ADPROCLUS: a graphical user interface for fitting additive profile clustering models to object by variable data matrices

Behavior Research Methods

e-ISSN 1554-3528
Volume 43
Number 1

Behav Res (2010) 43:56-65
DOI 10.3758/
s13428-010-0033-0



Springer

Your article is protected by copyright and all rights are held exclusively by Psychonomic Society, Inc.. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your work, please use the accepted author's version for posting to your own website or your institution's repository. You may further deposit the accepted author's version on a funder's repository at a funder's request, provided it is not made publicly available until 12 months after publication.



Author's personal copy

Behav Res (2011) 43:56–65
DOI 10.3758/s13428-010-0033-0

ADPROCLUS: a graphical user interface for fitting additive profile clustering models to object by variable data matrices

Tom F. Wilderjans · Eva Ceulemans · Iven Van Mechelen · Dirk Depril

Published online: 24 November 2010
© Psychonomic Society, Inc. 2010

Abstract In many areas of psychology, one is interested in disclosing the underlying structural mechanisms that generated an object by variable data set. Often, based on theoretical or empirical arguments, it may be expected that these underlying mechanisms imply that the objects are grouped into clusters that are allowed to overlap (i.e., an object may belong to more than one cluster). In such cases, analyzing the data with Mirkin's additive profile clustering model may be appropriate. In this model: (1) each object may belong to no, one or several clusters, (2) there is a specific variable profile associated with each cluster, and (3) the scores of the objects on the variables can be reconstructed by adding the cluster-specific variable profiles of the clusters the object in question belongs to. Until now, however, no software program has been publicly available to perform an additive profile clustering analysis. For this purpose, in this article, the ADPROCLUS program, steered by a graphical user interface, is presented. We further illustrate its use by means of the analysis of a patient by symptom data matrix.

Keywords Additive clustering · Overlapping clustering · Object-by-variable data · ADCLUS

T. F. Wilderjans (✉) · I. Van Mechelen · D. Depril
Research Group of Quantitative Psychology and Individual Differences, Department of Psychology, Katholieke Universiteit Leuven, Tiensestraat 102, Box 3713, 3000, Leuven, Belgium
e-mail: tom.wilderjans@psy.kuleuven.be

E. Ceulemans
Department of Educational Sciences, Katholieke Universiteit Leuven, Leuven, Belgium

Introduction

In psychological research, information is often gathered about a set of variables for a set of objects, resulting in an object by variable data matrix. In the clinical domain, for example, such data are encountered when a group of patients is evaluated by a clinician with respect to the extent to which these patients exhibit a number of symptoms. As a second example, in the field of developmental psychology one often collects data to measure in which phase (regarding, e.g., cognitive or moral development) a group of children are.

The goal of analyzing such data is often to uncover their underlying structural mechanisms. In many cases, theoretical or empirical arguments can be given in favor of underlying mechanisms that imply a grouping of the objects into clusters. In the clinical example, the clusters may represent groups of patients that suffer from the same syndrome(s) or disease(s). In the example from developmental psychology, the children may be grouped into clusters of children that are in the same developmental phase.

If a grouping/clustering of the objects is implied by the underlying mechanisms, three types of groupings/clusterings can be distinguished. These are illustrated in Fig. 1.

The first type pertains to the case where each object belongs to one cluster only (see top panel of Fig. 1), which implies a partitioning of the objects into a number of distinct groups or non-overlapping clusters. In the developmental psychology example, such a partitioning could be relevant, with the clusters corresponding to different developmental phases. It can be expected that each child is in one of the considered developmental phases only, and hence that it belongs to one cluster only. To disclose such a partitioning, well-known clustering methods such as k-means (MacQueen, 1967) or latent class modeling (Goodman, 1974; Lazarsfeld & Henry, 1968) can be used.

Author's personal copy

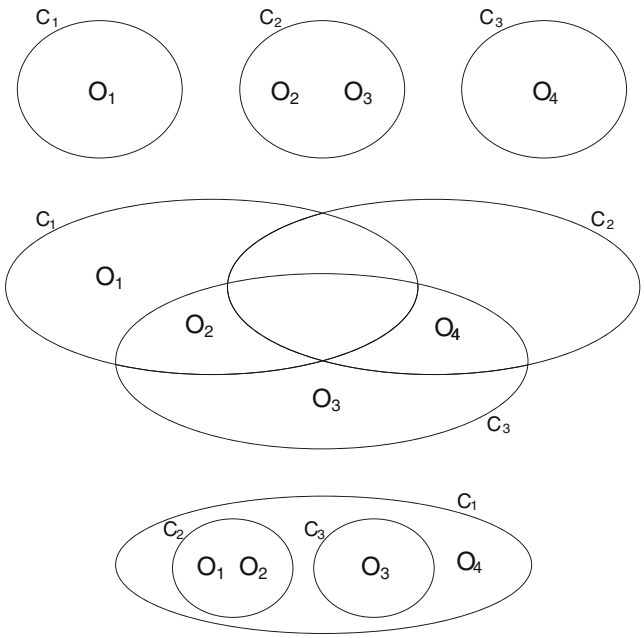


Fig. 1 Three types of groupings/clusterings: partitioning (*top*), overlapping clustering (*middle*), and hierarchical clustering (*bottom*). Objects are denoted by O_1, O_2, etc.; clusters are denoted by C_1, C_2, and C_3

A second type of clustering pertains to the situation in which each object may belong to more than one cluster (see middle panel of Fig. 1), implying that clusters are allowed to overlap. For instance, the clinical diagnosis case may call for a clustering, the clusters of which correspond to syndromes or diseases (e.g., schizophrenia, depression). Obviously, a patient may suffer from more than a single syndrome at the same time (i.e., syndrome comorbidity); as a consequence, each patient may belong to more than one cluster.

