CMPT-740 Fall 2003
Foundations of Data Mining
Martin Ester

Assignment 2

Total marks: 60
Due date: October 1, 2003

Assignment 2.1

The task of clustering has intuitively been described as finding clusters such that objects within a cluster are as similar as possible while objects of different clusters are as dissimilar as possible. As a necessary condition to accomplish this goal, we may specify the following requirement for clusterings:

Let D be a database of objects and $dist(x, y)$ a distance function for pairs of objects. A clustering is a set $C = \{C_1,...,C_k\}$ where $C_i \subseteq D$ and where each object $x \in D$ satisfies the following condition: $x \in C_i \Rightarrow \forall C_j \in C, C_j \neq C_i : dist(x, C_i) \leq dist(x, C_j)$.

(a) Propose two different alternatives for the definition of $dist(x, C_i)$, the distance between an object and a cluster (set of objects).
(b) For each proposed definition $dist(x, C_i)$, discuss which of the following types of clustering algorithms finds clusterings satisfying this definition: *k*-means, Single-Link, Average-Link, DBSCAN. Note that your answer may depend on the given distance function $dist(x, y)$ for pairs of objects.
(c) Draw two datasets of two-dimensional points, one where both alternatives (as discussed in part a) result in the same clustering of $k = 2$ clusters and another one where the resulting clusterings with two clusters differ from each other.

Assignment 2.2

We want to develop a top-down (divisive) hierarchical clustering method. This method starts with the whole dataset as one cluster. In each step of the recursive procedure, it (1) chooses one of the clusters to split and (2) splits the chosen cluster into two clusters.

(a) How would you choose the cluster to be split?
(b) How would you determine the split of the chosen cluster?
To answer this question, first propose a (inefficient) method to determine the optimum split, i.e. a split that maximizes the distance between the two resulting sub-clusters. What is the runtime complexity of this method in terms of the number *n* of elements of the cluster to be split? Then, suggest an efficient method to find a good split approximating the optimum split.