MASTER IN ENTREPRENEURSHIP INNOVATION MANAGEMENT
IN COLLABORATION WITH MIT SLOAN

IN COLLABORATION WITH
MIT MANAGEMENT SLOAN SCHOOL

UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

MASTER MEIM 2022-2023

Digital AI
Unsupervised techniques: Dimensionality Reduction

LECTURE 5

prof. Antonino Staiano

M.Sc. In Applied Computer Science of University Parthenope of Naples

MASTER IN ENTREPRENEURSHIP
INNOVATION MANAGEMENT
IN COLLABORATION WITH MIT SLOAN

IN COLLABORATION WITH
MIT MANAGEMENT SLOAN SCHOOL

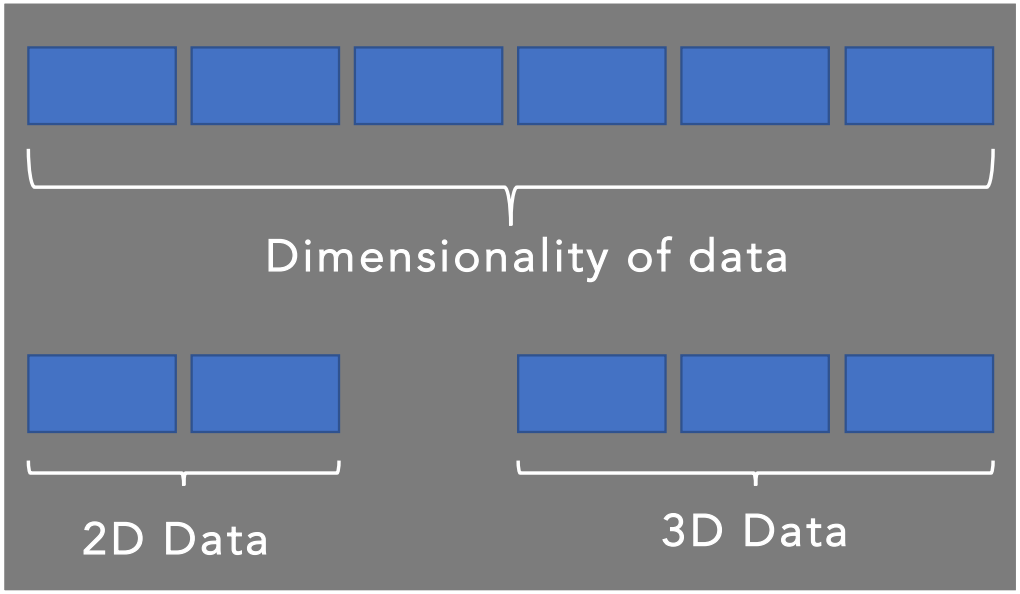
UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

