***Clustering: Means***

***What is clustering***

This is a form of unsupervised learning where we don’t have a specific task that we want to learn. We have the input features however we don’t have the classes that they belong to, therefore we tell the program to try to learn the patterns and classify them on their own.

A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”.

A *cluster* is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters.



the similarity criterion is *distance*: two or more objects belong to the same cluster if they are “close” according to a given distance (in this case geometrical distance). This is called *distance-based clustering*.  
Another kind of clustering is *conceptual clustering*: two or more objects belong to the same cluster if this one defines a concept *common* to all that objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures.

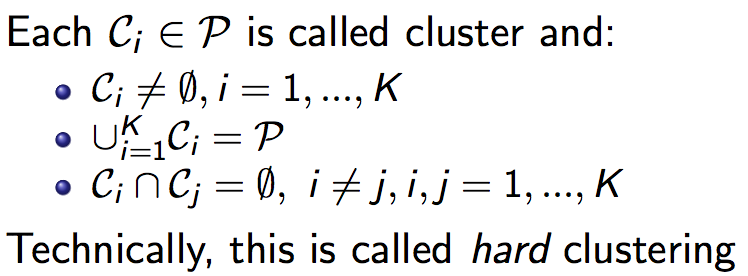
Clustering is a very well-known form of unsupervised learning. There are no desired/target signals to be learnt. However, many researchers believe that clustering is an ill-posed problem as a lot of the time it may not be clear what to do with the data, so it could be an intrinsically arbitrary process, however they do have several applications.

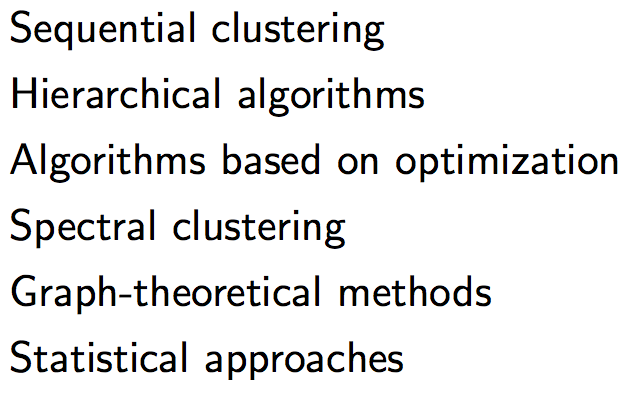
Clustering can be used to perform explorative analysis

***Some Definitions***

Let X = {x1, x2, ..., xN} ⊂ Rd be an input dataset in some dimensional space.

