Unsupervised Learning

20_KIN2 – Artificial Intelligence and Machine Learning

Lecture Contents

- Definition of unsupervised learning
- Clustering – K-Means algorithm
- Association – collocation extraction
- Dimensionality reduction – Principal Component Analysis

Unsupervised Learning

Unsupervised learning is the study of algorithms that discover hidden patterns or data groupings without the need for human intervention

- Used for data exploration and for learning concise representations
- The data is unlabeled and often high-dimensional
- The concept of distance plays a crucial role

Clustering

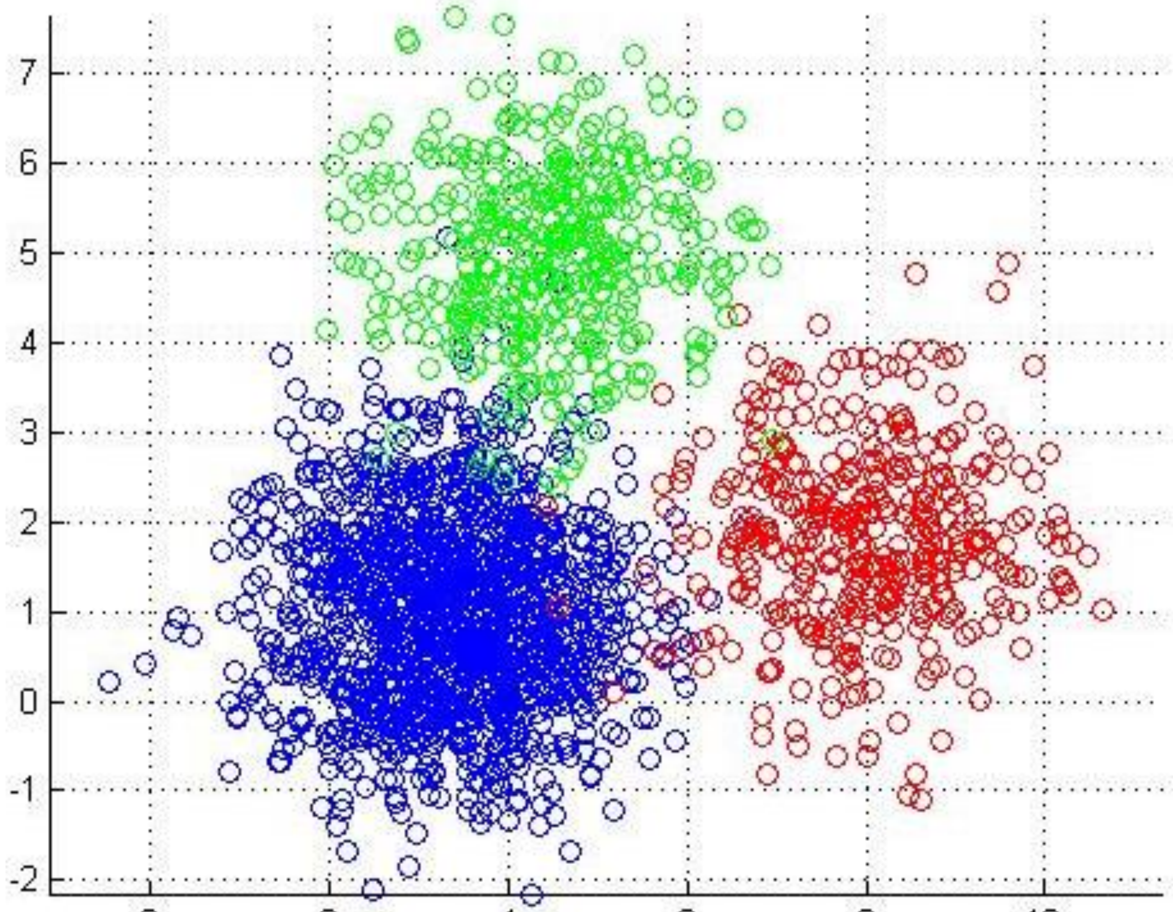
- Clustering seeks to group data based on similarity
  - objects in one cluster are similar among themselves and different from objects in another cluster

- Applications in many fields (pattern recognition, image analysis, bioinformatics, etc.)

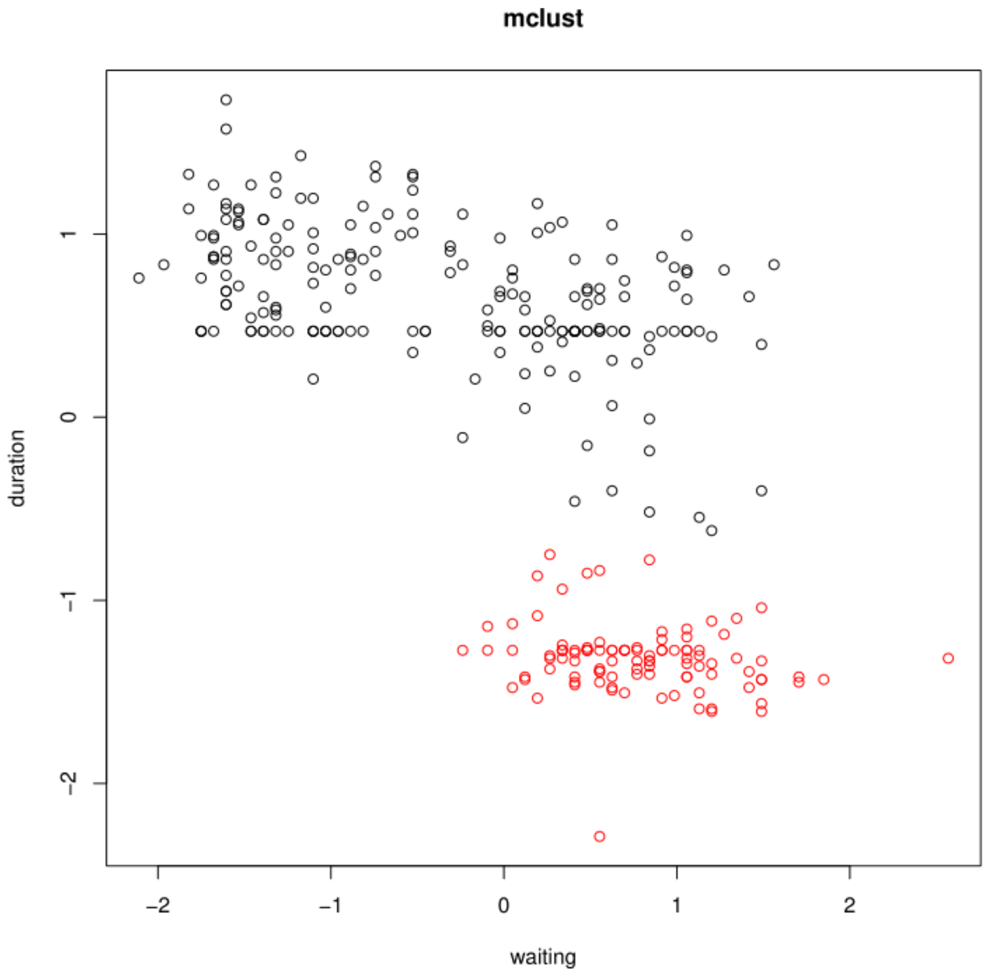
- Many different algorithms with own strengths and weaknesses

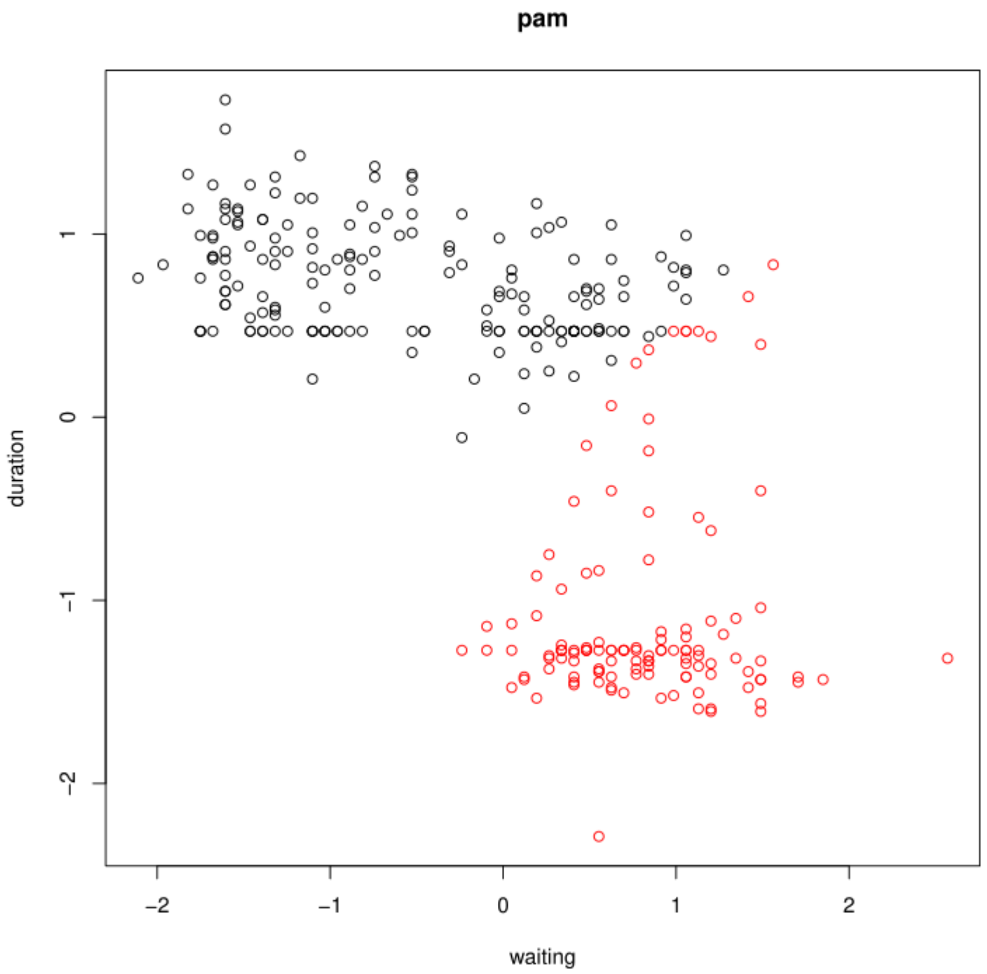
- Very important in big data – runtime is critical

