Lecture 15. Unsupervised Learning

COMP90051 Statistical Machine Learning

Semester 2, 2015
Lecturer: Andrey Kan

THE UNIVERSITY OF MELBOURNE

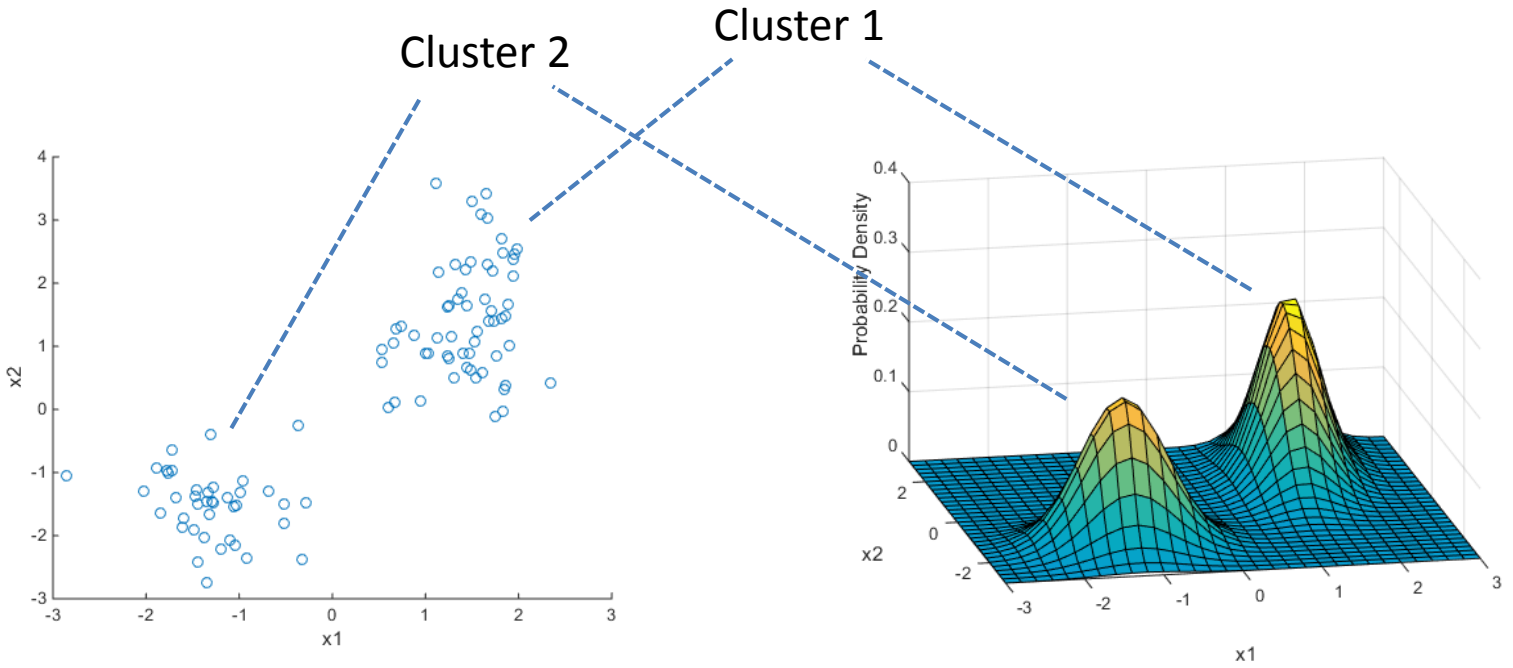
Supervised vs unsupervised learning

Supervised	Unsupervised
Each training example is a combination of features and labels	a set of data points (not distinguishing between "predictors" and "labels")
Application is to predict labels from features	Applications: clustering, association rules, identifying correlations
Given samples $[x_1, \dots, x_n, y_1, \dots, y_n]$, learn properties of $\Pr(Y\|X)$ (discriminative approach)	Given samples $[x_1, \dots, x_n]$, learn properties of joint density $\Pr(X)$ (generative approach)
Usually interested in $\mu(\boldsymbol{x}) = E(Y\|X = \boldsymbol{x})$	E.g., modes of the distribution, covariance matrix

Clustering: probabilistic interpretation

Clustering can be viewed as identification of components in a mixture probability density function

