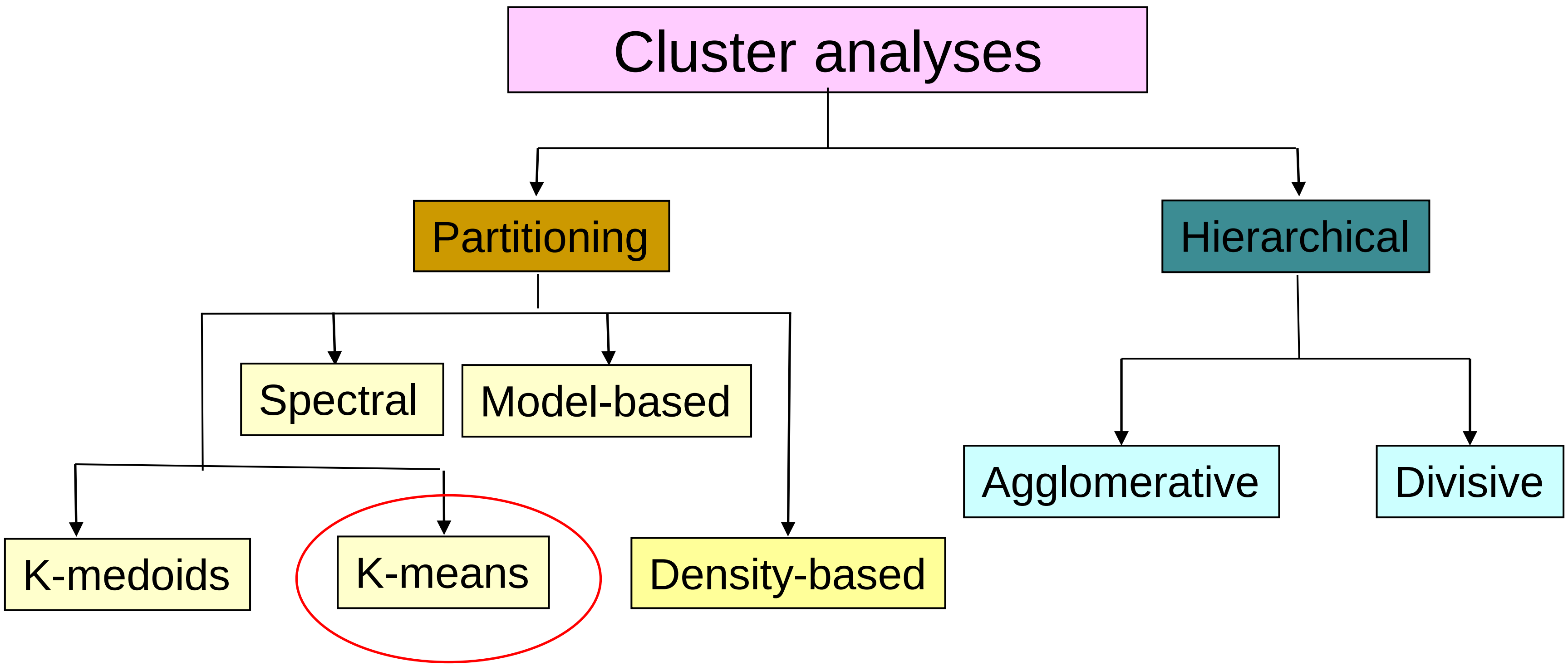
Cluster analysis

(a brief introduction focusing on k-means)

Radoslav Harman
KAMS FMFI UK

Structure of cluster analyses



Applications: Image segmentation, Recommender systems, Anomaly detection, Identification of groups in social networks, Market research, Medical imagining, Categorization of astronomic objects,…

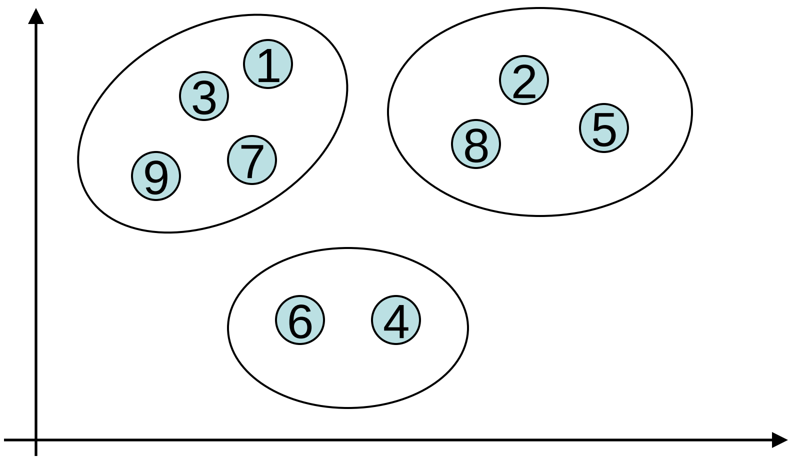
Partitioning cluster analysis

Finds a decomposition of objects 1,...,n into k disjoint clusters $C_1,...,C_k$ of „similar“ objects:

$$C_1 \cup \ldots \cup C_k = \{1, \ldots, n\},\ i \neq j \Rightarrow C_i \cap C_j = \varnothing$$

