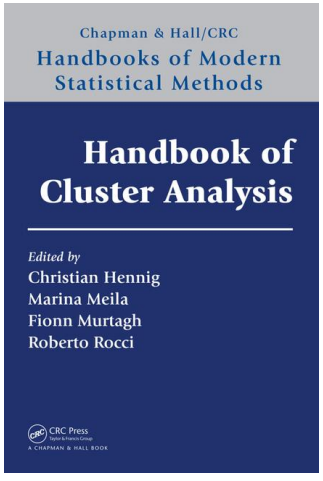
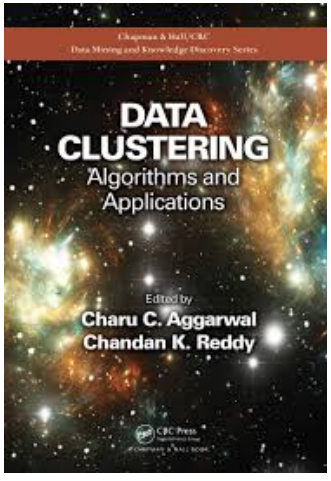
Clustering

Stéphane Canu
asi.insa-rouen.fr/enseignants/~scanu
scanu@insa-rouen.fr

INSA Rouen Normandie





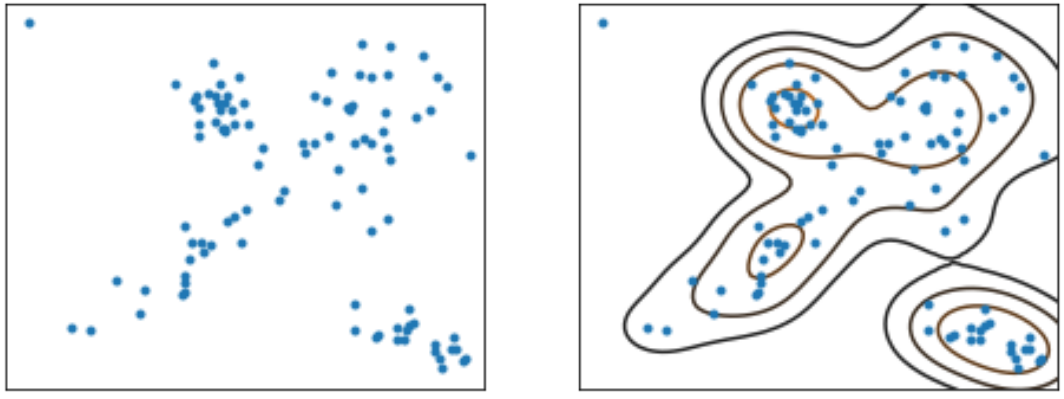
AML, 16 Décembre 2024

The 3 main kinds of machine learning

Motivation

Unsupervised learning data and model:

$$\{x_i\}_{i=1,\dots,n} \quad \sim \quad \mathbb{P}(x)$$



- density estimation: $\widehat{\mathbb{P}}(x) \sim \mathbb{P}(x)$
- visualization: $\phi : \mathbb{R}^p \to \mathbb{R}^q, \quad q < p$
- clustering: provide groups of alike objects

https://scikit-learn.org/stable/modules/clustering.html

Clustering applications

Marketing: customer segmentation

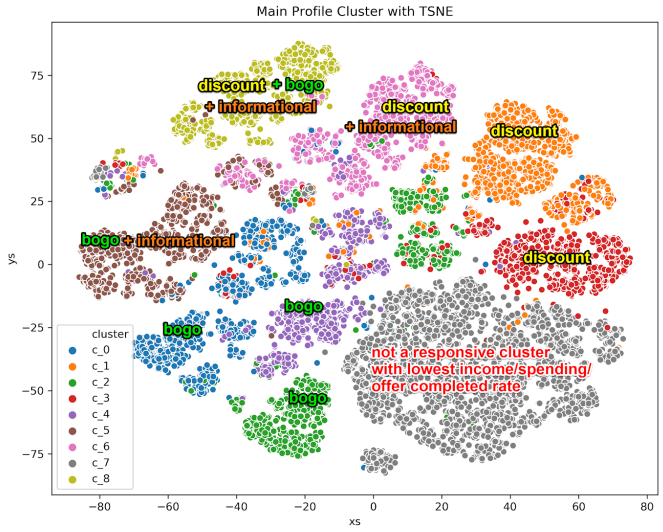
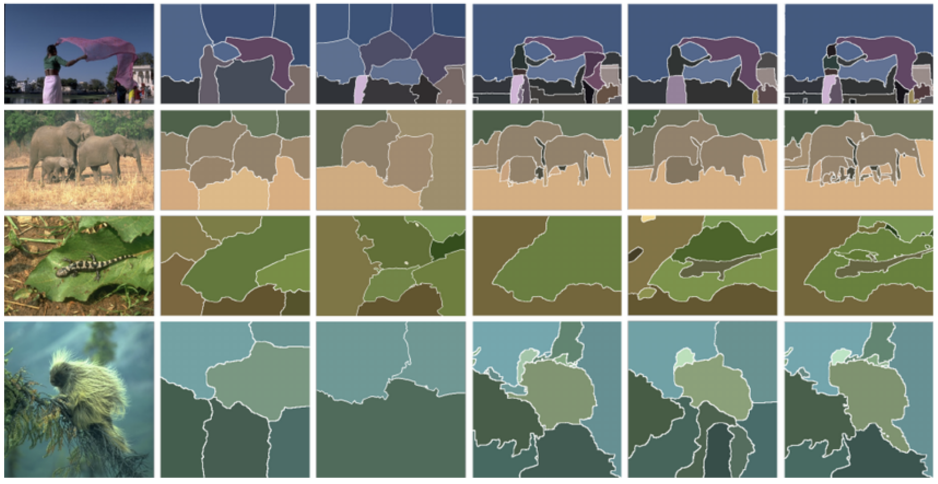


Image segmentation .



- Galaxy types: retreive galaxy types, Doccument analysis: medical records, Pizza hut position (delivery store optimization), quantification...
- Clustering with different data type: Categorical variables, Text, images, Multimedia, Time-Series, Discrete Sequences, Network Data and bi clustering, co clustering...
- Clustering also provides representations (prototype, quantification...)

Clustering at sklearn

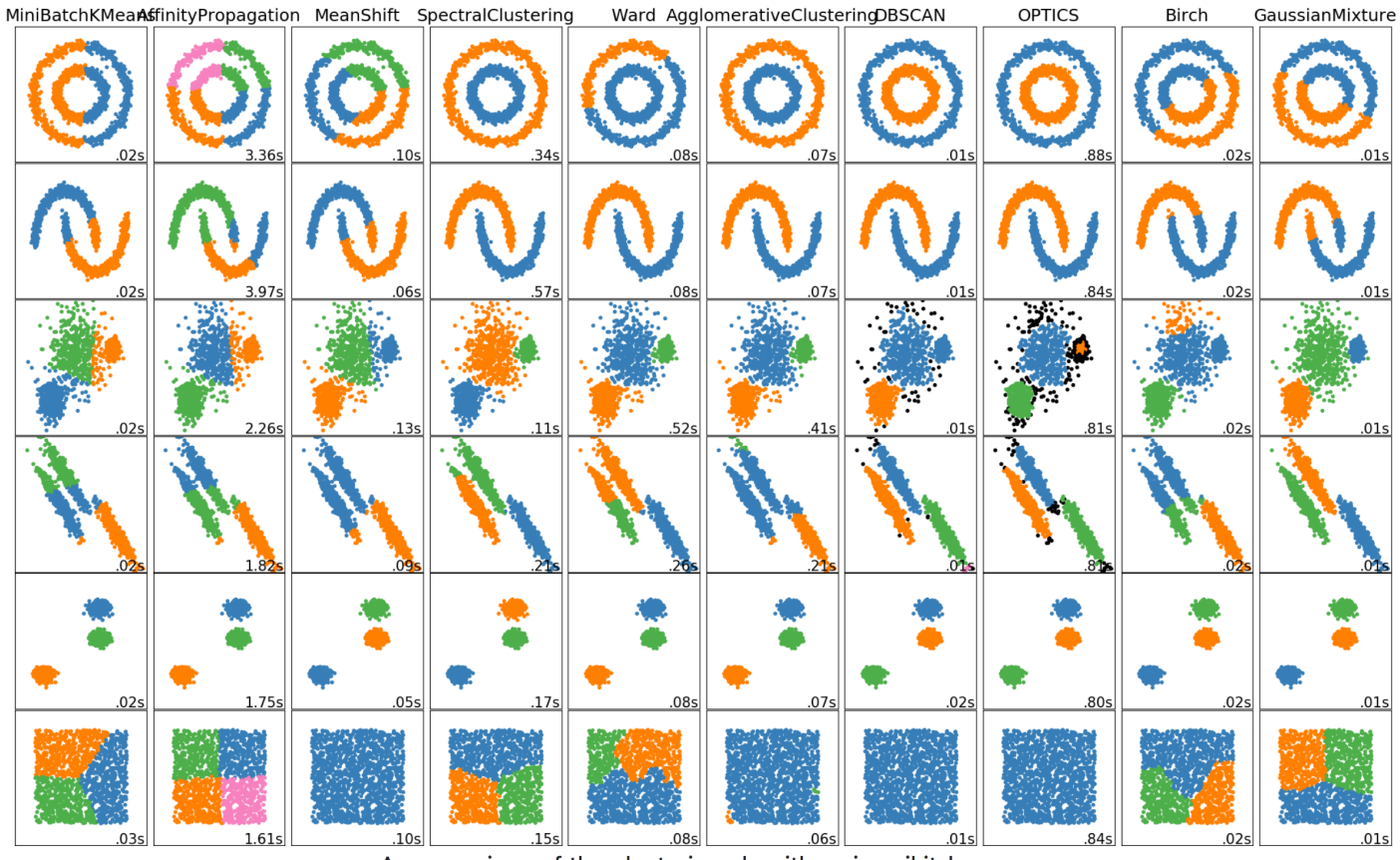
scikit learn Install User Guide API Examples More

Prev Up Next

scikit-learn 0.22.1
Other versions

Please **cite us** if you use the software.