Clustering

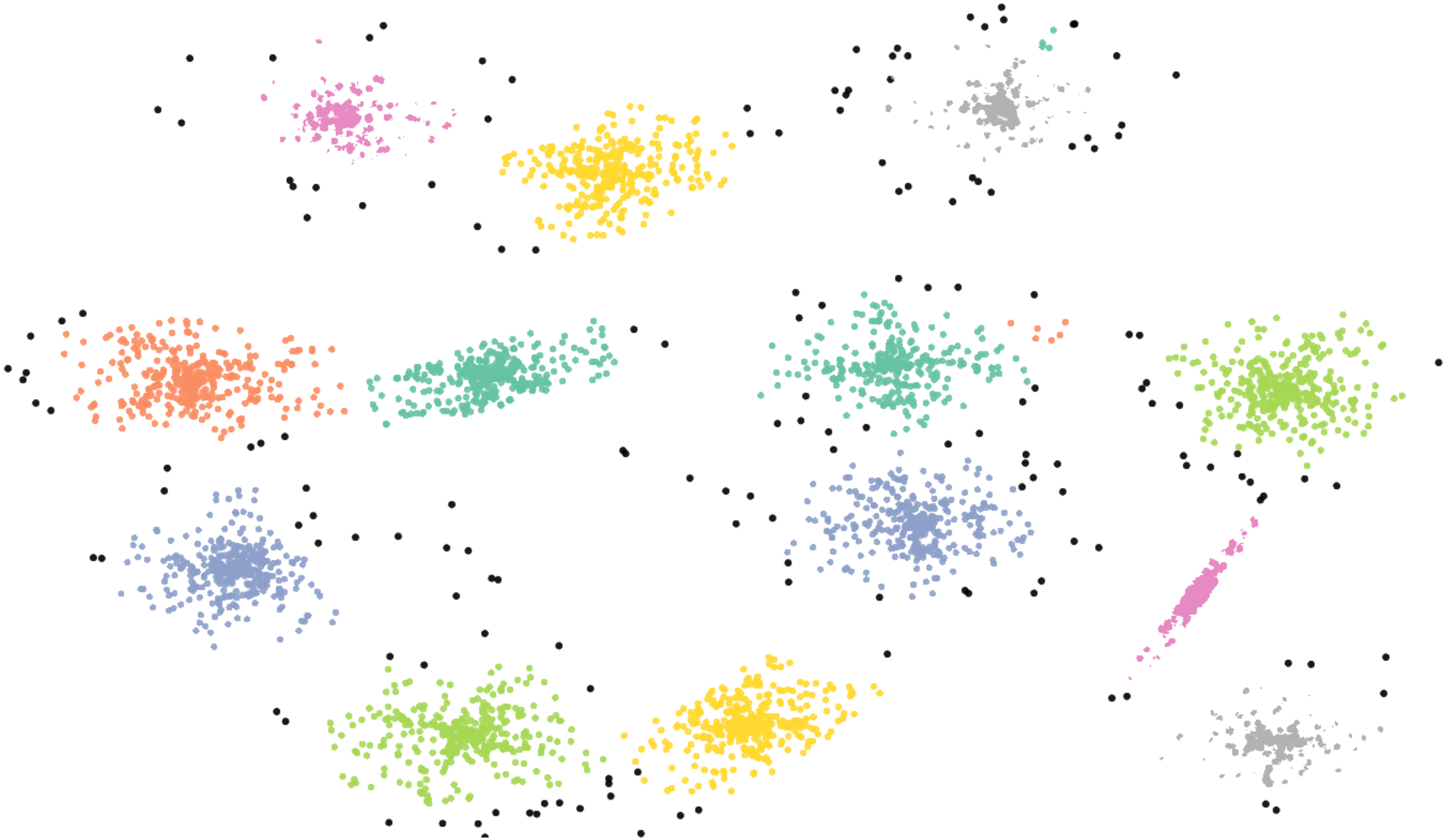


Clustering

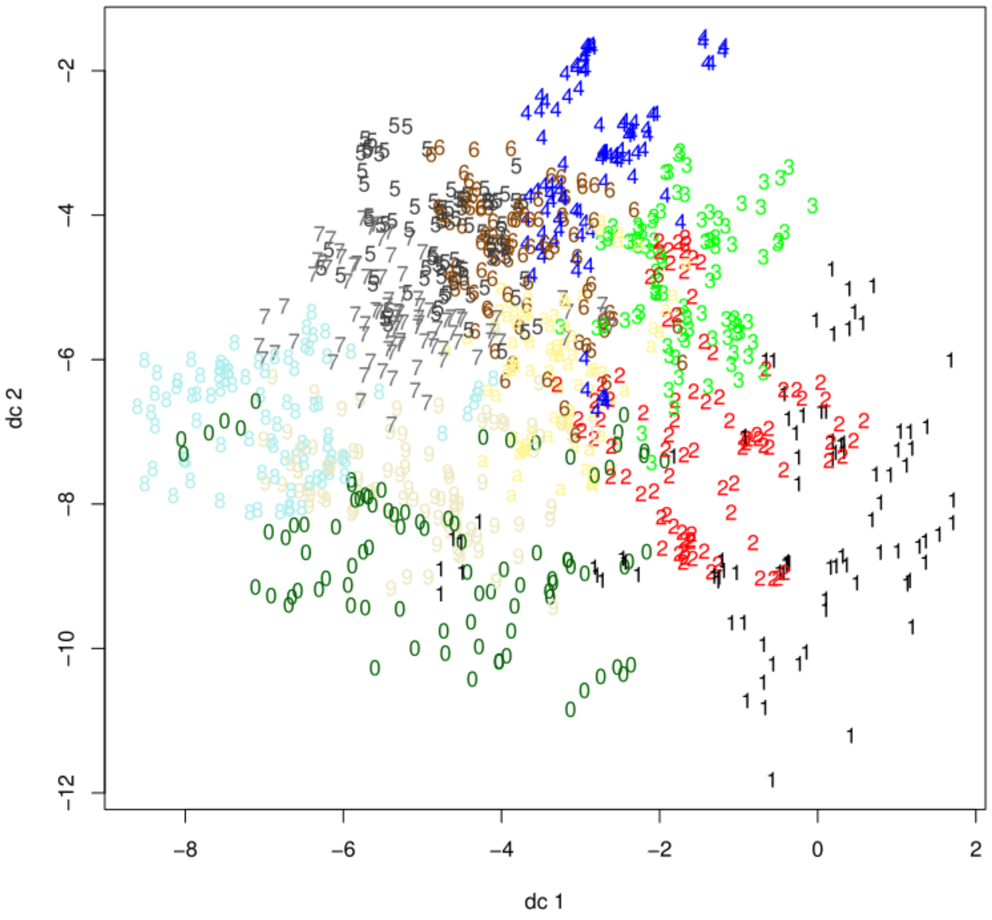

mclust
duration
waiting
−2
−1
0
1
2
−2
−1
0
1

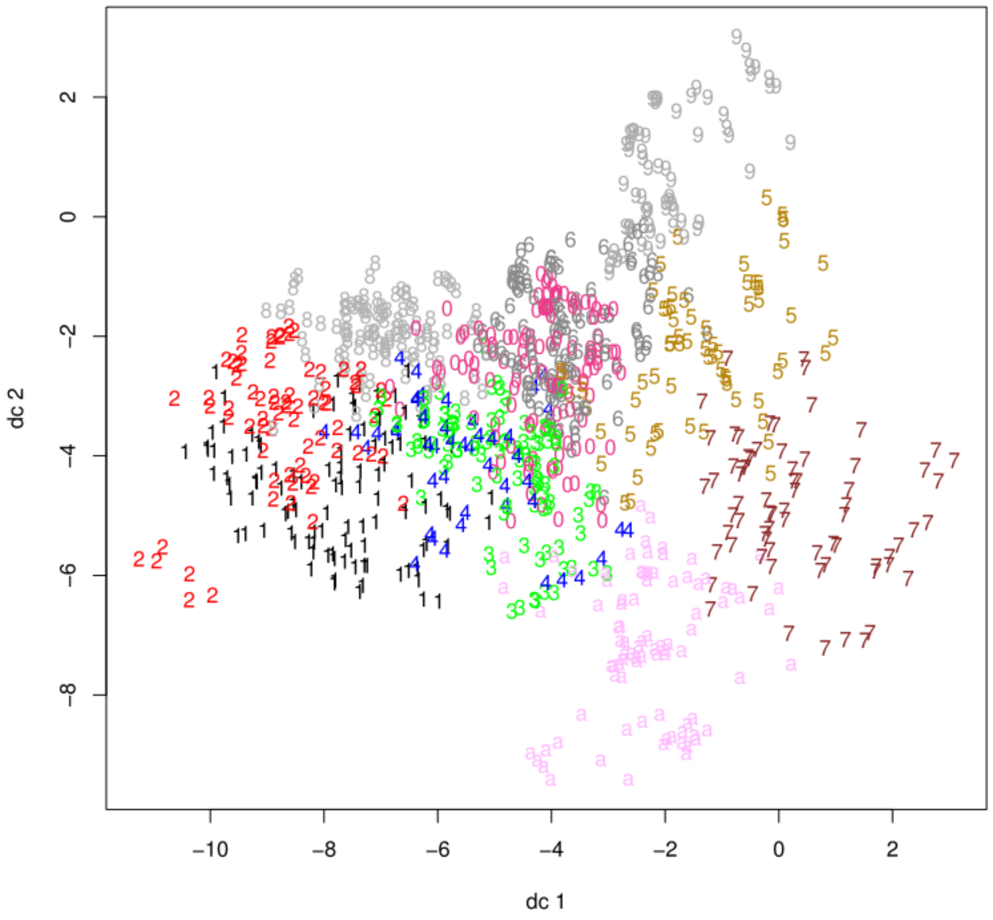

pam
duration
waiting
−2
−1
0
1
2
−2
−1
0
1

Clustering

