DIMACS
Center for Discrete Mathematics &
Theoretical Computer Science

DIMACS EDUCATIONAL MODULE SERIES

MODULE 03-1
Introduction to Clustering Algorithms:
Hierarchical Clustering

Date prepared: March 17, 2003

Alexander Kheyfits [1]
Bronx Community College of the City University of New York
University Avenue at 181st Street, Bronx, NY 10453
email: alexander.kheyfits@bcc.cuny.edu

DIMACS Center, CoRE Bldg., Rutgers University, 96 Frelinghuysen Road, Piscataway, NJ 08854-8018
TEL: 732-445-5928 • FAX: 732-445-5932 • EMAIL: `center@dimacs.rutgers.edu`
Web: `http://dimacs.rutgers.edu/`

Founded as a National Science Foundation Science and Technology Center and a Joint Project of Rutgers University, Princeton University, AT&T Labs - Research, Bell Labs, NEC Laboratories America and Telcordia Technologies with affiliated members Avaya Labs, HP Labs, IBM Research, Microsoft Research.

[1] Supported by the National Science Foundation (Grant DUE 9752776)

DIMACS EDUCATIONAL MODULE SERIES
Description of Module 03-1

Title:

Introduction to Clustering Algorithms: Hierarchical Clustering

Author:

Alexander Kheyfits

Abstract: Clustering algorithms separate (partition) discrete data sets into disjoint groups-clusters, such that every cluster contains only the elements close to each other in a precisely defined sense. These procedures are widely used in mathematical taxonomy, management, and many other applications of mathematics. In this module we discuss the simplest and commonly used hierarchical algorithms for clustering — Hubert's single-link and complete-link algorithms. These algorithms are called hierarchical because they build up an hierarchy of larger and larger clusters. The algorithms are based on the properties of a graph describing an initial collection of objects, and the terms *single-link*, *complete-link* refer to the methods of combining two subgraphs in one larger subgraph.

The module is aimed at freshman and sophomore students studying finite mathematics, introductory discrete mathematics, or statistics. So, only a minimal, high-school background in mathematics is assumed. In particular, we do not expect any knowledge of graph theory or probability theory. All relevant graph-theoretical definitions (like spanning trees, etc.) are discussed and illustrated by examples. The algorithms considered are simple and can be examined in an introductory computer science course. We do not perform any formal analysis of the algorithms, however we present them in a pseudocode form. The module can be used in the classroom and for the students' projects.

It is supposed that after active studying the module and working the included exercises, the reader

- will learn some basic concepts of graph theory together with their simple applications,
- will learn basic concepts of cluster theory and clustering algorithms,
- will be able to apply these algorithms for clustering small (since we do not discuss any software issues) arrays of data,
- will have enough background to learn and apply computer software for clustering real data,
- will be able to study more advanced literature on classification and clustering.

Informal Description:

In this module we discuss how to split a set of different objects into several groups of more or less similar entities. For example, when a manager of a big supermarket decides where to place various goods, she has to solve a problem of cluster analysis. Cluster analysis is used in any field concerned with classification of data. Examples are numerical taxonomy, design of the Internet, classification of natural languages, or image processing, to name just a few.

The readers of the module will learn how to form clusters step by step, first in an informal way, considering a model example, and then using some simple graph theory. These necessary concepts of the graph theory are also introduced in the module. In the last section a real case study is considered.

Target Audience:

The module is aimed at freshman and sophomore students studying finite mathematics, introductory discrete mathematics, or statistics. It may be used by well-motivated high-school students.

Prerequisites:

Only a minimal, high school background in mathematics is assumed. In particular, we do not expect any knowledge of graph theory or probability theory. All relevant graph-theoretical concepts, such as spanning trees, are discussed and illustrated by examples. The algorithms considered are simple and can be examined in an introductory computer science course. We do no formal analysis of the algorithms, however we present them in a standard pseudocode form.

Mathematical Field:

Graph theory; Classification theory; Cluster analysis.

Applications Areas:

Classification theory; Mathematical and biological taxonomy.

Mathematics Subject Classification:

MSC (2000) Primary 91C20; Secondary 05C90, 90B80, 92B10.

Contact Information:

Alexander Kheyfits
Dept. of Math. and Computer Science
Bronx Community College/CUNY
University Avenue at W. 181st Street
Bronx, NY 10453

Tel.: (718) 289-5616
E-mail: alexander.kheyfits@bcc.cuny.edu

Other DIMACS modules related to this module:

Module 03-7