2.3. Clustering
2.3.1. Overview of clustering methods
2.3.2. K-means
2.3.3. Affinity Propagation
2.3.4. Mean Shift
2.3.5. Spectral clustering
2.3.6. Hierarchical clustering
2.3.7. DBSCAN
2.3.8. OPTICS
2.3.9. Birch
2.3.10. Clustering performance evaluation



A comparison of the clustering algorithms in scikit-learn

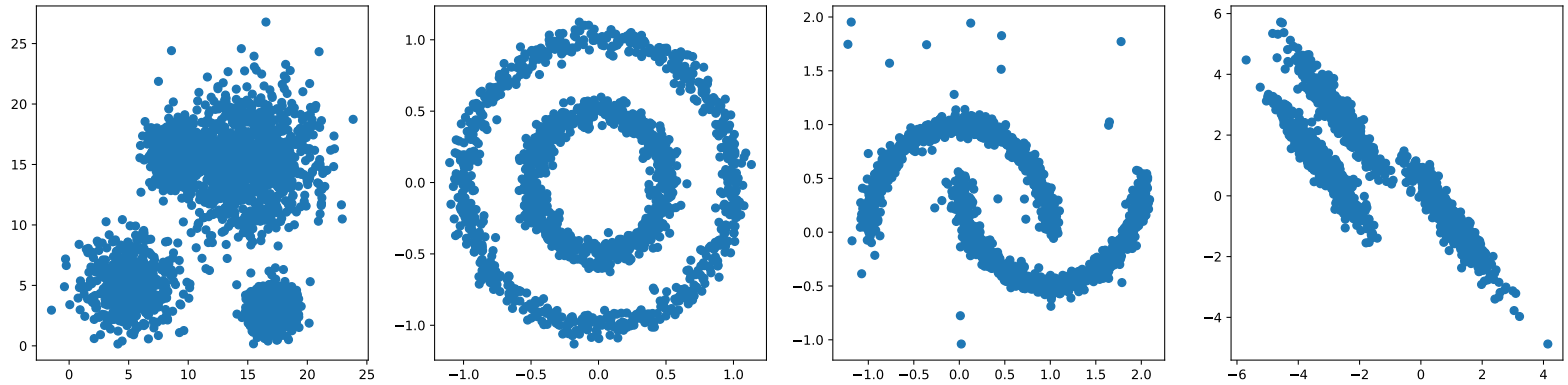
https://scikit-learn.org/stable/modules/clustering.html

Lecture road map

1. The clustering problem
2. Distance, proximity graph, densities and spanning trees
3. Parametric clustering
4. Partitional and Hierarchical Clustering Algorithms
5. Density-Based Clustering

The Clustering problem

Clustering is the task of **grouping a set of objects** in such a way that objects in the same group are more similar (in some sense) to each other than to those in other clusters (Wikipedia)



Clustering as a bi objective optimization problem:

- minimize some intra-cluster energy (distance)
- maximize some inter-cluster energy (distance)

Different representations of clustering

Hard affectation

- $z_{ik} = \begin{cases} 1 & \text{if observation } i \text{ belongs to cluster } k \\ 0 & \text{else} \end{cases}$

  Z is a $n \times \kappa$ membership matrix
- $a_{ij} = \begin{cases} 1 & \text{if observations } i \text{ and } j \text{ belong to the same cluster} \\ 0 & \text{else} \end{cases}$

  A is a $n \times n$ symmetric adjacency matrix

Soft affectation

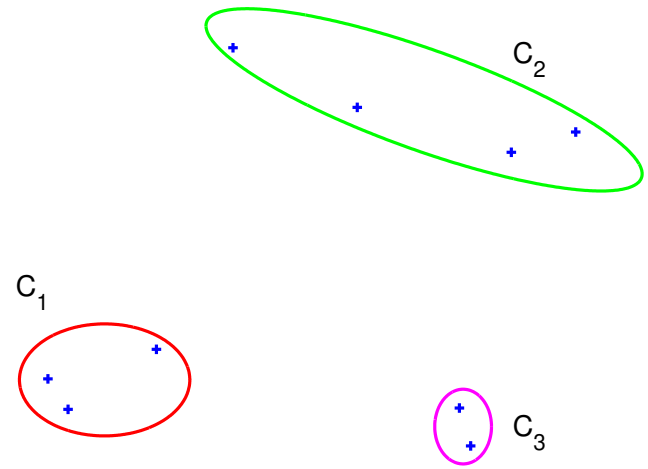
- $p_{ik} =$ probability that point i belongs to cluster k
- $\alpha_{ij} =$ probability that points i and j belong to the same cluster

What about outliers?

Clustering as a partition problem

$$z_{ik} = \begin{cases} 1 & \text{if observation } i \text{ belongs to cluster } k \\ 0 & \text{else} \end{cases}$$

Observation	Cluster 1	Cluster 2	Cluster 3
x_1	1	0	0
x_2	0	1	0
x_3	1	0	0
x_4	0	0	1
x_5	0	1	0
x_6	0	0	1
x_7	0	1	0
x_8	1	0	0
x_9	0	1	0



Minimize some energy and maximize some entropy

Enumerate of all possible $\left(\{0,1\},\{0,1\},\{0,1\}\right)^n$ configurations such that each point belongs to only one cluster.

This is a $k = 3$-partition problem.

Clustering as a Mixed interger program

Grötschell-Wakabayashi formulation (1989) with k cluster

$$
\begin{cases}
\min\limits_{A\in\{0,1\}^{n^2},r\in\{0,1\}^n} & \displaystyle\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} a_{ij}\|\mathbf{x}_i-\mathbf{x}_j\|^2 & \\
\text{with} & a_{ij}+a_{j\ell}-a_{i\ell}\leq 1 & i=1,n, j=1,n, \ell=1,n \\
& a_{ij}=a_{ji} & i=1,n, j=1,n \\
& \displaystyle\sum_{i=1}^{n} r_i = k & \\
& r_j + a_{ij} \leq 1 & j=1,n, i=1,j-1 \\
& r_j + \sum_{i=1}^{j-1} a_{ij} \geq 1 & j=1,n.
\end{cases}
$$

$r_j = 1$ si x_j est le plus petit point de sa partition

Clustering issues

1. How to deal with computational issues?
   - distance matrix D is $n \times n$
2. How to assess the quality of a partition?
3. How to represent a cluser?
   - prototype, center, shape...
4. How to decide the number of cluster?
   - Why not put each data point into a separate class?
   - What is the payoff for clustering things together?
5. What if each data vector can be classified in many different ways?
6. What for?

Clustering is an ill posed problem

Data clustering has been used with different objectives:

- Identify an underlying structure
- Retrieve a "natural" classification
- Data compression via prototypes

What is good clustering?

AK Jain, Data clustering: 50 years beyond K-means, 2010

Clustering performance evaluation

Evaluation of clustering results is as difficult as the clustering itself

Popular approaches involve :

- "internal" evaluation: clustering is summarized to a single score
  - Silhouette coefficient
  - Calinski-Harabasz Index
  - Davies–Bouldin index
  - Dunn index
- "external" evaluation, clustering is compared to some "ground truth",
  - Purity
  - Rand index (William M. Rand 1971)
  - F-measure
  - ...
- "manual" evaluation by a human expert,
- "indirect" evaluation by evaluating the utility of the clustering

https://scikit-learn.org/stable/modules/clustering.html

Internal evaluation

Silhouette Coefficient: the mean over all examples x_i

- a mean distance between x_i and all other points in the same class.
- b mean distance with all other points in the nearest cluster.

$$\frac{b-a}{max(a,b)}$$

Calinski-Harabasz Index: the ratio of the between-clusters dispersion mean and the within-cluster dispersion

$$B = \sum_k n_k (c_k - c)(c_k - c)^\top$$
$$W = \sum_k \frac{1}{n_k} \sum_{x_i \in C_k} (x_i - c_k)(x_i - c_k)^\top$$

$$\frac{tr(B)}{tr(W)} \frac{n-\kappa}{\kappa - 1}$$

Davies-Bouldin Index:

s_k is the average distance between each point of cluster k and c_k the centroid of that cluster – also know as cluster diameter.

$$\frac{1}{\kappa} \sum_k \max_{q \neq k} \frac{s_k + s_q}{\|c_k - c_q\|}$$

Lecture road map

1. The clustering problem
2. **Distance, proximity graph, densities and spanning trees**
3. Parametric clustering
4. Partitional and Hierarchical Clustering Algorithms
5. Density-Based Clustering

Distance matrix and similarity graph

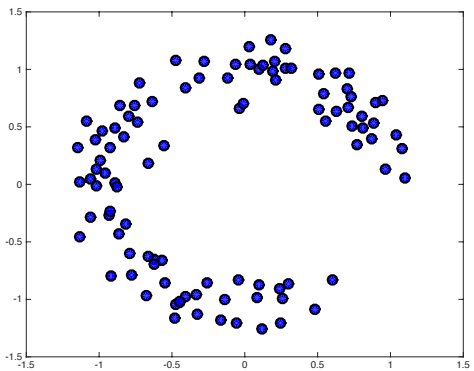
Distance matrix D. e.g. using the Euclian metric $d_{ij} = ||x_i - x_j||$
The associated similarity matrix S

$$S_{ij} = -D_{ij} \quad \text{or} \quad \frac{1}{1 + d_{ij}} \quad \text{or} \quad \exp^{-D_{ij}} \quad \text{or} \quad \ldots$$

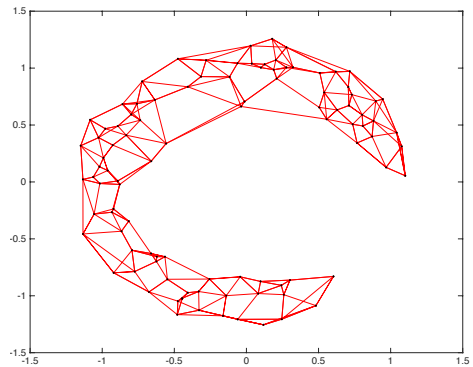
The corresponding connection graphs:

- a proximity graph
- the minimum spanning tree
- the dendogram ...