Identifying cluster centroids can be viewed as finding modes of distributions



Association rules: probabilistic interpretation

Given a list of supermarket transactions:
1: beer, chips, water
2: nappies, baby wipes, bread
3: tomatoes, potatoes, beer, chips
…

Identify frequent itemsets, e.g., "beer, chips"

- Each item as a binary random variable (e.g., x_1 for beer, … , x_4 for nappies, etc.)
- Each transaction is a sample with <u>all</u> variables.
- These set to 1 for items in the transaction, and set to 0 for all other items

Frequent itemsets = variables with high joint probability of ones

Data analysis

Hypothesis driven
Formulate a research question/hypothesis
Design and run experiments, collect data that can address hypothesis
Can be very expensive (e.g., Large hadron collider)
See if data supports or contradicts the hypothesis

Exploratory analysis
Data is there
Data is usually free/cheap (e.g., connection logs)
“Fishing expedition”: try to discover facts (trends, patterns, clusters)
See if discovered information is useful

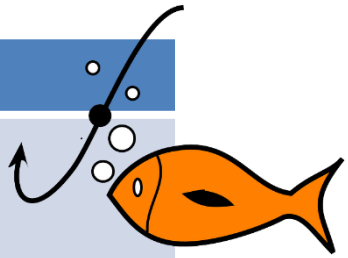
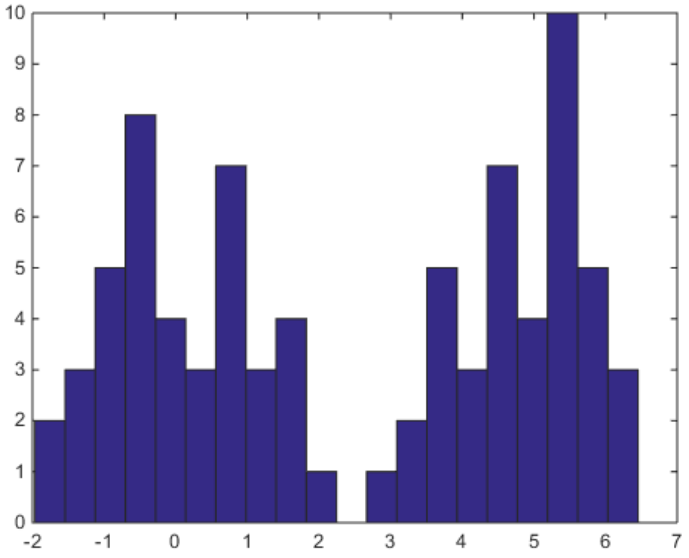


Image: ClkerFreeVectorImages@pixabay.com (CC0)