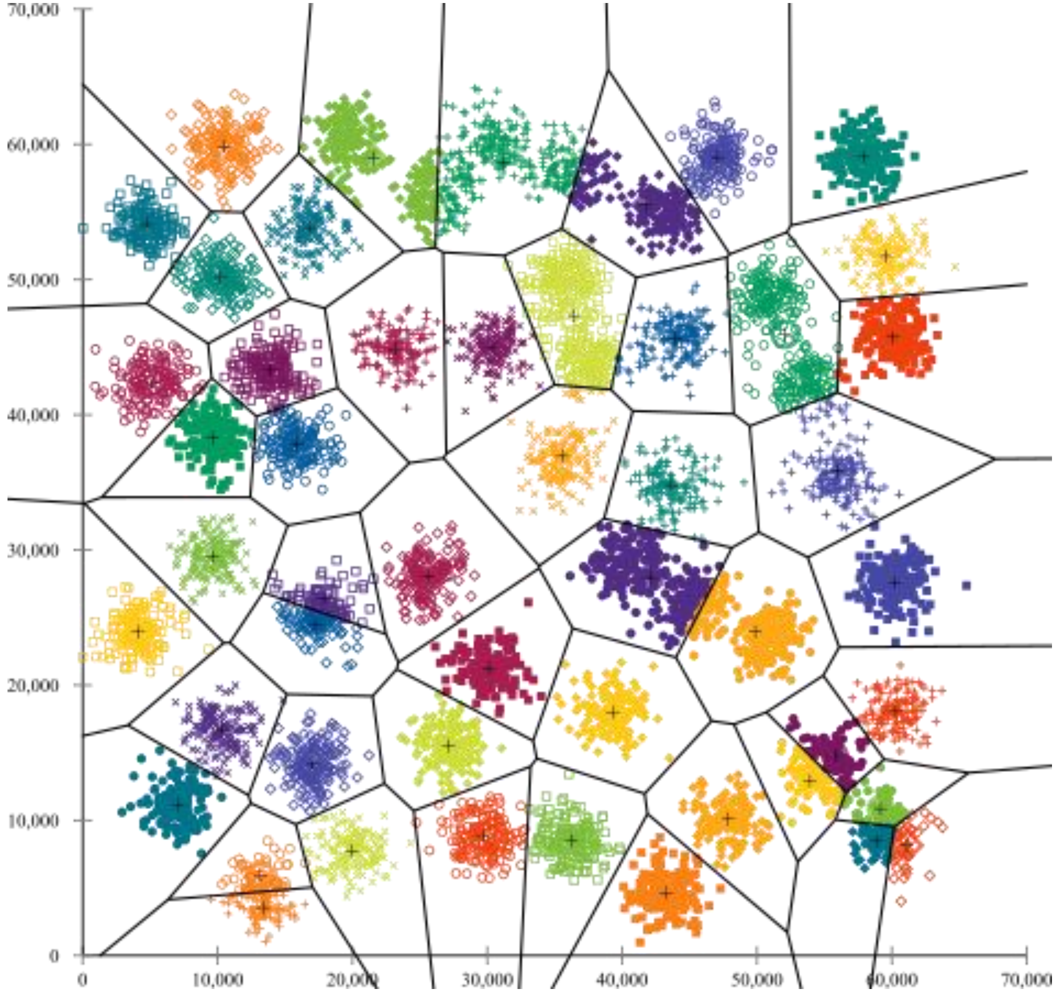


Clustering

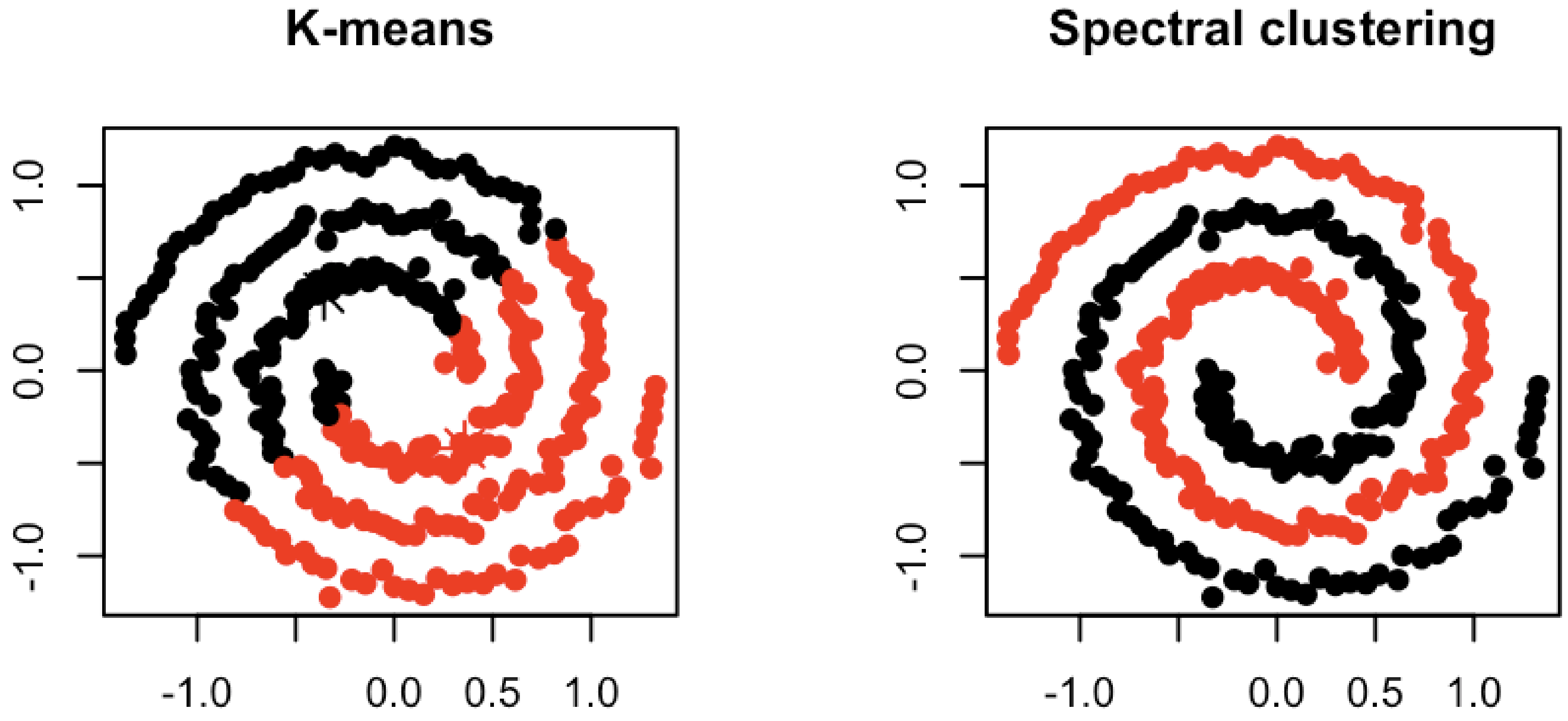




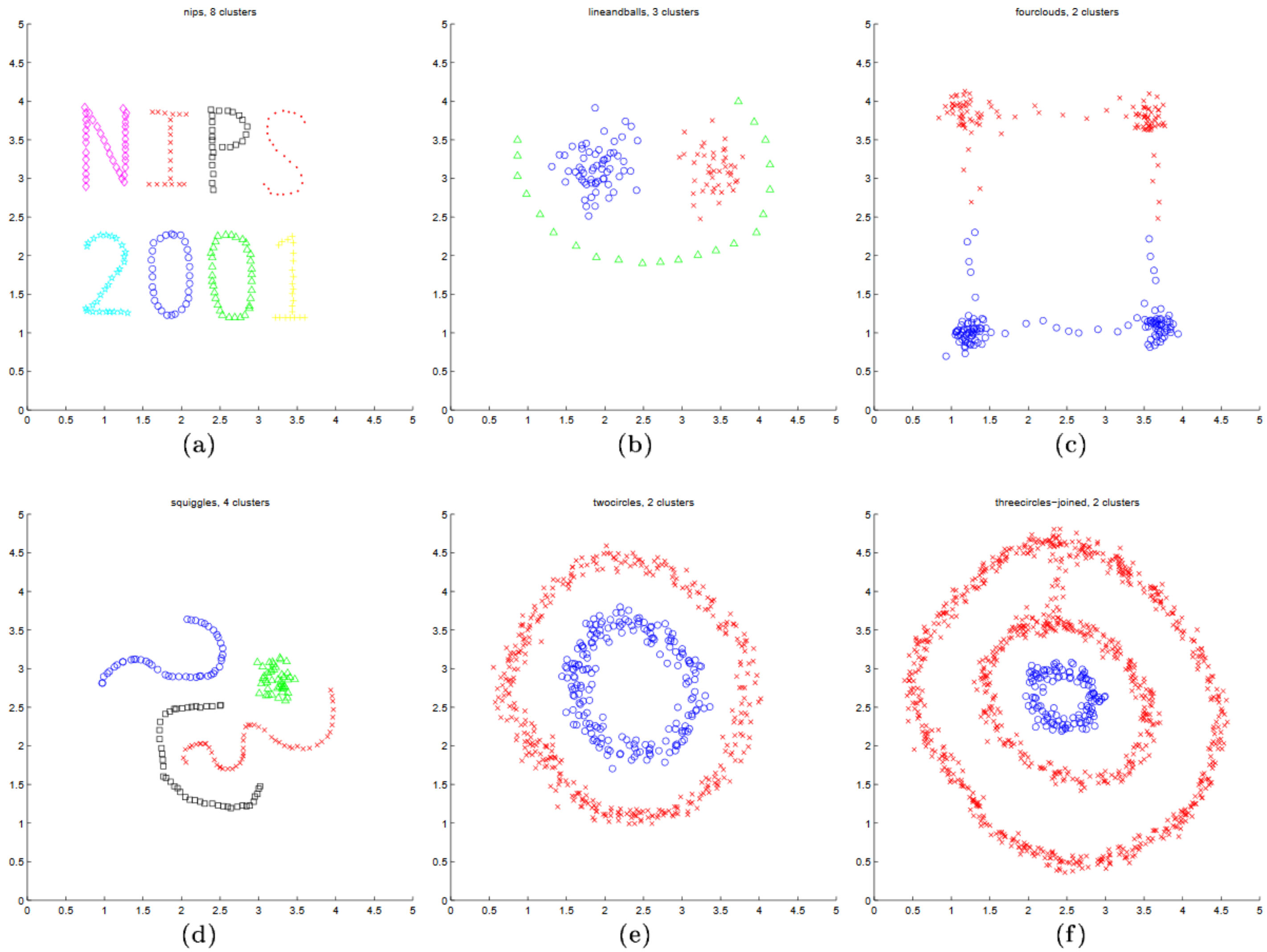
Clustering


70,000
60,000
50,000
40,000
30,000
20,000
10,000
0
0
10,000
20,000
30,000
40,000
50,000
60,000
70,000

