02450: Introduction to Machine Learning and Data Mining

K-means and hierarchical clustering

DTU Compute
Department of Applied Mathematics and Computer Science

Reading Material

Reading material:
C16
Feedback Groups of the day:

- Bryden Fogelman, Alex Genuario, Sydnee Mizuno
- Miguel Martínez Montaña, Stefano Savian
- Kristoffer Olesen, Lorenzo Belgrano, Benjamin Jüttner
- Agla Hardardottir, Finnur Kolbeinsson, Vidar Fridriksson
- Jens Urup, Kristian Breddam
- Carlos Corchado Miralles, Hakon Adalsteinsson
- Patrick Evers Bjørkman, Amalia Matei, Noah Reinert Sturis
- Jonas Nylev, Andreas Motzfeldt Jensen

Tue Herlau, Mikkel N. Schmidt and Morten Mørup

Introduction to Machine Learning and Data Mining

Course notes fall 2016, version 1

August 29, 2016

Technical University of Denmark

Lecture Schedule

1. Introduction
   30 August: C1

Data: Feature extraction, and visualization

2. Data and feature extraction
   6 September: C2, C3
3. Measures of similarity and summary statistics
   13 September: C4
4. Data Visualization and probability
   20 September: C5, C6

Supervised learning: Classification and regression

5. Decision trees and linear regression
   27 September: C7, C8 **(Project 1 due before 13:00)**
6. Overfitting and performance evaluation
   4 October: C9
7. Nearest Neighbor, Bayes and Naive Bayes
   11 October: C10, C11
8. Artificial Neural Networks and Bias/Variance
   25 October: C12, C13
9. AUC and ensemble methods
   1 November: C14, C15

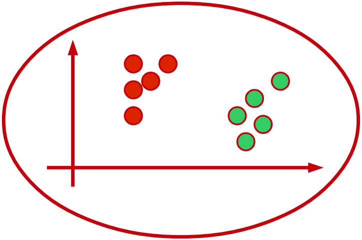
Unsupervised learning: Clustering and density estimation

10. **K-means and hierarchical clustering**
    8 November: C16 (Project 2 due before 13:00)
11. Mixture models and density estimation
    15 November: C17, C18
12. Association mining
    22 November: C19

Recap

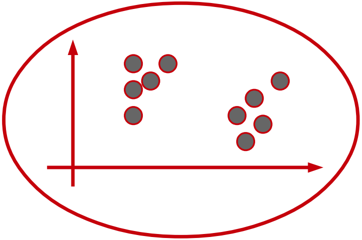
13. Recap and discussion of the exam
    29 November: C1-C19 **(Project 3 due before 13:00)**

Supervised and Unsupervised learning



Supervised Learning
Input data $\mathbf{x}_n$ and output y_n

(Classification and Regression)



Unsupervised Learning
Input data $\mathbf{x}_n$ alone

(Exploratory analysis)

Imagine you observe the world for the first time!

We humans are skilled at dividing objects into groups (clustering), but how do we make computers do the same?

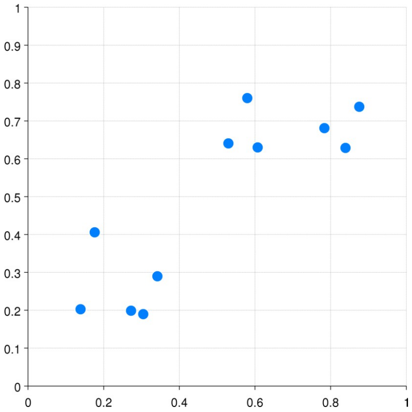
Unsupervised learning

- **Supervised learning**
  - Use the data to learn the output values
- **Unsupervised learning**
  - No output variables available
  - Sometimes called exploratory analysis
  - What to learn from the data?
    - Structure
    - Regularities
    - Hidden information
    - Etc.

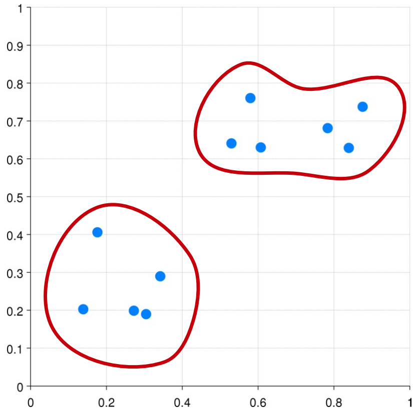
Clustering

- Divide data into groups (subsets/clusters) that are
  - **Meaningful**: Capture the natural structure of the data
  - **Useful**: Depends on purpose
- Observations in the same cluster are **similar in some sense**
- Unsupervised classification