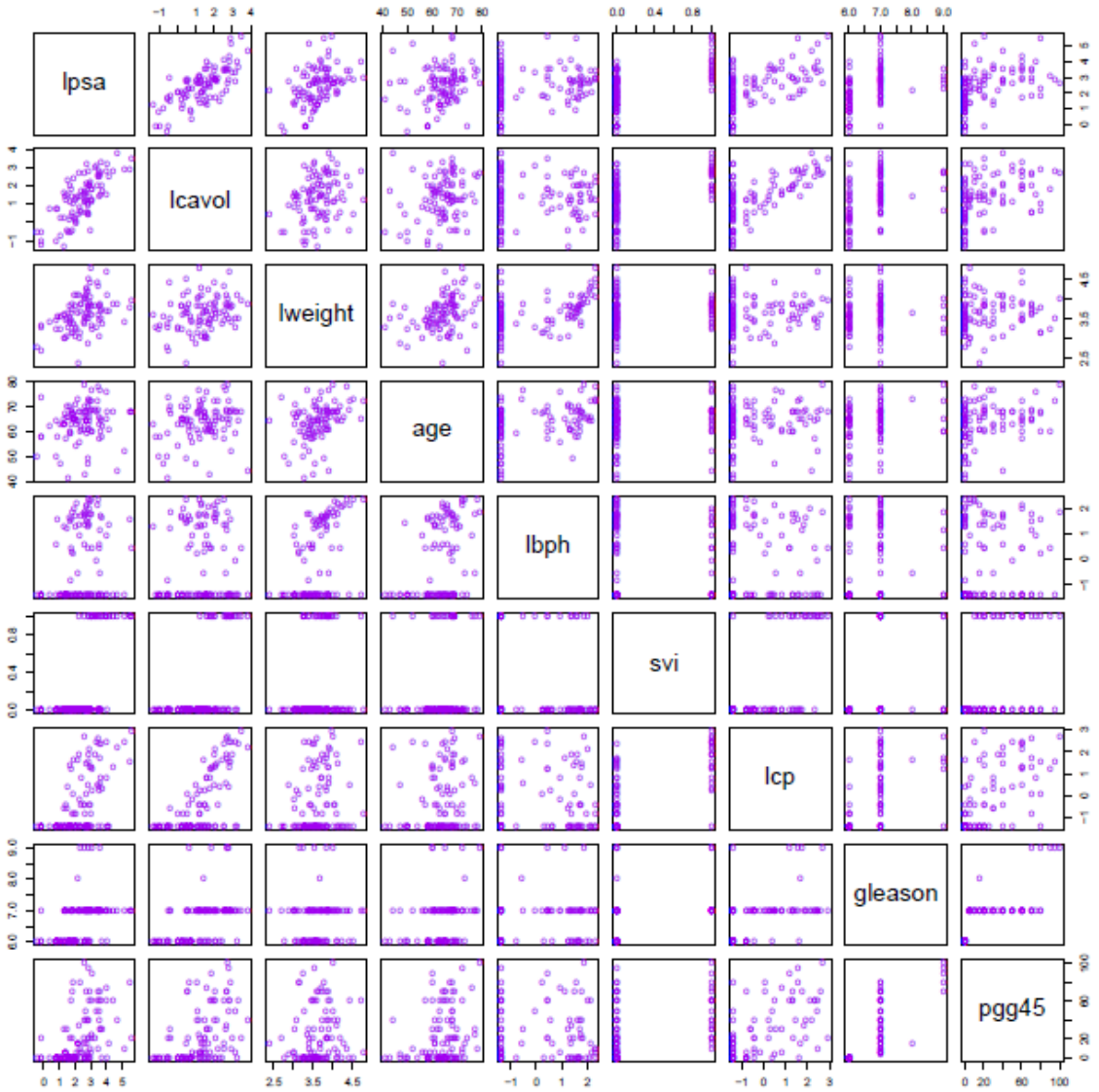
Visualise your data (Example 1)

- Consider a high dimensional dataset. Focus on a distribution of a particular feature x_i.
- Here are the feature values for 80 data points. Can you see the pattern?
- Plotting a histogram can make any patterns apparent

0.170	5.82	3.73	0.450
5.42	5.28	5.43	4.97
1.77	3.52	3.52	5.26
4.44	-1.51	3.21	3.96
1.80	6.45	-0.240	5.38
0.590	5.60	-0.850	4.46
-0.640	4.67	4.49	5.20
3.14	1.18	4.31	2.03
5.59	3.83	-1.97	-0.760
0.0700	-1.27	5.65	0.880
6.30	-0.650	-0.560	0.600
5.04	-0.280	1.78	0.790
0.180	2.98	-0.570	-1.87
4.72	0.800	-1.05	-0.290
4.65	-1.11	1.27	-0.200
-0.490	0.590	0.930	-0.850
4.51	5.45	5.06	3.95
-1.37	1.18	3.79	5.93
1.40	4.84	5.43	5.90
6.11	5.61	-0.320	-0.420



Visualise your data (Example 2)



Another useful way to visualize: pairwise scatterplots

Plot from Hastie et al. The Elements of Statistical Learning, 2013

Dimensionality reduction

- A generic term for converting a high dimensional dataset into a low dimensional representation
- Reduces computational time and storage requirements
  - A form of compression: often data can be approximated with a lower dimensional representation
- Often used for visualization
  - Dimensionality reduction does not always imply visualization
  - Low dimensional representation can have more than 3 dimensions
- There are many techniques for dimensionality reduction
  - Principle component analysis
  - Multidimensional scaling