In the third type of clustering (see bottom panel of Fig. 1), as in the second type, an object may be a member of more than one cluster. However, unlike for the second type, the way the clusters overlap is now restricted in that two clusters are allowed to overlap only in the sense that one of the two clusters is a subset of the other one. As an example, one may consider a researcher who wants to study how a person categorizes a set of animals based on data regarding their characteristic features. The implicit animal taxonomy of the person in question then could comprise a set of clusters (e.g., Labradors, mammals, fish, dogs). The clusters in question may overlap (e.g., a Labrador is also a dog and a mammal). However, whenever two clusters overlap, one of the two clusters is a subset of the other one (e.g., the cluster of the dogs is a subset of the cluster of the mammals). In the literature, a broad range of hierarchical clustering methods is described to disclose such a nested type of clustering from object by variable data.

In the present paper, we will focus on the second type of clustering, that is, clusterings of the unconstrained overlapping type. To derive an overlapping clustering underlying some object by variable data set, one could consider using the additive clustering (ADCLUS) method as proposed by Shepard and Arabie (1979). ADCLUS, indeed, is based on an overlapping clustering model. However, the original ADCLUS method has the disadvantage that it can only operate on object-by-object similarity data. Consequently, to apply the original ADCLUS method to object by variable data, prior to the actual analysis, these data need to be converted into object-by-object similarities. Such a conversion implies a number of non-trivial and arbitrary choices, which may be fairly consequential for the subsequent clustering results. Furthermore, ADCLUS provides neither a reconstruction of the original object by variable data, nor a comprehensive representation of the mechanisms underlying them.

As an alternative solution, Mirkin (1987, 1990) proposed the additive profile clustering method. This method operates directly on object by variable data. Furthermore, the associated model implies a full reconstruction of the data entries in terms of an underlying structural mechanism.

To fit the additive profile clustering model to data, Depril, Van Mechelen, and Mirkin (2008) proposed an effective alternating least squares algorithm. Up to now, however, no software program has been publicly available to use this algorithm in data-analytic practice. In this paper, we present an implementation of this program along with a MATLAB graphical user interface, called ADPROCLUS; for users not experienced with MATLAB, a standalone version is provided. The main (user-related) features of this program are: (a) It can be downloaded freely from the internet (http://ppw.kuleuven.be/okp/software/ADPROCLUS), (b) it is flexible in use in that it allows the user to specify different options for the analysis, (c) it supports the user in selecting an appropriate model that describes the data well, without being overly complex, and (d) the results of the analysis can be saved in different formats, in order to enable a further processing of the obtained output (e.g., plotting the variable profiles for each cluster) with general purpose software packages like SPSS and SAS.

The remainder of this paper is organized in two main sections. In section 'Additive profile clustering', the theory of additive profile clustering is recapitulated. In section 'The ADPROCLUS software program' discusses how the ADPROCLUS program can be used in practice to perform an additive profile cluster analysis of an object by variable data matrix.

Additive profile clustering

Model

The additive profile clustering model (Mirkin, 1987) is a model for object by variable data. In the model, objects are

Author's personal copy

grouped into clusters that are allowed to overlap. Furthermore, for each cluster an associated variable profile is specified. The data (i.e., the scores on the variables) for each object then can be reconstructed by summing up the variable profiles of the clusters that object belongs to.

In particular, in additive profile clustering an $I \times J$ object by variable data matrix $\mathbf{X}$ is approximated by an $I \times J$ model matrix $\mathbf{M}$. This matrix $\mathbf{M}$ can be further decomposed into an $I \times K$ binary cluster membership matrix $\mathbf{A}$ and a $J \times K$ real-valued cluster profile matrix $\mathbf{P}$,

$$\mathbf{M} = \mathbf{A}\mathbf{P}', \tag{1}$$

with K indicating the number of clusters (which is to be specified by the user in advance; see below). Entry a_{ik} of $\mathbf{A}$ indicates whether object i belongs to cluster k ($a_{ik} = 1$) or not ($a_{ik} = 0$), with the clusters in question being allowed to overlap. The columns of matrix $\mathbf{P}$ contain the variable profiles for each cluster. The decomposition rule then implies that the reconstructed data values for object i in $\mathbf{M}$ can be computed as the sum of the profiles of the clusters that object i belongs to:

$$m_{ij} = \sum_{k=1}^{K} a_{ik} p_{jk}. \tag{2}$$

To illustrate the above, we will make use of the hypothetical model matrix $\mathbf{M}$ as displayed in Table 1. This matrix contains the (real-valued) reconstructed scores of 15 patients on 11 symptoms, which indicate to which extent each patient suffers from each symptom. In Table 2 (cluster memberships) and Table 3 (cluster profiles) an additive profile clustering model with three clusters is presented for the model matrix $\mathbf{M}$ in Table 1. Note that the matrices in Tables 1, 2 and 3 were obtained by applying the additive profile clustering algorithm (see section 'Algorithm') to the data set presented in Fig. 5 (see further). The clusters can be interpreted as syndromes. When looking at the associated symptom profiles (Table 3), the first cluster represents an affective disorder syndrome, with heightened levels of anxiety, social isolation, suicide, and depression. The second cluster seems to represent addiction, as evidenced by an increased level of substance abuse. The third cluster is associated with heightened levels of hallucinations, suspicion, agitation, and inappropriate affect, which all are typical symptoms for a paranoid schizophrenic disorder. Note that all three syndromes are also associated with increased levels of impairment of daily routine and leisure time activities.