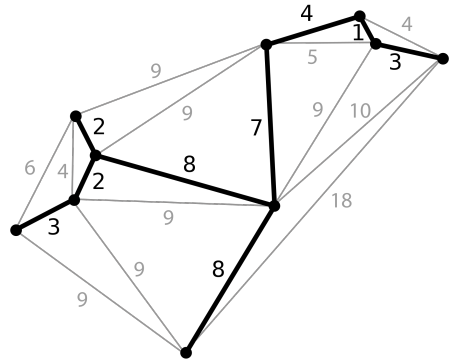
cloud of points



proximity graph



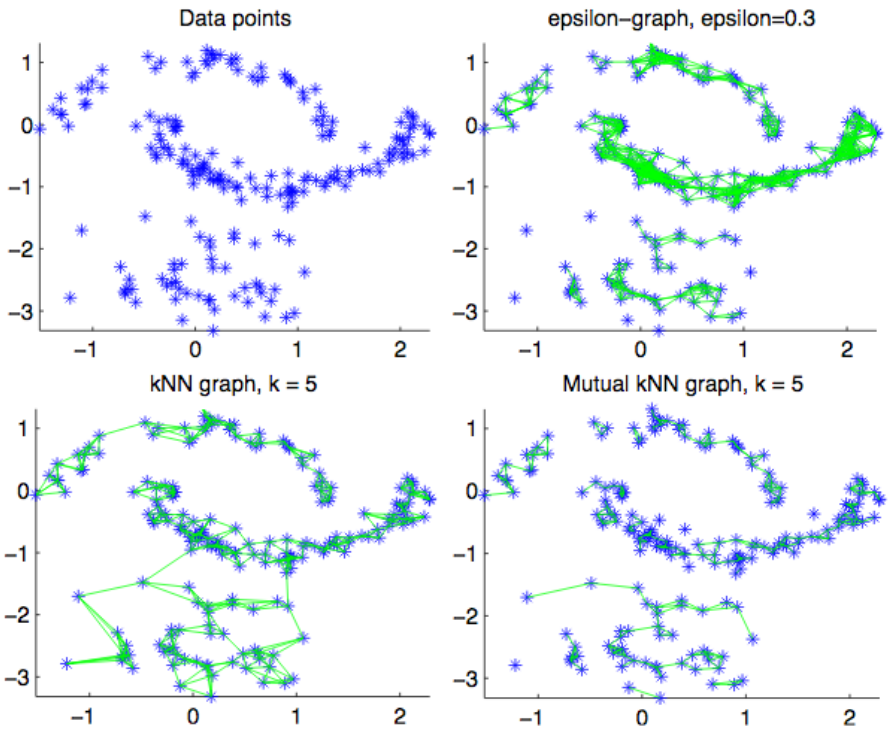
minimum spanning tree



Distance matrix and proximity graph

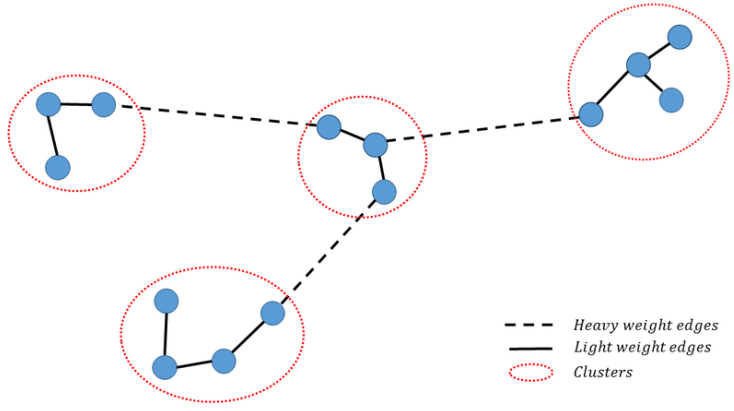
- Represent data with a similarity graph $G = (V, E)$ whose vertices are data points ($|V| = n$)
  - k-nearest neighbor graphs
  - ε-neighborhood graph
- a graph is associated with an adjacency matrix A with $A_{ij} \in \{0, 1\}$
- The matrix of edge weights W from a similarity graph

$$W_{ij} = S_{ij} \qquad \text{for instance } = \exp^{\frac{-D(i,j)}{c}}$$



Minimum spanning tree for clustering

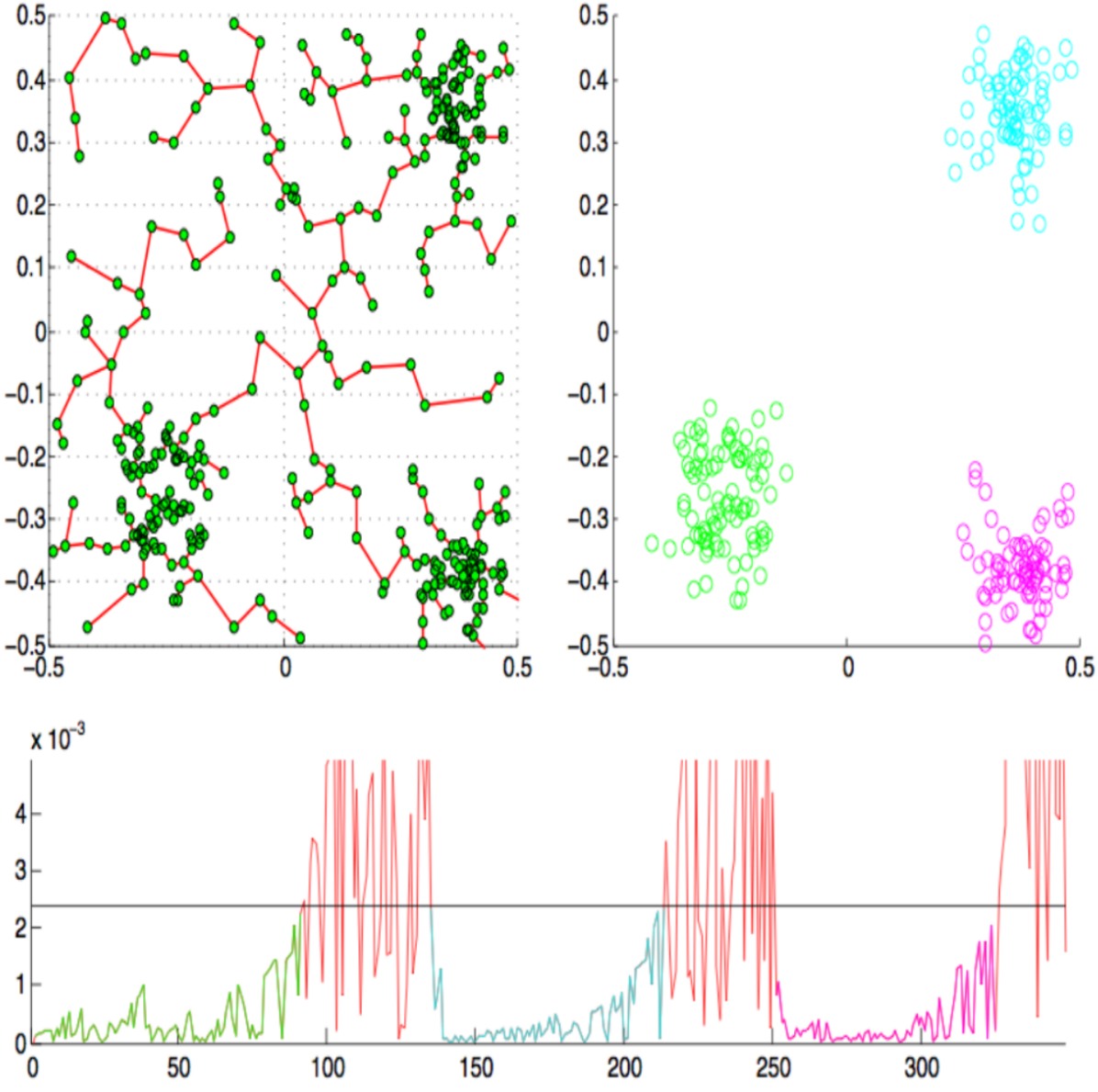
A minimum spanning tree (MST) or minimum weight spanning tree is a subset of the edges of a connected, edge-weighted undirected graph that connects all the vertices together, without any cycles and with the minimum possible total edge weight



Algorithms:

- Prim's (also known as Jarník's) algorithm
- Kruskal's algorithm can be shown to run in $\mathcal{O}(ElogE)$
- fast MST (B. Chazelle, 2000)

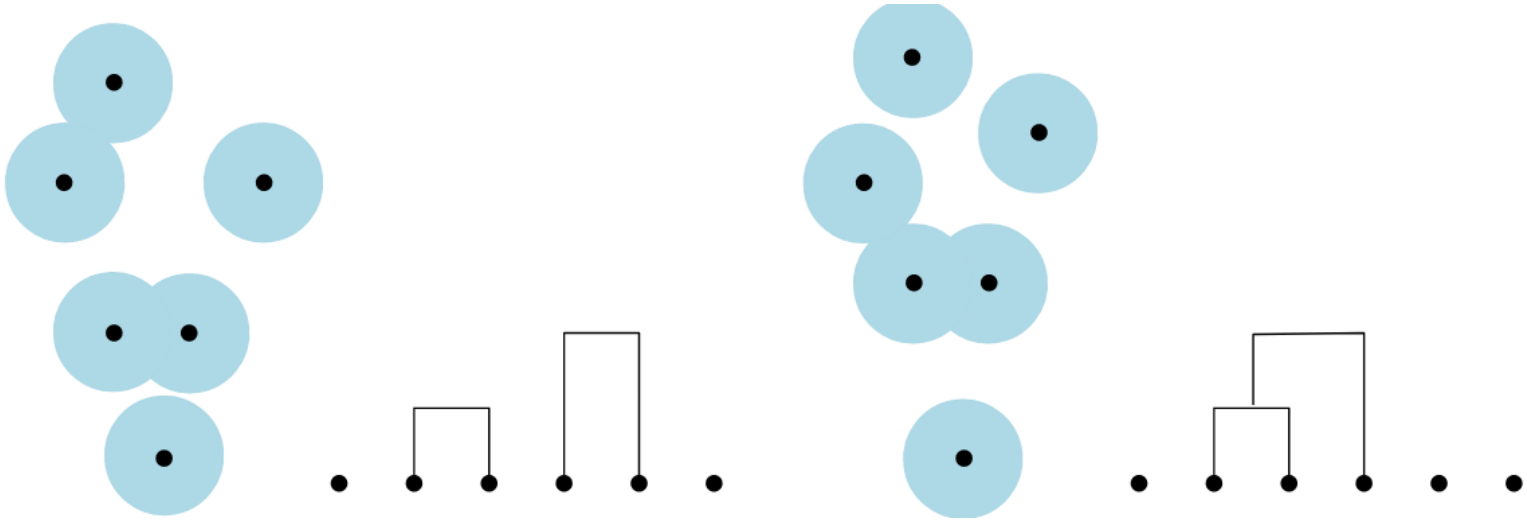
Distance matrix and minimum spanning tree



L. Galluccio et al. 2010

Dendogram

A dendrogram is a diagram representing a tree
Data groups are connected depending on their distance (sort n^2 distances)



bottom up or top down.
rather unstable!