Clustering


K-means
1.0
0.0
-1.0
-1.0
0.0
0.5
1.0
Spectral clustering
1.0
0.0
-1.0
-1.0
0.0
0.5
1.0

Clustering



(a) (b) (c)

(d) (e) (f)

Clustering

General approach

- Attempt to find meaningful groups based on some similarity
- The concept of a centroid plays an important role
- Cluster membership usually determined based on distance to centroid

Data structures

- Data matrix
- Distance (dissimilarity) matrix

Clustering

Cluster centroid

- Coined in 1814 with initial meaning “center of gravity” or “center of mass”.
- The concept is much older (Archimedes found the centroids of geometric plane figures).
- The centroid of a cluster is the mean point (its parameter values are the mean of the parameter values of all points in the cluster)

Distance – usually Euclidean distance when dealing with “flat geometry” (otherwise, graph distance, Mahalanobis distance)

Clustering

Distance-based Clustering

1. Assign a distance measure between data
2. Find a partition such that
   - Distance between objects within partition (ie., same cluster) is minimized
   - Distance between objects from different clusters is maximized

Potential issues

- Requires a distance
- Exponential number of possible partitions
- Relative weighting of attributes

Clustering

In general

- No “one size fits all”
- Clustering can have different goals depending on application
- If these aims carry different weights, they should be measured separately

Clustering

Typical clustering goals

- Between-cluster separation
- Within cluster homogeneity (low distances)
- Within cluster homogeneous distributional shape
- Good representation of data by centroids
- Good representation of dissimilarity by clustering-induced metric
- Clusters are regions of high density
- Uniform cluster sizes

Clustering

No size fits all

Pattern recognition in images requires separation

Clustering for information reduction requires good representation by centroids

Groups in social network analysis shouldn’t have large within-cluster gaps

Underlying “true” classes (biological species) may cause homogeneous distributional shapes

Measuring clustering quality is an ongoing research topic

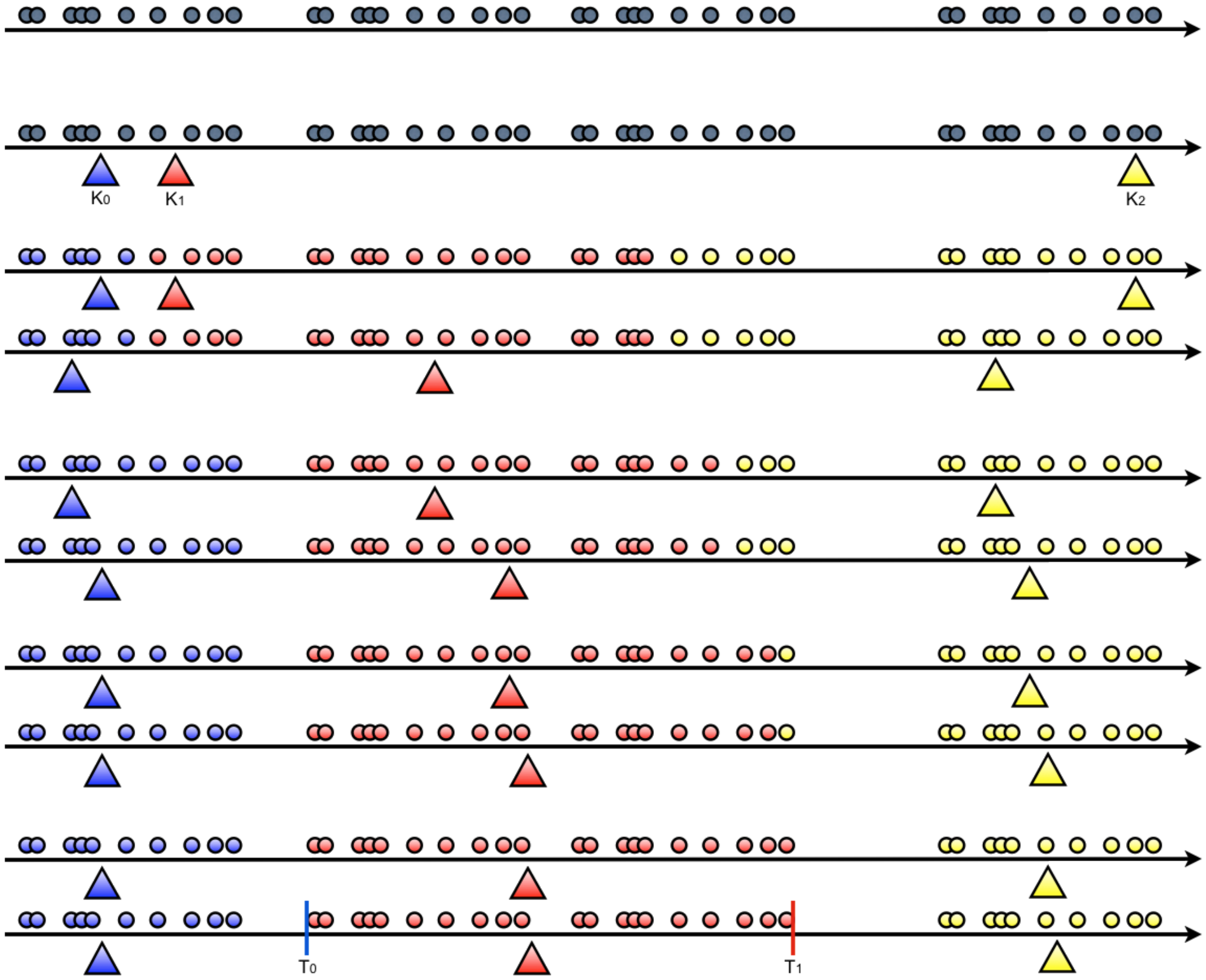
Evaluation/validation of clustering results is as difficult as clustering itself

Clustering

K-Means Algorithm

1. Specify desired number of clusters *n*
2. Pick *n* centroids (typically randomly)
3. Assign nearest points to corresponding centroids
4. Update centroids
5. If not converged go to step 3, otherwise stop

Clustering



Clustring

K-Means

Pros

- Simple, reasonably fast
- Widely available
- Decent results
- Building block for other clustering methods

Cons

- Non-deterministic
- Can have empty clusters
- Vulnerable to noise and outliers
- Tends to pick spherical (globular) groups

Data reduction

Motivation

- Storage space (some datasets consist of terabytes of data)
- Algorithm efficiency and improved modeling results

Reduction concept

- Obtain reduced representation while preserving information
- Preserve ability to learn from data

Strategies

Compression, numerosity reduction, dimensionality reduction, discretization

Principal component analysis

Unsupervised method - ignores class labels, invented in 1901 by **Karl Pearson**

Can reveal the structure of the data in a way that best explains the variance

- Used as a tool in exploratory data analysis
- Used as a preprocessing step before finding predictive models
- Often used to visualize distance and relatedness

Project a feature space onto a smaller subspace that represents the data well, by means of a linear transformation

Principal component analysis

Operating principle

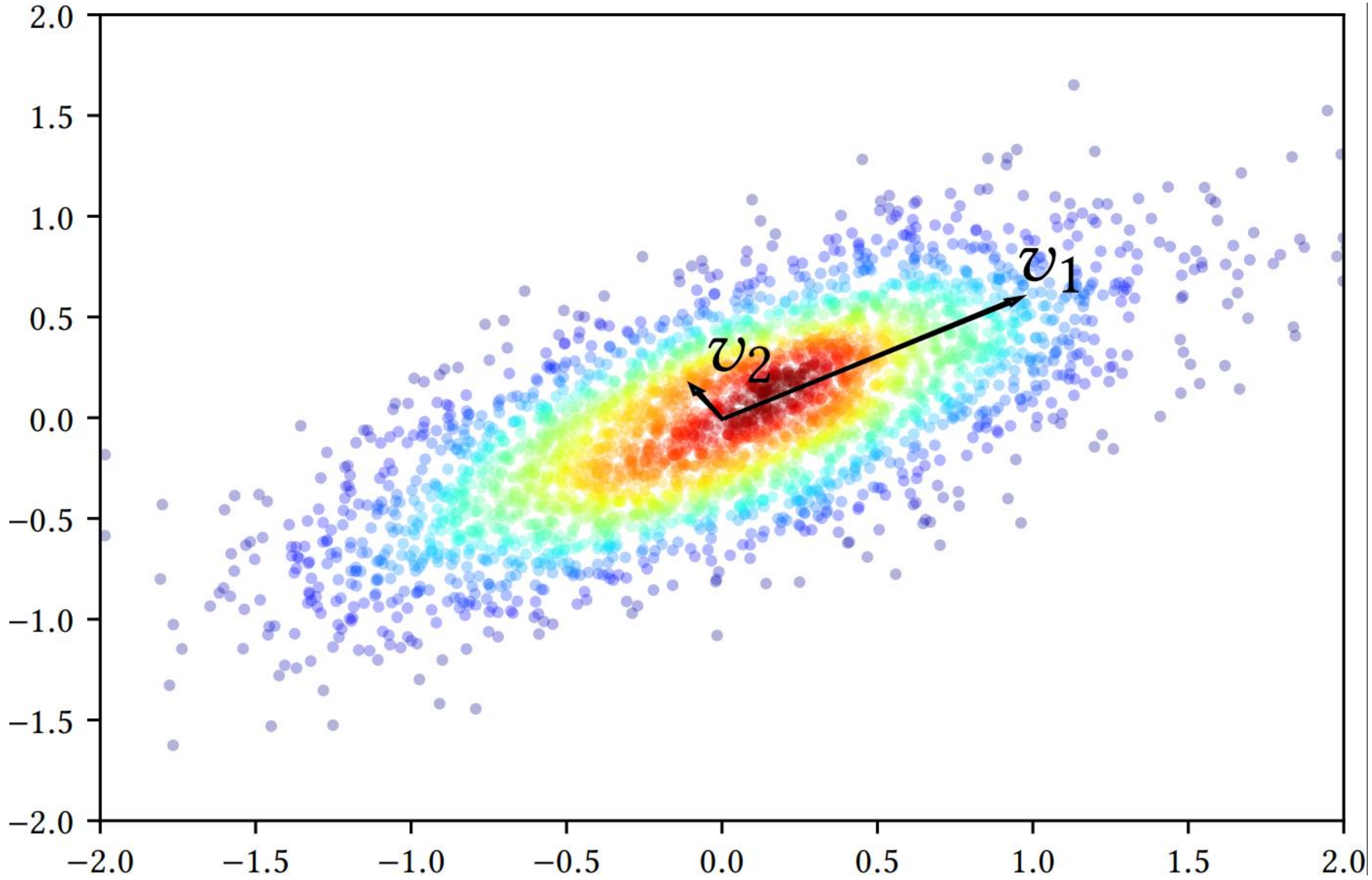
Eigenvalue decomposition of the data covariance or correlation matrix

