Decision Support System Methodology Using a Visual Approach for Cluster Analysis Problems

Ran M. Bittmann

School of Business Administration

Ph.D. Thesis

Submitted to the Senate of Bar-Ilan University

Ramat-Gan, Israel July, 2008

This work was carried out under the supervision of Dr. Roy Gelbard and Prof. Abraham Carmeli (School of Business Administration), Bar-Ilan University.

Table of Contents

Abstract

Classification and clustering decisions arise frequently in business applications such as recommendations concerning products, markets, human resources, etc. Currently, decision makers must analyze diverse algorithms and parameters on an individual basis in order to establish preferences on the decision-issues they face, because there is no supportive model or tool which enables comparing different result-clusters generated by these algorithms and parameters' combinations.

The suggested methodology is using Multi-Algorithm-Voting (MAV) a method developed to analyze and visualize results of multiple algorithms, were each one of them suggests a different decision. The visualization uses a Tetris like format in which all distributions (decisions) are ordered in a Matrix, where each distribution suggested by a specific algorithm is presented in a column of the said Matrix, and each data component (case) is presented in a row of the Matrix. "Local decisions" (of each specific algorithm, concerning each case) are presented as "Tags" in the cells of the said Matrix.

The MAV method associates the "arbitrary Tags" to each other, using an optimized algorithm, based on the local search algorithm framework, for the association of multiple distribution, developed for that purpose. Each association is presented in a visual form, for example using color codes. The colors are consistent over the said Matrix and similar colors, even on different rows, represent similar classification (decision). While used for the analysis of clustering using multiple algorithms, the analysis and the presentation methods can be used to associate and analyze multiple distributions in general.

The MAV method calculates the quality of each association for each row, representing a data component. The quality can be calculated, but is not limited to, as the Homogeneity (or Heterogeneity) of the association of a single data component over all the algorithms used in the analysis. Then it pinpoints the best association based on the quality meter used.

The MAV method enables not only visualization of results produced by diverse algorithms, but also as quantitative analysis of the results.

Preface

1. Introduction

The problem of analyzing datasets and classifying them into clusters based on known properties is a well known problem with implementations in fields such as finance (e.g. fraud detection), computer science (e.g. image processing), marketing (e.g. market segmentation), medicine (e.g. diagnostics), among others (Clifford & Stephenson, 1975; Erlich, Gelbard, & Spiegler, 2002; Jain, Murthy, & Flynn, 1999; Shamir & Sharan, 2002). Cluster analysis research studies evaluate different algorithms by performing them on known datasets with known true results and comparing their output, and the algorithms' accuracy, to the true classification. The commercial products running these algorithms neither show the resulting clusters of multiple methods nor give the researcher tools with which to analyze and compare the outcomes of the different tools.

Within this context, the this work presents a methodology that provides:

- A visual presentation of multiple classification suggestions, resulting from diverse algorithms.
- A comparison of the different results.
- A comparison of the results when different numbers of clusters are evaluated.
- An evaluation of the different results not only when the true classification is known, but also when the true classification is unknown.

Studies that compare different algorithms (Erlich, Gelbard, & Spiegler, 2002; Shamir & Sharan, 2002) find it difficult to give an exclusive contingency approach as to which method is preferable, since such a contingency approach needs to cover all problem types, data types and result types. This is complicated to define mathematically.

Within this context, this work is among the first to:

- Suggest a methodology and provide tools to recommend a preferred method for a given problem.

- Suggests a methodology and provide tools to recommend a preferred number of clusters for a given problem.
- Provide a visual approach to accompany the mathematical processing that enables for a presentation of the full spectrum of results to acquaint the researcher with the classification tools' possible outcomes.
- Provide an immediate indication of the areas of contention between the different algorithms.
- Effect analysis by using different numbers of clusters for the classification problem.

The conventional approach is to apply an algorithm from a set of algorithms tuned by the algorithm parameters based on the dataset properties' criteria and the researcher's expertise. This approach, however, limits the result to the effectiveness of the chosen algorithm and leaves the researcher totally in the dark when the classification of the dataset is unknown. It does not show us which samples are hard to classify or how effective the chosen properties are for the desired classification.

Furthermore, visualization of the dataset and its classification is virtually impossible when more than three properties are used, since displaying the dataset in this case will require giving up on some of the properties in order to display the dataset, or using some other method to display the dataset's distribution over four dimensions or more. This makes it very difficult to relate to the dataset samples and understand which of these samples are difficult to classify (in some cases, even when they are classified correctly), and which samples and clusters stand out clearly (Boudjeloud & Poulet, 2005; de Olivera & Levkowitz, 2003; Shultz, Mareschal, & Schmidt, 1994).

Even when the researcher uses multiple algorithms in order to classify the dataset, there are only a few tools that allow him or her to use the outcome of the algorithms' application.

This work suggests a methodology and provides measures that provide the researcher with tools to combine the power of multiple algorithms; compare their results and present them in a clear visual manner. The result is the foundation for a Decision Support System (DSS) that can be used to analyze datasets with both known and unknown classifications.

2. Research Objectives

This work outlines a methodological process and indicates criteria for cluster analysis decision-making using a visual approach. It explains how to best utilize the known set of tools, mainly algorithms that allow us to build a DSS. Using the suggested methodology, the DSS is helpful when trying to decide upon:

- The preferred cluster analysis algorithm
- The preferred number of clusters to divide the dataset into
- Evaluating the classification properties
- Identifying inconsistent samples

As a visual aid, the process uses a clear visualization that encompasses the cluster analysis into a comprehensible, two-dimensional perspective. This view allows a comparison of the affect of the different methods on the dataset, the effectiveness of the classification properties and the classification consistency of the individual samples.

The output includes a set of measures that quantifies the results and directs the researcher in making the decisions outlined above.

3. Theoretical Background

3.1. Cluster Analysis – Algorithms

In order to classify a dataset of samples according to a given set of properties, the researcher uses algorithms that process the properties of the dataset samples and associate them with suggested clusters. The association is performed by calculating a likelihood measure that indicates the likelihood of a sample to be associated with a certain cluster. Below is a short description of commonly used algorithms.

3.1.1. Two Step

This algorithm is used for large datasets and is applicable to both continuous and categorical properties. It is based, as its name implies, on two passes on the dataset. The first pass divides the dataset into a coarse set of sub-clusters, while the second pass groups the sub-clusters into the desired number of clusters. This algorithm is dependent on the order of the samples and may produce different results based on the initial order of the samples. The desired number of clusters can be determined automatically, or it can be a predetermined fixed number of clusters.