Unsupervised Learning

Dimensionality Reduction

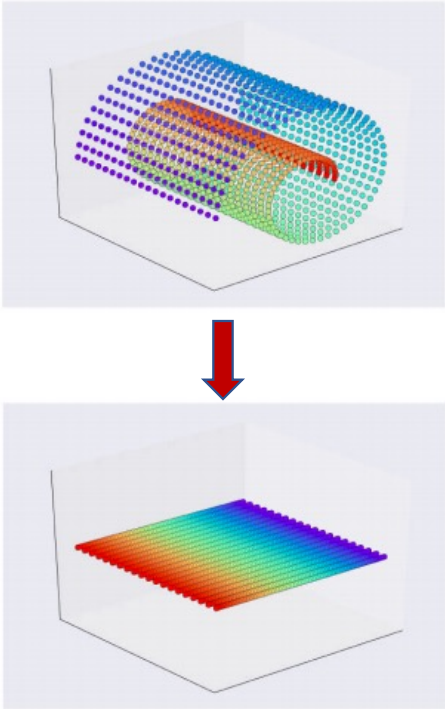
Data dimensionality

- The number of features in a dataset determines its data dimensionality



Unsupervised learning: Dimensionality reduction

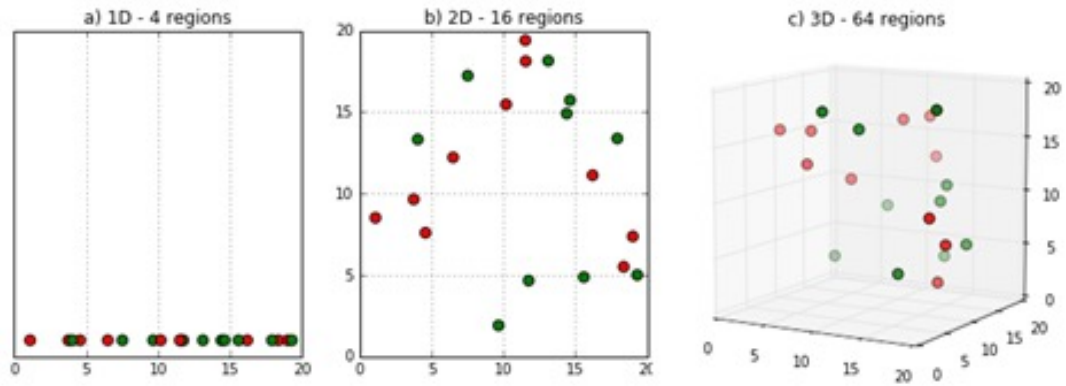
- The data used in machine learning applications often have many variables (features)
  - If your dataset has two features, then it is two-dimensional data. If it has three features, then it has three features and so on
- One aims at using as many features as possible to capture the characteristics of her data, but she also doesn't want the dimension to be too high
- Most of these dimensions may or may not matter in the context of our application with the questions we are asking
- Reducing such high dimensions to a more manageable set of related and useful variables improves the performance and accuracy of our analysis



Curse of Dimensionality

- Data in only one dimension is relatively packed
- Adding a dimension "stretch" the points across that dimension, making them further apart
- Adding more dimensions will make the points further apart – high dimensional data is extremely sparse
- Distance measure becomes meaningless – due to equidistance

Data gets increasingly sparse

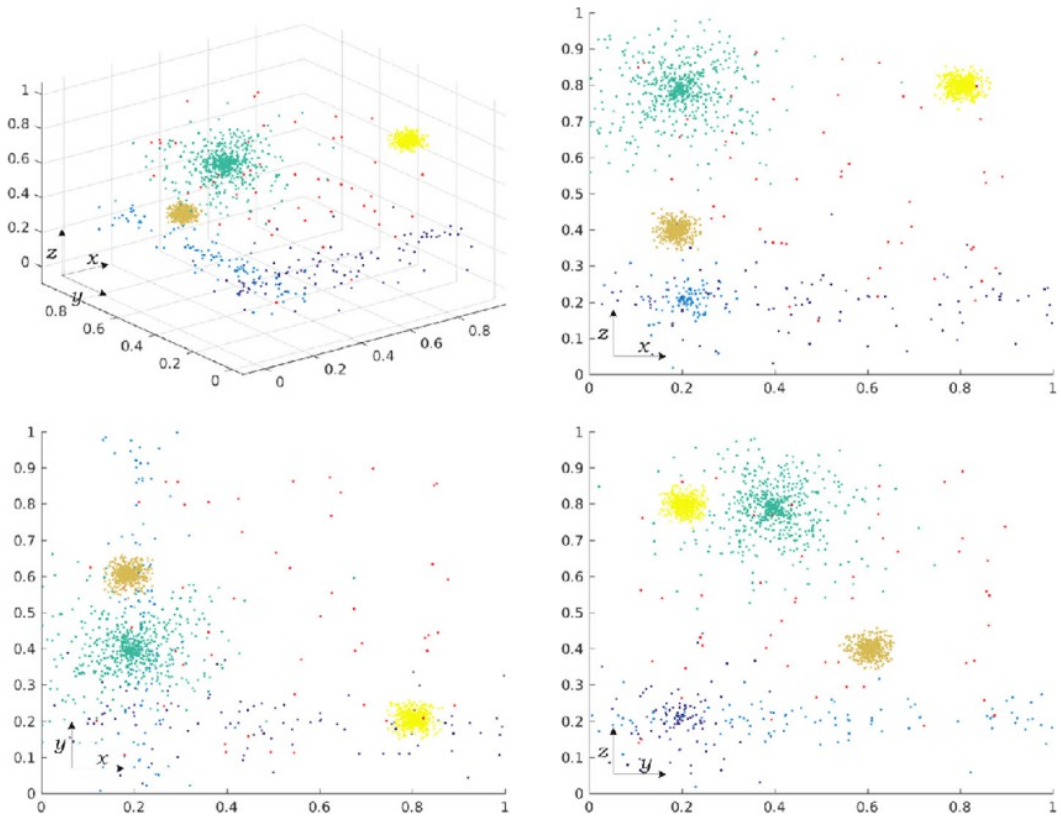


Analysis results degrade

Dimensionality reduction

3D -> 2D

- Data visualization is a further significant motivation behind dimensionality reduction