The entries a_{ik} of the cluster membership matrix $\mathbf{A}$ (as displayed in Table 2) indicate whether or not person i suffers from syndrome k. One can see that syndrome comorbidity is present in that some affective disordered patients also suffer from addiction (i.e., patients 1, 6, 8, and 13), while one patient combines paranoid schizophrenia with addiction (i.e., patient 2) and another patient combines an affective disorder with paranoid schizophrenia (i.e., patient 10). The predicted degree to which a particular patient exhibits the different symptoms then can be determined by summing up the symptom profiles corresponding to the syndromes that patient suffers from. For example, the predicted symptom values for patient 2 (see the values in $\mathbf{M}$ in Table 1), who suffers from paranoid schizophrenia and addiction, are obtained by summing up the symptom profiles associated with the second (i.e., addiction) and third (i.e., paranoid schizophrenia) cluster.

Data analysis

Aim

The model matrix $\mathbf{M}$ in Table 1 contains exactly reconstructed symptom scores for each person (i.e., noise-free data). In practice, however, data always contain noise. The goal of the data analysis then is to find a model matrix $\mathbf{M}$ that fits the data matrix $\mathbf{D}$ as closely as possible in the least squares sense.

In particular, the aim of an additive profile cluster analysis with K ($< I$) clusters of an $I \times J$ data matrix $\mathbf{D}$ is to estimate an $I \times K$ cluster membership matrix $\mathbf{A}$ and a $J \times K$ cluster profile matrix $\mathbf{P}$ which are such that the value of the loss function

$$f_1(\mathbf{A}, \mathbf{P}) = \|\mathbf{X} - \mathbf{A}\mathbf{P}'\| = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{ij} - \sum_{k=1}^{K} a_{ik} p_{jk}\right)^2, \tag{3}$$

is minimized, with $\|.\|$ indicating the sum of the squared entries. Given a membership matrix $\mathbf{A}$, the associated conditionally optimal profile matrix $\mathbf{P}$ is given by $\boldsymbol{P} = \mathbf{X}'\mathbf{A}\,(\mathbf{A}'\mathbf{A})^{-1}$. Plugging this estimate in (3) yields

$$f_1(\mathbf{A}) = \left\|\mathbf{X} - \mathbf{A}(\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\mathbf{X}\right\|. \tag{4}$$

Obviously, minimizing (4) over $\mathbf{A}$ is equivalent to minimizing (3) over $\mathbf{A}$ and $\mathbf{P}$. Note that in (3), because the clusters are overlapping, the profiles in $\mathbf{P}$ do not correspond to the variable means for each cluster, as would be the case for a partition (i.e., first type of clustering). Note further that when the membership matrix $\mathbf{A}$ is restricted to a partition, loss function (3) and the k-means loss function coincide.

Algorithm

To minimize loss function (4), Depril et al. (2008) developed an alternating least squares (ALS) algorithm. In this algorithm, starting from a random or a (pseudo-)

Author's personal copy

Table 1 Patient by symptom model matrix **M** (obtained by applying an additive profile cluster analysis to the data in Fig. 5)

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
P_1	2.16	1.72	2.17	1.81	2.42	0.17	–0.07	–0.03	1.38	2.89	3.49
P_2	0.30	0.32	0.33	0.13	2.03	1.88	2.07	1.55	1.84	2.62	3.77
P_3	0.54	0.05	0.30	–0.13	0.52	0.04	0.40	–0.10	1.60	1.70	1.96
P_4	1.62	1.67	1.87	1.94	1.90	0.13	–0.47	0.07	–0.22	1.19	1.53
P_5	–0.24	0.27	0.03	0.26	1.51	1.84	1.67	1.65	0.24	0.92	1.81
P_6	2.16	1.72	2.17	1.81	2.42	0.17	–0.07	–0.03	1.38	2.89	3.49
P_7	1.62	1.67	1.87	1.94	1.90	0.13	–0.47	0.07	–0.22	1.19	1.53
P_8	2.16	1.72	2.17	1.81	2.42	0.17	–0.07	–0.03	1.38	2.89	3.49
P_9	1.62	1.67	1.87	1.94	1.90	0.13	–0.47	0.07	–0.22	1.19	1.53
P_{10}	1.38	1.94	1.90	2.20	3.41	1.97	1.20	1.72	0.02	2.11	3.34
P_{11}	0.54	0.05	0.30	–0.13	0.52	0.04	0.40	–0.10	1.60	1.70	1.96
P_{12}	1.62	1.67	1.87	1.94	1.90	0.13	–0.47	0.07	–0.22	1.19	1.53
P_{13}	2.16	1.72	2.17	1.81	2.42	0.17	–0.07	–0.03	1.38	2.89	3.49
P_{14}	–0.24	0.27	0.03	0.26	1.51	1.84	1.67	1.65	0.24	0.92	1.81
P_{15}	–0.24	0.27	0.03	0.26	1.51	1.84	1.67	1.65	0.24	0.92	1.81

S_1 = suicide, S_2 = depression, S_3 = anxiety, S_4 = social isolation, S_5 = inappropriate affect, S_6 = hallucinations, S_7 = agitation, S_8 = suspicion, S_9 = substance abuse, S_{10} = impairment in leisure time activities, S_{11} = impairment in daily routine activities. P_1 = patient 1, …, P_{15} = patient 15.

rational initial estimate for **A**, each row of **A** is alternatingly updated conditionally upon the other rows of **A**. This updating procedure is repeated until there is no further decrease in the loss function.