The objects are (mostly) characterized by „vectors of features“ $x_1, \ldots, x_n \in \Re^p$

p=2
k=3
n=9



$C_1 = \{1,3,7,9\}$ $|C_1| = 4$

$C_2 = \{2,5,8\}$ $|C_2| = 3$

$C_3 = \{4,6\}$ $|C_3| = 2$

How do we understand „decomposition into clusters of similar objects“?

How is this decomposition calculated?

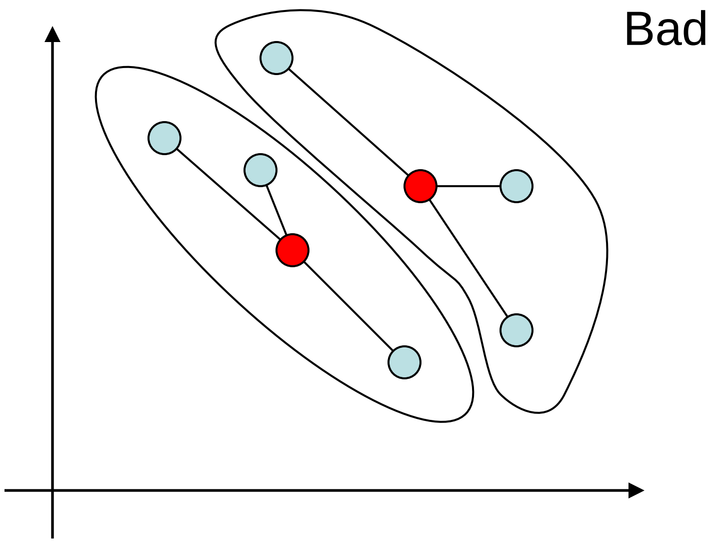
Many different principles and algorithms: **k-means**, **k-medoids**, DBScan...

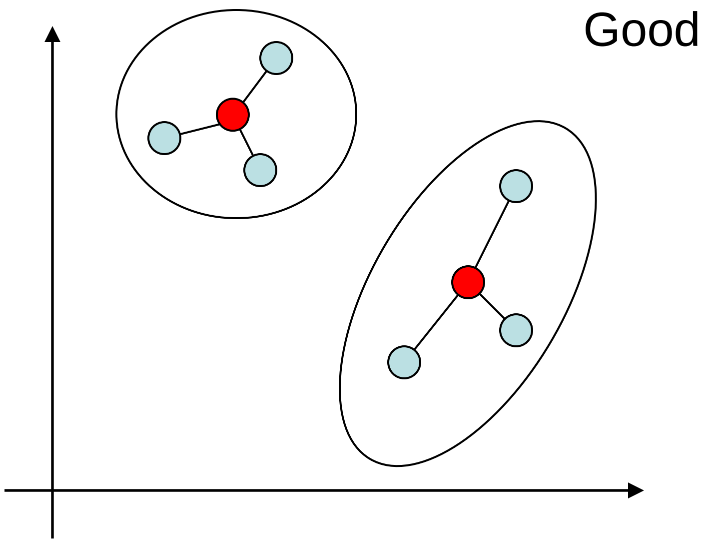
K-means clustering

The objective function to be minimized with respect to the selection of clusters is the „within-cluster sum of squares“:

$$\sum_{i=1}^{k} \sum_{r \in C_i} \rho^2 (x_r, c_i) \quad \text{where} \quad c_i = \frac{1}{|C_i|} \sum_{r \in C_i} x_r \quad \text{is the centroid of } C_i.$$

ρ is the Euclidean distance





K-means clustering

Computing the clustering that minimizes the k-means objective function is a difficult problem. Nevertheless, there are many efficient heuristics able to find a „good“ (not always optimal) solution, such as:

Lloyd’s Algorithm

- Create a random initial clustering $C_1,\ldots, C_k$.
- Until a maximum number of iterations is reached, or no reassignment of objects occurs do:
  - Calculate the centroids $c_1,\ldots, c_k$ of clusters.
  - For every i=1,...,k :
    - Form the new cluster C_i from all the points that are closer to c_i than to any other centroid.