Goals of data dimensionality reduction

- Preserve as much significant structure or information of the data present in the high-dimensional data as possible in the low-dimensional representation
- Increase the interpretability of the data in the lower dimension
- Minimizing information loss of data due to dimensionality reduction

MASTER IN ENTREPRENEURSHIP
INNOVATION MANAGEMENT
IN COLLABORATION WITH MIT SLOAN

IN COLLABORATION WITH
MIT MANAGEMENT SLOAN SCHOOL

UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

Dimensionality reduction

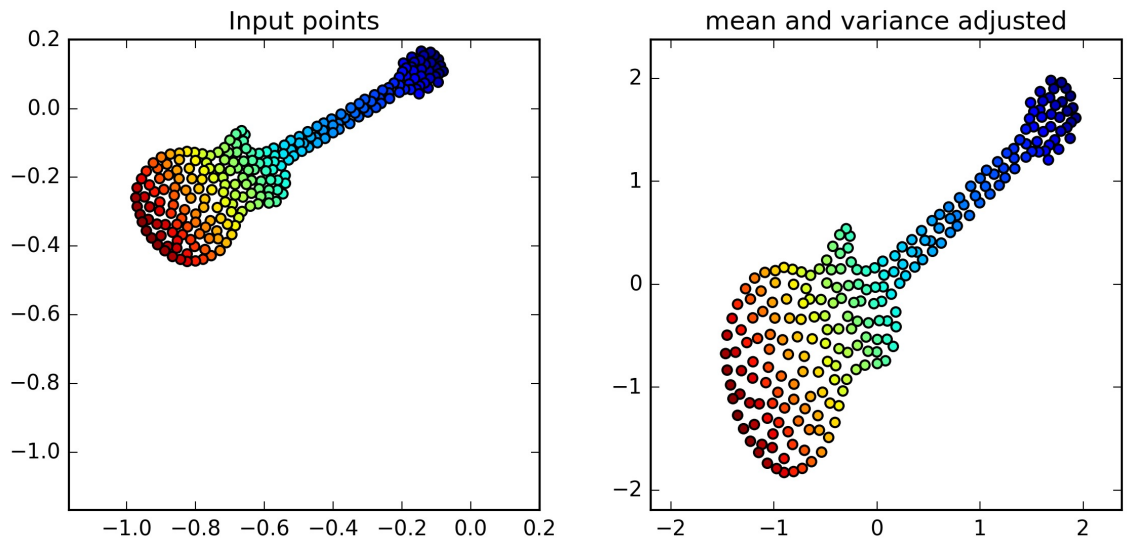
Principal Component Analysis

Principal Component Analysis (PCA)

- An unsupervised, deterministic algorithm used for feature extraction as well as visualization
- Applies a linear dimensionality reduction technique where the focus is on keeping the dissimilar points far apart in a lower-dimensional space
- Transforms the original data to a new data by preserving their variance using eigenvalues
- Outliers impact PCA

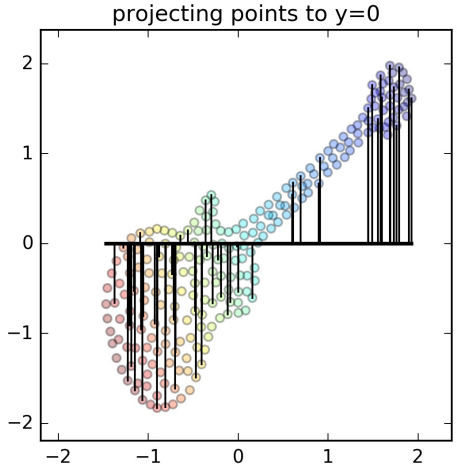
Principal component analysis (PCA)