Lecture road map

k-means

Cost function:

$$\min_{z,c} \sum_i \sum_k z_{ik} ||x_i - c_k||^2$$

Associated probabilist model:

$$\mathbb{P}(x) = \sum_k \mathbb{P}(k)\mathbb{P}(x|k) = \frac{1}{Z} \sum_k \omega_k \exp^{-\frac{1}{2}||x-c_k||^2}$$

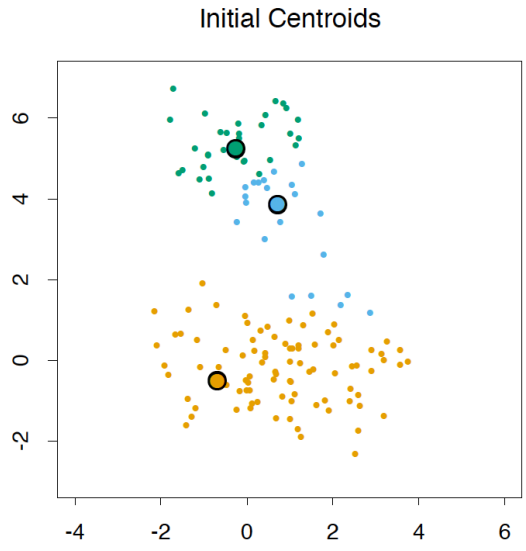
EM algorithm: iterate until convergence

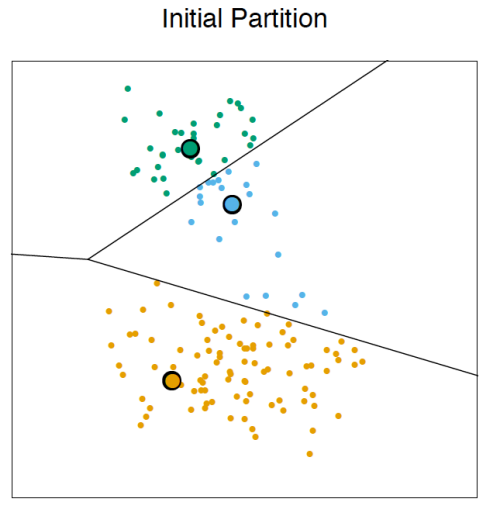
1. assignment: E-step $z_{ik} = 1$ if $k = \arg\min_j ||x_i - c_j||^2$
2. refitting: M-step $c_k = \frac{\sum_i z_{ik} x_i}{\sum_i z_{ik}}$

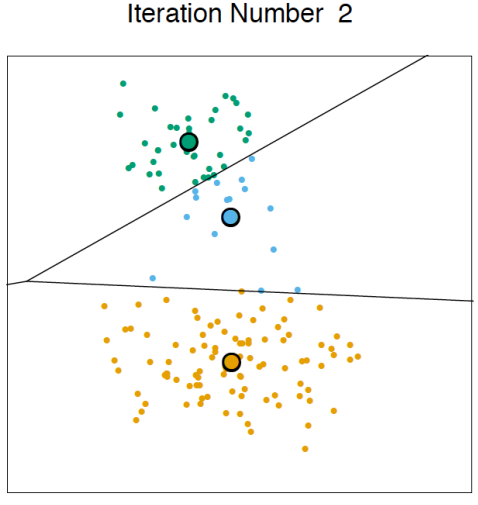
Algorithm 14.1 *K-means Clustering.*

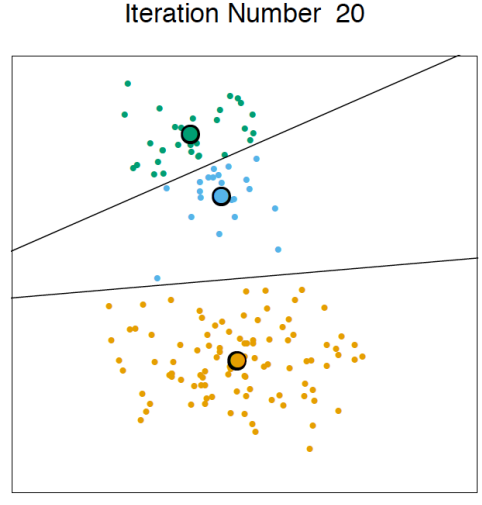
1. For a given cluster assignment C, the total cluster variance (14.33) is minimized with respect to $\{m_1, \ldots, m_K\}$ yielding the means of the currently assigned clusters (14.32).
2. Given a current set of means $\{m_1, \ldots, m_K\}$, (14.33) is minimized by assigning each observation to the closest (current) cluster mean. That is,
$$C(i) = \underset{1 \le k \le K}{\operatorname{argmin}} ||x_i - m_k||^2. \qquad (14.34)$$
3. Steps 1 and 2 are iterated until the assignments do not change.

k-means at work

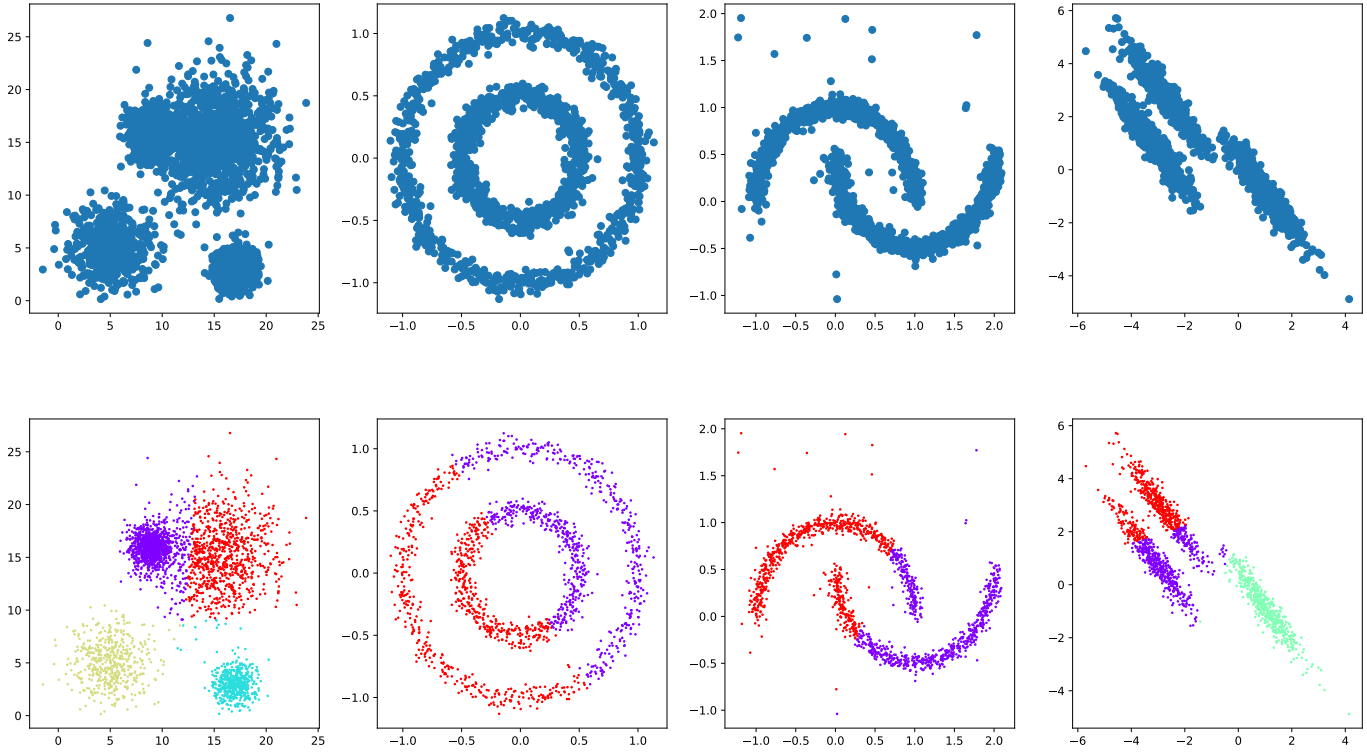








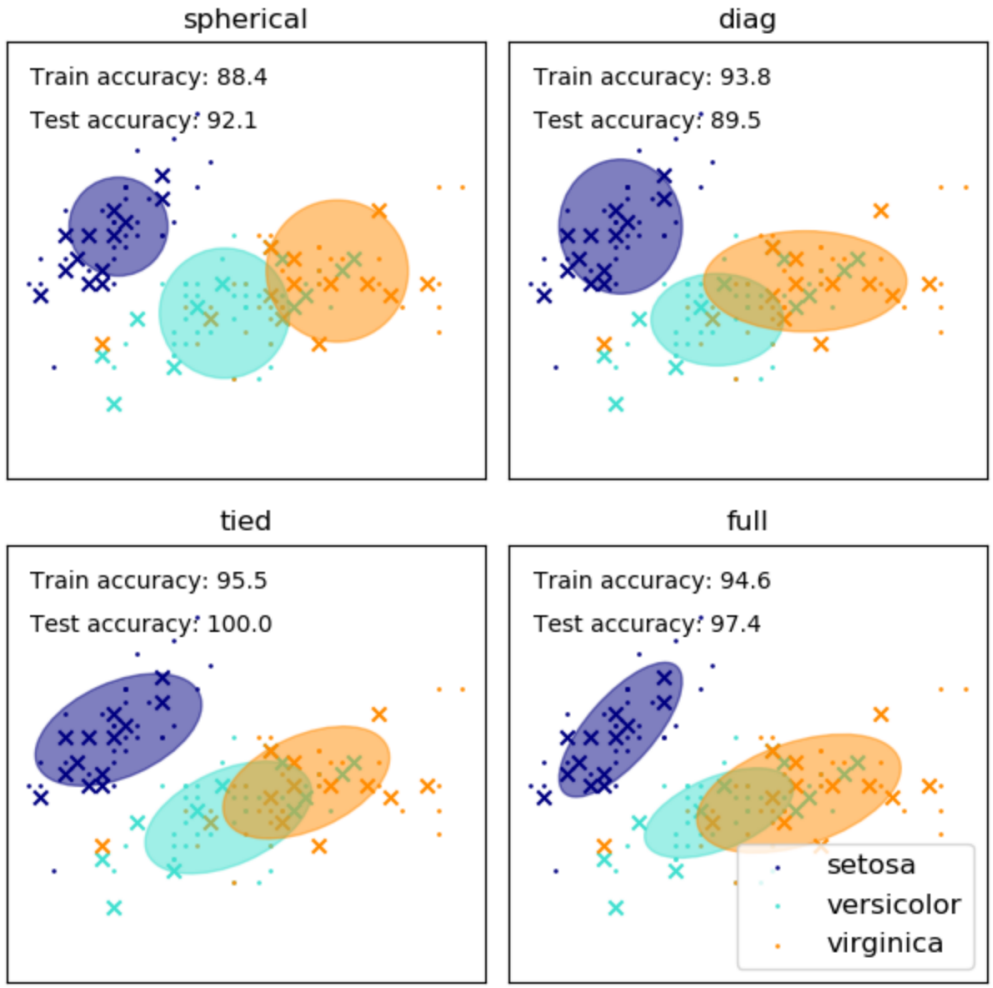
k-means at work



$k = 4$ $k = 2$ $k = 2$ $k = 3$

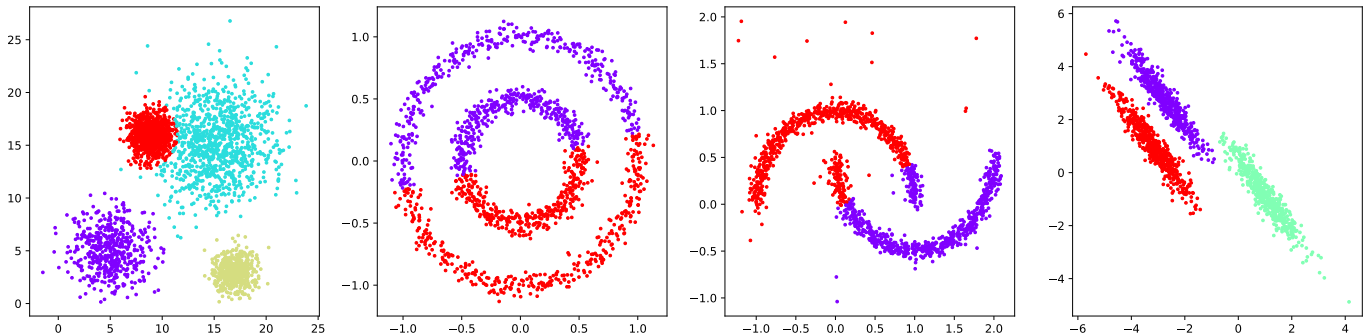
Gaussian Mixture

$$\mathbb{P}(x) = \sum_k \mathbb{P}(k)\mathbb{P}(x|k) = \frac{1}{Z} \sum_k \omega_k \exp^{-\frac{1}{2}(x-c_k)\Sigma_k^{-1}(x-c_k)}$$