Illustration of the Lloyds' algorithm

Choose an initial clustering

p=2

k=3

n=11

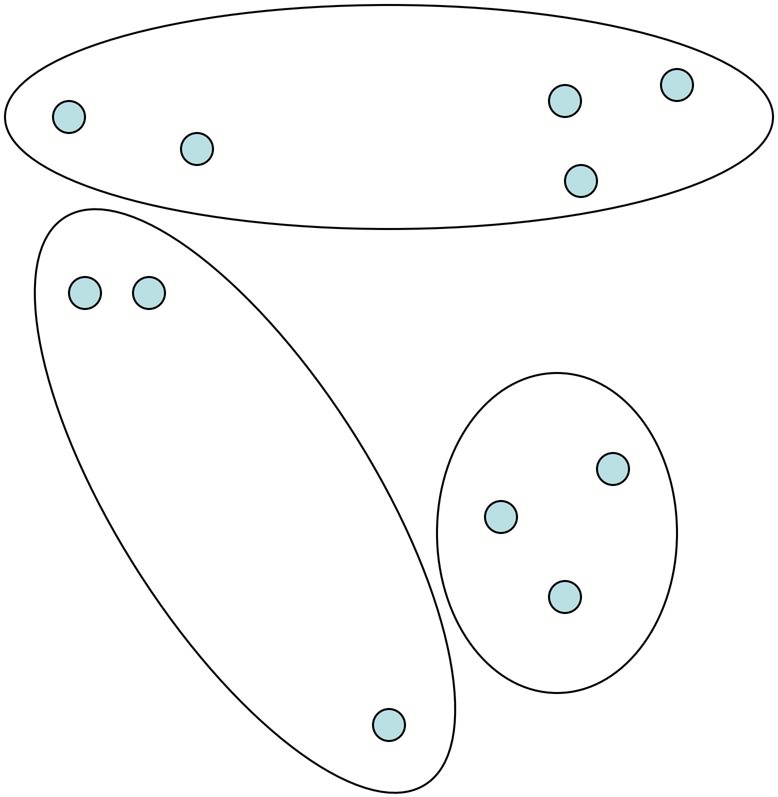


Illustration of the Lloyds' algorithm

Calculate the centroids of clusters

p=2

k=3

n=11

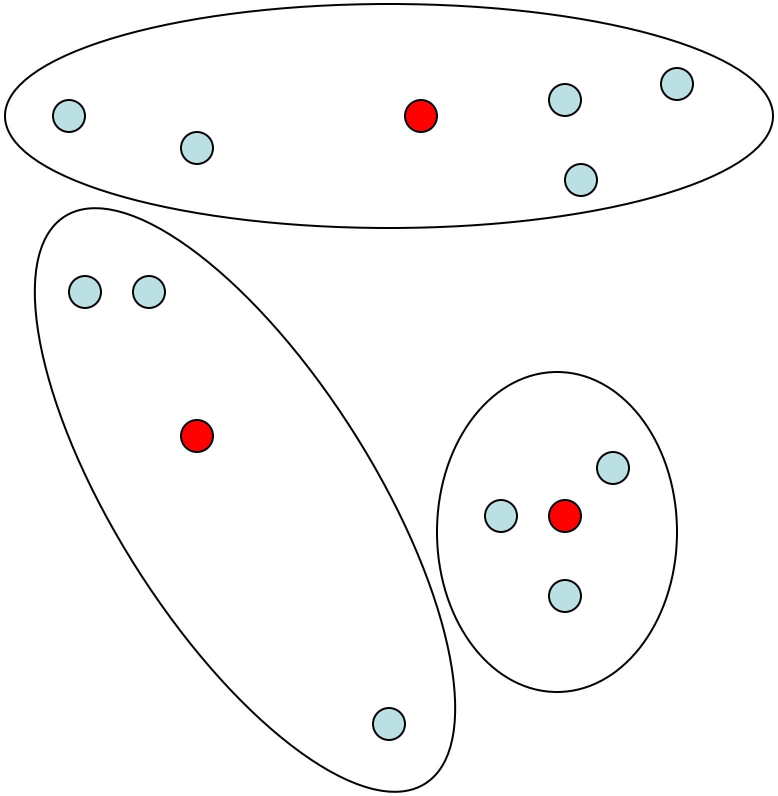


Illustration of the Lloyds' algorithm

Assign the points to the closest centroids

p=2

k=3

n=11

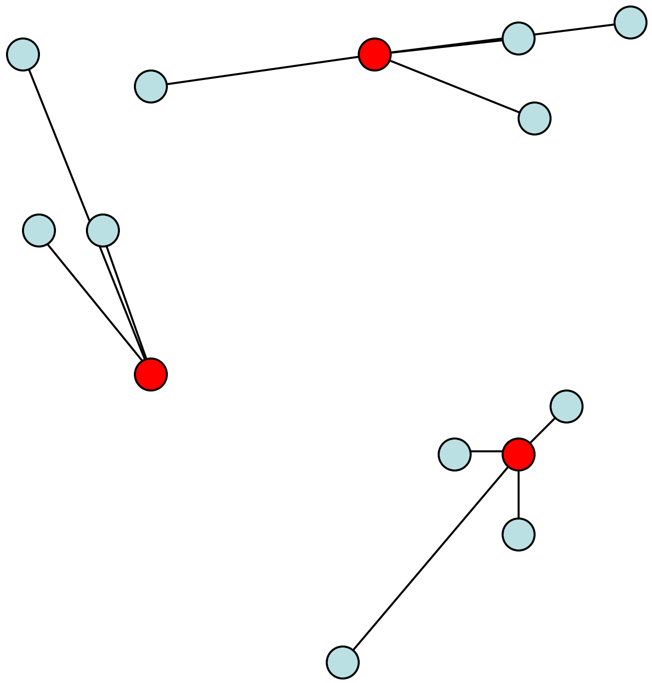


Illustration of the Lloyds' algorithm

Create the new clustering

p=2

k=3

n=11

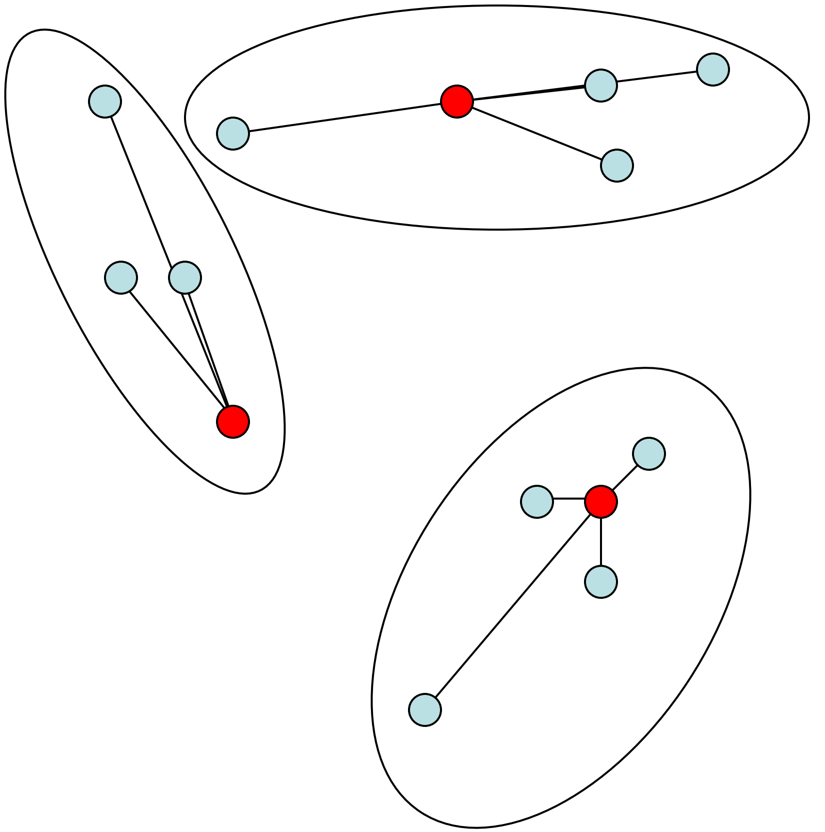