A random initial estimate of **A** is obtained by independently drawing entries from a Bernoulli distribution with parameter $\pi = .5$. As a rational initial estimate of **A**, one can take the clustering of the objects that results from subjecting the data to Mirkin's (1987, 1990) Principal Cluster Analysis (PCL) algorithm, in which the clusters are extracted one by one from the (residual) data. A pseudo-rational initial estimate of **A** can be determined by randomly drawing K data points (i.e., rows) from **X** as an initialization of **P** and by subsequently calculating the associated conditionally optimal membership matrix **A**.

To update a row of **A**, all possible binary patterns are checked (enumeratively) and the pattern that yields the lowest value on loss function (4) is retained.

Table 2 Cluster (membership) matrix of an additive profile clustering solution with three clusters for the model matrix **M** in Table 1

Patient	Cluster 1	Cluster 2	Cluster 3
P_1	1	1	0
P_2	0	1	1
P_3	0	1	0
P_4	1	0	0
P_5	0	0	1
P_6	1	1	0
P_7	1	0	0
P_8	1	1	0
P_9	1	0	0
P_{10}	1	0	1
P_{11}	0	1	0
P_{12}	1	0	0
P_{13}	1	1	0
P_{14}	0	0	1
P_{15}	0	0	1

Data-analytical choices

Preprocessing ADPROCLUS solutions are sensitive to multiplicative and additive transformations of the variables. Regarding multiplicative transformations (e.g., division of the variables by their standard deviation or range) one should take into account that, just as in the case of non-overlapping clustering (e.g., k-means), variables with a large variance may dominate the clustering. With respect to additive transformations (e.g., centering), one should bear in mind that, unlike in the non-overlapping case, such a transformation of one or more variables cannot be compensated by an additive transformation of the cluster profiles because of the cluster overlap.

With regard to additive transformations, one could argue, on substantive grounds that the mechanism as formalized by the ADPROCLUS model Eqs. 1 and 2 might operate at the level of deviations from some normative level instead of at the level of the raw scores. As an example, one may consider patient by symptom data with one symptom variable pertaining to body temperature or fever. When a

Author's personal copy

Table 3 Profile matrix of an additive profile clustering solution with three clusters for the model matrix **M** in Table 1

Symptom	Cluster 1	Cluster 2	Cluster 3
Suicide	1.62	0.54	–0.24
Depression	1.67	0.05	0.27
Anxiety	1.87	0.30	0.03
Social isolation	1.94	–0.13	0.26
Inappropriate affect	1.90	0.52	1.51
Hallucinations	0.13	0.04	1.84
Agitation	–0.47	0.40	1.67
Suspicion	0.07	–0.10	1.65
Substance abuse	–0.22	1.60	0.24
Impairment in leisure time activities	1.19	1.70	0.92
Impairment in daily routine activities	1.53	1.96	1.81

patient simultaneously suffers from two syndromes, it does not make sense to assume that his/her body temperature equals the sum of the body temperatures as associated with the two syndromes in question (as contained in the profile matrix **P** of the ADPROCLUS model). Yet, it would be more reasonable to assume that his/her increase in body temperature (compared to that of a healthy person) equals the sum of increases in body temperature as associated with the two syndromes involved.

In conclusion, based on substantive and/or technical arguments, it may be advisable to preprocess the raw data, prior to subjecting them to an ADPROCLUS analysis. Possible forms of preprocessing include: (1) dividing the variables by, for example, their range or standard deviation (Gordon, 1999; Milligan & Cooper, 1988) in order to undo between-variable differences in variance, and (2) converting raw scores into deviations from a mean, from a reference level, or from a normative score (e.g., the score of a healthy person as in the body temperature example).

In section 'The ADPROCLUS software program', to illustrate the ADPROCLUS software, a data set is used as a guiding example; this data set, which is displayed in Fig. 5, is well approximated by the reconstructed scores in Table 1. In order to remove differences in scales between the variables, prior to the analysis, these data were rescaled by dividing each variable by their standard deviation. No preprocessing in terms of additive transformations was needed, because the patients' scores on the symptoms were already expressed as deviations from the scores of a healthy individual.

For comparative purposes, we also performed a k-means analysis (MacQueen, 1967) with three clusters and a Ward hierarchical cluster analysis (Ward, 1963) on the same data (see Fig. 5). From Table 4, in which the resulting partitioning is displayed, it appears that k-means analysis is not able to disclose the grouping as implied by the additive profile cluster analysis. For example, the patients suffering from addiction only are categorized in two different groups, while the patients that suffer from schizophrenia only and those that combine schizophrenia with an addiction are categorized in the same cluster. From the dendrogram in Fig. 2 it can be seen that the hierarchical clustering solution is far from parsimonious and hard to interpret.