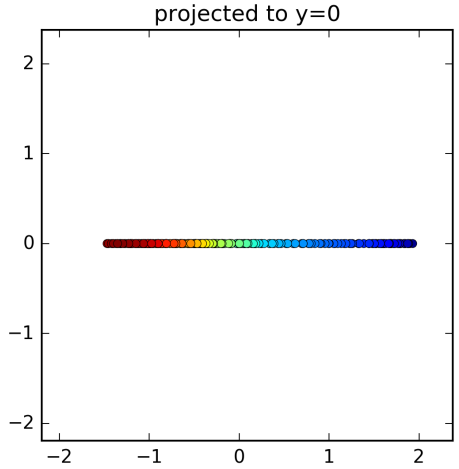
- PCA is a mathematical technique for reducing the dimensionality of data
- Goal
  - To reduce high-dimensional to low-dimensional data in some way



Principal component analysis (PCA)

- Let's draw a horizontal line on the X axis
  - Projection line
- Project each data point to its closest spot on the projection line

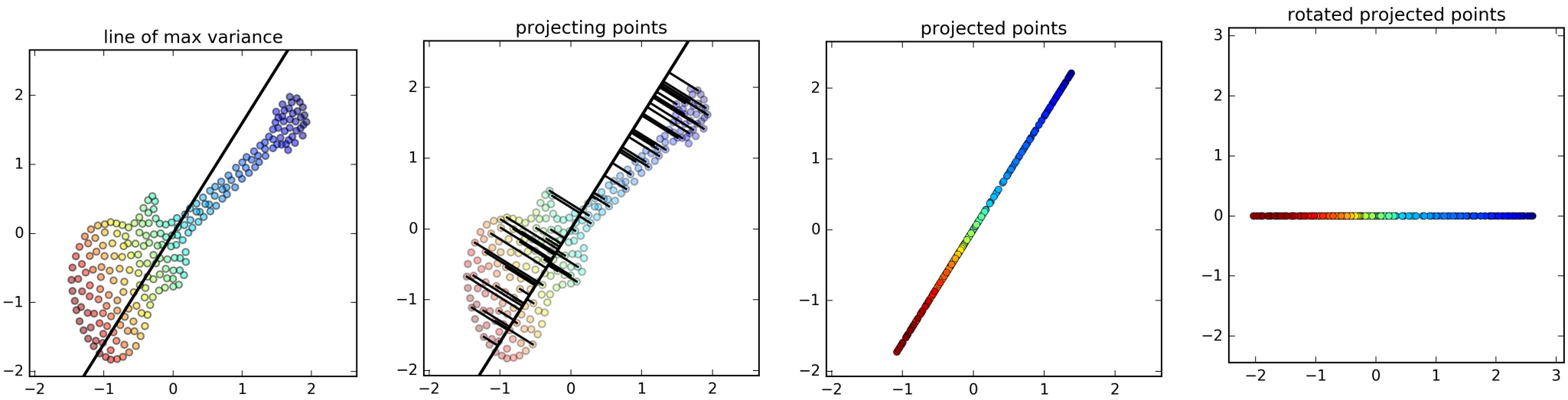




- That's the wrong way to proceed!

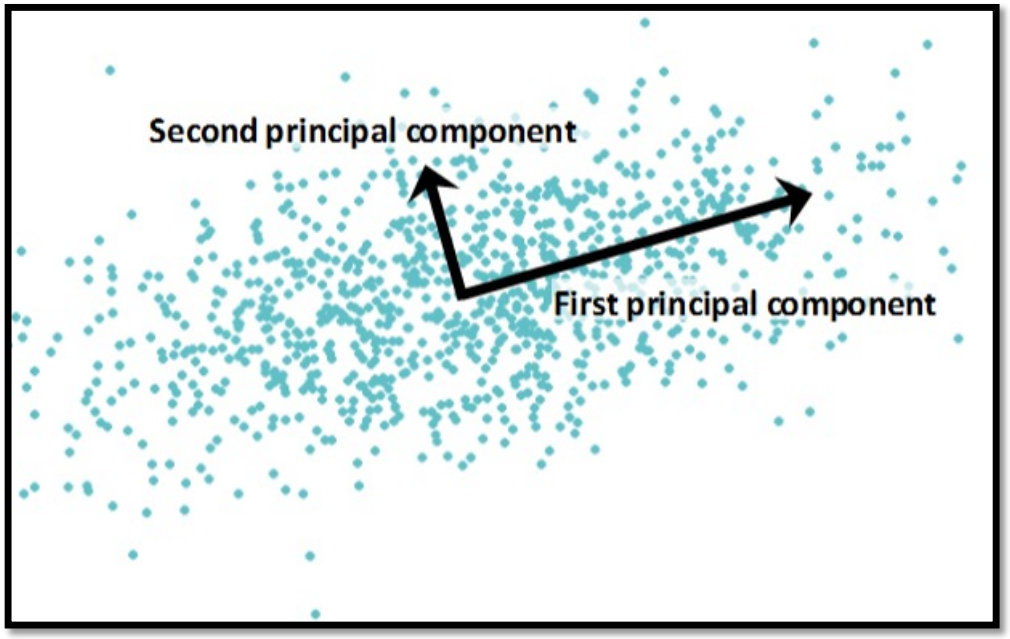
Principal component analysis (PCA)