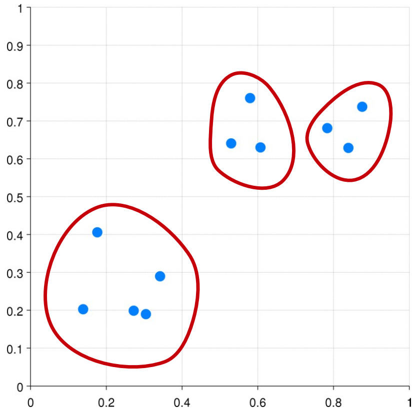
Clustering



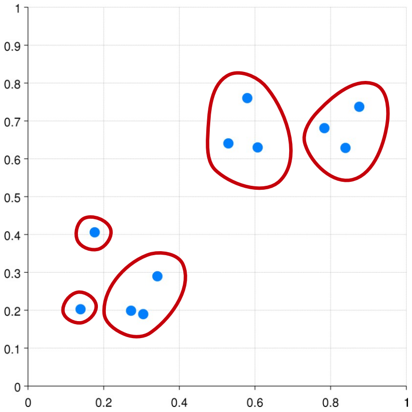
Clustering



Clustering

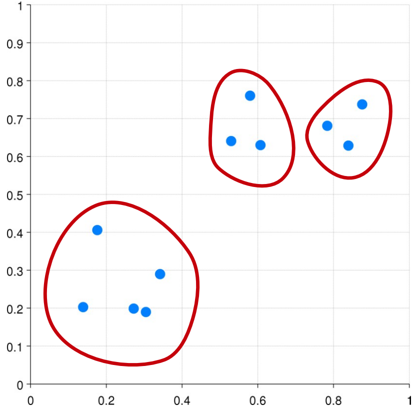


Clustering

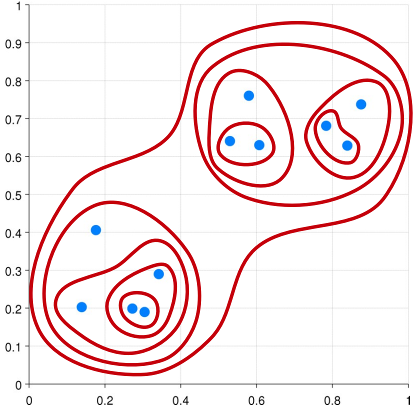


Partitional / hierarchical clustering

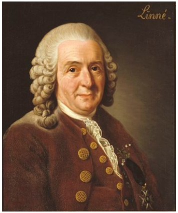
Partitional



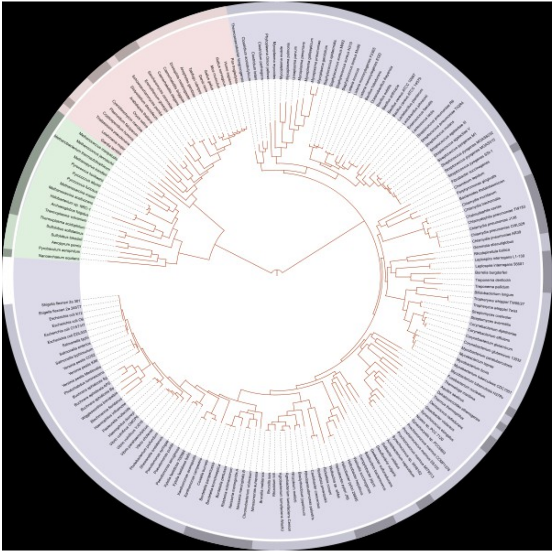
Hierarchical



Phylogenetic trees may be considered a type of hierarchical clustering



Carl Linnaeus
(1707 – 1778)
http://en.wikipedia.org/wiki/Carl_Linnaeus

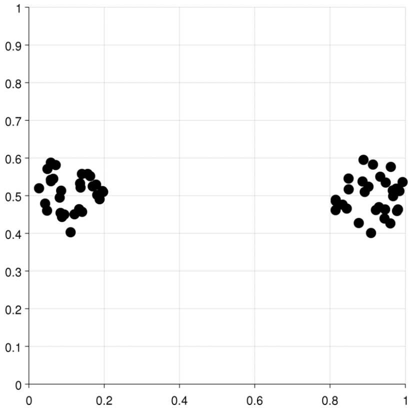


http://en.wikipedia.org/wiki/File:Tree_of_life_SVG.svg

Types of clustering

Well-separated

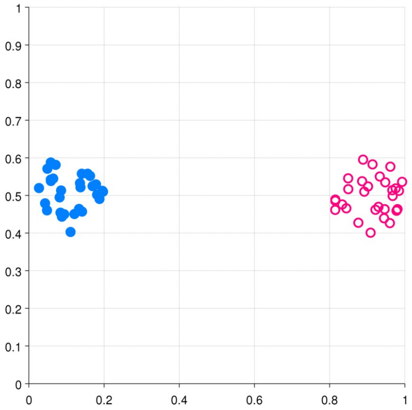
- Each point is closer to all points in its cluster than any point in another cluster



Types of clustering

Well-separated

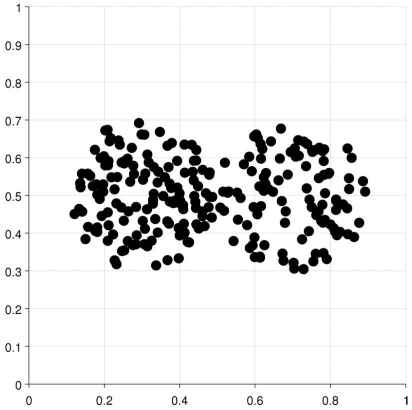
- Each point is closer to all points in its cluster than any point in another cluster



Types of clustering

Center-based

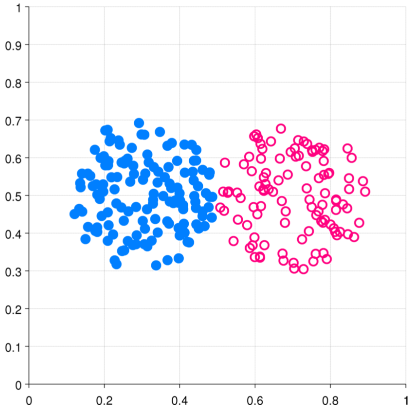
- Each point is closer to the center of its cluster than to the center of any other cluster



Types of clustering

Center-based

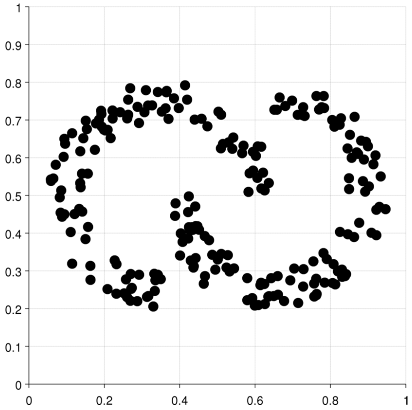
- Each point is closer to the center of its cluster than to the center of any other cluster



Types of clustering

Contiguity-based

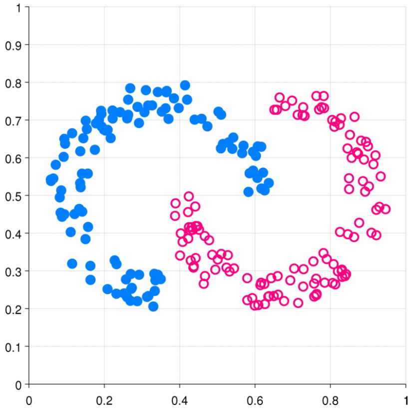
- Each point is closer to at least one point in its cluster than to any point in another cluster



Types of clustering

Contiguity-based

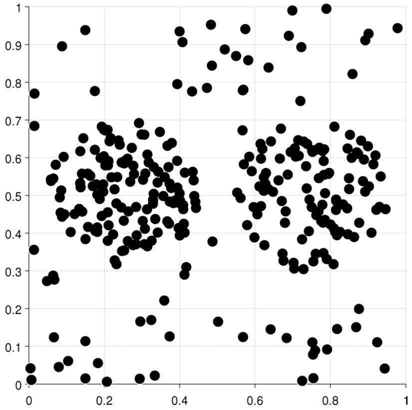
- Each point is closer to at least one point in its cluster than to any point in another cluster