https://scikit-learn.org/stable/auto_examples/mixture/plot_gmm_covariances.html

Gaussian Mixture



$k = 4$ $k = 2$ $k = 2$ $k = 3$

Vector quantization

FIGURE 14.9. *Sir Ronald A. Fisher* (1890 – 1962) *was one of the founders of modern day statistics, to whom we owe maximum-likelihood, sufficiency, and many other fundamental concepts. The image on the left is a* 1024×1024 *grayscale image at* 8 *bits per pixel. The center image is the result of* 2×2 *block VQ, using* 200 *code vectors, with a compression rate of* 1.9 *bits/pixel. The right image uses only four code vectors, with a compression rate of* 0.50 *bits/pixel*

Lecture road map

Hierarchical clustering

Two main types of hierarchical clustering

- Agglomerative: Start with the points as individual clusters
- Divisive: Start with one, all-inclusive cluster

Linkage criteria

- Average linkage clustering

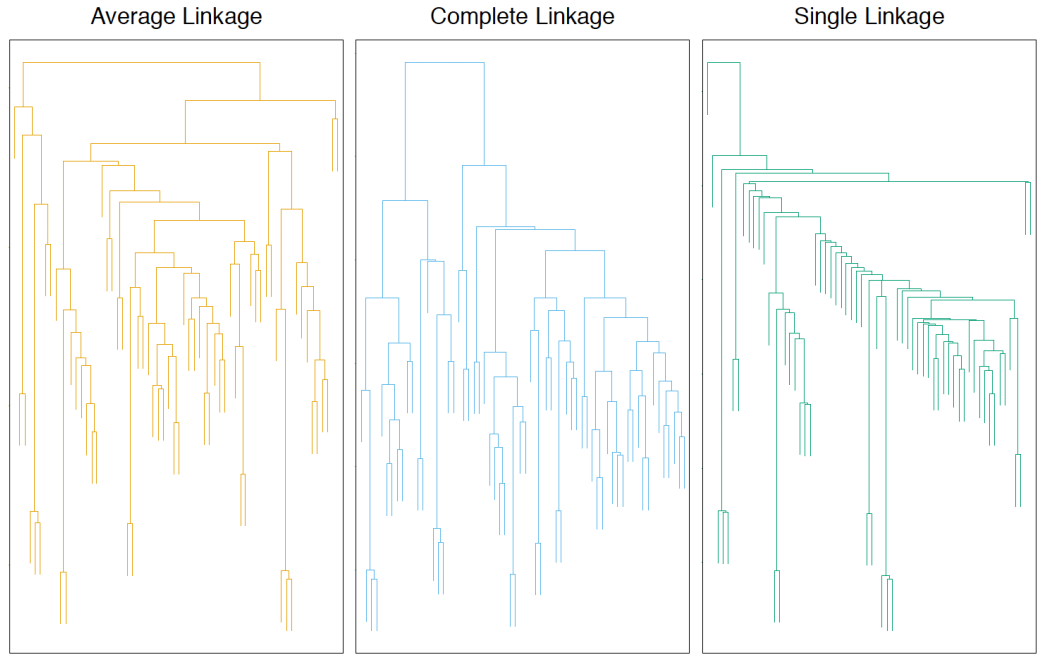
$$\frac{1}{|A||B|} \sum_{x_i \in A} \sum_{x_j \in B} D(x_i, x_j)$$

- Complete-linkage clustering

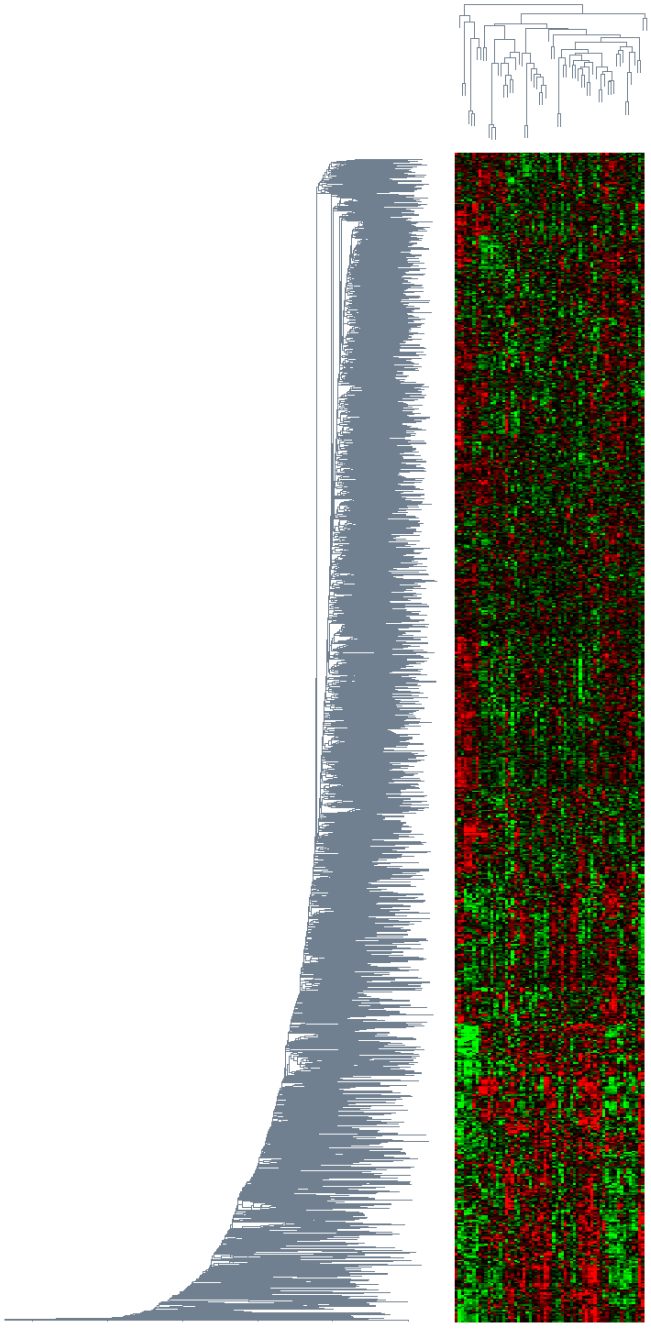
$$\max(D(x_i, x_j) : x_i \in A, x_j \in B)$$