- Here is how PCA proceeds



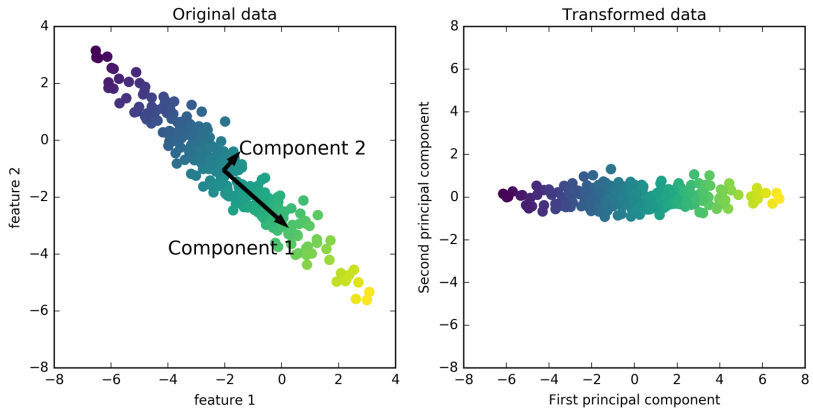
Principal Component Analysis

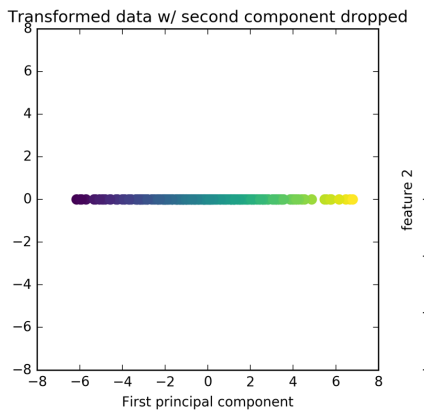
- Finds a new coordinate system such that few new axes captures the greatest variance
- Define lower-dimensional space for data
- Note
  - Original dimensions have a natural interpretation
    - E.g., Income, age, occupation, etc
  - New dimensions more difficult to interpret!
  - In general, there are as many principal components as original features



PCA algorithm main steps

- Starts by first finding the direction of maximum variance (Component 1)
  - This is the direction (or vector) in the data that contains most of the information, or in other words, the direction along which the features are most correlated with each other
- Next, it finds the direction that contains the most information while being orthogonal to the first direction (Component 2)
  - In two dimensions, there is only one possible orientation, but in higher-dimensional spaces there would be (infinitely) many orthogonal directions
- The directions found using this process are called principal components, as they are the main directions of variance in the data