3.1.2. K-Means

This algorithm is used for large datasets and is applicable to both continuous and categorical properties. It requires that the number of clusters used to classify the dataset will be pre-determined. It is based on determining arbitrary centers for the desired clusters, associating the samples with the clusters by using a pre-determined distance measurement, iteratively changing the center of the clusters and then re-associating the samples. The length of the process is very much dependent on the initial setting of the clusters’ centers and can be improved if knowledge exists regarding the whereabouts of these clusters’ centers.

3.1.3. Hierarchical methods

This is a set of algorithms that work in a similar manner. These algorithms take the dataset properties that need to be clustered and start initially by classifying the dataset so that each sample represents a cluster. Next, it merges the clusters in steps. Each step merges two clusters into a single cluster until there is only one cluster (the dataset) remaining. The algorithms differ in the way in which distance is measured between clusters, mainly by using two parameters: the distance or likelihood measure, e.g. Euclidean, Dice, etc. and the cluster method, e.g. between group linkage, nearest neighbor, etc.

This work uses these well-known Hierarchical Methods to classify the datasets:

- *Average Linkage (within groups)* – This method calculates the distance between two clusters by applying the likelihood measure to all the samples in the two clusters. The clusters with the best average likelihood measure are then united.
- *Average Linkage (between groups)* – This method calculates the distance between two clusters by applying the likelihood measure to all the samples of one cluster and then comparing it with all the samples of the other cluster. Once again, the two clusters with the best likelihood measure are then united.
- *Single Linkage (nearest neighbor)* – This method, as in the Average Linkage (between groups) method, calculates the distance between two clusters by applying the likelihood measure to all the samples of one cluster and then comparing it with all the samples of the other cluster. The two clusters with the best likelihood measure, from a pair of samples, are united.
- *Complete Linkage (furthest neighbor)* – This method, like the previous methods, calculates the distance between two clusters by applying the likelihood measure to all the samples of one cluster and then comparing it with all the samples of another cluster. For each pair of clusters the pair with the worst likelihood measure is taken. The two clusters with the best likelihood measure of those pairs are then united.
- *Centroid* – This method calculates the centroid of each cluster by calculating the mean average of all the properties for all the samples in

each cluster. The likelihood measure is then applied to the means of the clusters and the clusters with the best likelihood measure between their centroids are united.

- *Median* – This method calculates the median of each cluster. The likelihood measure is applied to the medians of the clusters and the clusters with the best median likelihood are then united.
- *Ward* – This method calculates the centroid for each cluster and the square of the likelihood measure of each sample in the cluster and the centroid. The two clusters, which when united have the smallest (negative) affect on the sum of likelihood measures, are the clusters that need to be united.

3.1.4. Likelihood Measure

The likelihood measure is used to measure the similarities of the samples that form a specific cluster. This similarity is measured by the distance between the samples, the smaller the distance, the more likely that the samples belong to the cluster.

Among the common measures for the distance used as a likelihood measure are:

- *Euclidean distance* – This distance is calculated as the geometric distance in the multidimensional space. This distance is calculated as:

$$\sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2}$$

- *Squared Euclidean distance* – In some cases it is desired to give more weight to distant samples, in this case this measure is taken. This distance is calculated as:

$$\sum_{i=1}^{n}(X_i - Y_i)^2$$

- *Chebychev distance* – This distance measurement defines the distance between two samples as the largest distance on one of its dimensions. This distance is calculated as:

$$\max|X_i - Y_i|$$

3.2. Cluster Analysis – Visualization

Currently, datasets are analyzed according to the following method:

- The researcher selects the best classification algorithm based on his or her experience and knowledge of the dataset.
- The researcher tunes the chosen classification algorithm by determining parameters such as the likelihood measure.
- The researcher applies the algorithm to the dataset using one of the following options:
  - Predetermination of a fixed number of clusters to divide the dataset into (supervised classification).
  - Deciding on the preferred number of clusters to classify the dataset based on the algorithm output (unsupervised classification).

3.2.1. Dendrogram

When hierarchical classification algorithms are applied, the researcher may use a dendrogram, depicted in Figure 1, which is a tree-like graph that presents the merger of clusters from the initial case, where each sample is a different cluster, to the total merger, where the whole dataset is one cluster. The connecting lines in a dendrogram represent clusters that are joined, while their distance from the base represent the likelihood coefficient for the merger. The shorter the distance, the more likely the clusters will merge. Though the dendrogram provides the researcher with some sort of a visual representation, it is limited to only a subset of the algorithms used. Furthermore, the information in the dendrogram relates only to the used algorithm and does not compare or utilize additional algorithms. The information itself serves as a visual aid to joining clusters, but does not provide a good indication of inconsistent samples in the sense that their position in the dataset spectrum according to the chosen properties is misleading, and likely to be wrongly classified