- Single-linkage clustering

$$\min(D(x_i, x_j) : x_i \in A, x_j \in B)$$

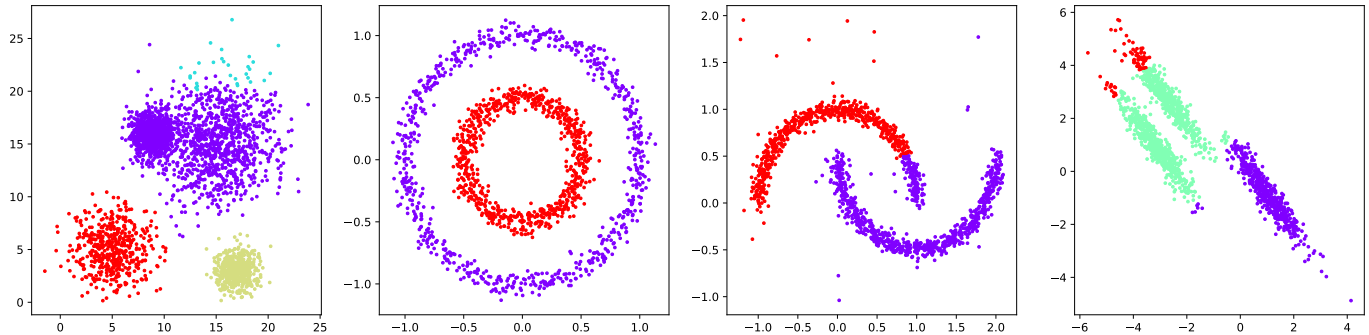


Hierarchical clustering

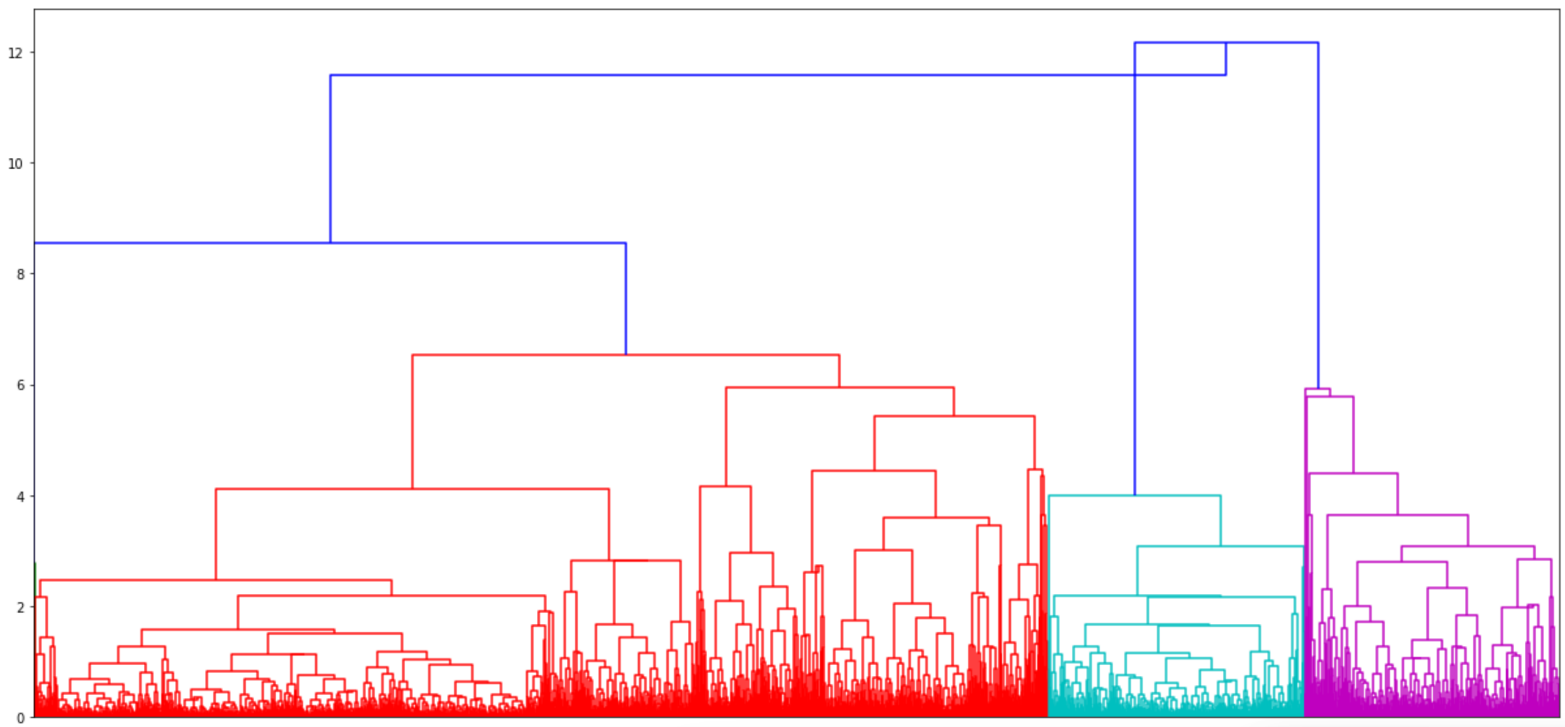


DNA microarray data: average linkage hierarchical clustering has been applied independently to the rows (genes) and columns (samples), determining the ordering of the rows and columns (see text). The colors range from bright green (negative, under-expressed) to bright red (positive, over-expressed).

Hierarchical clustering



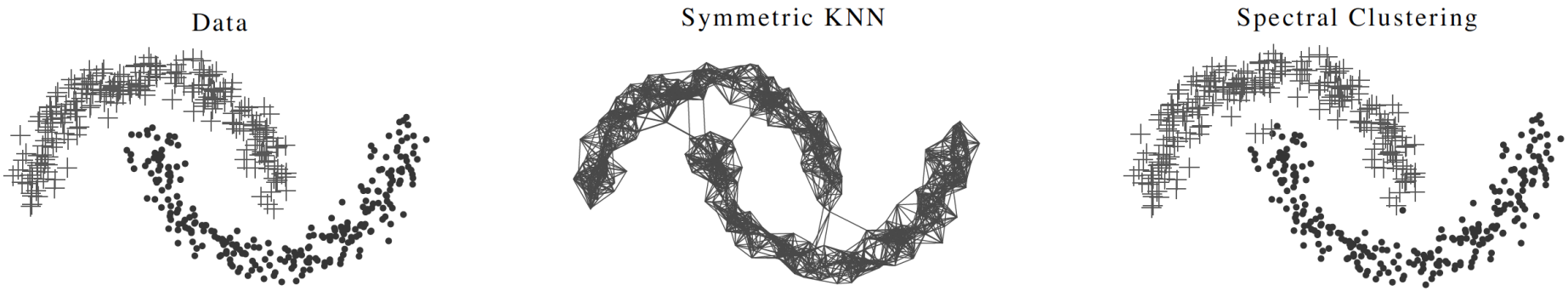
$k = 4$ $k = 2$ $k = 2$ $k = 3$.



scipy.cluster.hierarchy.dendrogram

Spectral clustering (Donath and Hoffman, 1973)

1. represent data with a similarity graph $G = (V, E)$
   use adjacency (or affinity) matrix W to describe G
   - K-nearest neighbor graphs
   - ε-neighborhood graph
   - Fully connected graph
2. The data points are embedded in a space, in which the clusters are more "obvious," with the use of the eigenvectors of the graph Laplacian
3. Finally, use the k-means is applied to partition the embedding (v_2 the eigen vector associated wit the second smallest eigen value or more. . .)



Example illustrating three steps of spectral clustering

Spectral clustering (step 2)

- The matrix of edge weights W from a similarity graph

$$W_{ij} = \exp^{\frac{-D(i,j)}{c}} \quad \text{or} \quad 0$$

- The degree matrix G is the diagonal matrix satisfying $G_{ii} = \sum_{j=1}^{n} W_{ij}$
- The unnormalized graph Laplacian L is defined by

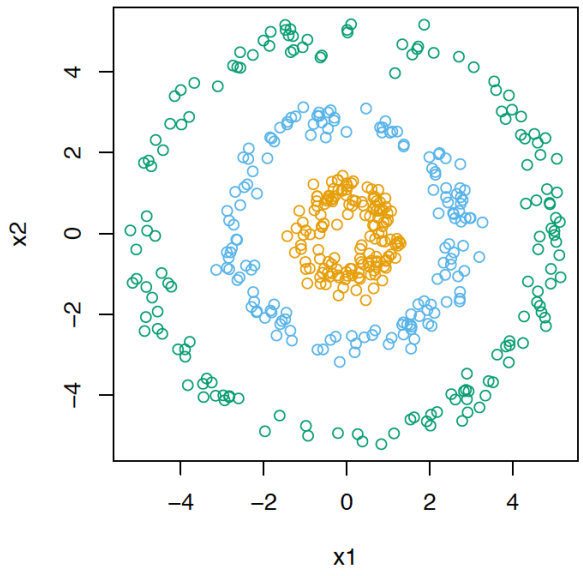
$$L = G - W$$

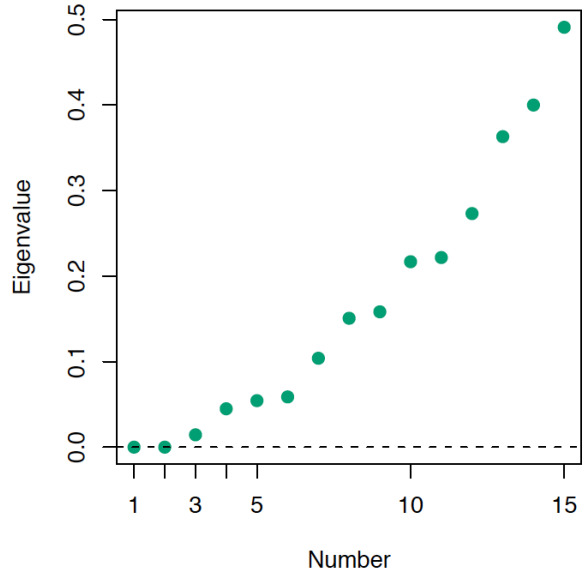
- The normalized graph Laplacians (there exists different)

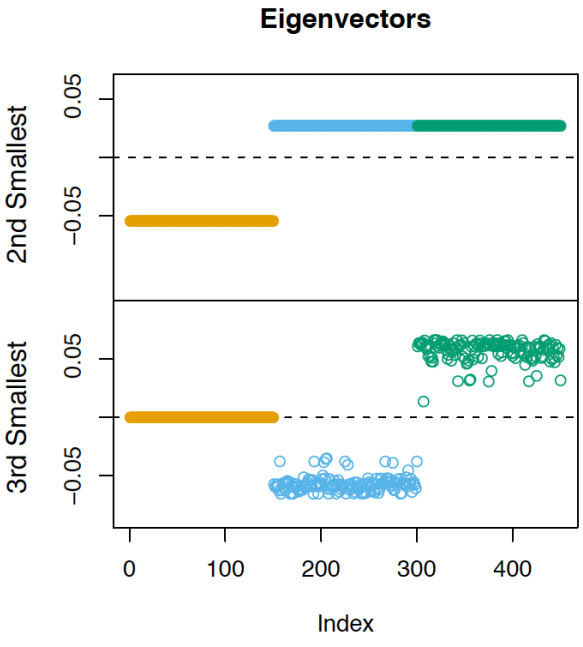
$$L_N = G^{-1}L = I - G^{-1}W$$

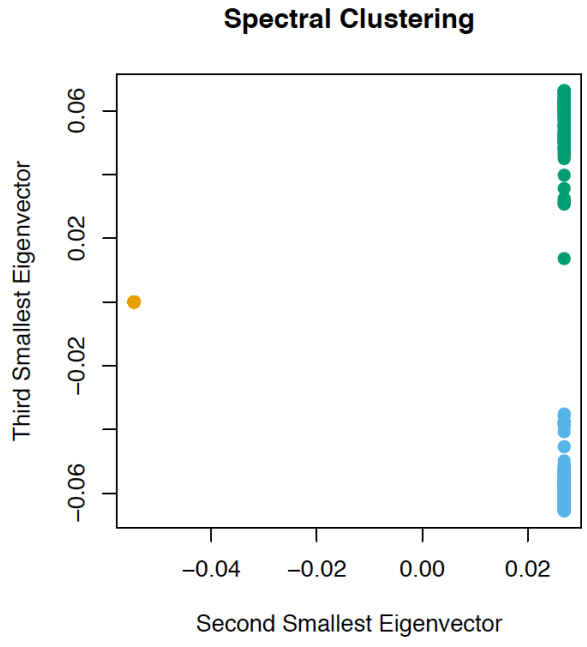
- The matrix L satisfies the following properties:
  - L is symmetric and positive semi-definite.
  - L has n non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \cdots \leq \lambda_n$
  - $\forall x \in \mathbb{R}^n, \quad x^\top Lx = \frac{1}{2}\sum_{ij} w_{ij}(x_i - x_j)^2$
  - the multiplicity of the eigenvalue 0 of L equals the number of connected components