Dimensionality reduction

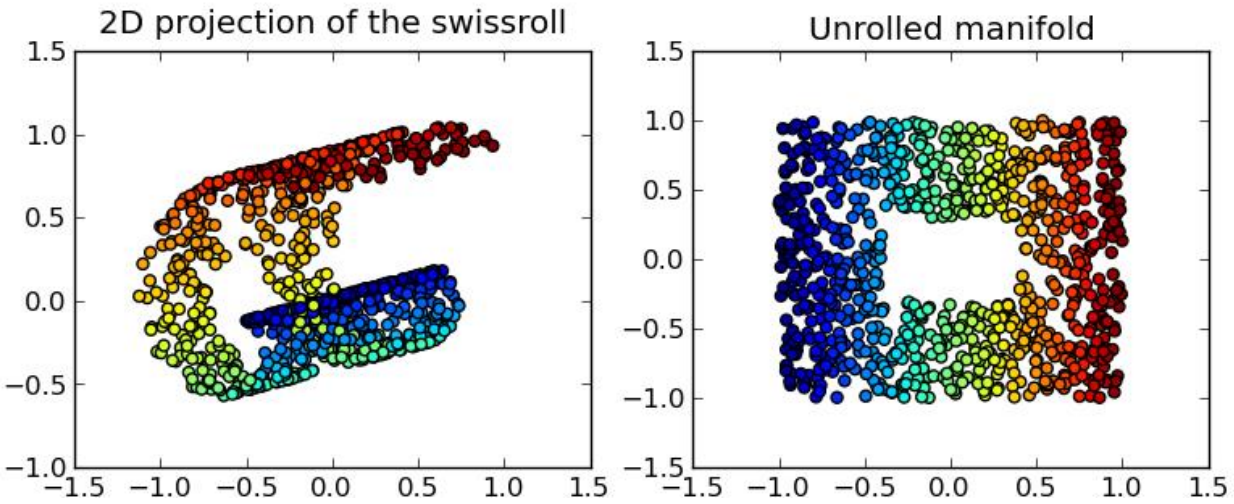
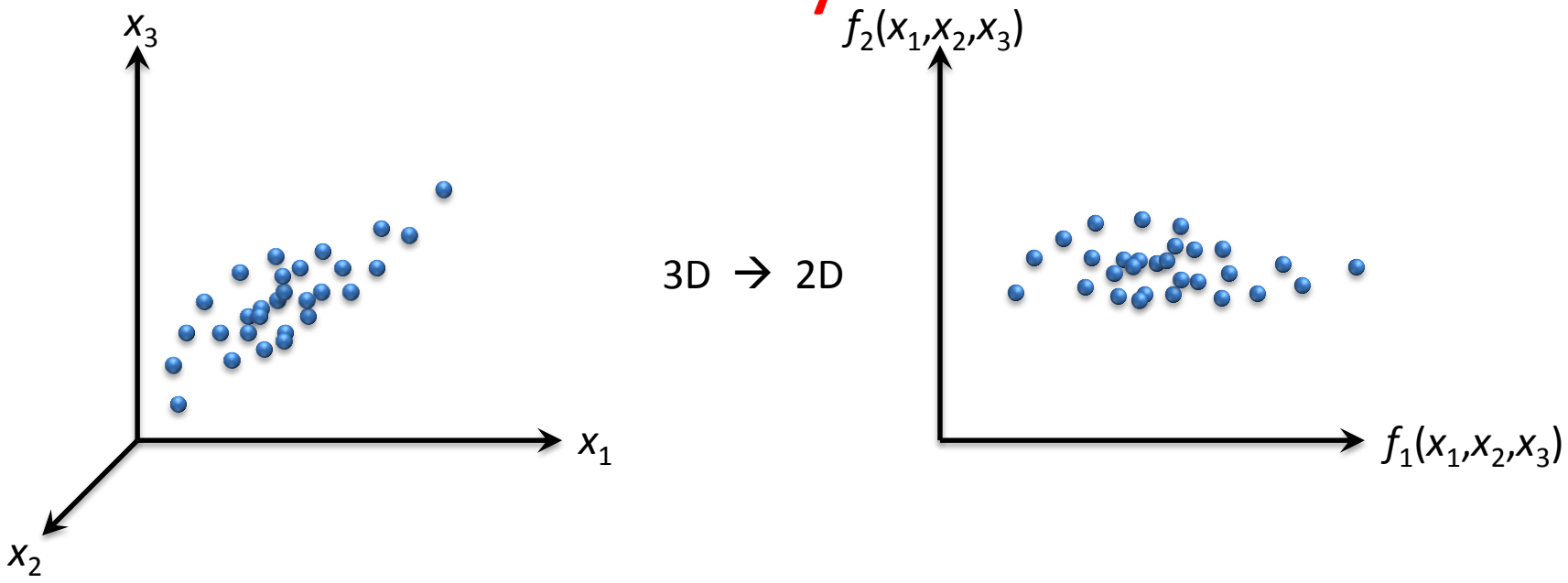