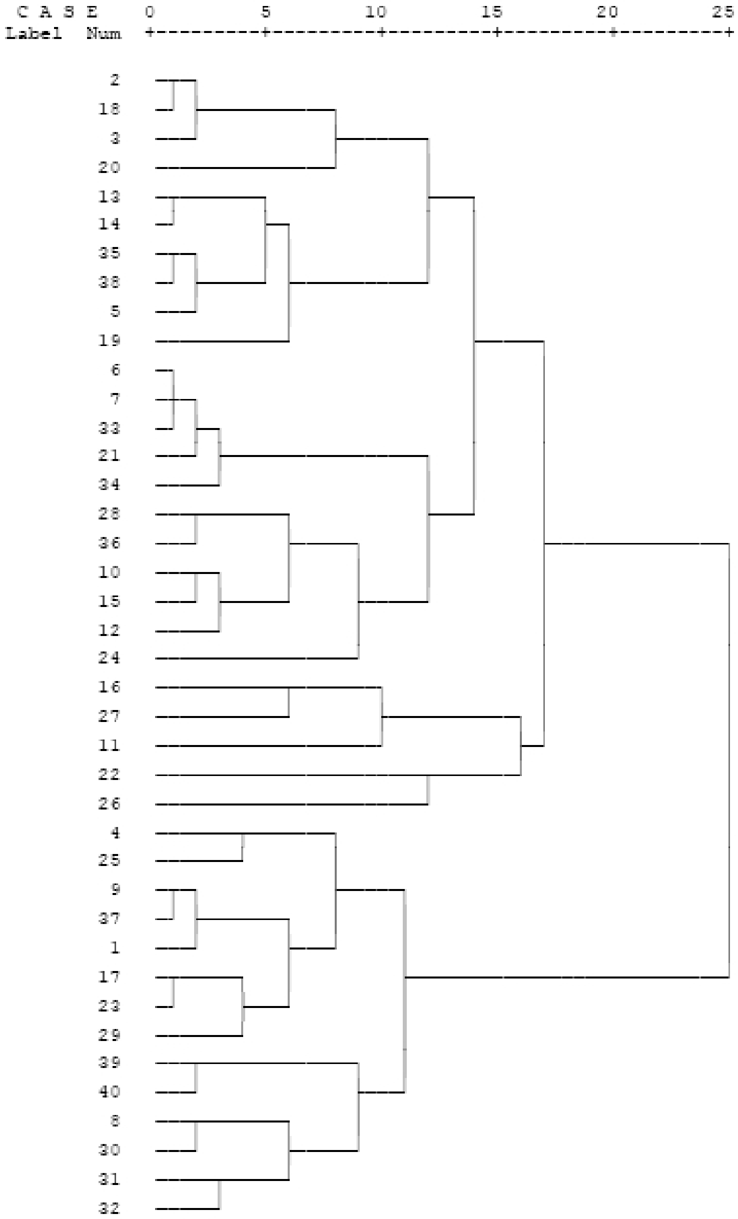


Figure 1: Sample Dendrogram

3.2.2. Discriminant Analysis & Factor Analysis

The problem of clustering may be perceived as finding functions applied to the variables that discriminate between samples and decide on cluster membership. Since usually there are more than two or even three variables it is difficult to visualize the samples in such multidimensional spaces. Some methods use discriminating functions, which are a transformation of the original variables, and present them on two- dimensional plots. Discriminant function analysis is analogous to multiple regressions. Two-group discriminant analysis is also called Fisher linear discriminant analysis (Fisher, 1936). In general, in this approach we fit a linear equation of the type:

$$Group = a + b_1 \cdot x_1 + b_2 \cdot x_2 + \cdots + b_m \cdot x_m$$

Where:

- a - is a constant
- $b_1 \ldots b_m$ - are regression coefficients

The variables (attributes) with significant regression coefficients are the ones that contribute most to the prediction of group membership. However, these coefficients do not tell us which groups the respective functions discriminate. The means of the functions across groups identify the group's discrimination. This can be visualized by plotting the individual scores for the discriminant functions, as illustrated in Figure 2 (Abdi, 2007).

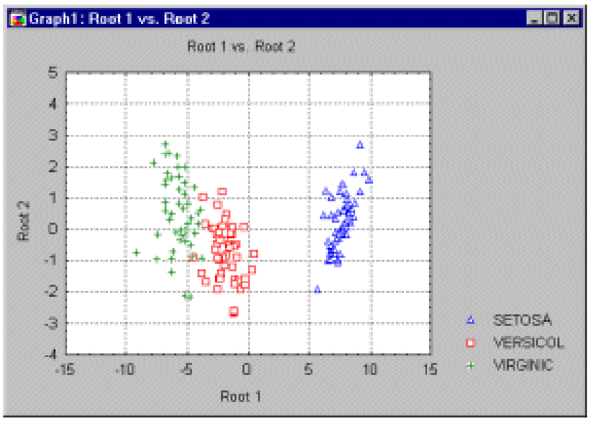


Figure 2: Discriminant Analysis of Fisher's Iris Dataset

3.2.3. Factor Analysis

Factor analysis is another way to determine which variables (attributes) define a particular discriminant function. The former correlations can be regarded as factor loadings of the variables on each discriminant function. Figure 3 (Raveh, 2000) illustrates the visualization of both correlations between the variables in the model (using adjusted Factor Analysis), and discriminant functions using a tool that combines these two methods. Each ray represents one variable (property). The angle between any two rays presents the correlation between these variables (possible factors).

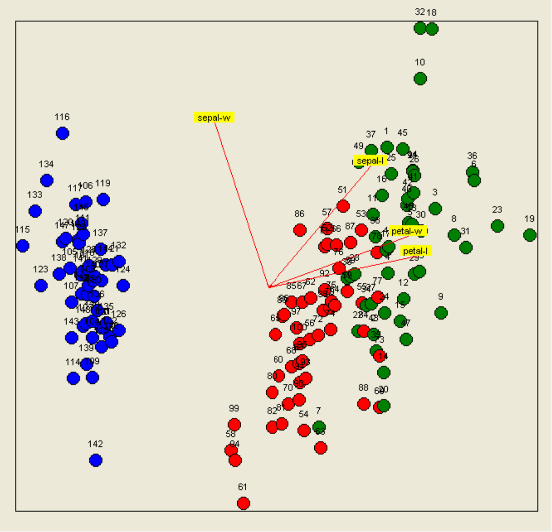


Figure 3: Factor Analysis of Fisher's Iris Dataset

These methodologies are usually incapable of making comparisons between different algorithms and leave the decision-making, regarding which algorithm to choose, to the researcher. This leaves the researcher with very limited visual assistance and prohibits the researcher from having a full view of the relations between the samples and a comparison between the dataset classifications based on the different available tools.

3.3. Local Search Algorithms

Local search algorithms are algorithms that allow for optimize computationally complex problems (Henderson, 2001; Kanungo, Mount, Netanyahu, Piatko, Silverman, & Wu, 2002; Sensen, 1999) like the problem of finding the best association of different classifications. These algorithms are incomplete in the sense that the solution they provide may not be the best solution. The researcher may control the probability of finding the best or good solution with adjusting the number of iteration that the algorithm performs. This results are a tradeoff between the resources and the probability to achieve a good result.

These algorithms are a family of algorithms that search for the best results in the neighborhood of the current position. Improving the results with each step until a stop criteria is met. A framework for local search algorithms can be seen below (Sensen, 1999):

```
$N :=$ Number of repetitions
$\tilde{s} := \emptyset$;
for $i := 1$ to $N$ do
    $s :=$ initial solution;
      while there is a better neighbor of $s$ with better quality do
        $s :=$ one arbitrary neighbor of $s$ with better quality;
    end while
    if $s$ is better than $\tilde{s}$ then
        $\tilde{s} := s$ ;
    end if
end for
return $\tilde{s}$ ;
```

In order to perform the algorithm a neighborhood needs to be defined as well as the quality of the results and the way to compare between them. A stop criteria signals when the algorithm stops searching for a better solutions. This can be based on the quality of the results, but also on the number of iterations the researcher decided to invest in the process.

4. Research Hypothesis

This work is an analytical research that is verified in addition to its analytical tools by the implementation of a prototype that has been activated over well known datasets (Fisher, 1936; PACE New Car & Truck 1993 Buying Guide, 1993; Consumer Reports: The 1993 Cars - Annual Auto Issue, 1993).

This work is not an empirical study and therefore the hypotheses used in the research are:

1. A decision support system methodology using visual approach for cluster analysis problems can be modeled.
2. Intuitive control measures based on visualization of a research cluster analysis can be modeled to assist the human decision maker.
3. This model can be verified in multiple research environments.