Spectral clustering: L eigenvalues

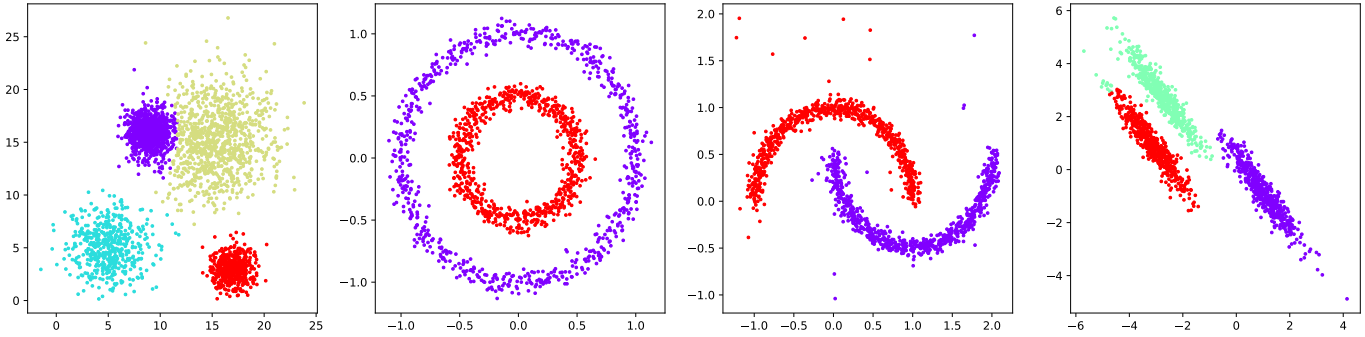








Spectral clustering



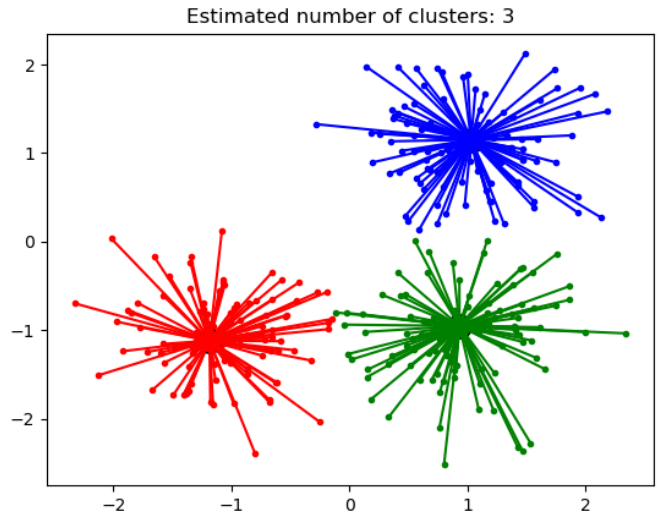
$k = 4$ $k = 2$ $k = 2$ $k = 3$

Affinity propagation (Frey and Dueck, 2007)

- Given
  - data $(x_1, \ldots, x_n)$
  - the "distance" matrix D with $D_{ij} = ||x_i - x_j||^2$
  - and modified diagonal $D_{ii} = d^\star$
- Split the data into 2 sets:
  - exemplar data points (cluster center) $\{\varphi(i), i = 1, \ldots, n\}$
  - and non-exemplar (other data points related with an exemplar)

find $\varphi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$

$$\min_{\varphi} \sum_{i=1}^{n} D(i, \varphi(i))$$



- $d^\star = 0$: n cluster (each pojnt is a cluster)
- $d^\star = \infty$: only one big cluster

Affinity propagation

- To split the data into 2 sets, build
  - The "responsibility" matrix R
    $r(i,k)$ quantify how well-suited x_k is to serve as the exemplar for x_i,

$$r(i,k) = -D(i,k) - \max_{j \neq k} \big(-D(i,j) + a(i,j)\big)$$

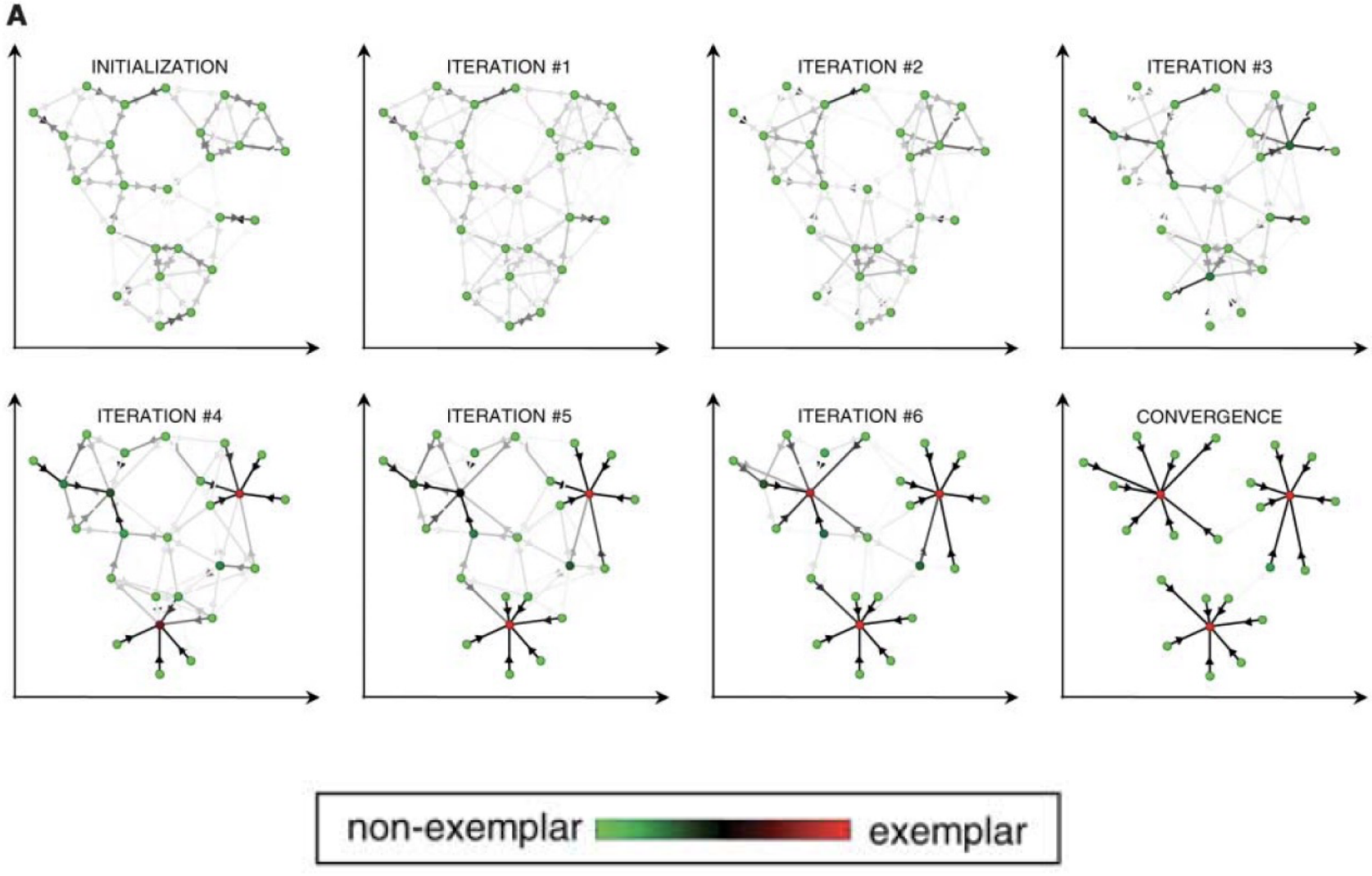
  - The "availability" matrix A
    $a(i,k)$ represent how "appropriate" for x_i to pick x_k as its exemplar
- The exemplars are extracted from the final matrices as those whose 'responsibility + availability' for themselves is positive

$$r(i,i) + a(i,i) > 0$$

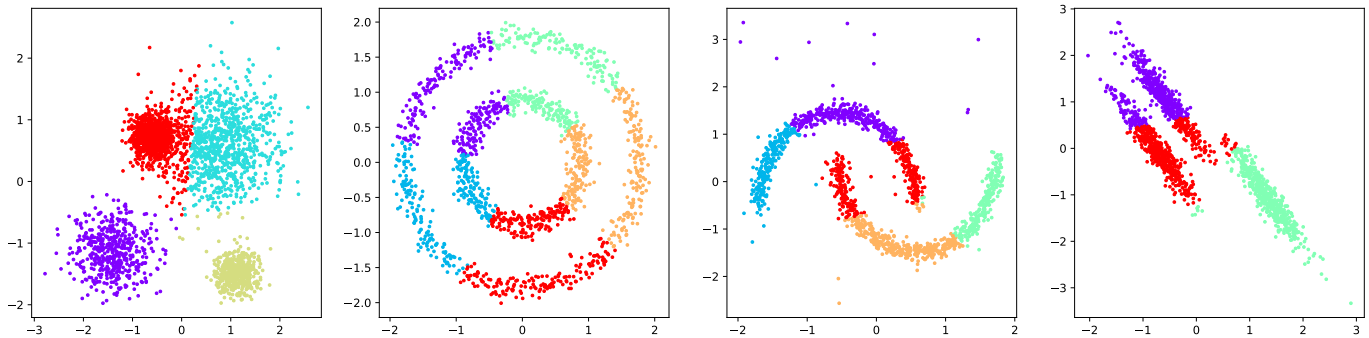
- Hyperparameters:
  - diagonal terms D_{ii} controls how many classes the algorithm produces
  - the damping factor λ (momentum)

$$R^{(k+1)} = \lambda R^{(k)} + (1-\lambda) R^{(k+1)}$$

Affinity propagation



Affinity propagation



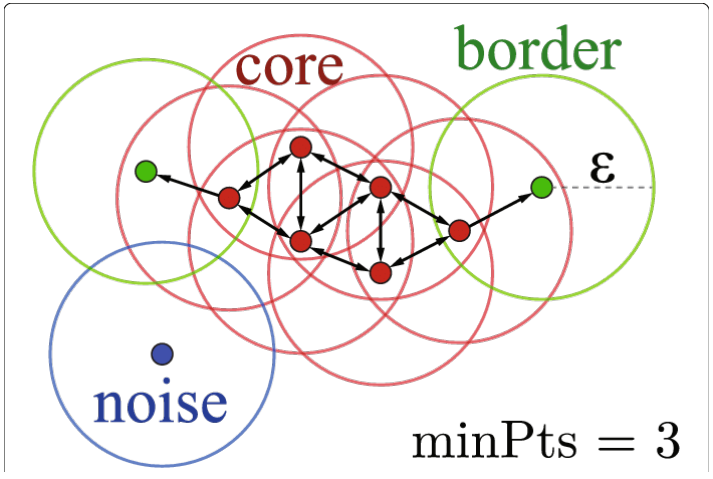
Lecture road map

DBSCAN

Density-based spatial clustering of applications with noise (1996)
Characterization of points (ε, minPts):

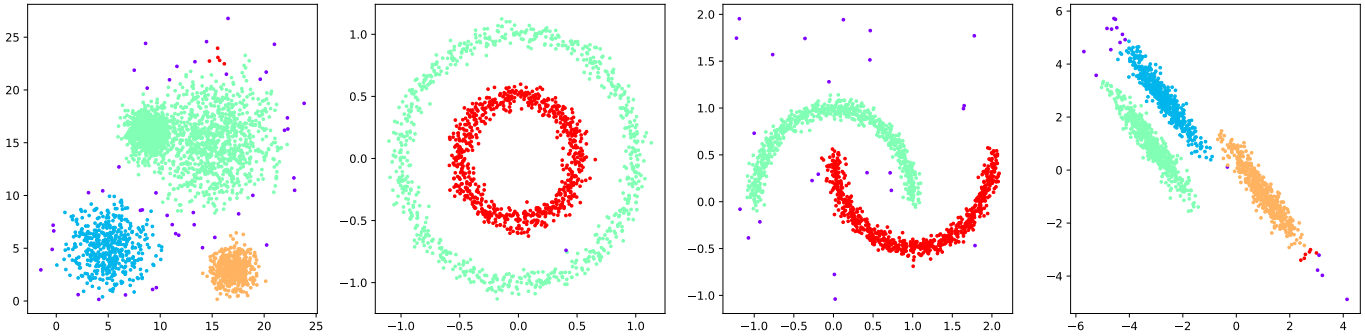
ε the maximum distance (radius) to consider,

MinPts the number of points required to form a cluster.