- Eigenvectors are vectors which are fixed in direction under a given linear transformation
- The scaling factors of these eigenvectors are called the eigenvalues

Suppose we have a set of data with a certain distribution

- Eigenvectors v_i tell us the orientation of the distribution
- Eigenvalues λ_i tell the amount of variance in each dimension

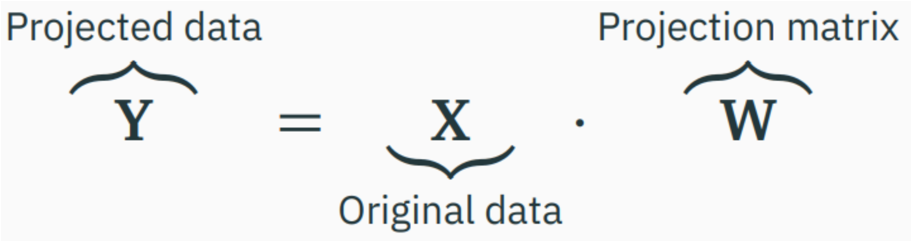
Principal component analysis



Principal component analysis

Step-by-step guide

1. Standardize the data (zero-mean, unit-variance)
2. Calculate *eigenvectors* and *eigenvalues*
3. Choose *k* principal components based on the *k* largest *eigenvalues*
4. Construct projection matrix W from the selected *k eigenvectors*
5. Project original feature space X using W to obtain *k*-dimensional feature subspace Y



$$\overbrace{\mathbf{Y}}^{\text{Projected data}} = \underbrace{\mathbf{X}}_{\text{Original data}} \cdot \overbrace{\mathbf{W}}^{\text{Projection matrix}}$$

PCA Example

R. A. Fisher, *The Use of Multiple Measurements in Taxonomic Problems*, 1936

- Data describing morphologic variation of Iris flowers
- Three related species (class labels): setosa, versicolor, virginica
- Four features: sepal length and width, petal length and width (in cm)

setosa

versicolor

virginica

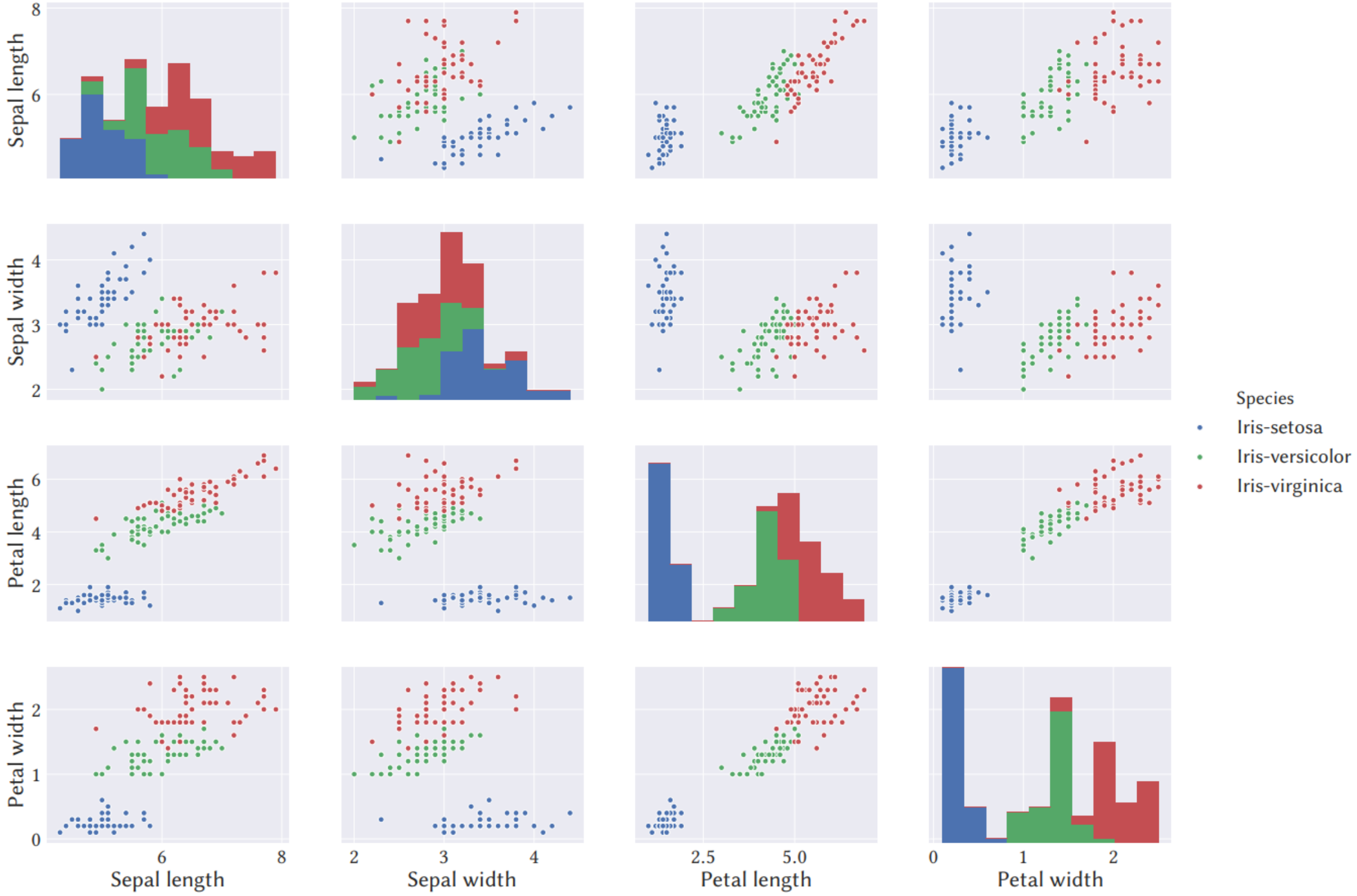
PCA Example

Input data X

$$\mathbf{x}^T = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \text{sepal length} \\ \text{sepal width} \\ \text{petal length} \\ \text{petal width} \end{bmatrix}$$

- 150 instances (50 for each class), no missing values
- Predicted attribute: species

PCA Example


Sepal length
Sepal width
Petal length
Petal width
8
6
4
3
2
6
4
2
2
1
0
6
8
2
3
4
2.5
5.0
0
1
2
Sepal length
Sepal width
Petal length
Petal width
Species
● Iris-setosa
● Iris-versicolor
● Iris-virginica

PCA Example

1. **Standardization (Z-score normalization)**

$$x' = \frac{x - \bar{x}}{s}$$

- important step for many ML algorithms (e.g., k-means, KNN, SVM, LDA)
- when in doubt, standardize

PCA Example

2. Eigenvectors and eigenvalues

- Compute covariance matrix Σ from data
- Eigendecomposition of covariance matrix Σ
  (also possible to use correlation matrix)
- In practice, we prefer singular value decomposition (more efficient)