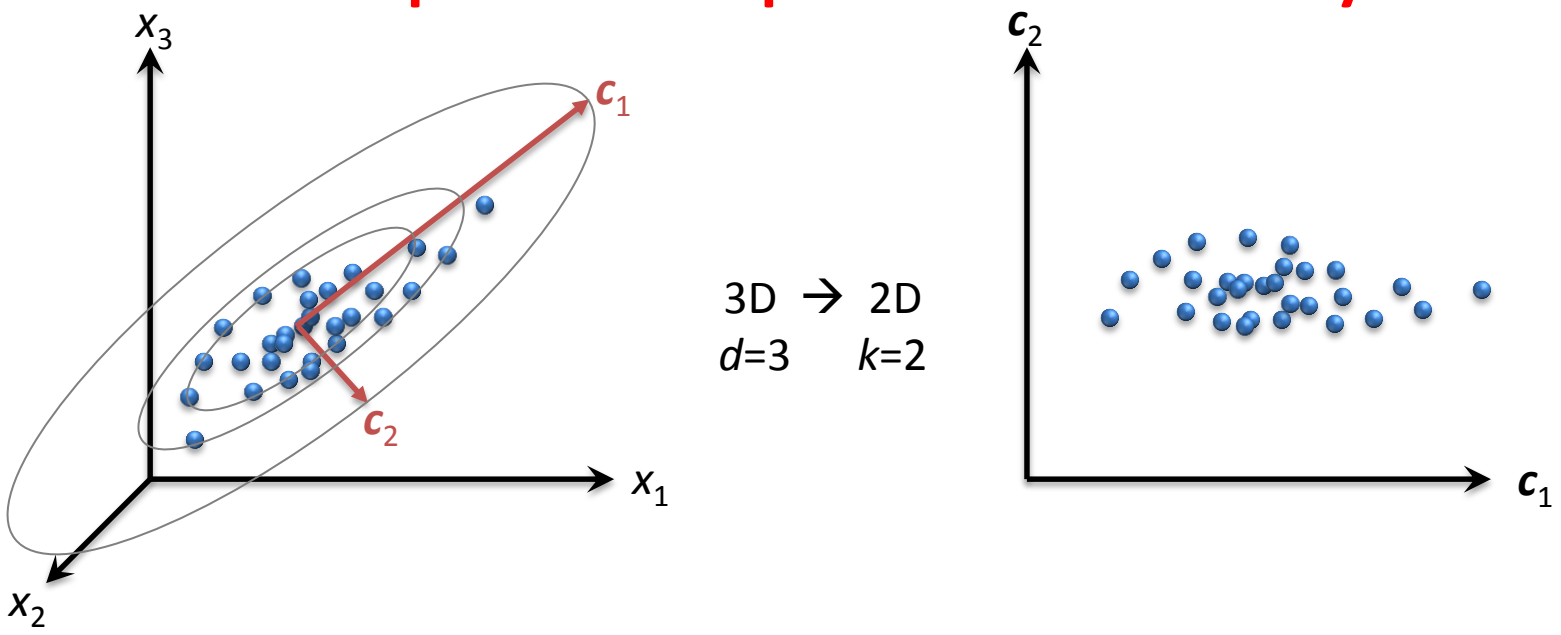
Image adapted from Wikipedia, original image: Olivier Grisel

Dimensionality reduction



- Visualisation
- Feature selection, before (e.g.) classification

Principal Components Analysis



- Chooses new dimensions as directions of **max variance** – the data's **principal components**
- PC's chosen to be orthogonal; use top $k<d$
- Same as k-dim plane that minimises RSS

PCA Algorithm

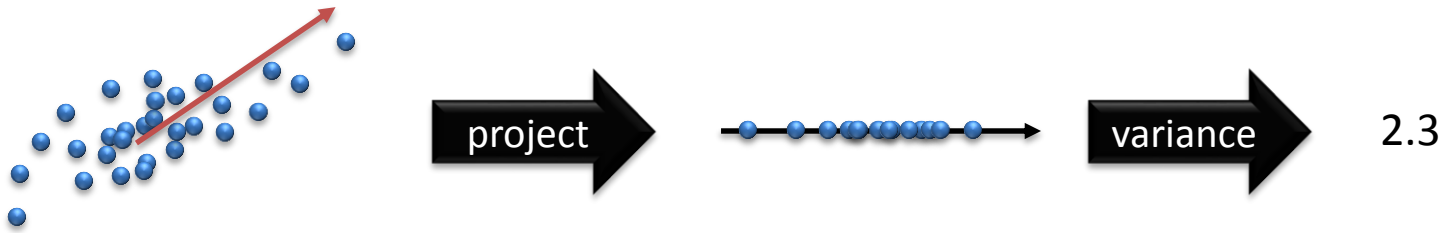
- Find direction $\boldsymbol{c}$ of max variance