Multi-start procedure Alternating least squares algorithms, such as the ADPROCLUS algorithm, may result in a locally rather than a globally optimal solution. To minimize the risk of ending up in a local minimum, it is recommended to use a multi-start procedure. In such a procedure, the algorithm is performed multiple times, each time with different initial estimates for the parameters, and the best encountered solution across all runs is retained. Choices to be made on this level pertain to the type and the number of starts. Regarding the type of starts, Depril et al. (2008) advise to use a hybrid starting strategy that includes a number of random and a number of pseudo-rational starts

Table 4 Cluster (membership) matrix of a k-means analysis with three clusters for the data in Fig. 5

Patient	Cluster 1	Cluster 2	Cluster 3
P_1	0	1	0
P_2	0	0	1
P_3	0	0	1
P_4	1	0	0
P_5	0	0	1
P_6	0	1	0
P_7	1	0	0
P_8	0	1	0
P_9	1	0	0
P_{10}	0	1	0
P_{11}	1	0	0
P_{12}	1	0	0
P_{13}	0	1	0
P_{14}	0	0	1
P_{15}	0	0	1

Author's personal copy

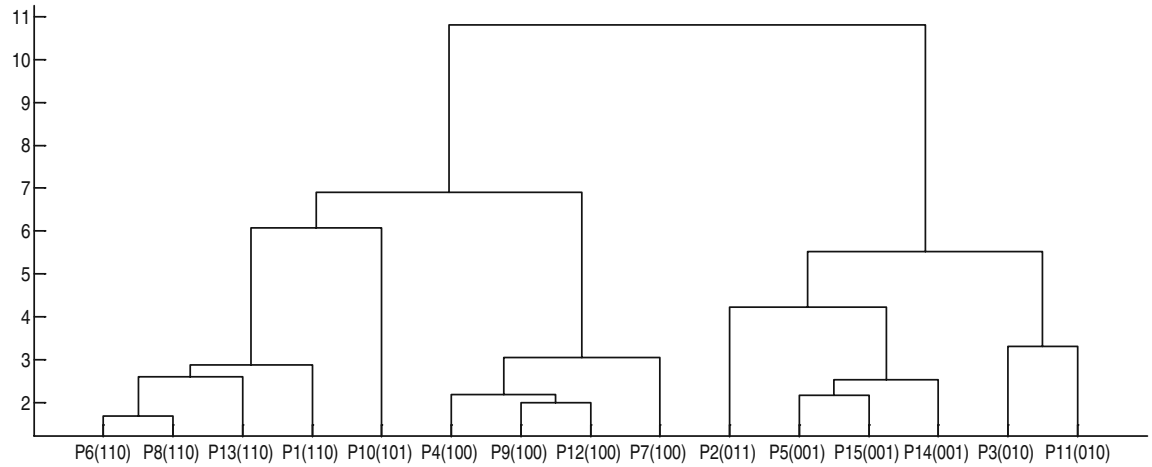


Fig. 2 Dendrogram resulting from applying a Ward hierarchical cluster analysis to the data set in Fig. 5. For comparative purposes, for each patient (indicated by $\mathbf{P_1}$, …, $\mathbf{P_{15}}$) the membership pattern from the overlapping clustering solution (see Table 2) is presented in *parenthesis*

(see section 'Algorithm') in addition to a rational start as obtained from Mirkin's PCL algorithm. Regarding the number of starts, one should note that increasing this number will, in general, improve the quality of the retained solution, yet at the expense of computation time.

Model selection Prior to the actual ADPROCLUS analysis, the user has to decide about the number of clusters *K*. In practice, this number is almost never known beforehand. To deal with this, one usually performs analyses with an increasing number of clusters. To determine the optimal number of underlying clusters, one then may rely on some model selection heuristic (like, e.g., a scree plot) that identifies the solution that has the best balance between model fit (i.e., the loss function value) and model complexity (i.e., the number of clusters). As an example, Fig. 3 displays a scree plot for the psychiatric diagnosis data. On the basis of this plot, one may select the solution with three clusters. Of course, when determining the number of clusters, interpretability and stability of the solution should also be taken into account.

The ADPROCLUS software program

Program handling

The ADPROCLUS software can be downloaded from the website http://ppw.kuleuven.be/okp/software/ADPROCLUS. The software is available in a MATLAB version and a standalone version for Microsoft Windows (*ADPROCLUS.exe*). To install the MATLAB version, one needs to store all the ADPROCLUS software files (i.e., the MATLAB figure file *ADPROCLUS* and a set of MATLAB m-files) in the same folder, and one needs to set the current MATLAB directory to this folder (or to make this folder part of the MATLAB path). Next, the software can be launched in MATLAB by typing *ADPROCLUS* at the command prompt:

>> ADPROCLUS < ENTER >

To launch the standalone version, the ADPROCLUS application needs to be installed (see the instructions file *ReadMe_Standalone.txt* at the above-mentioned Web site) and the user needs to double click on the *ADPROCLUS.exe* icon.

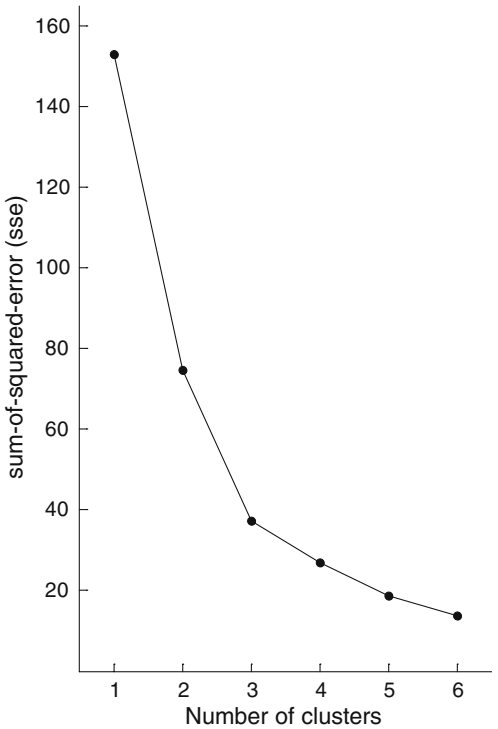


Fig. 3 Scree plot of the results of ADPROCLUS analyses for the patient by symptom data set

Author's personal copy

In both versions, a graphical user interface (GUI), which is displayed in Fig. 4, appears. This GUI consists of three compartments that enable the user to steer the analysis (i.e., Data description and data files, Analysis options, and Output files). Note that in Fig. 4 the boxes of the GUI already contain the information concerning the guiding example. To perform an analysis, one needs to specify the necessary information in the compartments and click the *Run analysis*-button. In the following sections, the three compartments will be discussed, together with error handling.