Illustration of the Lloyds' algorithm

Create the new clustering

p=2

k=3

n=11

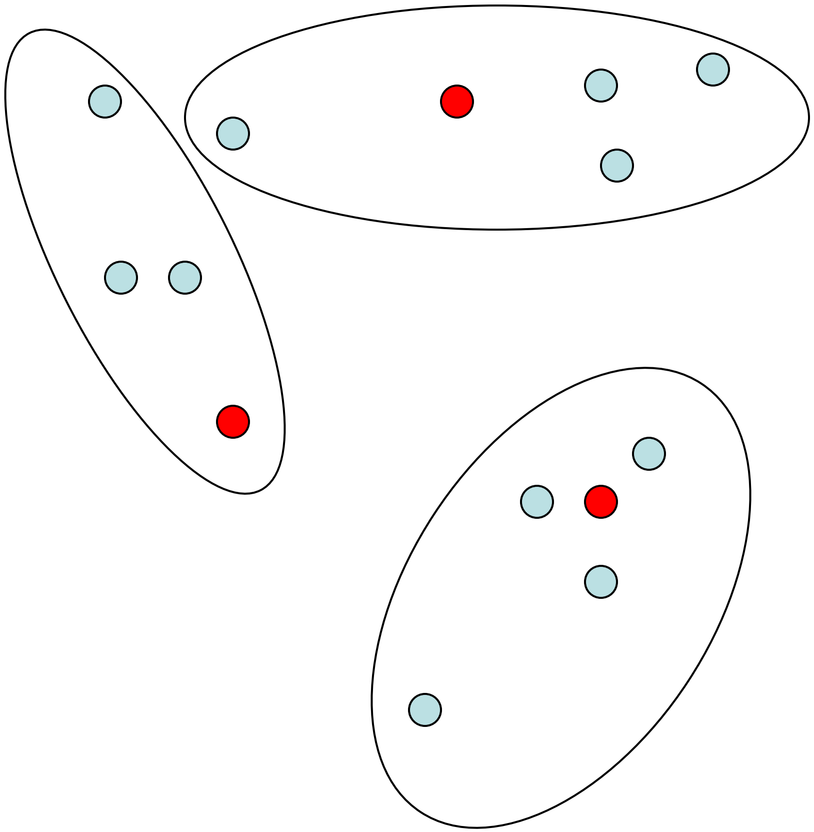


Illustration of the Lloyds' algorithm

Calculate the new centroids of clusters

p=2

k=3

n=11

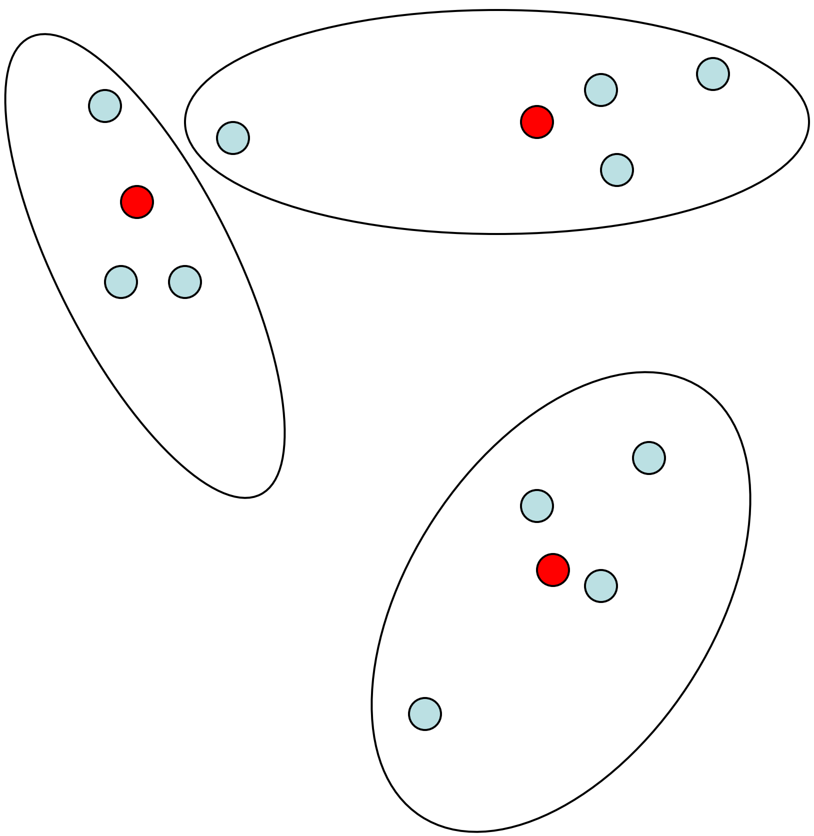


Illustration of the Lloyds' algorithm

Assign the points to the closest centroids

p=2

k=3

n=11

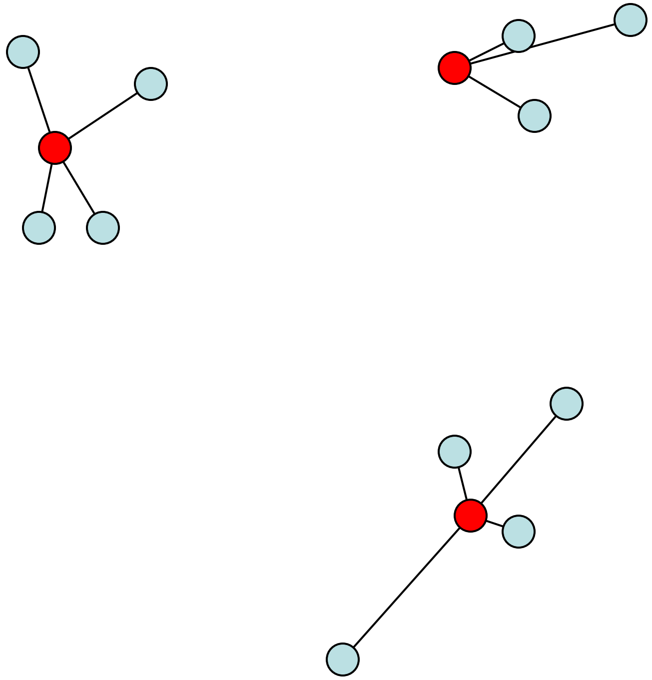


Illustration of the Lloyds' algorithm

Create the new clustering

p=2
k=3
n=11

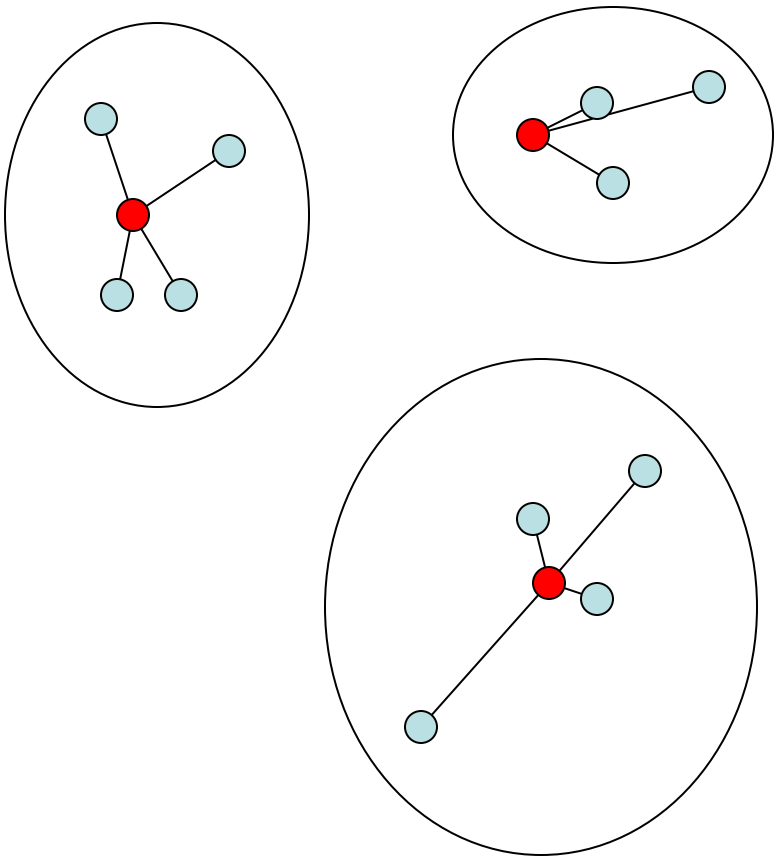