Four features (four dimensions) → four eigen vector/value pairs

Values represent the amount of variance explained by each principal component

PCA Example

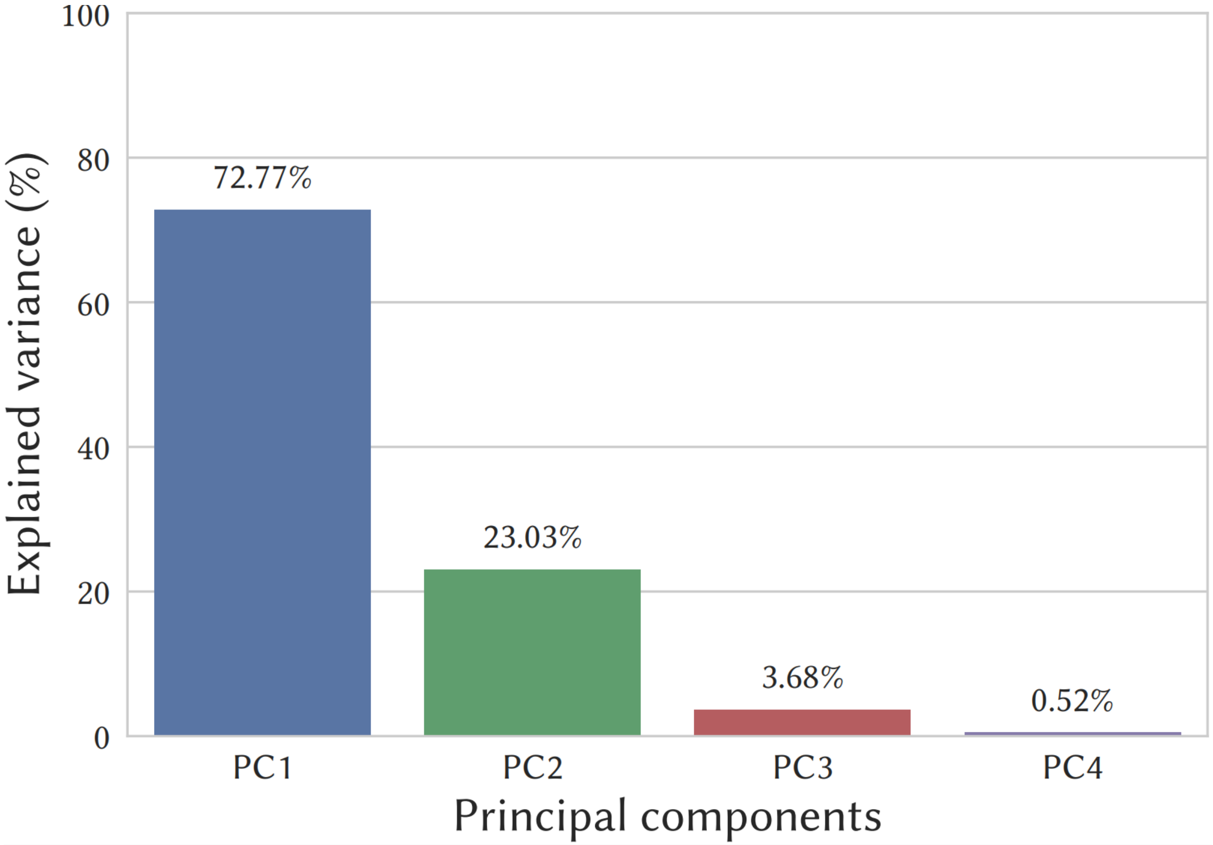
2. Eigenvectors and eigenvalues

$$\Sigma = \begin{bmatrix} 1.0067 & -0.1184 & 0.8776 & 0.8234 \\ -0.1184 & 1.0067 & -0.4313 & -0.3686 \\ 0.8776 & -0.4313 & 1.0067 & 0.9693 \\ 0.8234 & -0.3686 & 0.9693 & 1.0067 \end{bmatrix}$$

$$V = \begin{bmatrix} 0.5211 & -0.3774 & -0.7196 & 0.2613 \\ -0.2693 & -0.9233 & 0.2444 & -0.1235 \\ 0.5804 & -0.0245 & 0.1421 & -0.8014 \\ 0.5649 & -0.0669 & 0.6343 & 0.5236 \end{bmatrix}, \lambda = \begin{bmatrix} 2.9381 \\ 0.9202 \\ 0.1477 \\ 0.0209 \end{bmatrix}$$

PCA Example

3. Choose principal components



PCA Example

4. **Construct the projection matrix**

   We keep the first two *eigenvectors* corresponding to the two principal components, and we obtain

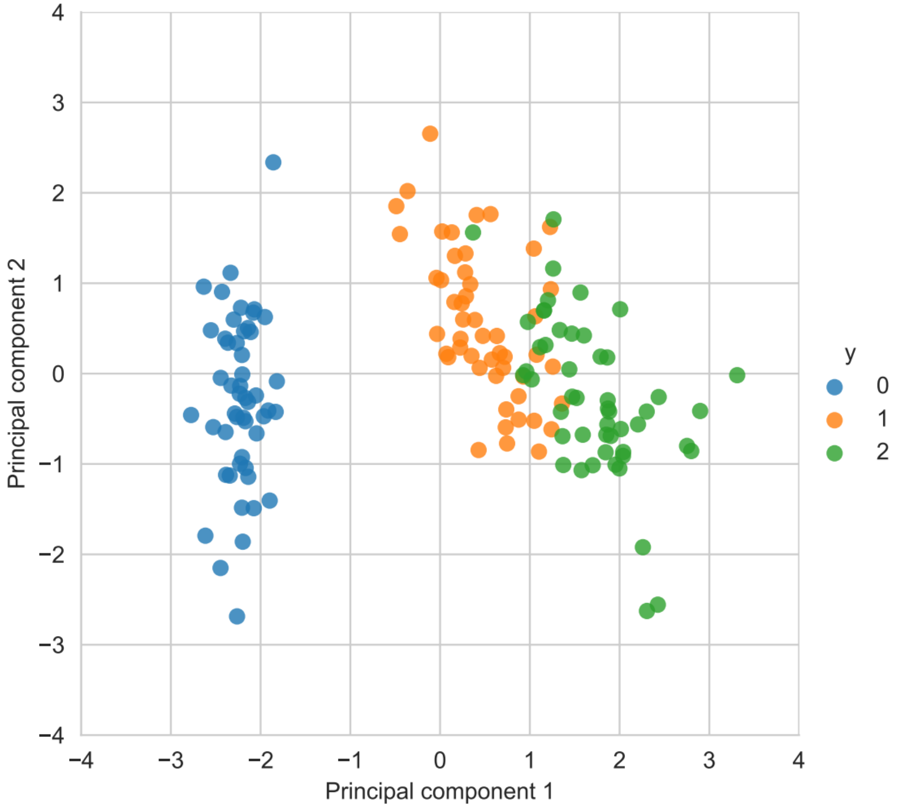
$$\boldsymbol{W} = \begin{bmatrix} 0.5211 & -0.3774 \\ -0.2693 & -0.9233 \\ 0.5804 & -0.0245 \\ 0.5649 & -0.0669 \end{bmatrix}$$

5. **Project the feature space**

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}$$

PCA Example

Samples scatterplot in the lower-dimensional (projected) feature space



PCA Example

How to reconstruct the original features?

Use $\boldsymbol{W}^{\boldsymbol{T}}$ to map the data back to the original dimensions

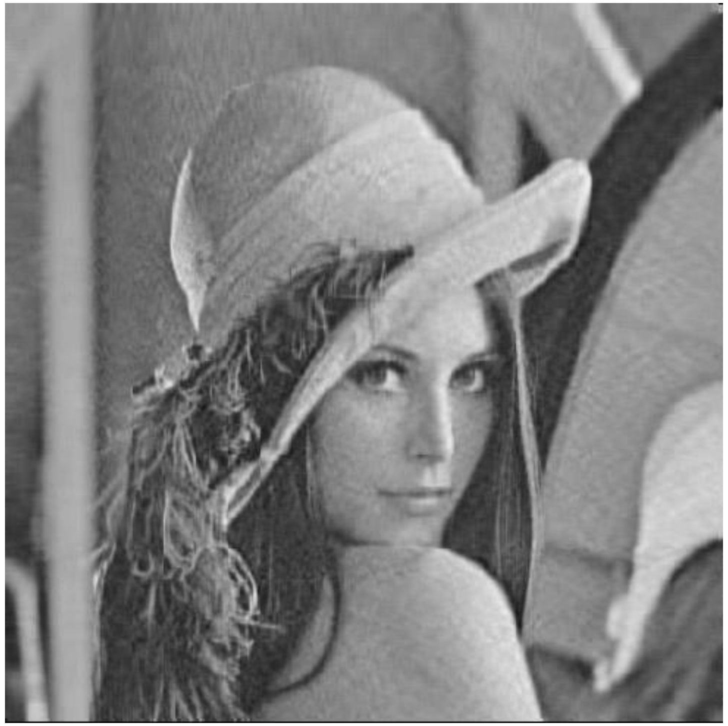
$$\widehat{\boldsymbol{X}} = \boldsymbol{Y}\boldsymbol{W}^{\boldsymbol{T}}$$

Due to this feature, PCA can also be used for compression

PCA Example

Image compression

Original image



Reconstruction from 50 PC

Collocation Extraction

Automatically extract collocations from a corpus

- Find direct connections and quantify strength of connection
- No indirect connections
- Cannot infer causal relationship

Usually a collocation matrix is built, in a given context.

Example

Words which occur together more often than expected by chance

Collocation Extraction

	Apple	Facebook	Tesla
Elon Musk	10	15	300
Mark Zuckerberg	500	10000	500
Tim Cook	200	30	10

Big values → possibly related terms

Approach
"More often than expected by chance"
How can we investigate this aspect?

Collocation Extraction

How to find out if Elon Musk and Tesla are related?

1. Build a joint probability table

	Apple	Facebook	Tesla	
Elon Musk	0.00086	0.00130	0.02594	**0.02810**
Mark Zuckerberg	0.04323	0.86468	0.04323	**0.95115**
Tim Cook	0.01729	0.00259	0.00086	**0.02075**
	0.06139	**0.86857**	**0.07004**	

Collocation Extraction

2. Assume people and companies are independent

In this case, *P(A,B) = P(A) x P(B).* Based on this assumption we get:

	Apple	Facebook	Tesla	
Elon Musk	0.00173	0.02441	0.00197	**0.02810**
Mark Zuckerberg	0.00197	0.82614	0.06662	**0.95115**
Tim Cook	0.00127	0.01802	0.00145	**0.02075**
	0.06139	**0.86857**	**0.07004**	

Collocation Extraction

What we actually observe

	Apple	Facebook	Tesla	
Elon Musk	0.00086	0.00130	**0.02594**	**0.02810**
Mark Zuckerberg	0.04323	**0.86468**	0.04323	**0.95115**
Tim Cook	**0.01729**	0.00259	0.00086	**0.02075**
	0.06139	**0.86857**	**0.07004**	

What we should see if people and companies were independent

	Apple	Facebook	Tesla	
Elon Musk	0.00173	0.02441	**0.00197**	**0.02810**
Mark Zuckerberg	0.00197	**0.82614**	0.06662	**0.95115**
Tim Cook	**0.00127**	0.01802	0.00145	**0.02075**
	0.06139	**0.86857**	**0.07004**	