$$\max_{\boldsymbol{c}} \ \mathrm{Var}(\mathbf{X}\boldsymbol{c})$$

$$\text{subject to } \|\boldsymbol{c}\| = 1$$

Project data onto candidate $\boldsymbol{c}$

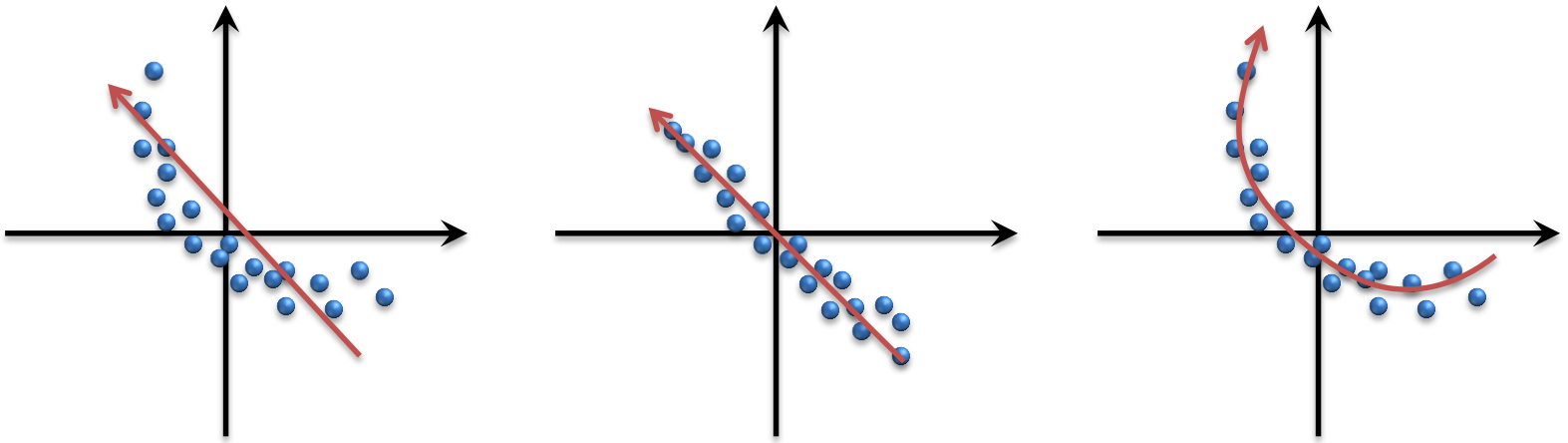
Variance is of scalar projections



- Simplify: first center data, then linear algebra…
  - $\boldsymbol{c}$'s given by eigenvectors of covariance matrix $\mathbf{X}\mathbf{X}^T$
  - Variance explained by PC i given by i^{th} eigenvalue

Kernel PCA

- Low-dim approximation need not be linear!

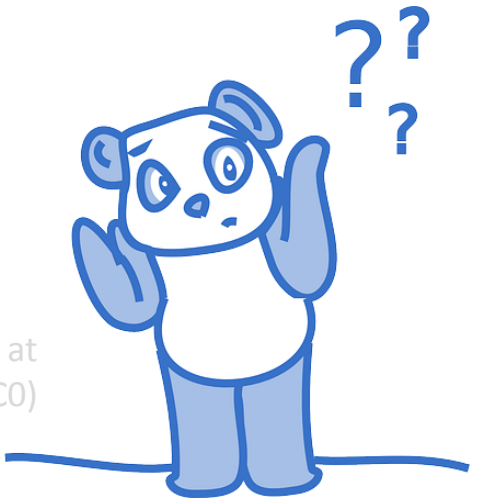


PCA | PCA in feature space | Kernel PCA

- Kernel PCA: map data to feature space, then run PCA
  - Modular! Just PCA on a matrix related to ***K***