5. The Proposed Model

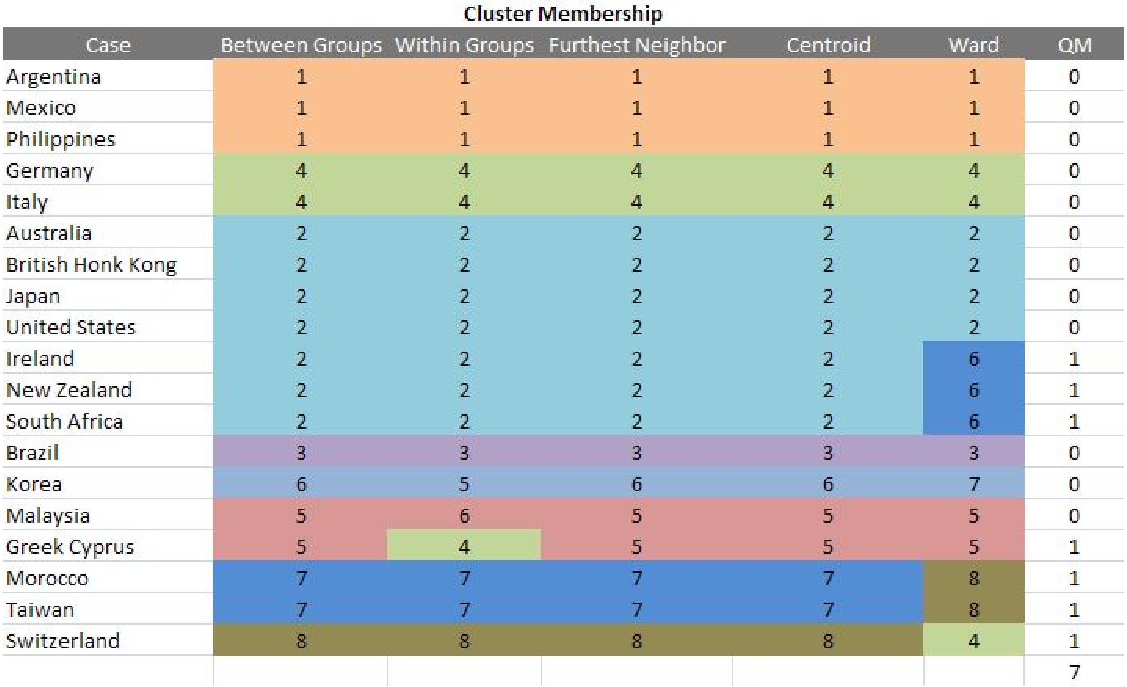
5.1. The Model Concept

The methodology presents a classification model from a clear, two-dimensional perspective, together with tools used for the analysis of this perspective.

5.1.1. Vote Matrix

The concept of the ‘Vote Matrix’ process recognizes that each algorithm represents a different view of the dataset and its clusters, based on how the algorithm defines a cluster and how the algorithm measures the distance of a sample from a cluster. Therefore, each algorithm is given a “Vote” as to how it perceives the dataset should be classified.

The methodology is based on a “Vote Matrix”, depicted in Figure 4, generated by the “vote” of each algorithm used in the process. Each row represents a sample and each column represents an algorithm and its vote for each sample about which cluster it should belong to, according to the algorithm’s understanding of clusters and distances.



Cluster Membership

Case	Between Groups	Within Groups	Furthest Neighbor	Centroid	Ward	QM
Argentina	1	1	1	1	1	0
Mexico	1	1	1	1	1	0
Philippines	1	1	1	1	1	0
Germany	4	4	4	4	4	0
Italy	4	4	4	4	4	0
Australia	2	2	2	2	2	0
British Honk Kong	2	2	2	2	2	0
Japan	2	2	2	2	2	0
United States	2	2	2	2	2	0
Ireland	2	2	2	2	6	1
New Zealand	2	2	2	2	6	1
South Africa	2	2	2	2	6	1
Brazil	3	3	3	3	3	0
Korea	6	5	6	6	7	0
Malaysia	5	6	5	5	5	0
Greek Cyprus	5	4	5	5	5	1
Morocco	7	7	7	7	8	1
Taiwan	7	7	7	7	8	1
Switzerland	8	8	8	8	4	1
						7

Figure 4: Sample Vote Matrix

5.1.2. Heterogeneity Meter

The methodology requires the association of clusters from different algorithms, i.e. since each algorithm divides the dataset into different clusters. Although the number of clusters in each case remains the same for each algorithm, tools are required to associate the clusters of each algorithm, e.g. cluster number two according to algorithm A1 is the same as cluster number 3 according to algorithm A2. To achieve this correlation, we calculate a measure called the *Heterogeneity Meter* for each row, i.e. the collection of votes for a particular sample, and sum it up for all the samples.

The *Heterogeneity Meter* can be calculated in multiple manners, two popular ways to calculate the *Heterogeneity Meter* are:

5.1.2.1. Squared Vote Error (SVE)

SVE is calculated as the square sum of all the votes that did not vote for the chosen classification. It is calculated as follows:

$$H = \sum_{i=1}^{n} (N - M_i)^2$$

Where:

- H - is the *Heterogeneity Meter*
- N - is the number of algorithms voting for the sample
- M - is the maximum number of similar votes according to a specific association obtained for a single sample
- i - is the sample number
- n - is the total number of samples in the dataset

5.1.2.2. Distance From Second Best (DFSB)

DFSB is calculated as the difference in the number of votes that the best vote, i.e. the vote common to most algorithms, received and the number of votes the second best vote received. The idea is to find out how much separates the best vote from the rest. This is actually a homogeneity meter as a higher score indicates less heterogeneity. It is calculated as follows:

$$H = \sum_{i=1}^{n} \left(B_i - SB_i \right)$$

Where:

- H - is the *Homogeneity Meter*
- B - is the best, i.e. the cluster voted most times; cluster for a given sample
- SB - is the second best cluster for a given sample
- i - is the sample number
- n - is the total number of samples in the dataset