Illustration of the Lloyds' algorithm

Create the new clustering

p=2
k=3
n=11

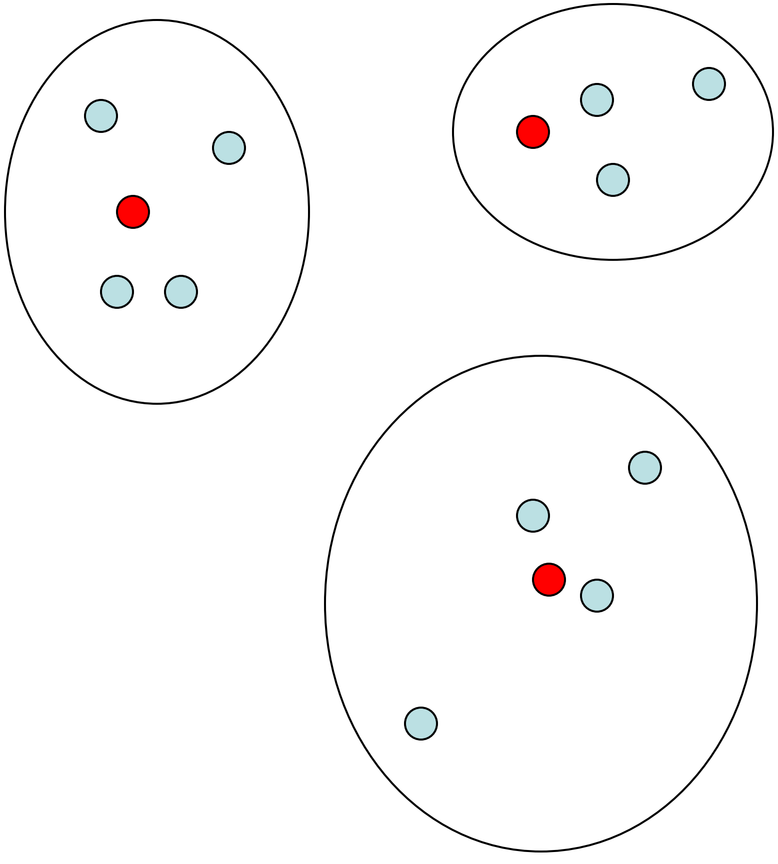


Illustration of the Lloyds' algorithm

Calculate the new centroids of clusters

p=2

k=3

n=11

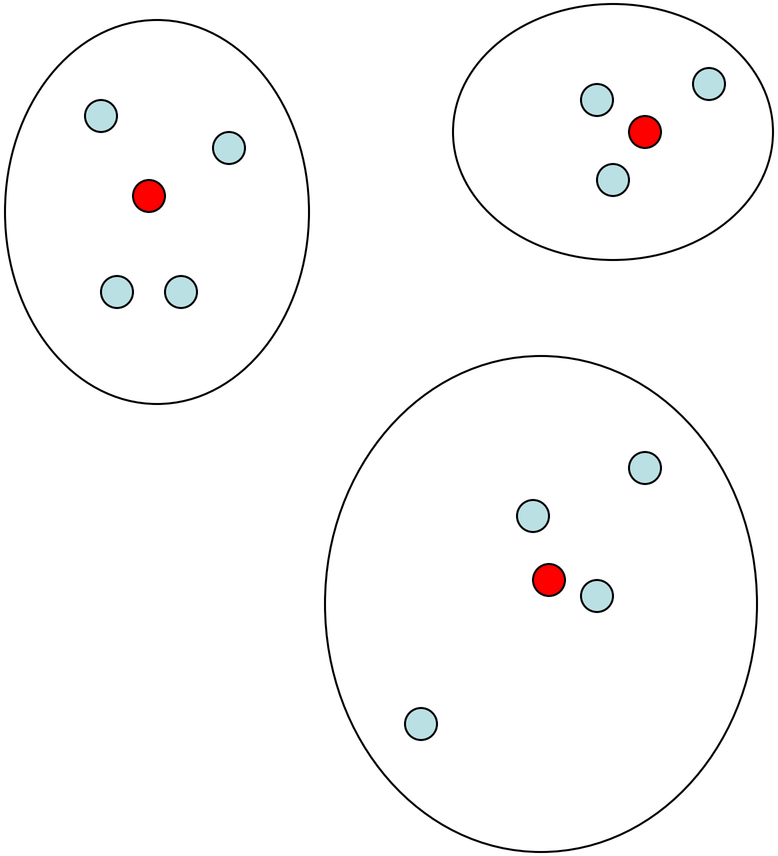


Illustration of the Lloyds' algorithm

Assign the points to the closest centroids

p=2

k=3

n=11

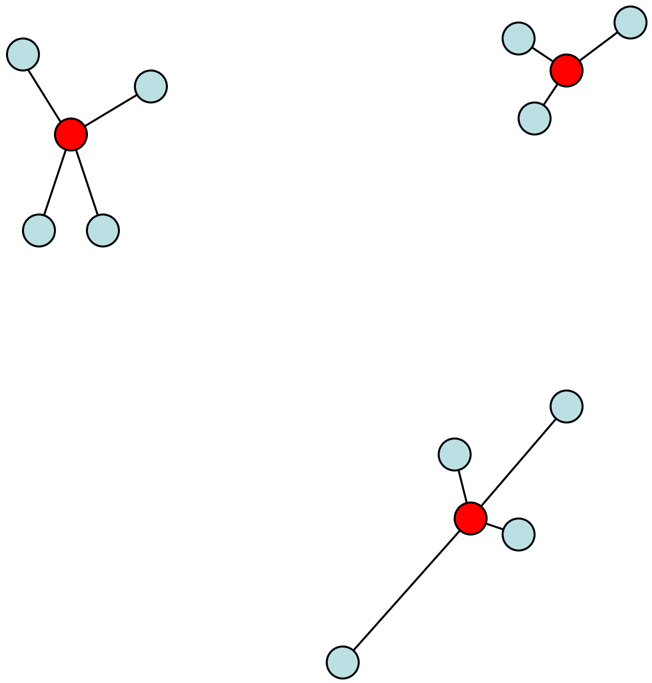


Illustration of the Lloyds' algorithm

Create the new clustering

p=2

k=3

n=11