Checkpoint

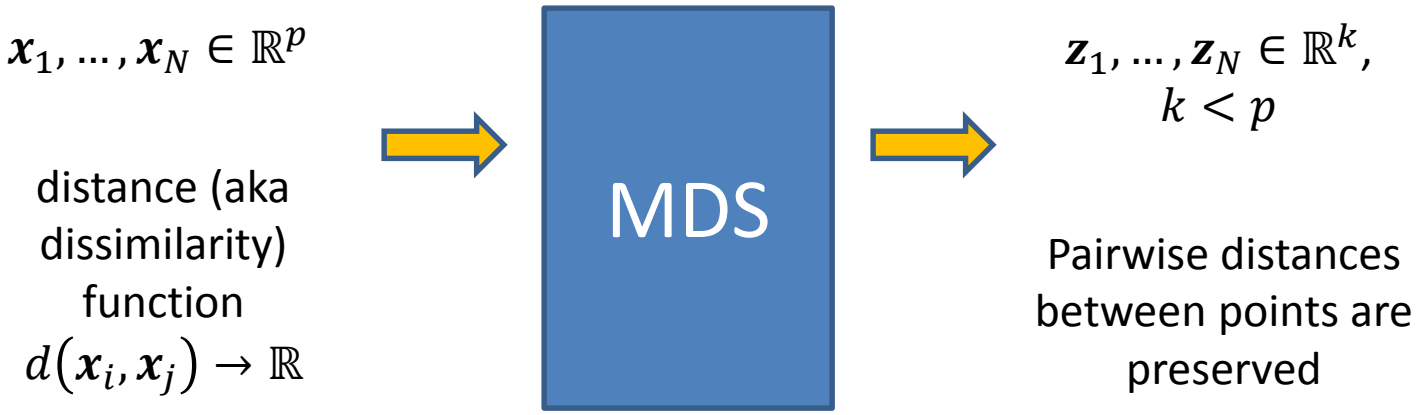
- Which of the following statements is true?
  -  Sum of two kernels is a kernel
  -  Kernel is any symmetric real valued function of two arguments $K(\boldsymbol{u}, \boldsymbol{v}) = K(\boldsymbol{v}, \boldsymbol{u})$
  -  Representer theorem provides two solutions to the soft margin SVM optimization problem



art: OpenClipartVectors at pixabay.com (CC0)

Multidimensional scaling (MDS)

- MDS is an approach to map data to a lower-dimensional space, such that pairwise distances are preserved
- MDS is a common name for a group of related methods



Least squares MDS

- Given a dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^p$ with $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denoting distance between points i and j, find values $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N \in \mathbb{R}^k$, $k < p$, to minimise

- Stress function

$$S_M(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N) = \sum_{i \neq j} \left(d(\boldsymbol{x}_i, \boldsymbol{x}_j) - \|\boldsymbol{z}_i - \boldsymbol{z}_j\| \right)^2$$

Least squares MDS with Sammon mapping

- Given a dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^p$ with $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denoting distance between points i and j, find values $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N \in \mathbb{R}^k$, $k < p$, to minimise

$$S_{Sm}(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N) = \sum_{i \neq j} \frac{\left(d(\boldsymbol{x}_i, \boldsymbol{x}_j) - \|\boldsymbol{z}_i - \boldsymbol{z}_j\|\right)^2}{d(\boldsymbol{x}_i, \boldsymbol{x}_j)}$$