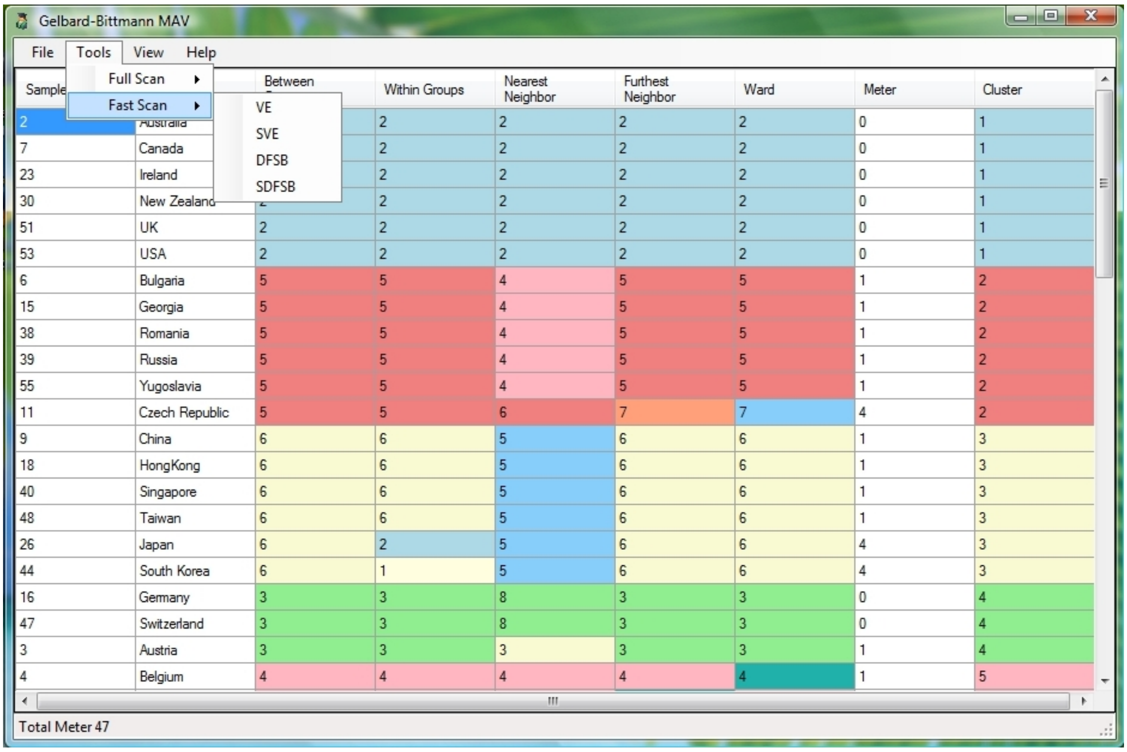
To maintain consistency in the association of the clusters a negative value for the DFSB meter is used changing it to a *Heterogeneity* meter.

The *SVE* usually yields clearer associated clusters than the *DFSB* meter that emphasizes the best associated samples. Using the *SVE* meter, the decision maker can identify which samples belong to which cluster with the highest significance, while using the *DFSB* the decision maker can identify more easily outstanding samples.

6. Research Methods

6.1. Research Tools

To implement the research on the common datasets we developed a tool that implements the methodology on different clustering results. We used SPSS Inc.'s SPSS statistical analysis software to perform the dataset clustering using multiple algorithms as described above. The developed tool performed the association of the algorithms as required by the methodology. An example of its output is depicted in Figure 5. The tool reads the clustering data from common data analysis programs such as Microsoft Excel. The tool performs the cluster association using different methods. When there is a small number of clusters and algorithms that allow passing on all possible options brute force association may be used. For other cases an accelerated association algorithm was developed.



Sample		Between	Within Groups	Nearest Neighbor	Furthest Neighbor	Ward	Meter	Cluster
2	Australia		2	2	2	2	0	1
7	Canada		2	2	2	2	0	1
23	Ireland		2	2	2	2	0	1
30	New Zealand	2	2	2	2	2	0	1
51	UK	2	2	2	2	2	0	1
53	USA	2	2	2	2	2	0	1
6	Bulgaria	5	5	4	5	5	1	2
15	Georgia	5	5	4	5	5	1	2
38	Romania	5	5	4	5	5	1	2
39	Russia	5	5	4	5	5	1	2
55	Yugoslavia	5	5	4	5	5	1	2
11	Czech Republic	5	5	6	7	7	4	2
9	China	6	6	5	6	6	1	3
18	HongKong	6	6	5	6	6	1	3
40	Singapore	6	6	5	6	6	1	3
48	Taiwan	6	6	5	6	6	1	3
26	Japan	6	2	5	6	6	4	3
44	South Korea	6	1	5	6	6	4	3
16	Germany	3	3	8	3	3	0	4
47	Switzerland	3	3	8	3	3	0	4
3	Austria	3	3	3	3	3	1	4
4	Belgium	4	4	4	4	4	1	5

Total Meter 47

Figure 5: Prototype Screenshot

6.1.1. Brute Force Clustering Association

Using Brute Force to test all the possible associations in order to reach the best association the following flow, depicted in Figure 6 is required:

Step 1 - Initialize all arguments including a permutation table that includes all possible permutations for different cluster association based on all algorithms.

Step 2 - The total of all the Heterogeneity meters is calculated for all the samples. The total is initialized in the beginning of the loop.

Step 3 - The Heterogeneity Meter is calculated for each sample.

Step 4 - The Heterogeneity Meter is added to the total for that permutation.

Step 5 - Steps 3 - 4 are performed in a loop over all samples.

Step 6 - The new total is compared to the best, .i.e. smallest, total so far.

Step 7 - If the total is better, i.e. smaller, than it replaces the best total so far and the permutation is saved for reference.

Step 8 - Steps 2-7 are performed in a loop over all permutations.

Step 9 - The user is presented with the best cluster association.

The complexity of finding the best association using the is calculated as follows:

$$O(D \cdot Q \cdot C!^{(A-1)})$$

Where:

- D - is the size of the dataset
- Q - is the complexity of calculating the Heterogeneity quality meter
- C - is the number of clusters
- A - is the number of algorithms “voting” for the clustering