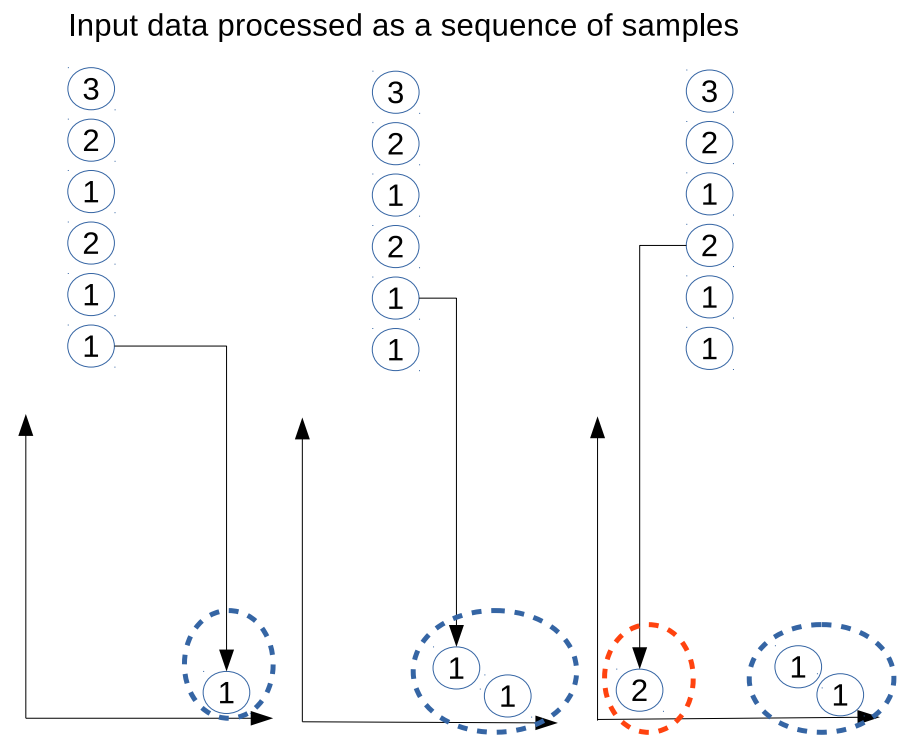
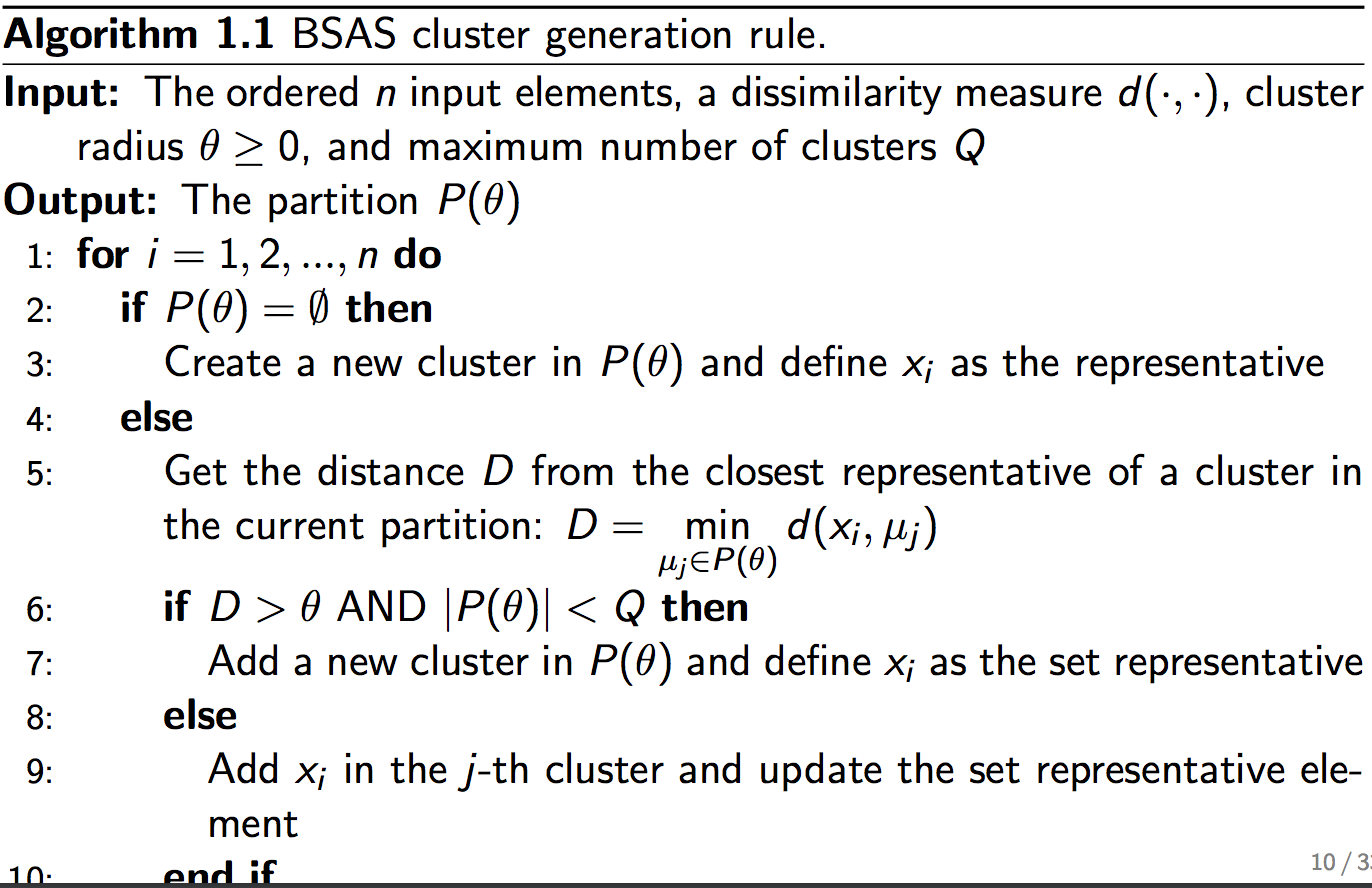
A partition P = {C1, ..., CK } of X is a grouping/clustering in K subsets. K is known as the order of the partition.