Types of clustering

Density-based

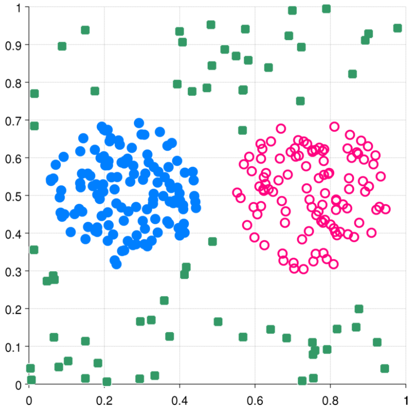
- Clusters are regions of high density separated by regions of low density



Types of clustering

Density-based

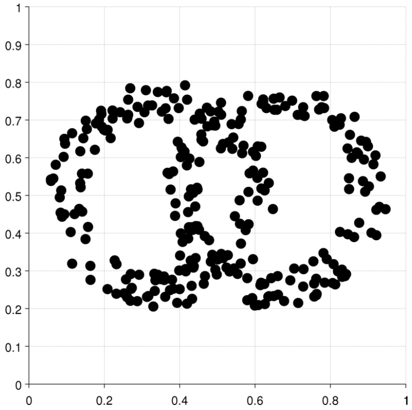
- Clusters are regions of high density separated by regions of low density



Types of clustering

Conceptual clusters

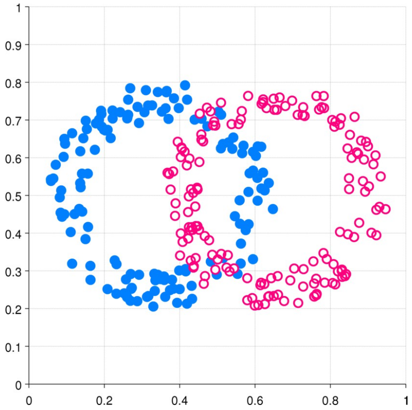
- Points in a cluster share some general property that derives from the entire set of points



Types of clustering

Conceptual clusters

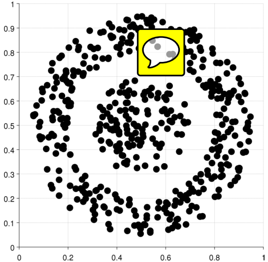
- Points in a cluster share some general property that derives from the entire set of points

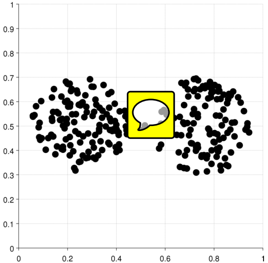


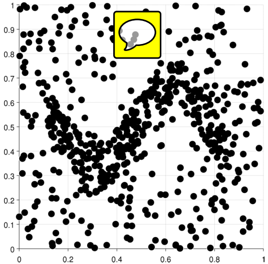
Group exercise

Using the five criteria

- How will these points be clustered?
- How many clusters?







Well-separated

- Each point is closer to all points in its cluster than any point in another cluster

Center-based

- Each point is closer to the center of its cluster than to the center of any other cluster

Contiguity-based

- Each point is closer to at least one point in its cluster than to any point in another cluster

Density-based

- Clusters are regions of high density separated by regions of low density

Conceptual clusters

- Points in a cluster share some general property that derives from the entire set of points

This taxonomy of cluster types taken from:
"Introduction to Data Mining" by Pang-Ning Tan, Michael Steinbach, Vipin Kumar

K-means clustering

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change

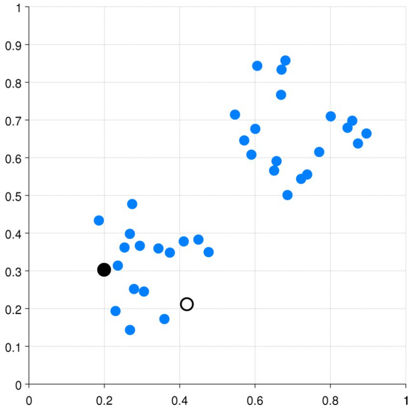
K-means clustering

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change



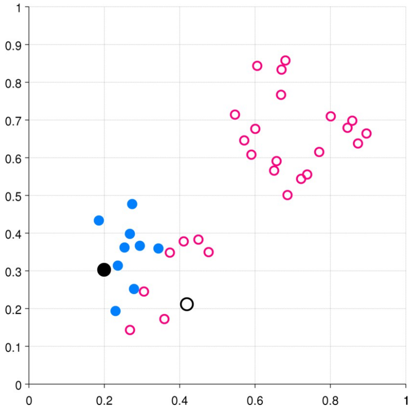
K-means clustering

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change



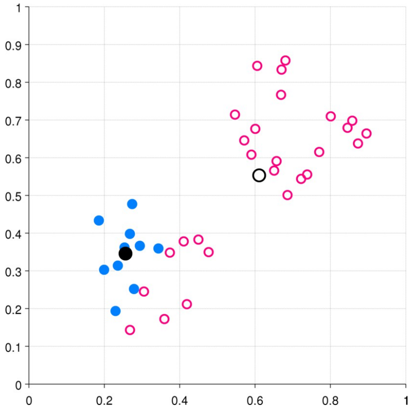
K-means clustering

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change



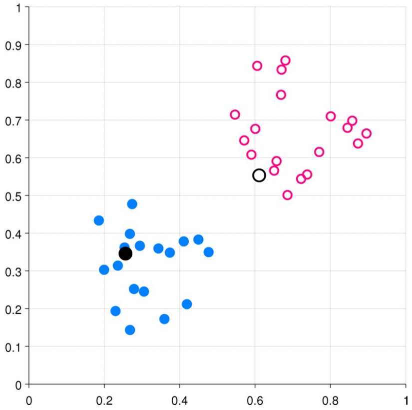
K-means clustering

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change



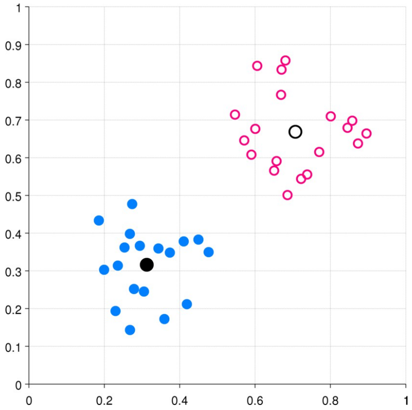
K-means clustering

Select K points as initial centroids

Repeat

- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change



K-means clustering

How do I

- Find the closest centroid?
  - Use a suitable **dissimilarity/similarity measure**
- Compute the cluster centroids
  - Depends on dissimilarity/similarity measure
  - For example, for Euclidean distance the mean is optimal