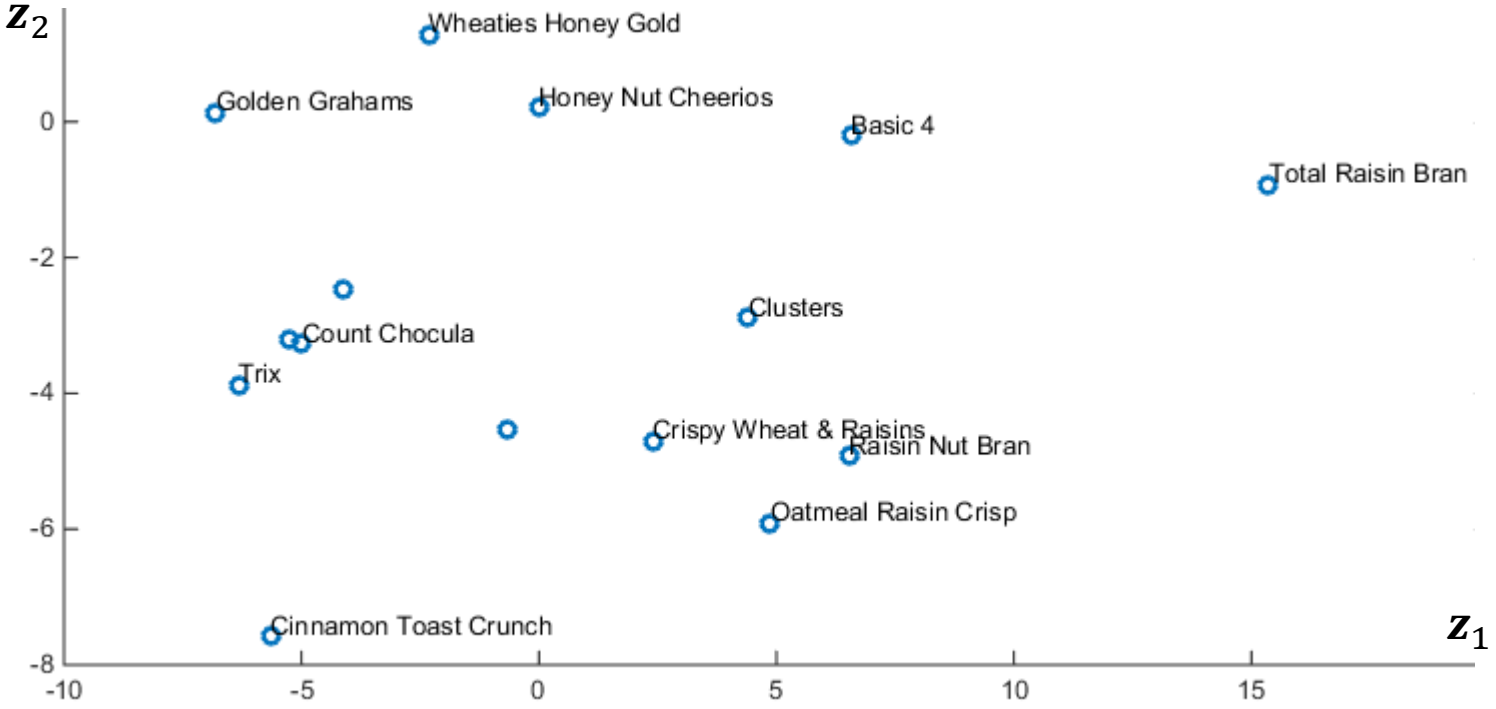
Classical MDS

- Given a dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^p$ with $s(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denoting similarity between points i and j, find values $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N \in \mathbb{R}^k$, $k < p$, to minimise

$$S_C(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N) = \sum_{i \neq j} \left(s(\boldsymbol{x}_i, \boldsymbol{x}_j) - \langle \boldsymbol{z}_i - \bar{\boldsymbol{z}}, \boldsymbol{z}_j - \bar{\boldsymbol{z}} \rangle \right)^2$$

MDS at work

- Cereals dataset (sample)
- Each original point has 22 dimensions
- MDS is used to convert it to 2D data



Based on nonclassical MDS example from Matlab documentation

Notes on MDS

- Most commonly used with $k = 2$ for visualization
- Least squares MDS is solved using gradient descent
- Classical MDS has an explicit solution (in terms of eigenvectors)
  - Classical MDS is equivalent to PCA (if similarities are centered inner-products)
- The three variations of MDS are not equivalent, objective functions are different

Pairwise distances/similarities

- MDS does not require original values for data points, only pairwise dissimilarities
  - Sometimes we don't even have original values (e.g., wine tasting experiment)
- Euclidean distance is commonly used as a dissimilarity measure
  - $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$
- Centered inner product is often used as a similarity measure
  - $s(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i - \overline{\boldsymbol{x}}, \boldsymbol{x}_j - \overline{\boldsymbol{x}} \rangle$
- Other distance and similarity measures can be used!

Summary

- The goal of unsupervised learning is to investigate properties of data distribution
- Two types of approaches: hypothesis driven and exploratory analysis
- Dimensionality reduction and visualization are useful in data analysis
- Principal Component Analysis and Multidimensional Scaling are popular tools for dimensionality reduction