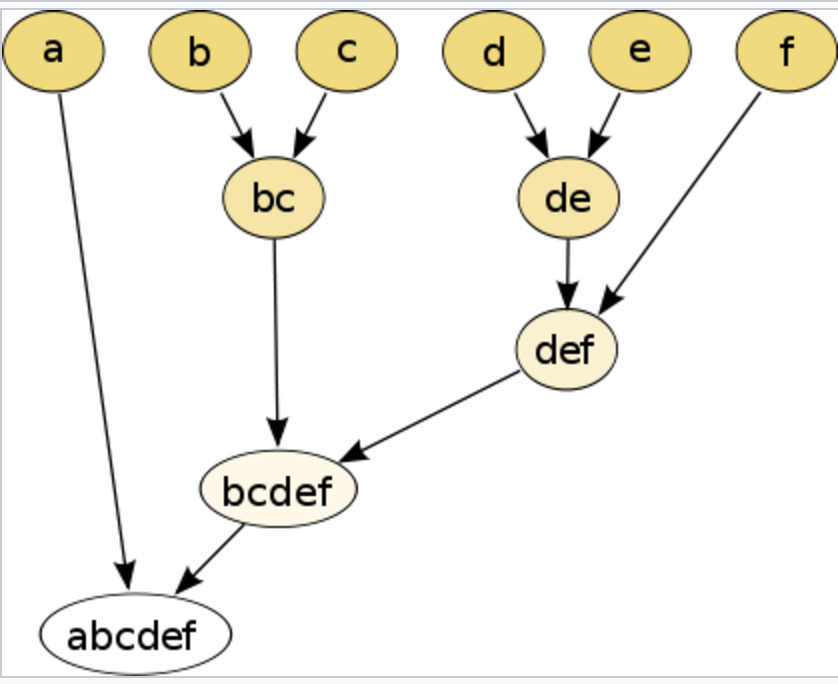
***Types of clustering algos***

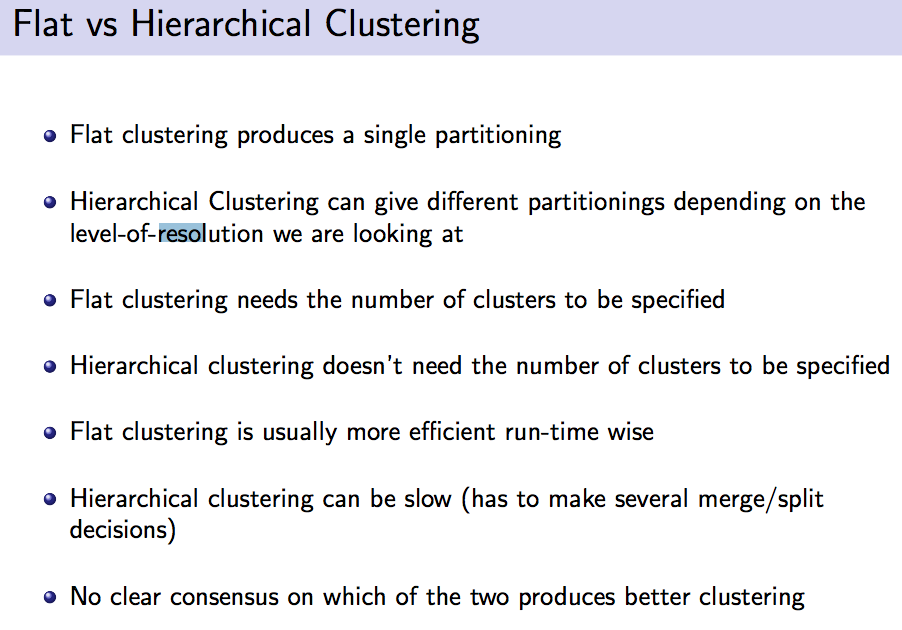
***Sequential Clustering***

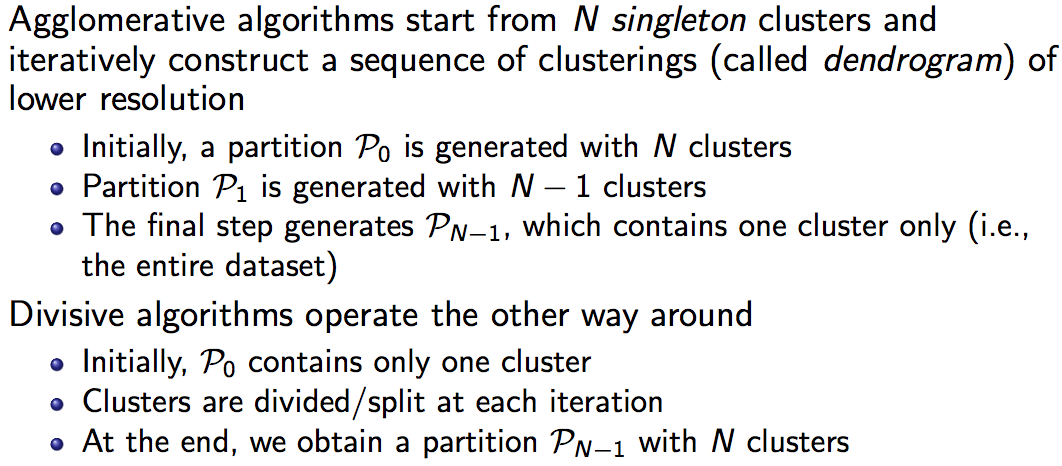
Here, data is like a steam, and it enters the system in sequential order, categorising the data one by one to form clusters

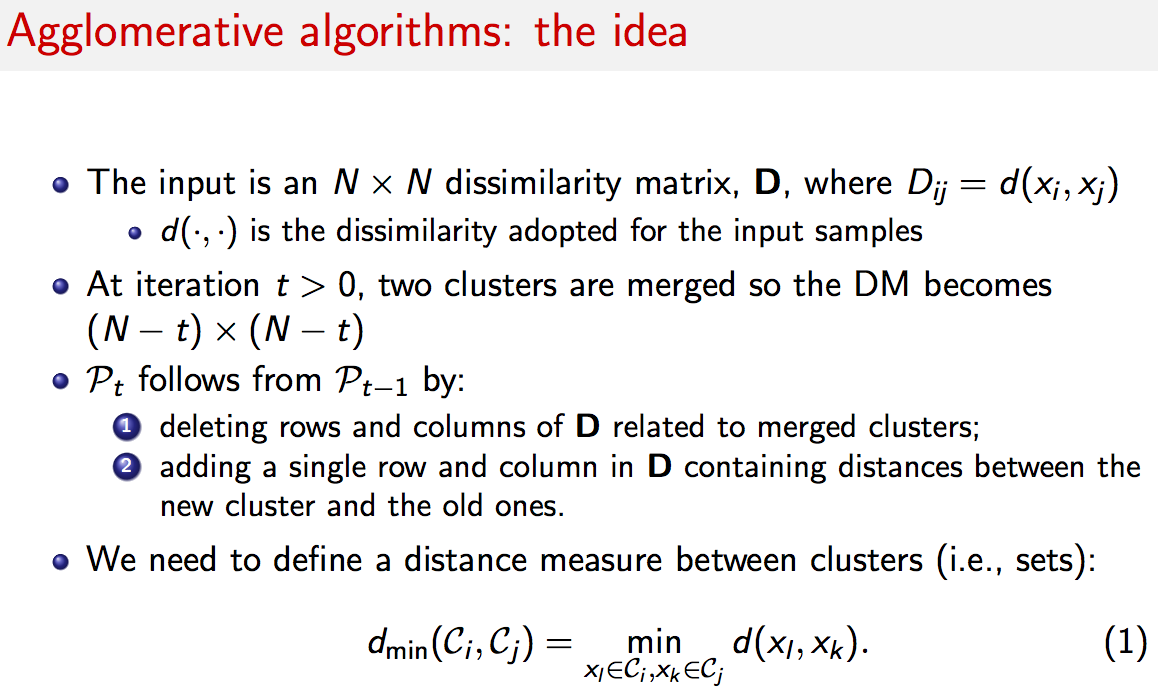
Here, order the n input data points. For each one of the data points, you loop over every single data point, if the partition which is a set of clusters is null, then create a new cluster partition and define the data point as the representative of that cluster. Otherwise calculate the distance D from the data point that we are considering the closest points from each cluster to that data point. Select the minimum distance and if the min distance is less than the cluster radius, then we put it into that cluster partition. If it is larger than the cluster radius, and the number of clusters is less than the max number of clusters, then create a new cluster and insert that data point into that new cluster, and repeat this process for every single data point.