MASTER IN ENTREPRENEURSHIP
INNOVATION MANAGEMENT
IN COLLABORATION WITH MIT SLOAN

IN COLLABORATION WITH
MIT MANAGEMENT SLOAN SCHOOL

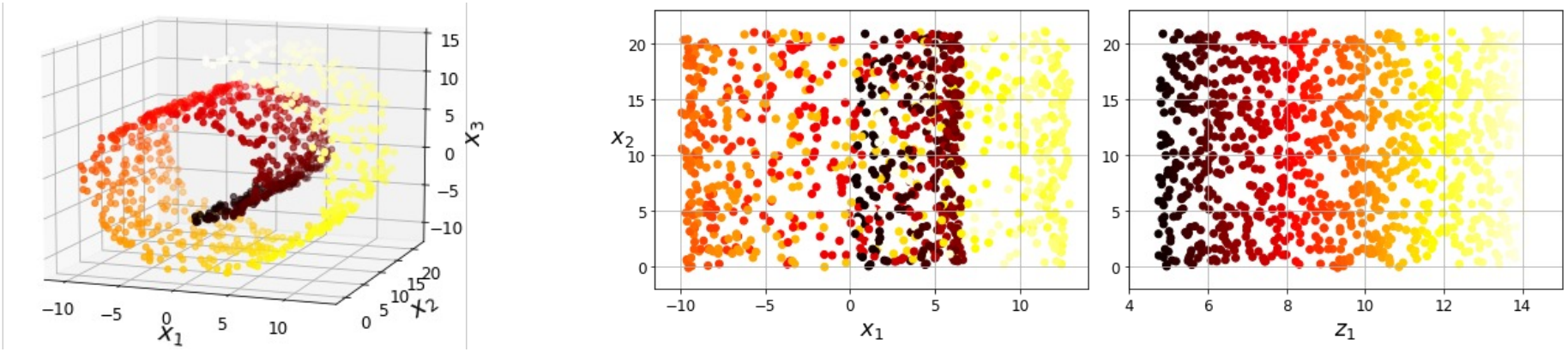
UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

Dimensionality reduction

Manifold Learning

Why Manifold learning ?

- Many dimensionality reduction algorithms work by modeling the manifold on which the training instances lie (manifold learning)
- It relies on the manifold assumption stating that most real-world high-dimensional datasets lie close to a much lower dimensional manifold



Manifold learning

- PCA is often a good first approach for transforming your data so that you might be able to visualize it using a scatter plot
- There is a class of algorithms for visualization called *manifold learning algorithms* that allow for much more complex mappings, and often provide better visualizations
- A particularly useful one is the t-SNE algorithm

t-Distributed Stochastic Neighbor Embedding(t-SNE)

- An unsupervised, randomized algorithm, used only for visualization
- Applies a non-linear dimensionality reduction technique where the focus is on keeping the very similar data points close together in lower-dimensional space
- Preserves the local structure of the data using student t-distribution to compute the similarity between two points in lower-dimensional space
  - Used for visualization purposes because it exploits the local relationships between datapoints and can subsequently capture nonlinear structures in the data
- Outliers do not impact t-SNE

t-SNE algorithm

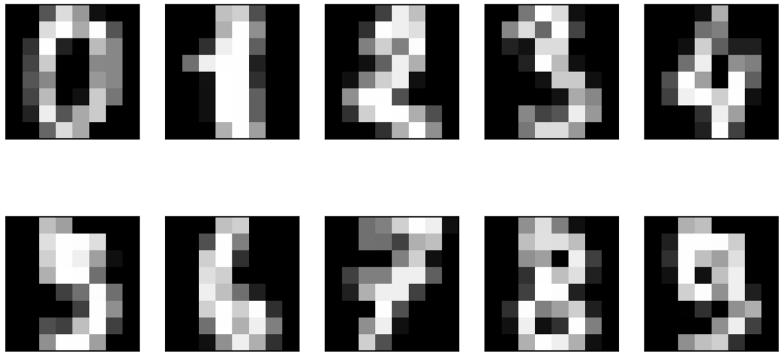
- Manifold learning algorithms are mainly aimed at visualization
  - rarely used for data transformation
- Manifold learning can be useful for exploratory data analysis but is rarely used if the final goal is supervised learning
- The idea behind t-SNE is to find a two-dimensional representation of the data that preserves the distances between points as best as possible
  - it starts with a random two-dimensional representation for each data point
  - then tries to make points that are close in the original feature space closer, and points that are far apart in the original feature space farther apart
- t-SNE puts more emphasis on points that are close by, rather than preserving distances between far-apart points
  - it tries to preserve the information indicating which points are neighbors to each other

t-SNE main steps

- The t-SNE algorithm can be roughly summarized as two steps:
  1. Create a probability distribution capturing the relationships between points in the high dimensional space
  2. Find a low dimensional space that resembles the probability distribution as well as possible