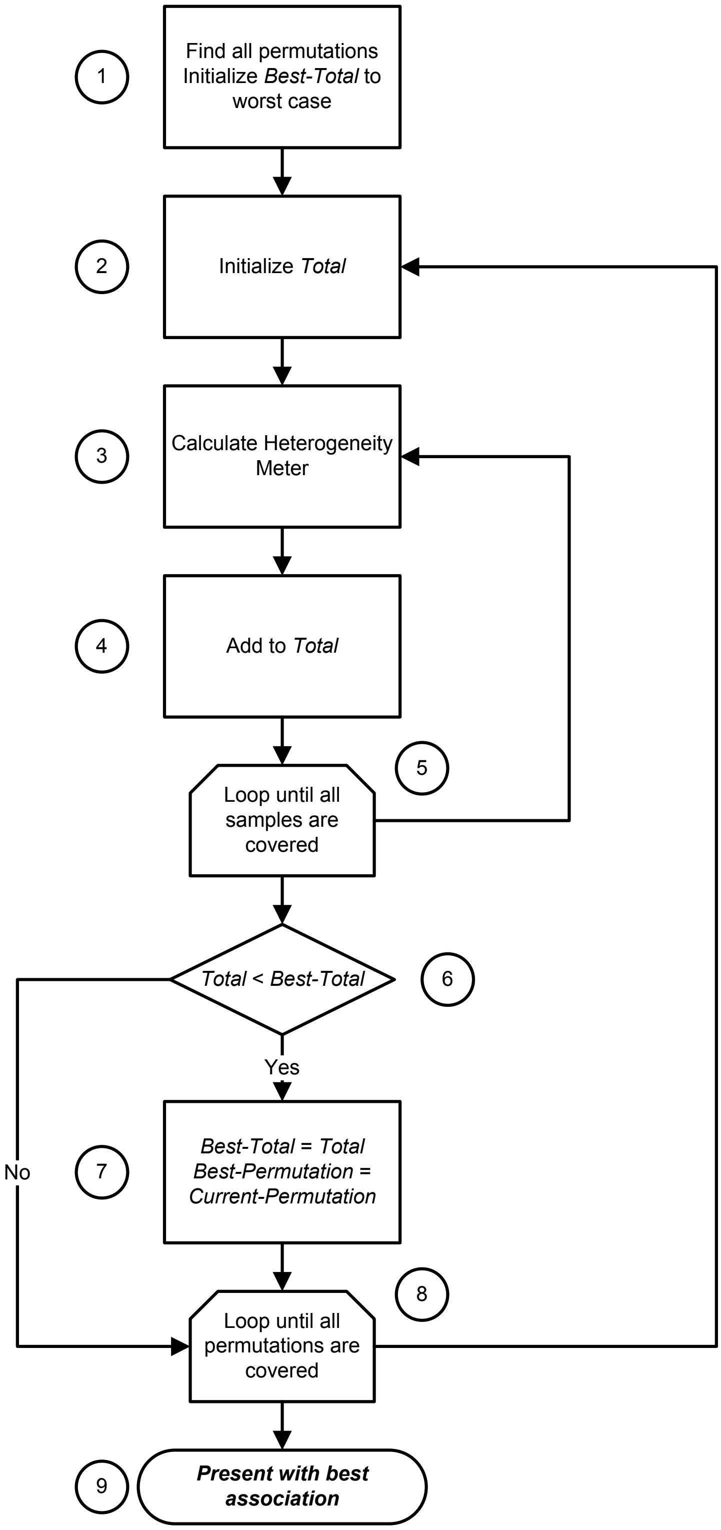

1
Find all permutations
Initialize *Best-Total* to worst case
2
Initialize *Total*
3
Calculate Heterogeneity Meter
4
Add to *Total*
5
Loop until all samples are covered
Total < *Best-Total*
6
Yes
No
7
Best-Total = *Total*
Best-Permutation = *Current-Permutation*
8
Loop until all permutations are covered
9
Present with best association

Figure 6: Brute Force Cluster Association Flow Chart

This complexity is calculated using the clustering of the first algorithm as a pivot to which all the other algorithms' clusters need to be associated. For each algorithm all possible cluster permutations are testes performing $C!^{(A-1)}$ operations, this is multiplied by the number of data components (rows) for which the quality meter needs to be calculated.

6.1.2. Accelerated Association Estimate

The brute force method is good to deliver the best results in the case when there is a small number of algorithms or clusters, but when the number of algorithms and clusters increase the calculation complexity rises significantly and may result impractical calculation times. For this case an accelerated cluster association estimate algorithm was developed. This algorithm is based on a local search for the best association as depicted in Figure 7:

Step 1 - Select a single arbitrary association.

Step 2 - Calculate the Quality meter.

Step 3 - If a better Quality meter is reached than start all over again using the current association as the initial association.

Step 4 - Perform Steps 2 - 3 on all single swaps from the initial association for a certain algorithm.

Step 5 - Perform Steps 2 – 4 on all algorithms.

Step 6 - After all cases with a single swap from the initial association are covered, the user is presented with the estimate for the best association.

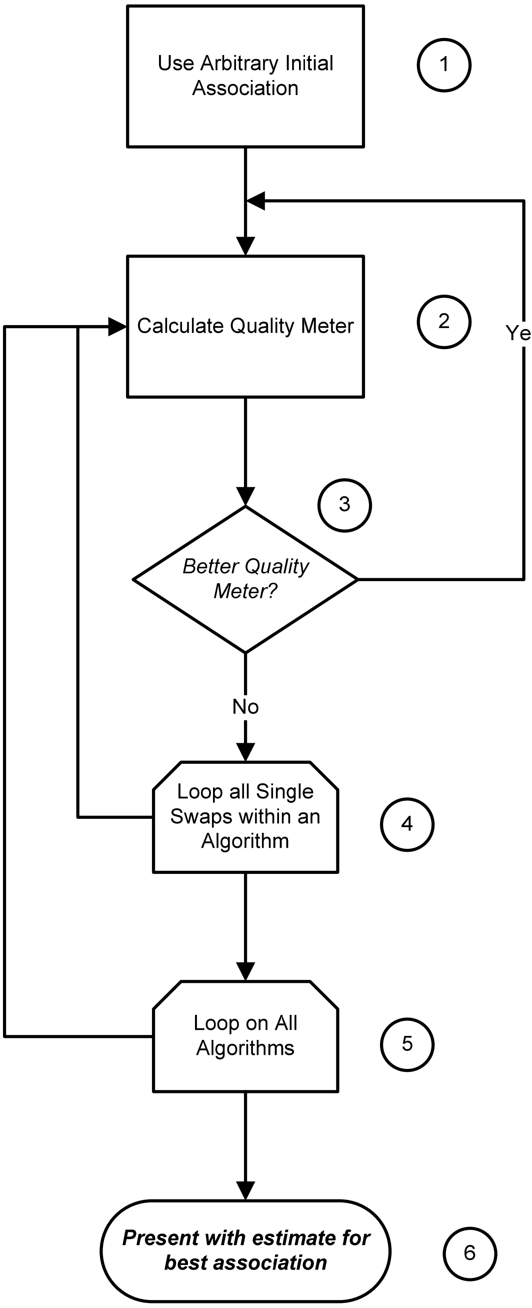

Use Arbitrary Initial Association
1
Calculate Quality Meter
2
Yes
3
Better Quality Meter?
No
Loop all Single Swaps within an Algorithm
4
Loop on All Algorithms
5
Present with estimate for best association
6

Figure 7: Accelerated Cluster Association Algorithm Flow Chart

The complexity of the accelerated association is:

$$O(D \cdot Q \cdot C^2 \cdot A \cdot \log(C!^{(A-1)}))$$

Where:

- D - is the size of the dataset
- Q - is the complexity of calculating the Heterogeneity quality meter
- C - is the number of clusters
- A - is the number of algorithms "voting" for the clustering

As in the case of brute force association the Quality meter *Q* is calculated for all data components *D*. Going over all the single swaps from an initial association requires C^2 swaps for each algorithm, over all the algorithms *A,* yielding $C^2 \cdot A$ calculations. Since each such case approximately divides the remaining associations to half, i.e. those with better Quality meters, and those with Quality meters not better than the best one so far, we effectively perform the calculations only on a log of all the possible associations $C!^{(A-1)}$.

The only case where this estimate will not reach the best association is if it converges into a local maximum of the quality meter.