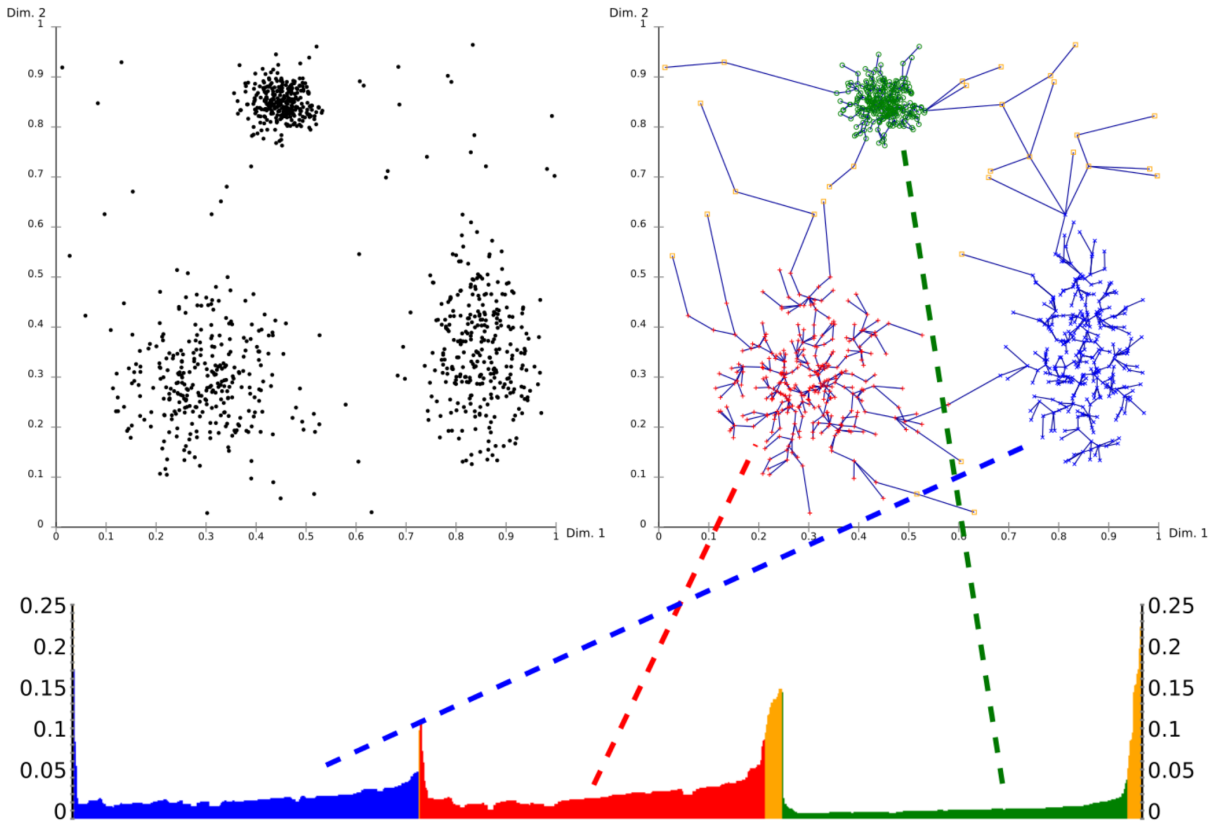
- core point: more than minPts within its ε-neighborhood
- border point: fewer neighbors than minPts, in the neighbohood of a core point
- noise point is any point that is not a core point or a border point.

DBSCAN


25
20
15
10
5
0
0 5 10 15 20 25
1.0
0.5
0.0
−0.5
−1.0
−1.0 −0.5 0.0 0.5 1.0
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0
6
4
2
0
−2
−4
−6 −4 −2 0 2 4

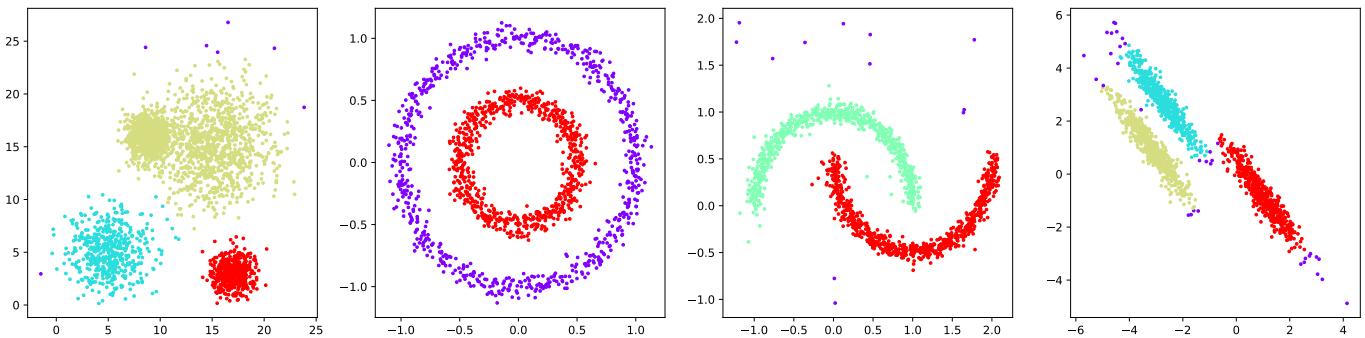
Optics

Ordering points to identify the clustering structure (OPTICS)
Characterization of points (ε, minPts) with variable ε



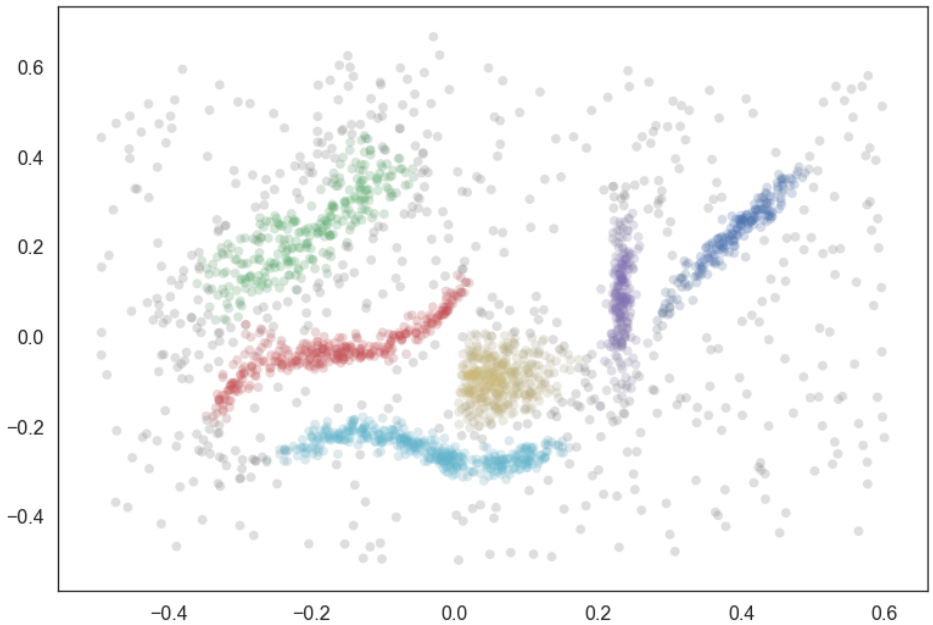
Ankerst et al, 1999

Optics


25
20
15
10
5
0
0 5 10 15 20 25
1.0
0.5
0.0
−0.5
−1.0
−1.0 −0.5 0.0 0.5 1.0
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0
6
4
2
0
−2
−4
−6 −4 −2 0 2 4

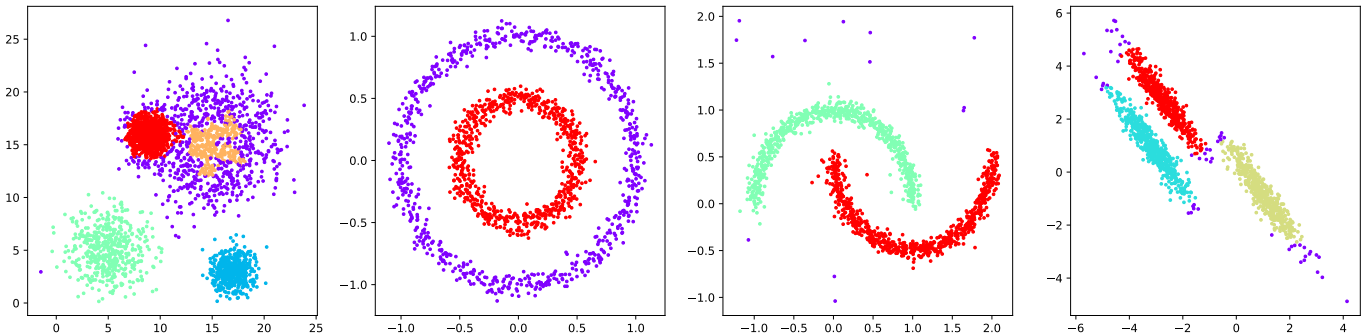
Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN)

Performs DBSCAN over varying ε values and integrates the result



Density-Based Clustering Based on Hierarchical Density Estimates Campello et al. 2013

HDBBSCAN


25
20
15
10
5
0
0 5 10 15 20 25
1.0
0.5
0.0
−0.5
−1.0
−1.0 −0.5 0.0 0.5 1.0
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0
6
4
2
0
−2
−4
−6 −4 −2 0 2 4

Comparizon

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
OPTICS	minimum cluster membership	Very large `n_samples`, large `n_clusters`	Non-flat geometry, uneven cluster sizes, variable cluster density	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers
Birch	branching factor, threshold, optional global clusterer.	Large `n_clusters` and `n_samples`	Large dataset, outlier removal, data reduction.	Euclidean distance between points

https://scikit-learn.org/stable/modules/clustering.html

Conclusions

1. And the winner is DBSCAN and spectral clustering and . . .
2. still an open issue (what is the application?)
3. other approaches. . .
   - PAM, CLARANS: Solutions for the k-medoids problem
   - BIRCH: Constructs a hierarchical tree that acts a summary of the data, and then clusters the leaves.
   - ROCK: clustering categorical data by neighbor and link analysis
   - LIMBO, COOLCAT: Clustering categorical data using information theoretic tools.
   - CURE: Hierarchical algorithm uses different representation of the cluster
   - CHAMELEON: Hierarchical algorithm uses closeness and interconnectivity for merging
4. No consensus or clear guidelines exist to guide these decisions. Cluster analysis always produces clustering, but whether a pattern observed in the sample data characterizes a pattern present in the population remains an open question. -Allison et al. (2006)