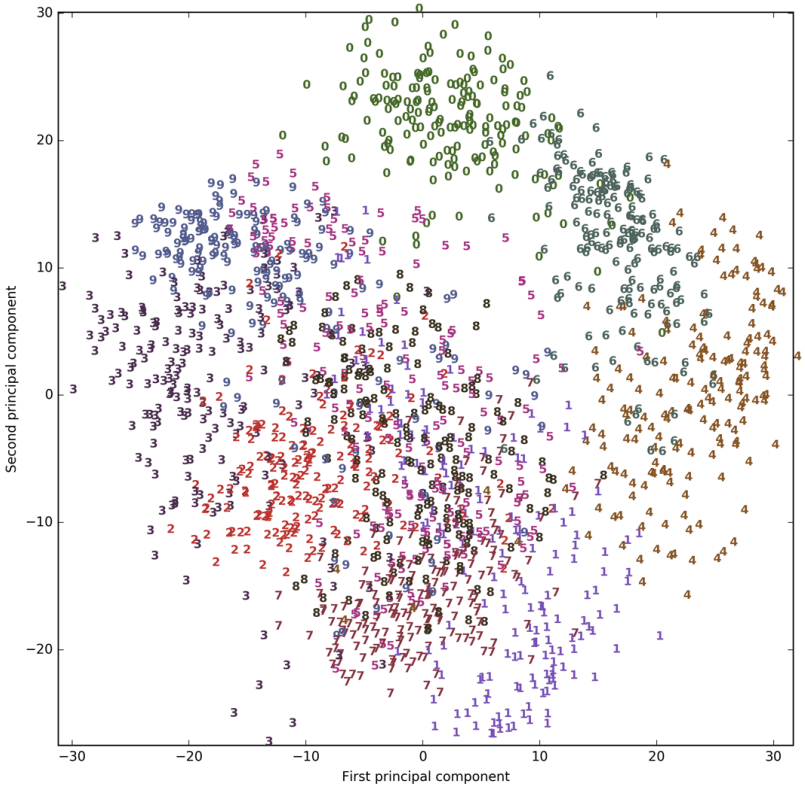
Visualization example

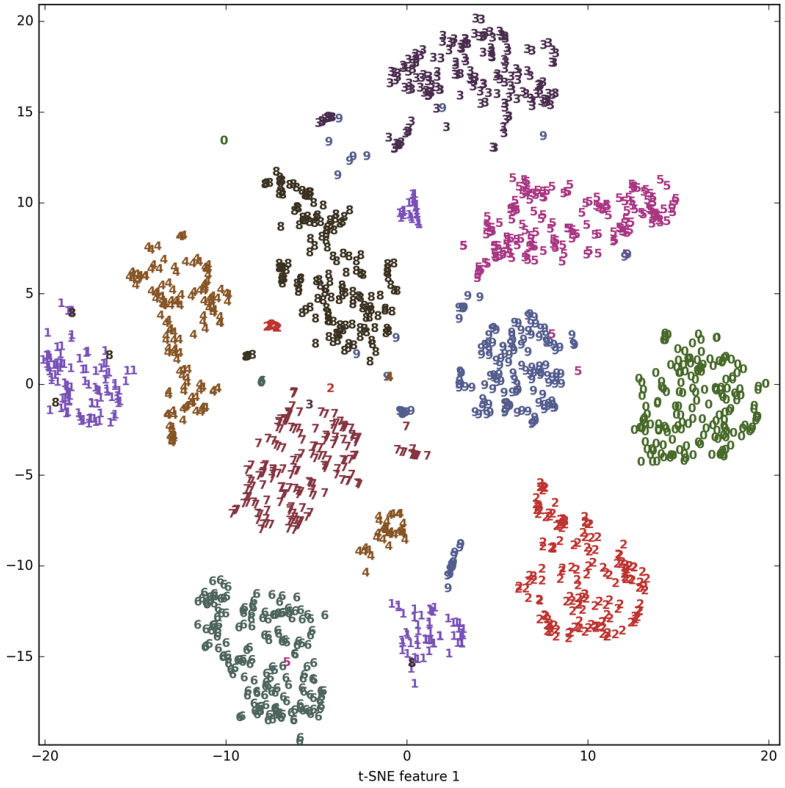
- Let's apply the t-SNE manifold learning algorithm on a dataset of handwritten digits
- Each data point is an 8×8 gray-scale image of a handwritten digit between 0 and 1
- Let's use PCA to visualize the data reduced to two dimensions
  - We plot the first two principal components, and color each dot by its class
- Let's also use t-SNE



Example images from handwritten digit dataset

PCA vs t-SNE


Second principal component
First principal component


t-SNE feature 1