A further improvement of this effective estimation for the cluster association is to perform the process multiple times or until no improvements are reached after a predetermined number of multiple activations of the process, starting each time from a new random arbitrary association. Since there are only a few local maximums if at all, performing the operation multiple times will improve the probability to start from a point converging into the best maximum.

6.2. Model Evaluation

To evaluate the methodology we performed the following steps:

6.2.1. Using a known dataset with a known classification

We implemented the methodology on a well known dataset, the Fisher Iris dataset (Fisher, 1936), commonly used in classification problems with known classification. This allowed us to confirm our methodology analysis results.

6.2.2. Using a dataset with unknown classification guided by an researcher

We worked with experts in the field of management to analyze datasets that characterize manager's behavior in different cultures and professions. We implemented it in two cases:

1. A research analyzing a given dataset under the supervision of the experts. The experts are in a position to evaluate the results. The results were consistent with the experts' evaluations.

2. A follow-up research on a dataset already analyzed. Again the expert researchers were in a position to analyze the methodology implementation outcome and the additional value resulting from applying it. The results were consistent with the researchers' findings.

6.2.3. Implementing the methodology to analyze a new case of a commonly researched problem

After receiving the positive feedback from the previous steps we implemented the methodology on well researched problems such as the case of car pricing using the dataset of cars sold in the US in 1993 (PACE New Car & Truck 1993 Buying Guide, 1993) analyzing the dataset and comparing the results to the known facts. Again the results were consistent with previous researches and were able to demonstrate the value of the developed methodology.

7. Research Structure and Publications

This research is based on papers presenting the suggested methodology. Its implementation in various fields related mainly, but not limited to, business administration research areas, and the development of tools to perform the required implementation. The research flow is depicted in Figure 8.

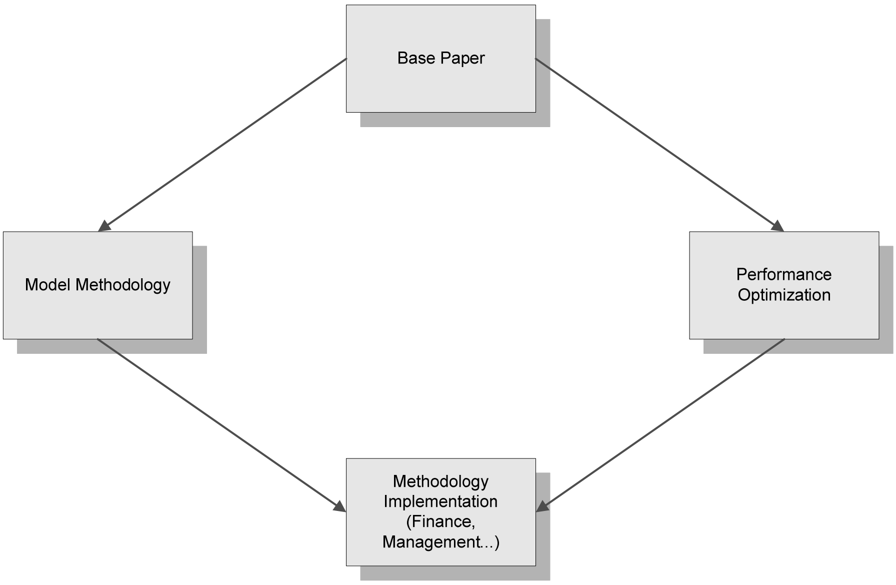


Figure 8: Research Structure Flow

The first paper sets the foundations of the suggested methodology while the following papers provide the tools to implement the methodology and shows its implementation on known problems. The research was done with the assistance of tools developed for the purpose of implementing the methodology in a practical manner by applying algorithms developed for that purpose that allowed for the practical implementation of the methodology on the selected problems.

8. Summary and Discussion

This work presents a solid methodology for visual presentation of multi-classifications. The methodology defines a detailed process in which the researcher performs cluster analysis using multiple algorithms and tools and harnesses the outcome to a clear two dimensional presentation that emphasizes the interesting characteristics of the researched dataset including:

- Which clustering algorithms are suitable for different tasks
- What is a good number of categories to classify the dataset into
- Which categories can be easily identified
- Which samples tend to be wrongly classified
- Which samples are difficult to classify

Both the methodology and its implementation are backed up by published papers.

The methodology is implemented in a prototype that allows the researcher to get the required information easily. Care was taken to the performance when implying the methodology and an algorithm to accelerate the results to allow practical use was developed and implemented.

The research opens the door to future research both by implementing the methodology in additional cases where clustering is required, and its visual presentation is needed to analyze and present the research outcome, and in the area of optimizing and utilizing the association techniques used in the methodology to perform the required visualization of the clustering results.

Essay 1

“Decision-making Method Using a Visual Approach, for Cluster Analysis Problems; Indicative Classification Algorithms and Grouping Scope”

Article

Decision-making method using a visual approach for cluster analysis problems; indicative classification algorithms and grouping scope

Ran M. Bittmann and Roy M. Gelbard

Information System Program, Graduate School of Business Administration, Bar-Ilan University, Ramat-Gan 52900, Israel
E-mail: gelbardr@mail.biu.ac.il; ran@bittmann.homedns.org

Abstract: *Currently, classifying samples into a fixed number of clusters (i.e. supervised cluster analysis) as well as unsupervised cluster analysis are limited in their ability to support 'cross-algorithms' analysis. It is well known that each cluster analysis algorithm yields different results (i.e. a different classification); even running the same algorithm with two different similarity measures commonly yields different results. Researchers usually choose the preferred algorithm and similarity measure according to analysis objectives and data set features, but they have neither a formal method nor tool that supports comparisons and evaluations of the different classifications that result from the diverse algorithms. Current research development and prototype decisions support a methodology based upon formal quantitative measures and a visual approach, enabling presentation, comparison and evaluation of multiple classification suggestions resulting from diverse algorithms. This methodology and tool were used in two basic scenarios: (I) a classification problem in which a 'true result' is known, using the Fisher iris data set; (II) a classification problem in which there is no 'true result' to compare with. In this case, we used a small data set from a user profile study (a study that tries to relate users to a set of stereotypes based on sociological aspects and interests). In each scenario, ten diverse algorithms were executed. The suggested methodology and decision support system produced a cross-algorithms presentation; all ten resultant classifications are presented together in a 'Tetris-like' format. Each column represents a specific classification algorithm, each line represents a specific sample, and formal quantitative measures analyse the 'Tetris blocks', arranging them according to their best structures, i.e. best classification.*

Keywords: cluster analysis, visualization techniques, decision support system

1. Introduction

The problem of analysing data sets and classifying them into clusters based on known properties is a well-known problem with implementations in fields such as finance (e.g. fraud detection), computer science (e.g. image processing), marketing (e.g. market segmentation) and medicine (e.g. diagnostics), among others (Jain & Dubes, 1988; Jain *et al.*, 1999; Erlich *et al.*, 2002; Sharan & Shamir, 2002; Clifford & Stevenson, 1975). Cluster analysis research studies evaluate different algorithms by performing them on known data sets with known true results and comparing their output, and the algorithms' accuracy, to the true classification. The commercial products running these algorithms neither show the resulting clusters of multiple methods nor give the researcher tools with which to analyse and compare the outcomes of the different tools.

