# Hierarchical methods

**CHAMELEON**

Hierarchical clustering using dynamic modeling. Measures the similarity based on a dynamic model. Can work with arbitrary shapes, not just spherical ones!

CHAMELEON is based on a KNN graph. Each vertex (punkt) of the graph represents a data object and there is an edge between two vertices if the objects if they are k nearest neighbors of each other.

We start with a constructing a sparse graph (KNN graph) for the data set. The graph is partitioned to get a large number of relatively small sub-clusters (sub-graphs). Edge cut , cut the graph to get disconnected components.

The sub-clusters are then merged by using an agglomerative hierarchical clustering algorithm to find the genuine clusters by combining sub-clusters. The method determines the pair of most similar sub-clusters by taking into account both the interconnectivity and closeness of the clusters.

Edge cut when dividing into two approx.. similar pieces.

Two clusters are merged only if the interconnectivity and closeness (proximity) between two clusters are high relative to the internal interconnectivity of the clusters and closeness of items within the clusters.

NOTE! In KNN graph an object is not a neighbor of itself!

Algorithm:

1. Use a graph partitioning algorithm, i.e. cluster objects into a large number of relatively small sub-clusters.
   1. Based on KNN graph
   2. There is an edge between two nodes if points corresponding to either of the nodes are among the k-most similar points of the points corresponding to the other node
   3. Edge weight is density of the region
   4. Dynamic notion of density: Regions with high density has small neighborhood radius and vice versa.
2. Use an agglomerative hierarchical clustering algorithm – find the genuine clusters by repeatedly combining these sub-clusters
   1. Interconnectivity between clusters and – normalized sum of edges weights (weight of edges that connects nodes in and )
   2. Closeness of clusters and – average similarity between points in that are connected to points in
   3. Merge if both measures are above user-defined thresholds