***Hierarchical Clustering***

Generate a hierarchy of partitions with different resolution levels. Here you are creating a hierarchy of clusters based on the resolution you want. These algorithms are good because they provide you with multiple solutions and allow for interaction with the user who can choose the resolution level. They are very heavy in terms of computation however they are flexible and therefore they can be applied to multiple datatypes, a lot of the time based on dissimilarity matrices.







***Algorithms based on Optimisation***

The problem is posed as an optimization problem; therefore, we have an objective function that we maximize/minimize. The objective function is typically an elaboration of the distances between the samples or a statistical measure (e.g., those derived from information theory) quantifying the dependence (e.g., mutual information). The goal is maximizing the intra-cluster similarity, while at the same time minimize the inter-cluster similarity. This is still a form of classification so it is still possible to be optimised.

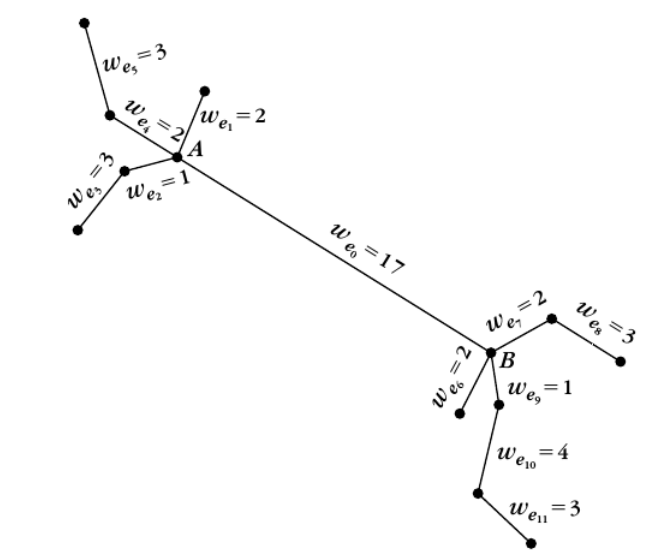
***Graph Theoretical Methods:***

The idea is to map the input data as a graph. If you have n data points, you end up with n vertices and connected edges.

A graph is constructed by defining the vertices as the samples and the edges represent the pairwise dissimilarity values

There are three main classes of algorithms based on topology (based on the connections – how vertices are connected), spectral methods and random walks.

These are very effective and supported by a solid mathematical framework, however they are time consuming and computationally expensive to run, and have a high complexity.



Minimum spanning tree