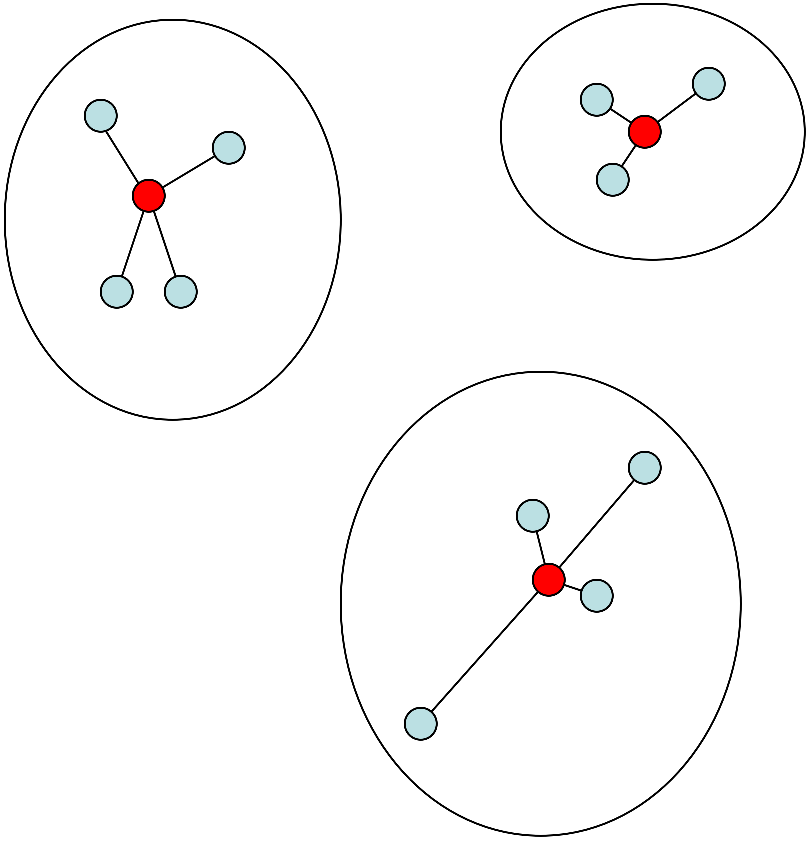
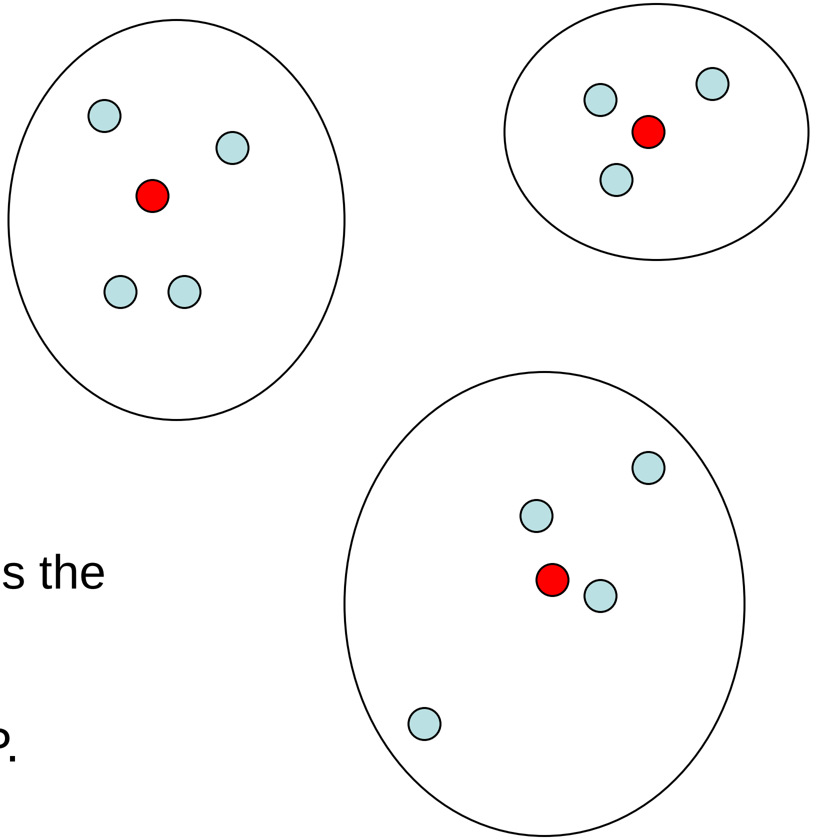


Illustration of the Lloyds' algorithm

Create the new clustering

p=2
k=3
n=11



The clustering is the same as in the previous step, therefore STOP.

Properties of the k-means as a method

+ Simple to understand
+ Many efficient heuristic methods (better than the Lloyds' algorithm)
- The number k of clusters must be given in advance
- The resulting clustering depends on the units of measurement
- Not suitable for finding clusters with nonconvex shapes
- The variables must be real vectors („dissimilarities" are not enough)

Properties of the Lloyds' algorithm

+ Simple to implement
+ Reasonably fast (always convergent in a finite number of steps)
+ Usually converges to a "good" solution
- Different initial clusterings can lead to different final clusterings. We often run the procedure several times with different (random) initial clusterings

Computation of k-means in R

In R (library `stats`):

```
kmeans(x, centers, iter.max, nstart, algorithm)
```

Dataframe of real vectors of features

The number of clusters or a set of centers

Maximum number of iterations

Number of restarts

The method used ("Hartigan-Wong", "Lloyd", "Forgy", "MacQueen")

Many packages contain clustering functions, e.g. `cluster`, `clusterR`

The “elbow” method to determine k

$$\alpha(k) = \sum_{i=1}^{k} \sum_{r \in C_i^{(k)}} \rho^2\left(x_r - c_i^{(k)}\right)$$

$C_1^{(k)}, \ldots, C_k^{(k)}$ … optimal clustering obtained by assuming k clusters

$c_1^{(k)}, \ldots, c_k^{(k)}$ … corresponding centroids