Group exercise

Using pen-and-paper k-means, cluster the following data objects

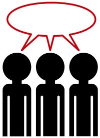
- Number of clusters
  - K=2
- Distance measure
  - Euclidean
- Computation of centroid
  - Mean of cluster members
- Initial centroids
  - For example the first two data objects
- In case of any ties, flip a coin to decide

- **Data objects**

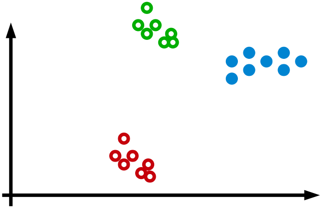
$$x = \{42, 60, 17, 48, 12\}$$

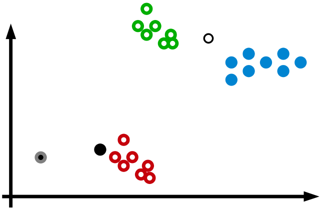
Select K points as initial centroids
Repeat
- Form K clusters by assigning each point to its closest centroid
- Recompute the centroids of each cluster

Until centroids do not change

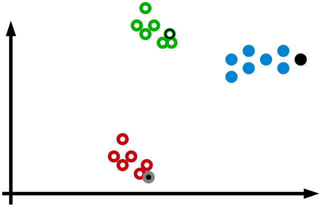


How will the data (top-left diagram) be clustered given the initialization of the three centroids shown at the right and at the bottom?





- What could we do if we have an empty cluster?
- What could be a good initialization procedure? (Farthest First)



Agglomerative hierarchical clustering

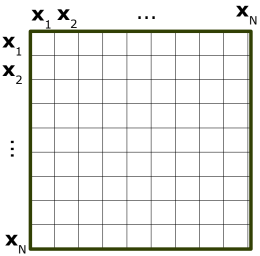
Initialize the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains

$$D_{ij} = \text{distance}(\mathbf{x}_i, \mathbf{x}_j)$$



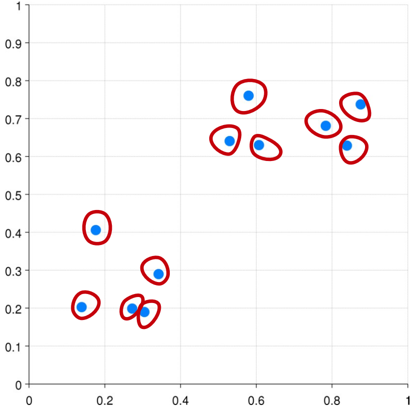
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



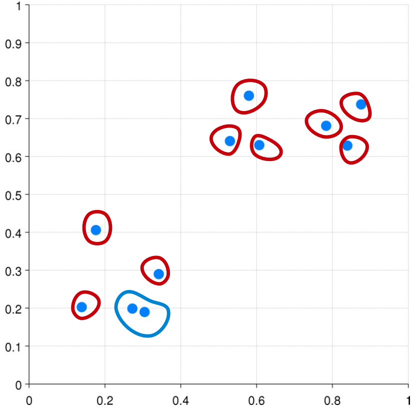
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



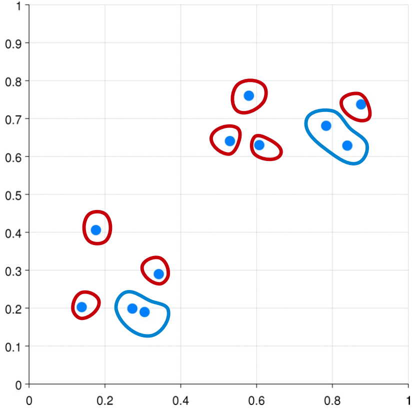
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



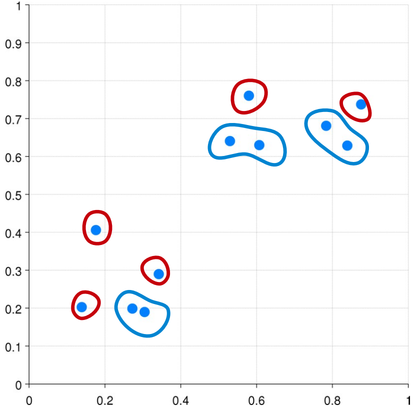
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



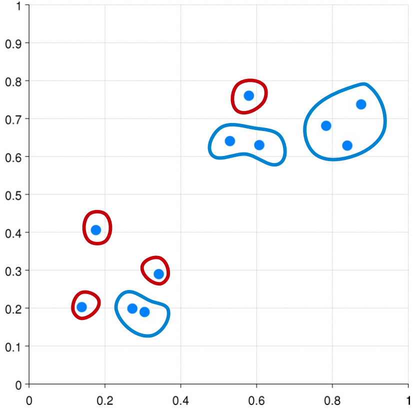
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



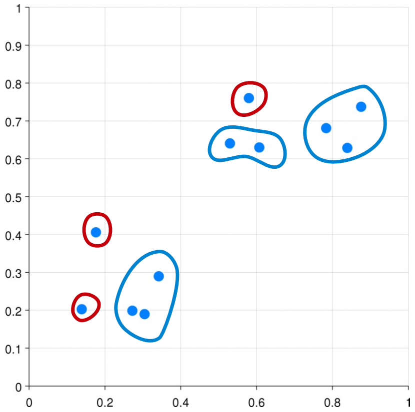
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



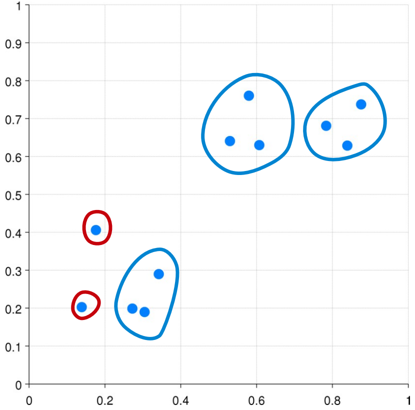
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