Data description and data files

Data description In the "Data description and data files" panel the user has to specify the number of rows and columns in the data matrix. The ADPROCLUS program is limited to data sets with maximally 10,000 rows and 10,000 columns. For our guiding example, as can be seen in Fig. 4, we specify that there are 15 rows (patients) and 11 columns (symptoms).

Data file The user has to specify the file that contains the (object by variable) data by clicking on the *Browse*-button and selecting the file in question. For the ADPROCLUS program, the selected data file should be an ASCII file (i.e., ".txt" file) that is organized as follows (see top panel of Fig. 5, in which the data file "data.txt" is displayed): The data file should contain as many rows and columns as specified in the data description panel, with no empty lines being allowed in between two rows of the data matrix (i.e., the rows may be separated by line breaks only). Within each row, the data elements may be separated by one or more spaces, commas, semicolons, tabs, or any combination of these. Each data element should be an integer or a real number, with decimal separators being denoted by a period and not by a comma.

Label file Optionally, the user may also provide labels for all row and column entities. This can be achieved by selecting in the "Object and variable labels" part of the "Data description and data files" compartment the "yes (specify the file in which the labels are stored)" option and by, subsequently, browsing for the file that contains the labels. This file should again be an ASCII file (".txt") containing two blocks of labels (i.e., the first block should pertain to the rows, and the second block to the columns), that may be separated by one or more empty lines. Within each block, the labels should be separated by line breaks only. Of course, the number and ordering of the row and column labels has to correspond to the number and ordering of the row and column entities in the data file. Each label should be a character string that may contain any kind of symbols. In the bottom panel of Fig. 5, the label file ("labels.txt") pertaining to the guiding example is shown. If the user does not want to provide row and column labels, the "no (no labels)" option should be selected. As a consequence, the