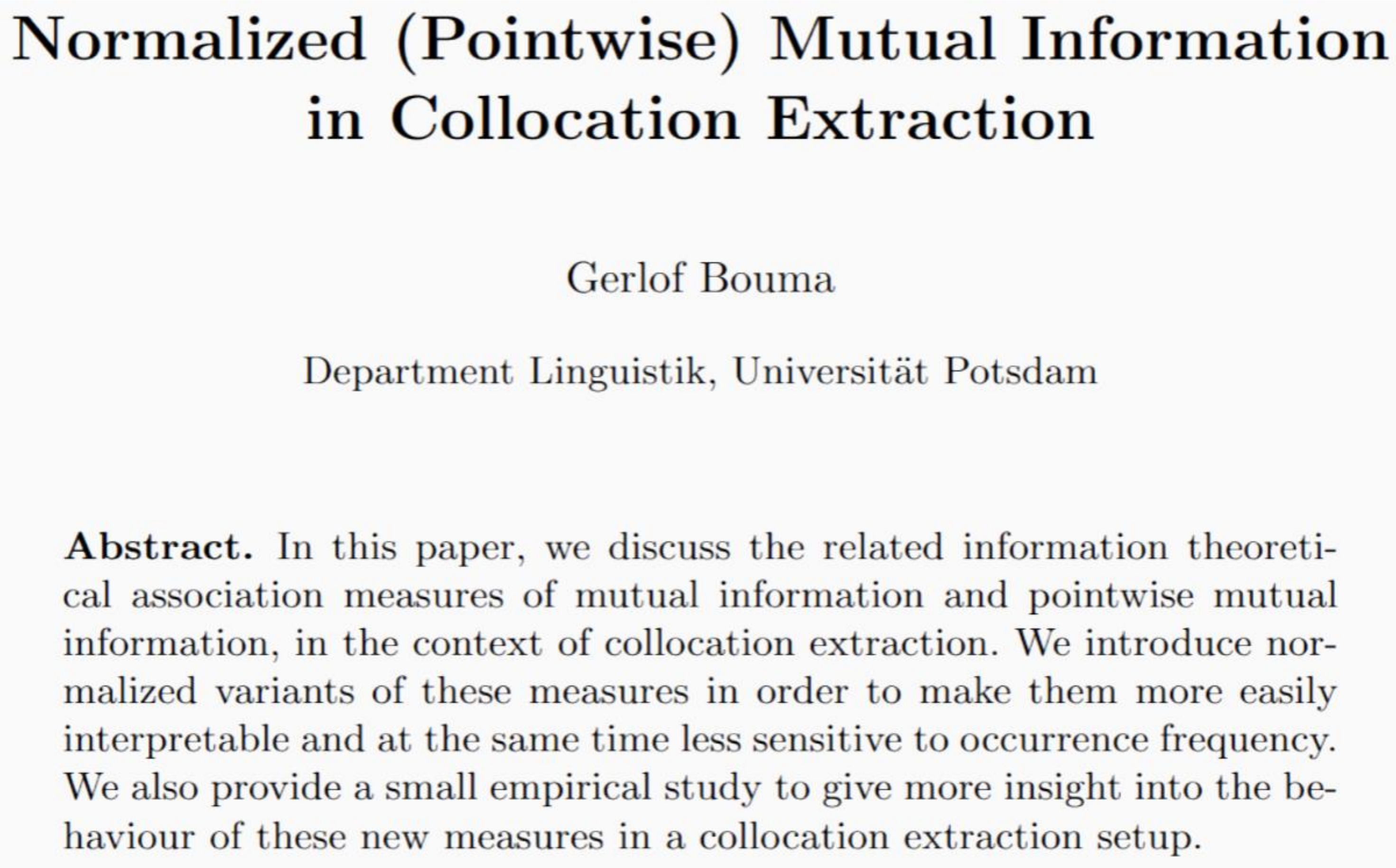
Collocation Extraction

We can quantify the observed differences using the pointwise mutual information (PMI) measure

$$\mathrm{PMI}(A,B) \equiv \ln \frac{P(A,B)}{P(A)\cdot P(B)} = \ln \frac{P(A|B)}{P(A)} = \ln \frac{P(B|A)}{P(B)}$$

	Apple	Facebook	Tesla
Elon Musk	-0.691	-2.935	**2.579**
Mark Zuckerberg	-0.301	**0.046**	-0.432
Tim Cook	**2.608**	-1.938	-0.519

Collocation Extraction



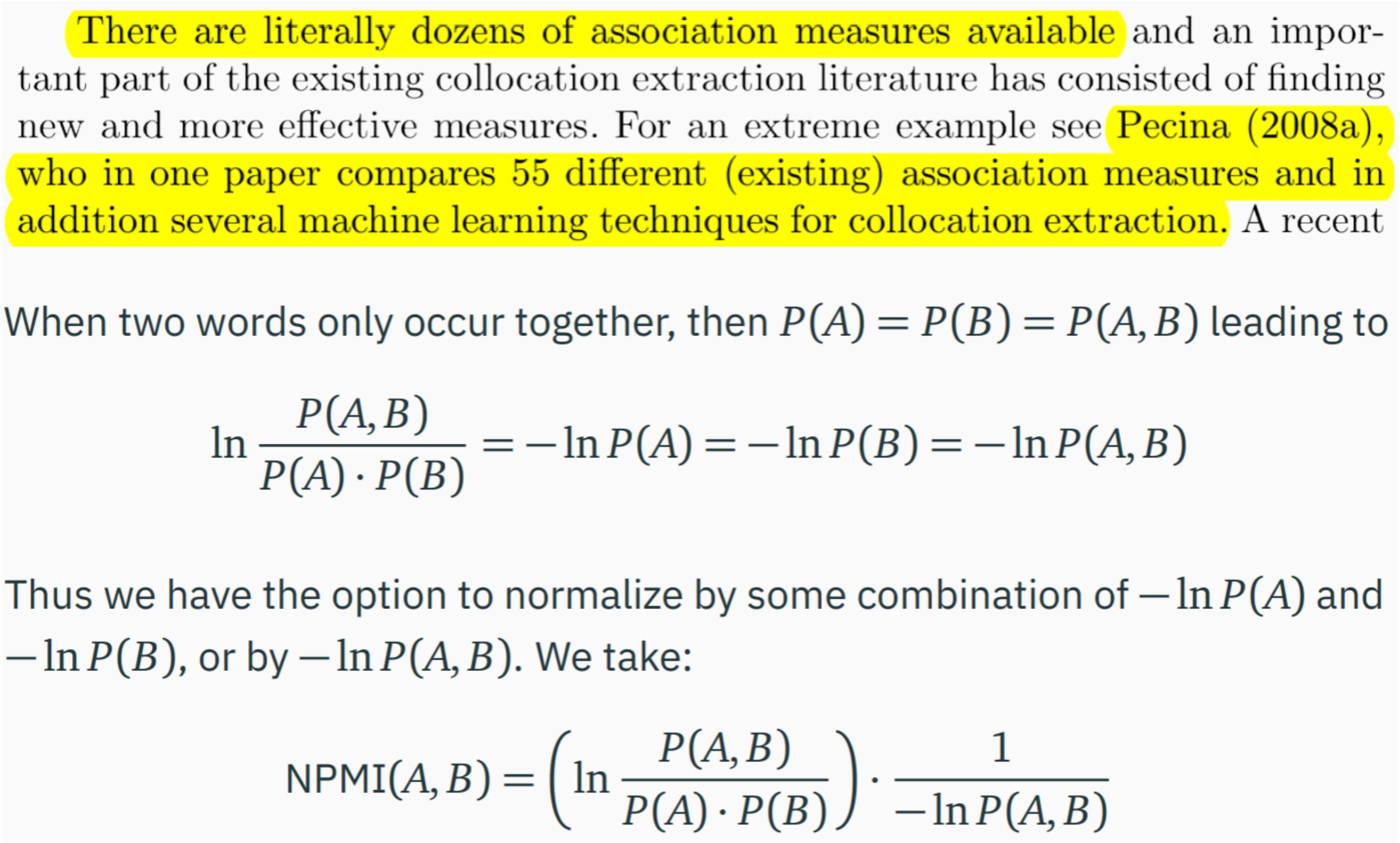
Normalized (Pointwise) Mutual Information in Collocation Extraction

Gerlof Bouma

Department Linguistik, Universität Potsdam

Abstract. In this paper, we discuss the related information theoretical association measures of mutual information and pointwise mutual information, in the context of collocation extraction. We introduce normalized variants of these measures in order to make them more easily interpretable and at the same time less sensitive to occurrence frequency. We also provide a small empirical study to give more insight into the behaviour of these new measures in a collocation extraction setup.