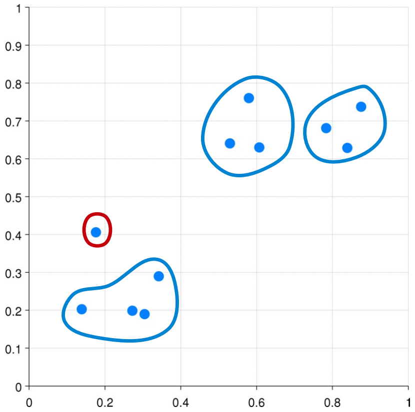
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



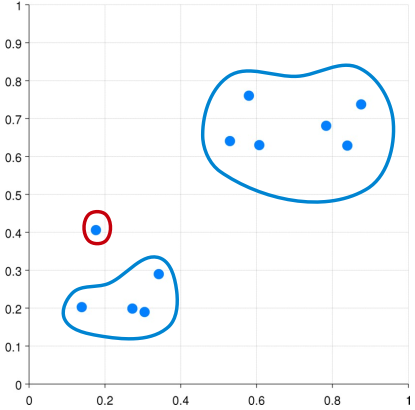
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



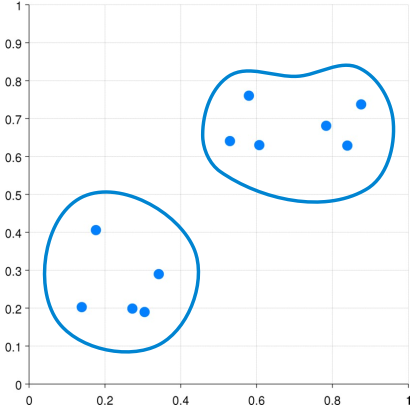
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



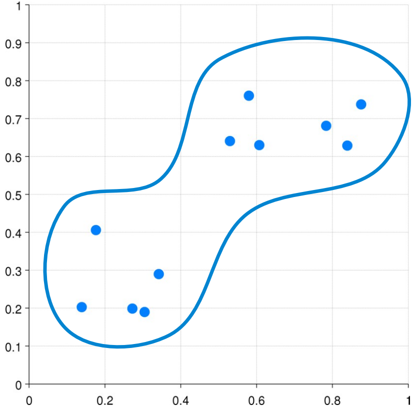
Agglomerative hierarchical clustering

Compute the proximity matrix

Repeat

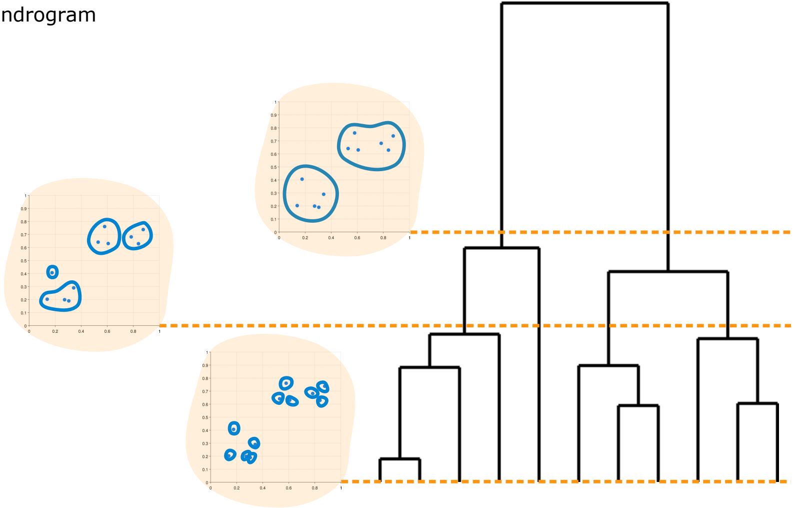
- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



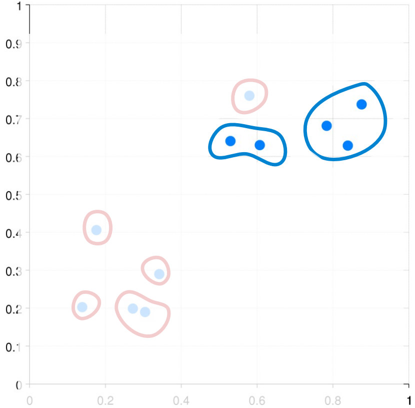
Agglomerative hierarchical clustering

- Dendrogram



Similarity between clusters

- The **key operation** in agglomerative hierarchical clustering is measuring **distance (dissimilarity) between clusters**

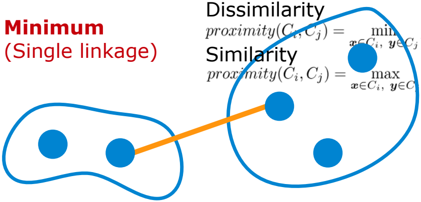


Proximity between clusters

- Can be computed using **proximity between objects**
- **Notice we need different definition if we are given a similarity or dissimilarity measure**
- In our example before we used Euclidian distance as proximity measure; i.e. it is the first definition which is relevant (dissimilarity)

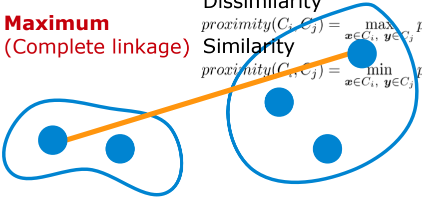
Minimum
(Single linkage)

Dissimilarity
$$proximity(C_i, C_j) = \min_{\boldsymbol{x}\in C_i,\ \boldsymbol{y}\in C_j} proximity(\boldsymbol{x}, \boldsymbol{y})$$
Similarity
$$proximity(C_i, C_j) = \max_{\boldsymbol{x}\in C_i,\ \boldsymbol{y}\in C_j} proximity(\boldsymbol{x}, \boldsymbol{y})$$

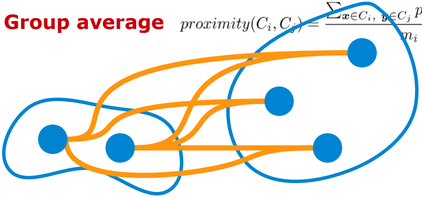


Maximum
(Complete linkage)

Dissimilarity
$$proximity(C_i, C_j) = \max_{\boldsymbol{x}\in C_i,\ \boldsymbol{y}\in C_j} proximity(\boldsymbol{x}, \boldsymbol{y})$$
Similarity
$$proximity(C_i, C_j) = \min_{\boldsymbol{x}\in C_i,\ \boldsymbol{y}\in C_j} proximity(\boldsymbol{x}, \boldsymbol{y})$$