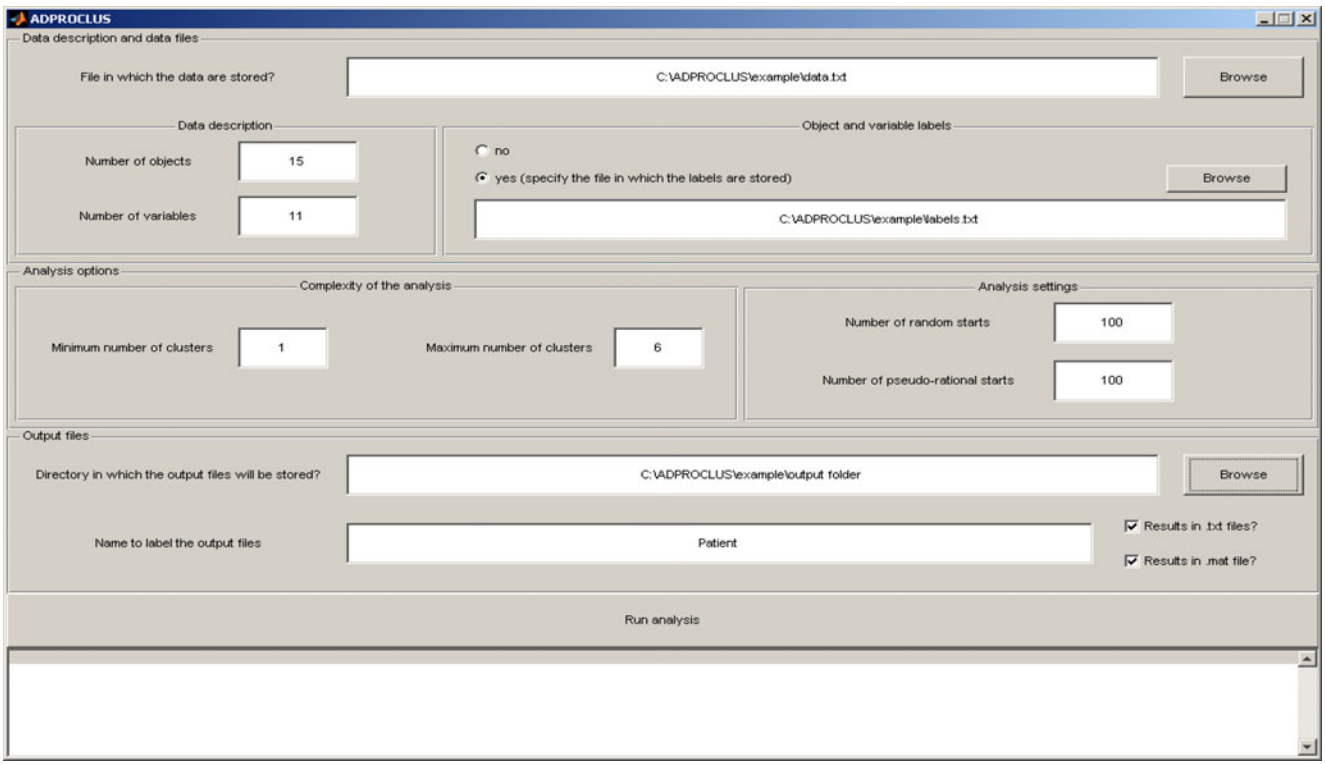


Fig. 4 Screenshot of the graphical user interface of the ADPROCLUS program

Author's personal copy

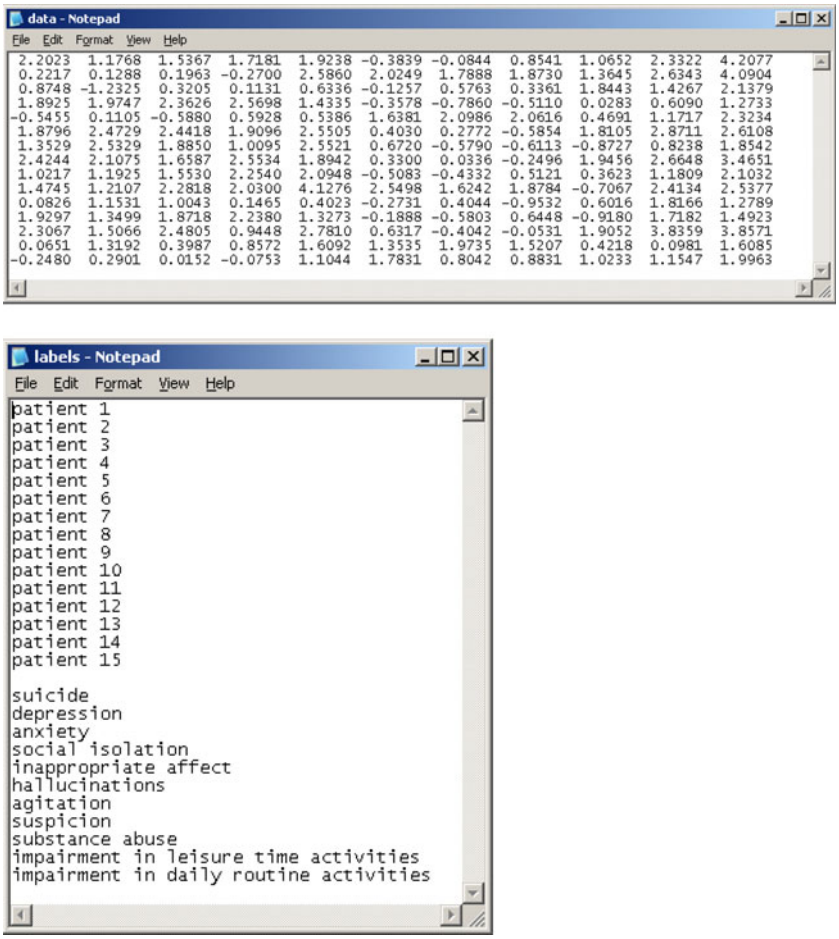


Fig. 5 Screenshot of the data file ("data.txt") and the file for the labels ("labels.txt")

ADPROCLUS program generates default row and column labels (e.g., "Row1", "Row2", "Column1", etc. …).

Analysis options

In this compartment, the user has to specify in the left-hand panel the number of clusters (i.e., the complexity of the analysis), and in the right-hand panel some analysis settings.

Complexity of the analysis The user has to decide about the minimum and maximum number of clusters. The minimum number of clusters should be greater than or equal to one, whereas the maximum number of clusters may not exceed the number of objects in the data (and, of course, should be greater than or equal to the minimum number of clusters). In Fig. 4, one can see that multiple analyses will be performed on the guiding data set with the number of clusters ranging from one to six.

Analysis settings To overcome the local minima problem, the ADPROCLUS program relies on a multi-start procedure. In the "Analysis settings" panel, the user has to specify how many pseudo-rational starts and how many random starts should be performed (see section 'Algorithm'). These numbers should be integers between 0 and 10,000. Note that increasing the number of pseudo-rational and/or random starts may improve the quality of the obtained solution (i.e., a lower loss function value), but at the expense of computation time. In the ADPROCLUS program, the default numbers of pseudo-rational and random starts equal 100.

Output files

In the "Output files" compartment the user has to specify the following information:

(1) The directory where all the standard output files that are generated by the ADPROCLUS program (i.e., ". mht" files) will be stored. To this end, the user clicks the *Browse*-button and selects the desired directory on the computer.
(2) A starting string, which should be typed into the corresponding box, for the names of all output files.

Author's personal copy

(3) Whether or not one wants additional output files in ".txt" format; this is achieved by checking/unchecking the "Results in .txt files?" option.
(4) Whether or not one wants in addition results stored in ".mat" format and into the MATLAB workspace (which is not possible for the standalone version); this is obtained by selecting/unselecting the "Results in .mat file?" option.

For the guiding example, one can see in Fig. 4 that all output files will be stored in the directory "C:\ADPROCLUS\example\output folder" and that the name of all these files will start with "Patient".

Standard output files As a result, the ADPROCLUS program will generate a series of ".mht" files, one for each complexity, of which the name (1) starts with the string that was specified in the "Output files" compartment and (2) contains the number of clusters. For instance, the results for the guiding example can be found in the files "Patient_1cluster.mht", "Patient_2clusters.mht", and so on. In each of these output files one may first find the analysis options. Second, the fit of the model (in terms of the sum-of-squared-error and the proportion of variance accounted for) is displayed. Third, the file contains the resulting cluster membership matrix (which implies an overlapping clustering for the objects). Fourth, one may find the obtained cluster profile matrix, which contains a variable profile for each cluster, along with a graphical representation of these profiles. The membership matrix and profile matrix obtained by applying ADPROCLUS to the data in Fig. 5 can be found in Table 2 and Table 3, respectively.