Collocation Extraction



There are literally dozens of association measures available and an important part of the existing collocation extraction literature has consisted of finding new and more effective measures. For an extreme example see Pecina (2008a), who in one paper compares 55 different (existing) association measures and in addition several machine learning techniques for collocation extraction. A recent

When two words only occur together, then $P(A) = P(B) = P(A, B)$ leading to

$$\ln \frac{P(A,B)}{P(A) \cdot P(B)} = -\ln P(A) = -\ln P(B) = -\ln P(A,B)$$

Thus we have the option to normalize by some combination of $-\ln P(A)$ and $-\ln P(B)$, or by $-\ln P(A,B)$. We take:

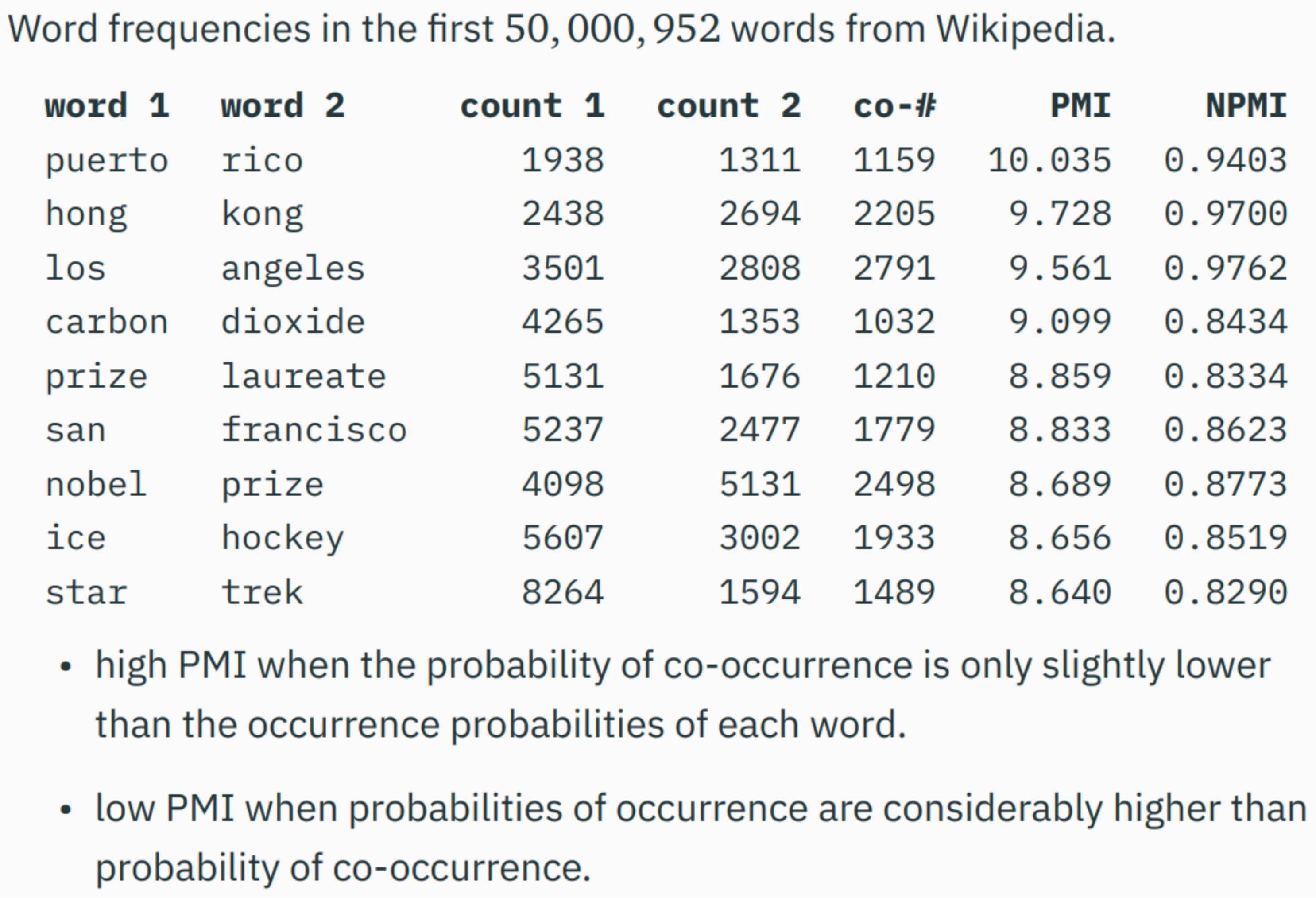
$$\text{NPMI}(A,B) = \left(\ln \frac{P(A,B)}{P(A) \cdot P(B)} \right) \cdot \frac{1}{-\ln P(A,B)}$$

Collocation Extraction

Normalized pointwise mutual information

	Apple	**Facebook**	**Tesla**
Elon Musk	-0.098	-0.441	**0.706**
Mark Zuckerberg	-0.096	**0.314**	-0.138
Tim Cook	**0.643**	-0.326	-0.074

Collocation Extraction

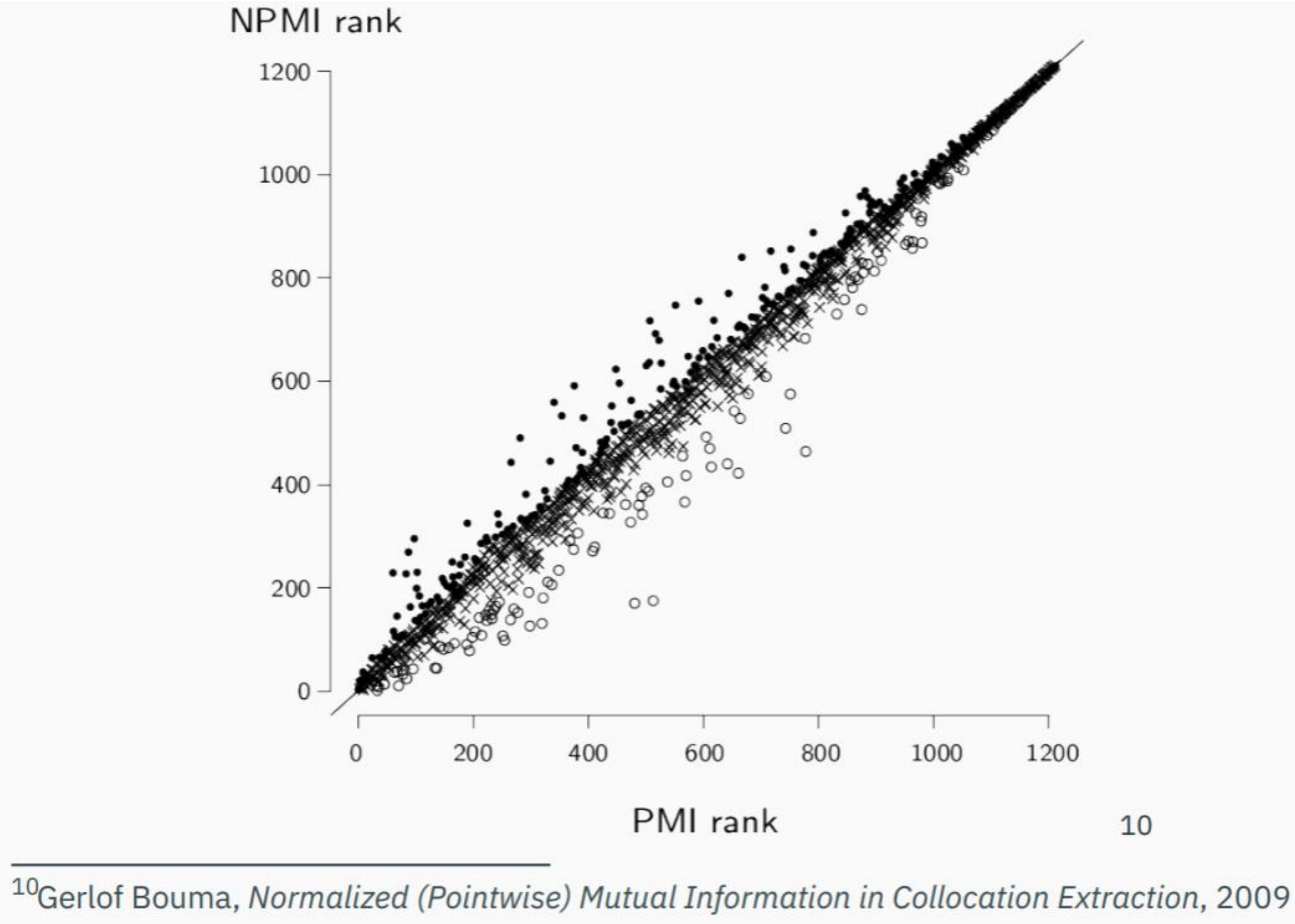


Word frequencies in the first 50, 000, 952 words from Wikipedia.

word 1	word 2	count 1	count 2	co-#	PMI	NPMI
puerto	rico	1938	1311	1159	10.035	0.9403
hong	kong	2438	2694	2205	9.728	0.9700
los	angeles	3501	2808	2791	9.561	0.9762
carbon	dioxide	4265	1353	1032	9.099	0.8434
prize	laureate	5131	1676	1210	8.859	0.8334
san	francisco	5237	2477	1779	8.833	0.8623
nobel	prize	4098	5131	2498	8.689	0.8773
ice	hockey	5607	3002	1933	8.656	0.8519
star	trek	8264	1594	1489	8.640	0.8290

- high PMI when the probability of co-occurrence is only slightly lower than the occurrence probabilities of each word.
- low PMI when probabilities of occurrence are considerably higher than probability of co-occurrence.

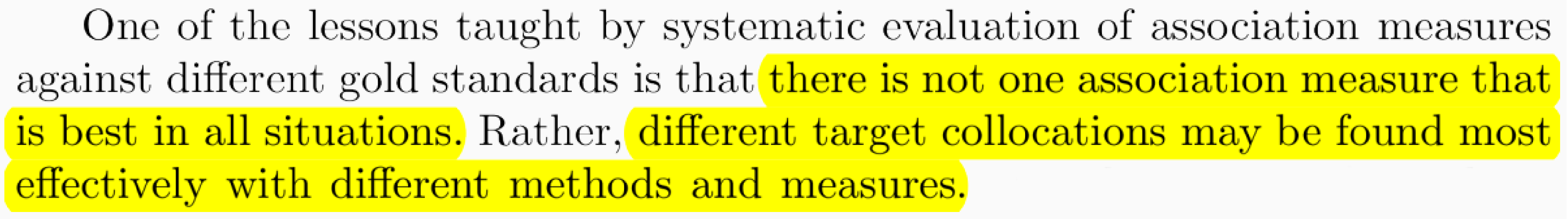
Collocation Extraction



[10] Gerlof Bouma, *Normalized (Pointwise) Mutual Information in Collocation Extraction*, 2009

Collocation Extraction

No one measure is best



One of the lessons taught by systematic evaluation of association measures against different gold standards is that there is not one association measure that is best in all situations. Rather, different target collocations may be found most effectively with different methods and measures.

Co-occurrrence analysis applications

- Products & dates Anticipate when certain products are likely to be purchased/rented/consumed more.
- Products & locations Anticipate where certain products are likely to be purchased/rented/consumed more.

Collocation Extraction

What we've learned

- We can find direct connections by mining text data
- We can quantify the strength of the connection

Shortcomings

- No indirect connections are captured
- We cannot infer causal relationship