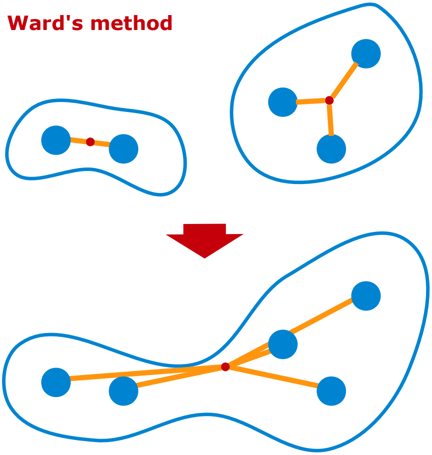
Group average
$$proximity(C_i, C_j) = \frac{\sum_{\boldsymbol{x}\in C_i,\ \boldsymbol{y}\in C_j} proximity(\boldsymbol{x}, \boldsymbol{y})}{m_i \cdot m_j}$$



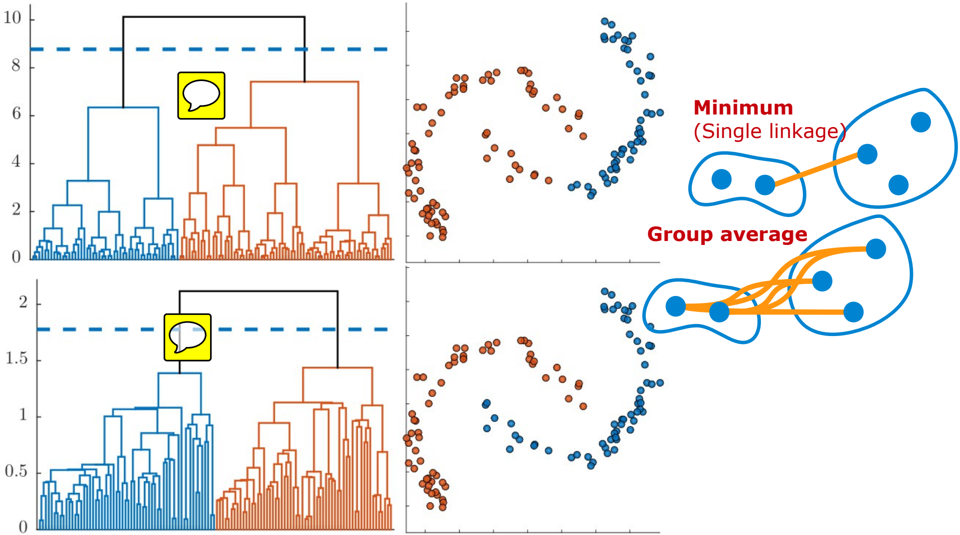
C_i: Observations in cluster i
C_j: Observations in cluster j
m_i: Number of observations in cluster i
m_j: Number of observations in cluster j

Similarity between clusters

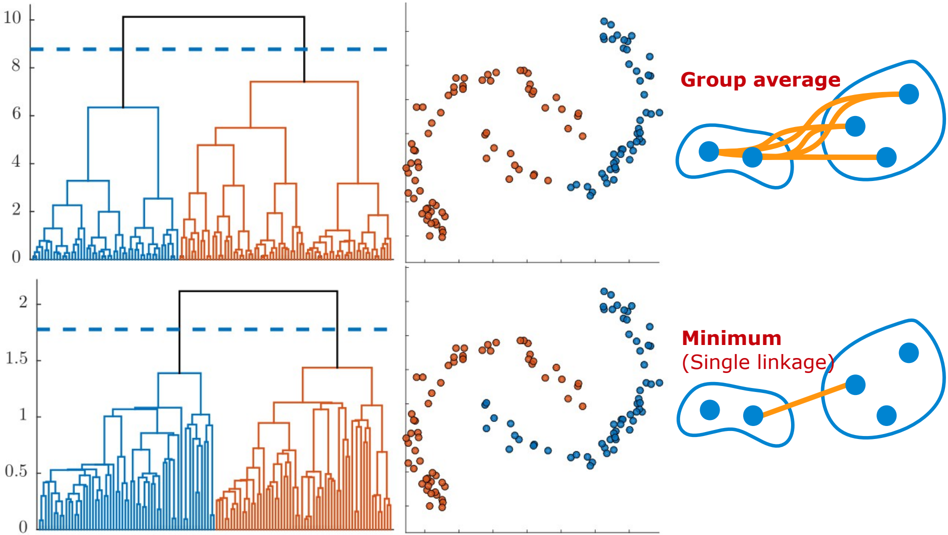
- Increase in sum of squared error after merging the two clusters should be as small as possible



