension (with $k < n$): find k vector(s) onto which data can be projected and the projection error is minimised.
- The principal components (PCs) are statistically uncorrelated. PC1 captures the biggest variance in the data, PC2 captures the biggest variance in the data among the remaining data after PC1, and so on...



[source]

Before applying PCA/any other dimensionality reduction methods

- Normalise the values of each feature so that each feature has zero mean
- Scale features if different features have different ranges of values

The drawback of PCA

- Compared to the original features, the PCs are difficult to interpret.
  - Original features: age, gender, income, education, ...
  - New features: PC1, PC2, ...
- May miss non-linear structure in the data.

Overview

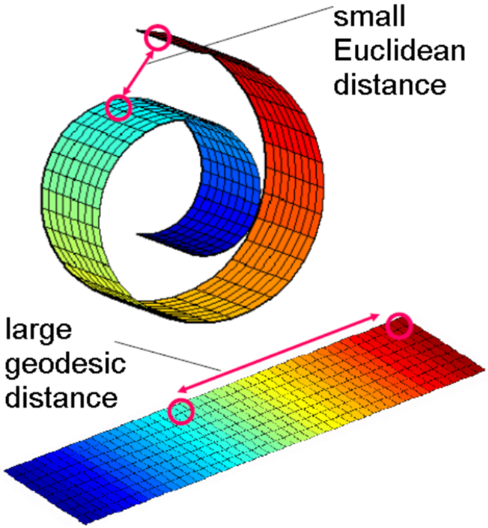
Why use dimensionality reduction?

Principal Component Analysis (PCA)

Dealing with non-linear structure

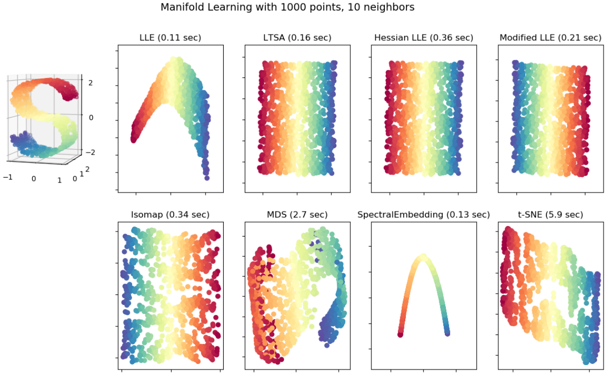
Review of today's lecture

Example of non-linear structure: the "Swiss roll" dataset



[source]

manifold learning: nonlinear dimensionality reduction



- see `https://scikit-learn.org/stable/modules/manifold.html`

t-distributed Stochastic Neighborhood Embedding

- t-SNE is a **nonlinear** dimensionality reduction method
- key idea: preserve **neighborhood** structure in reduced space

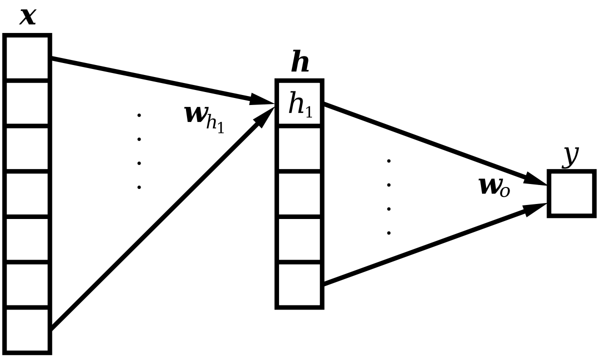
t-SNE: modeling probabilities of neighbors

- calculates the probability closeness of points in high dimensional space as well as in low dimensional space and then minimises the difference between both probabilities
- van der Maaten et al. (2008) *Visualizing Data using t-SNE*

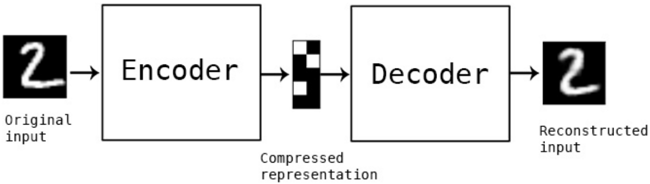
$$p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_i - x_k\|^2 / 2\sigma_i^2\right)}$$

$$q_{j|i} = \frac{\exp\left(-\|y_i - y_j\|^2\right)}{\sum_{k \neq i} \exp\left(-\|y_i - y_k\|^2\right)}$$

recall: neural networks do dimensionality reduction internally



autoencoders: dimensionality reduction using a NN



- for several Keras implementations, see

  `https://blog.keras.io/building-autoencoders-in-keras.html`

Autoencoders NN

- Specific for the type of data it is trained on.
- Only need an encoding function, a decoding function, and a distance function
- Can be used together with t-SNE to visualise very high-dimensional data.

  `https://blog.keras.io/building-autoencoders-in-keras.html`

Overview

Why use dimensionality reduction?

Principal Component Analysis (PCA)

Dealing with non-linear structure

Review of today's lecture

Review of today's lecture

- Explain the idea of PCA.
- Discuss the difference between PCA and linear regression!
- What might be the motivation for combining autoencoder NN with t-SNE?

Next lecture

- Unsupervised learning (e.g., clustering, anomaly detection)