Overview file Finally, when analyses in different complexities are performed, an overview file (in ".mht" format) of the results of these analyses is generated (and stored in the selected output directory). This overview file consists of three parts. The first part displays the selected analysis options. In the second part, information is presented regarding the fit of the model to the data for all considered complexities. The third part displays a scree plot in which the number of clusters is plotted against the sum-of-squared-error (see section 'Data-analytical choices').

Extra output files When the user asked for additional output files in ".txt" format, the obtained solutions are also stored in files, the name of which starts with the specified string in the "Output files" compartment, and ends with "…_1cluster.txt", "…_2clusters.txt", and so on (in the guiding example: "Patient_1cluster.txt", etc.). When the user selected the "Results in .mat file?" option, an object with the name "ADPROCLUSsolution" will be stored in the MATLAB workspace and saved in a ".mat" file with the specified string as a filename (for the example: "Patient.mat"). This file contains a vector of cells, one cell for each number of clusters for an ADPROCLUS analysis. Each cell contains the following fields: (1) "DataDescription", (2) "AnalysisOptions", (3) "FitInformation", and (4) "Solution". These fields can be accessed by typing in MATLAB "ADPROCLUSsolution.NumberOfClusters{*number of clusters*}".

Status of the analysis and error handling

Once the analysis has been started (by clicking the *Run analysis*-button), information concerning the status of the analysis will be displayed in the box at the bottom of the GUI screen. To notify the user that the analysis has finished, a screen will pop up with the message "The computation is complete !". Subsequently, after clicking the *OK*-button, the results can be consulted in the output files, which will have been stored in the selected directory.

When the data or label file is incorrectly specified or does not comply with the required format, one or more error screens will appear with information on the problems encountered, and the analysis will stop. After clicking the *OK*-button(s), the user is given the opportunity to correctly specify the input files and/or the analysis options. To aid the user in this, the content of the error messages will also be displayed in the box at the bottom of the GUI screen. Once all errors have been corrected, the user should again click the *Run analysis*-button.

Conclusions

In this paper, we discussed additive profile clustering, which is a method for revealing the underlying mechanisms of an object by variable data matrix. Overlapping clustering methods (e.g., additive profile clustering) disclose these underlying mechanisms in terms of a set of overlapping clusters, whereas partitioning methods (e.g., k-means and latent class analysis) imply a set of non-overlapping clusters, and hierarchical clustering methods imply a set of overlapping clusters with constraints on the overlap. Note that any overlapping clustering (with K groups) of a set of objects as obtained from a hierarchical or an unconstrained overlapping cluster analysis, implies a partitioning of the objects (with minimum K and at most 2^K groups). As a consequence, in general, partitioning methods will need a larger number of clusters to fully represent the grouping underlying an object by variable data set than overlapping or hierarchical clustering methods (e.g., for the data in Fig. 5, a partitioning method will need six clusters to fully represent the six groups of patients—the six different membership patterns that are present in Table 2—that are implied by the overlapping clustering of the patients).

Author's personal copy

Further, in this paper we introduced ADPROCLUS, a program, along with a graphical user interface, to fit an additive profile clustering model to an object by variable data matrix. This program is freely available and allows the user to control the analysis in a flexible way. By means of a scree plot, it further assists the user in determining the appropriate number of clusters. Finally, it offers the user the possibility to store the obtained solutions and associated fit information in different file formats, which facilitates a further processing of the results by means of general-purpose statistical software programs like SAS and SPSS.

Finally, we point to two limitations of the ADPROCLUS software. First, in its present form the program cannot handle missing data. Second, for very large data sets (i.e., data sets with a very large number of objects and/or variables) computation time may become a burden.

Acknowledgments The research reported in this paper has been partially supported by the Research Council of K. U. Leuven (GOA/2010/002), by the Belgian Federal Science Policy (IAP/P6/03), and by IWT Flanders (IWT/060045/SBO Bioframe). Requests for reprints should be sent to Tom F. Wilderjans. We would like to thank Kristof Meers for his helpful comments regarding the programming of the graphical user interface.

References

Depril, D., Van Mechelen, I., & Mirkin, B. G. (2008). Algorithms for additive clustering of rectangular data tables. *Computational Statistics and Data Analysis, 52*, 4923–4938. doi:10.1016/j.csda.2008.04.014

Goodman, L. A. (1974). Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika, 61*, 215–231.

Gordon, A. (1999). *Classification* (2nd ed.). London: Chapman and Hall-CRC.

Lazarsfeld, P. F., & Henry, N. W. (1968). *Latent structure analysis*. Boston: Houghton Mill.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In L. M. Le Cam & J. Neyman (Eds.), *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, pp. 281–297). Berkeley: University of California Press.

Milligan, G. W., & Cooper, M. C. (1988). A study of standardization of variables in cluster analysis. *Journal of Classification, 5*, 181–204. doi:10.1007/BF01897163

Mirkin, B. G. (1987). The method of principal clusters. *Automation and Remote Control, 10*, 131–143.

Mirkin, B. G. (1990). A sequential fitting procedure for linear data analysis models. *Journal of Classification, 7*, 167–195. doi:10.1007/BF01908715

Shepard, R. N., & Arabie, P. (1979). Additive clustering: Representation of similarities as combinations of discrete overlapping properties. *Psychological Review, 86*, 87–123. doi:10.1037/0033-295X.86.2.87

Ward, J. H., Jr. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association, 58*, 236–244.