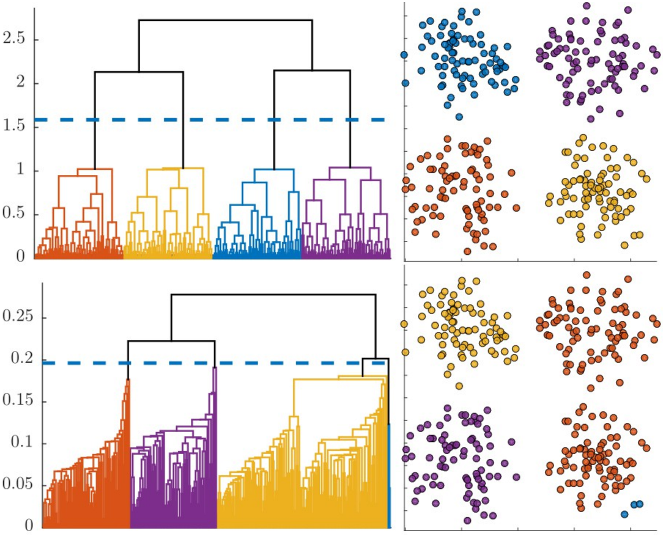
Clusterings and linkage function

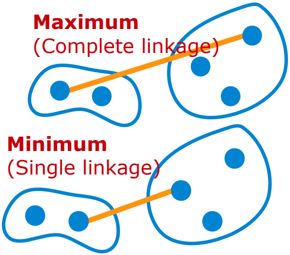


Clusterings and linkage function

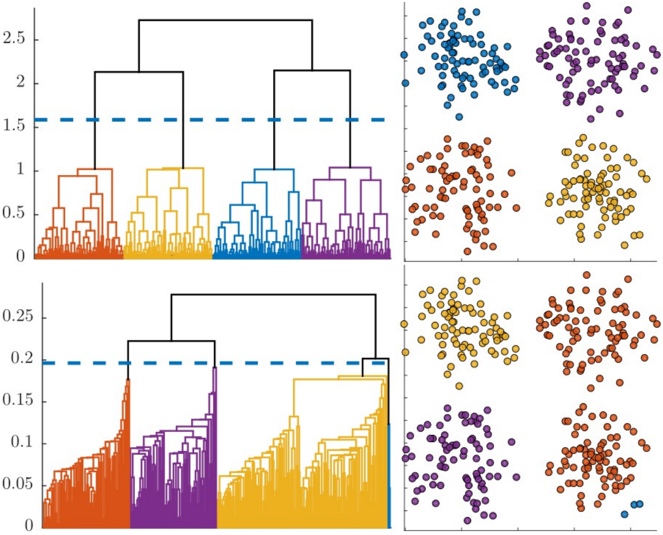


Clusterings and linkage function

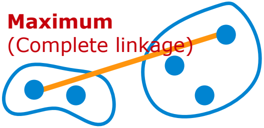




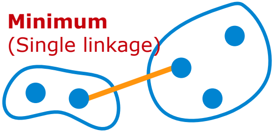
Clusterings and linkage function



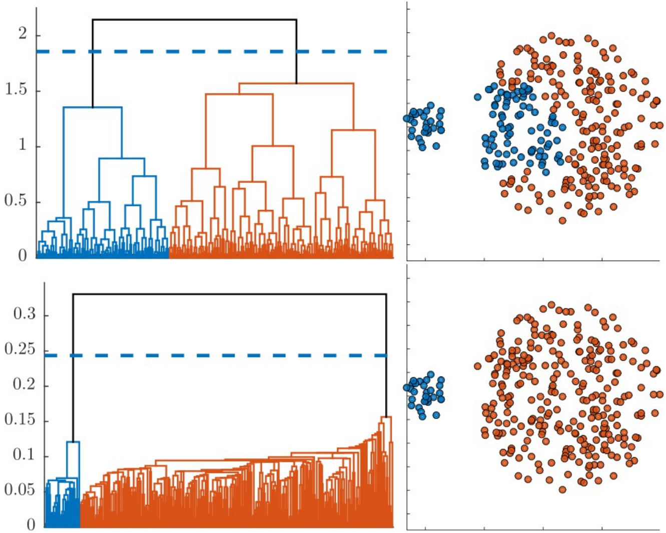
Maximum
(Complete linkage)

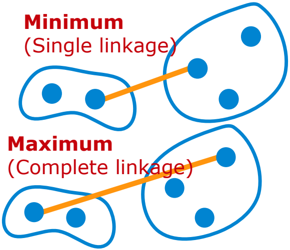


Minimum
(Single linkage)

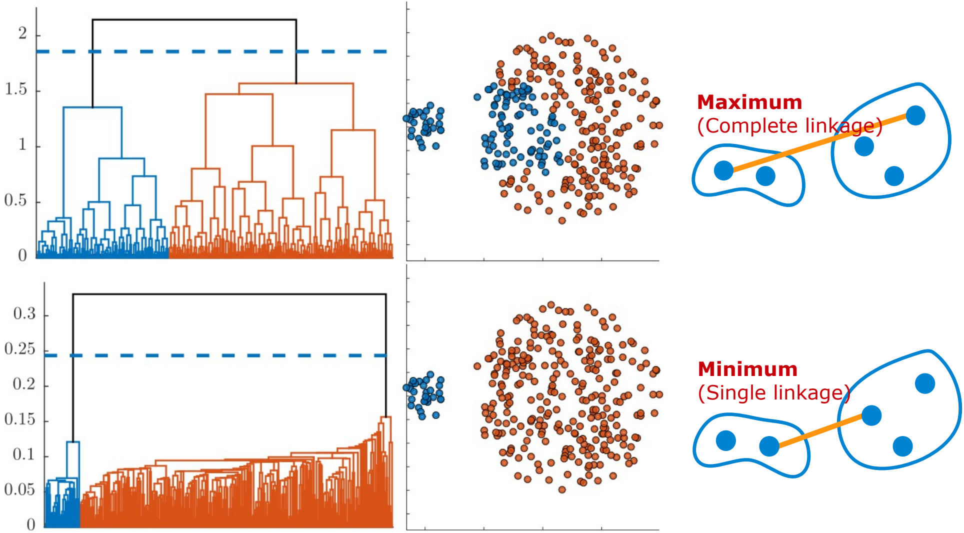


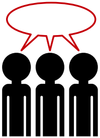
Clusterings and linkage function





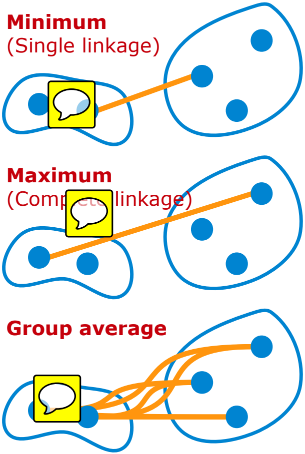
Clusterings and linkage function





Group exercise

Can the choice of linkage be related to the notion of what constitutes clusters?



Well-separated

- Each point is closer to all points in its cluster than any point in another cluster

Center-based

- Each point is closer to the center of its cluster than to the center of any other cluster

Contiguity-based

- Each point is closer to at least one point in its cluster than to any point in another cluster

Density-based

- Clusters are regions of high density separated by regions of low density

Conceptual clusters

- Points in a cluster share some general property that derives from the entire set of points

Group exercise

Using pen-and-paper agglomerative hierarchical clustering, **cluster** the following data objects and draw the **dendrogram**

- Distance measure
  - Euclidean
- Similarity between clusters
  - Minimum (Single linkage)

- **Data objects**

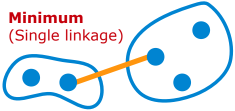
$$x = \{42, 60, 17, 48, 12\}$$

Compute the proximity matrix

Repeat

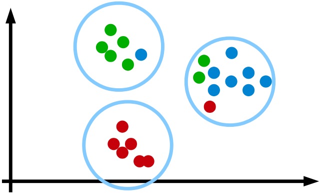
- Merge the two closest clusters
- Update the proximity matrix to reflect the proximity between the new cluster and the original clusters

Until only one cluster remains



How can we compare partitions?

Motivation: Evaluate the extent to which manual classification process can be automatically produced by cluster analysis by comparing clustering to “ground truth”



Supervised measures of cluster validity

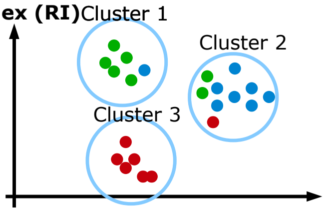
Binary similarity measures

- **Simple matching coefficient (SMC)/Rand index (RI)**

$$\mathrm{SMC}(x,y) = \frac{f_{00} + f_{11}}{K}$$

- **Jaccard coefficient**

$$\mathrm{J}(x,y) = \frac{f_{11}}{K - f_{00}}$$



K : Total number of **pairs of objects**, N·(N-1)/2

f_{00} : Number of object pairs in **different class** assigned to **different clusters**

f_{11} : Number of objects pairs in **same class** assigned to **same cluster**

What is SMC(x,y) and J(x,y) for the example given?

Supervised measures of cluster validity

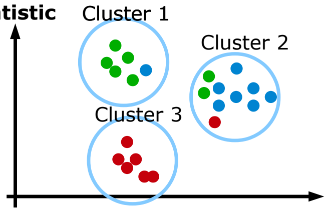
Binary similarity measures

- **Simple matching coefficient (SMC)/Rand statistic**

$$\mathrm{SMC}(x,y) = \frac{f_{00} + f_{11}}{K}$$

- **Jaccard coefficient**

$$\mathrm{J}(x,y) = \frac{f_{11}}{K - f_{00}}$$



K : Total number of **pairs of objects**, N·(N-1)/2

f_{00} : Number of object pairs in **different class** assigned to **different clusters**

f_{11} : Number of objects pairs in **same class** assigned to **same cluster**

In our example we find:

K=22·(22-1)/2=231

$f_{11}=(5\cdot(5-1)/2+1\cdot(1-1)/2)_{c1}+(7\cdot(7-1)/2+2\cdot(2-1)/2+1\cdot(1-1)/2)_{c2}+(6\cdot(6-1)/2)_{c3}=10+22+15=47$

$F_{00}=(5\cdot(7+1)+1\cdot(2+1)+0\cdot(2+7))_{c1\text{->}c2}+(5\cdot 6+1\cdot 6+0\cdot 0)_{c1\text{->}c3}+(2\cdot 6+7\cdot 6+1\cdot 0)_{c2\text{->}c3}=43+36+54=133$

SMC=(47+133)/231=180/231

Jaccard=47/(231-133)=47/98

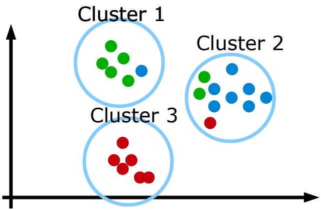
Normalized Mutual Information

Mutual information:

$$\mathrm{MI}[k,m] = \sum_{k=1}^{K}\sum_{m=1}^{M} P(k,m)\log\frac{P(k,m)}{P(k)P(m)}$$

$$H[m] = -\sum_{m} P(m)\log P(m)$$

$$\mathrm{NMI}[k,m] = \frac{\mathrm{MI}[k,m]}{\sqrt{H[k]}\sqrt{H[m]}}$$



	Cluster 1	Cluster 2	Cluster 3	Total
Blue	1	7	0	8
Green	5	2	0	7
Red	0	1	6	7
Total	6	10	6	**22**

Exam question examples

QUESTION I:

We have a one dimensional data set of size N = 5 with data examples
$x_1 = 1$, $x_2 = 3$, $x_3 = 6$, $x_4 = 7$ and $x_5 = 12$.
We run hierarchical clustering with a Euclidean dissimilarity between data points using group average linkage. We will use the following notation to summarize the dendrogram: (x y) means that x and y are joined in the binary tree. x and y can themselves be binary trees. What is the order we build the tree?

A. 12(34)5 → (12)(34)5 → ((12)(34))5 → (((12)(34))5).

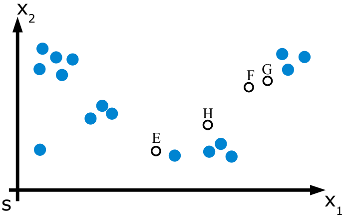
B. 12(34)5 → 12((34)5) → (12)((34)5) → (12((34)5)).

C. 12(34)5 → (12)(34)5 → (12)((34)5) → ((12)((34)5)).

D. 12(34)5 → 12((34)5) → 1(2((34)5)) → (1(2((34)5))).

QUESTION II:

Consider the clustering problem given to the right where blue dots are observations and black circles are the initial position of four centroids denoted E,F,G and H used to cluster the data by k-means using Euclidean distances as dissimilarity. Upon convergence of the k-means algorithm which one of the following statements is wrong?



A. Cluster formed by centroid F will be empty

B. Cluster formed by centroid E will contain 10 observations

C. Clusters formed by centroid H will contain 4 observations

D. Cluster formed by centroid G will contain 3 observations

Exam question examples

QUESTION I:

We have a one dimensional data set of size N = 5 with data examples $x_1 = 1$, $x_2 = 3$, $x_3 = 6$, $x_4 = 7$ and $x_5 = 12$.

We run hierarchical clustering with a Euclidean dissimilarity between data points using group average linkage. We will use the following notation to summarize the dendrogram: (x y) means that x and y are joined in the binary tree. x and y can themselves be binary trees. What is the order we build the tree?

A. 12(34)5 → (12)(34)5 → ((12)(34))5 → (((12)(34))5).

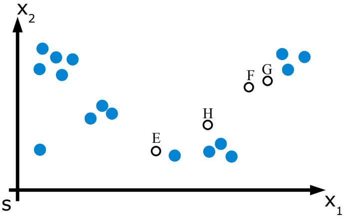
B. 12(34)5 → 12((34)5) → (12)((34)5) → (12((34)5)).

C. 12(34)5 → (12)(34)5 → (12)((34)5) → ((12)((34)5)).

D. 12(34)5 → 12((34)5) → 1(2((34)5)) → (1(2((34)5))).

QUESTION II:

Consider the clustering problem given to the right where blue dots are observations and black circles are the initial position of four centroids denoted E,F,G and H used to cluster the data by k-means using Euclidean distances as dissimilarity. Upon convergence of the k-means algorithm which one of the following statements is wrong?



A. Cluster formed by centroid F will be empty

B. Cluster formed by centroid E will contain 10 observations

C. Clusters formed by centroid H will contain 4 observations

D. Cluster formed by centroid G will contain 3 observations

(Solution: QI: A, QII: B)