Within this context, the current study aims to provide

- a visual presentation of multiple classification suggestions, resulting from diverse algorithms;
- a comparison of the different results;
- a comparison of the results when different numbers of clusters are evaluated;
- an evaluation of the different results not only when the true classification is known but also when the true classification is unknown.

Studies that compare different algorithms (Erlich *et al.*, 2002; Sharan & Shamir, 2002) find it difficult to give an exclusive contingency approach as to which method is preferable, since such a contingency approach needs to cover all problem types, data types and result types. This is complicated to define mathematically. Within this context, the current study is among the first to

- suggest a methodology and provide tools to recommend a preferred method for a given problem;
- suggest a methodology and provide tools to recommend a preferred number of clusters for a given problem;
- provide a visual approach to accompany the mathematical processing for a presentation of the full spectrum of results to acquaint the researcher with the classification tools' possible outcomes;
- provide an immediate indication of the areas of contention between the different algorithms;
- effect analysis by using different numbers of clusters for the classification problem.

The conventional approach is to apply an algorithm from a set of algorithms tuned by the algorithm parameters based on the data set properties' criteria and the researcher's expertise. This approach, however, limits the result to the effectiveness of the chosen algorithm and leaves the researcher totally in the dark when the classification of the data set is unknown. It does not show us which samples are hard to classify or how effective the chosen properties are for the desired classification.

Furthermore, visualization of the data set and its classification is virtually impossible when more than three properties are used, since displaying the data set in this case will require giving up on some of the properties in order to display the data set, or using some other method to display the data set's distribution over four dimensions or more. This makes it very difficult to relate to the data set samples and understand which of these samples are difficult to classify (in some cases, even when they are classified correctly) and which samples and clusters stand out clearly (Shultz *et al.*, 1994; De-Oliveira & Levkowitz, 2003; Boudjeloud & Poulet, 2005).

Even when the researcher uses multiple algorithms in order to classify the data set, there are no tools that allow him/her to use the outcome of the algorithms' application. In addition, the researcher has no tools with which to analyse the difference in the results.

This study suggests a methodology and provides measures that provide the researcher with tools to combine the power of multiple algorithms, compare their results and present them in a clear visual manner. The result is the foundation for a decision support system that can be used to analyse data sets with both known and unknown classifications.

The current research shows the implementation of the suggested methodology in two cases:

- the Fisher iris data set, where the true classification is known (Fisher, 1936);
- the user profiles data set where the true classification is unknown (Shapira *et al.*, 1999).

The rest of the paper is outlined as follows. Section 2 presents the problem's theoretical background, Section 3 presents the research objectives and Section 4 presents the suggested methodology and the measures used for its application. The research environment is presented in Section 5 and Section 6 shows the results when applying the methodology to the two data sets mentioned above. Section 7 summarizes the research and analyses the results.

2. Theoretical background

2.1. Cluster analysis – algorithms

In order to classify a data set of samples according to a given set of properties, a researcher uses algorithms that process the properties of the data set samples and associate them with suggested clusters. The association is performed by calculating a likelihood measure that indicates the likelihood of a sample being associated with a certain cluster. Below is a short description of the algorithms that were used in this study.

2.1.1. Two-step algorithm This algorithm is used for large data sets and is applicable to both continuous and categorical properties. It is based, as its name implies, on two passes on the data set. The first pass divides the data set into a coarse set of sub-clusters, while the second pass groups the sub-clusters into the desired number of clusters. This algorithm is dependent on the order of the samples and may produce different results based on the initial order of the samples. The desired number of clusters can be determined automatically, or it can be a predetermined fixed number of clusters. We used the fixed number of clusters option in our analysis so that we could use this algorithm in conjunction with the other algorithms chosen for the study.

2.1.2. k-means This algorithm is used for large data sets and is applicable to both continuous and categorical properties. It requires that the number of clusters used to classify the data set is predetermined. It is based on determining arbitrary centres for the desired clusters, associating the samples with the clusters by using a predetermined distance measurement, iteratively changing the centre of the clusters and then re-associating the samples. The length of the process is very much dependent on the initial setting of the clusters' centres and can be improved if knowledge exists regarding the whereabouts of these clusters' centres.

2.1.3. Hierarchical methods This is a set of algorithms that work in a similar manner. These algorithms take the data set properties that need to be clustered and start initially by classifying the data set so that each sample represents a cluster. Next, it merges the clusters in steps. Each step merges two clusters into a single cluster until there is only one cluster (the data set) remaining. The algorithms differ in the way in which distance is measured between clusters, mainly by using two parameters: the distance or likelihood measure, e.g. Euclidean, dice etc., and the cluster method, e.g. between group linkage, nearest neighbour etc.

In this study, we used the following well-known hierarchical methods to classify the data sets.

- *Average linkage (within groups)* – This method calculates the distance between two clusters by applying the likelihood measure to all the samples in the two clusters. The clusters with the best average likelihood measure are then united.
- *Average linkage (between groups)* – This method calculates the distance between two clusters by applying the likelihood measure to all the samples of one cluster and then comparing it with all the samples of the other cluster. Once again, the two clusters with the best likelihood measure are then united.
- *Single linkage (nearest neighbour)* – This method, as in the average linkage (between groups) method, calculates the distance between two clusters by applying the likelihood measure to all the samples of one cluster and then comparing it with all the samples of the other cluster. The two clusters with the best likelihood measure, from a pair of samples, are united.
- *Complete linkage (furthest neighbour)* – This method, like the previous methods, calculates the distance between two clusters by applying the likelihood measure to all the samples of one cluster and then comparing it with all the samples of another cluster. For each pair of clusters the pair with the worst likelihood measure is taken. The two clusters with the best likelihood measure of those pairs are then united.

- *Centroid* – This method calculates the centroid of each cluster by calculating the mean average of all the properties for all the samples in each cluster. The likelihood measure is then applied to the means of the clusters and the clusters with the best likelihood measure between their centroids are united.
- *Median* – This method calculates the median of each cluster. The likelihood measure is applied to the medians of the clusters and the clusters with the best median likelihood are then united.
- *Ward* – This method calculates the centroid for each cluster and the square of the likelihood measure of each sample in the cluster and the centroid. The two clusters that when united have the smallest (negative) effect on the sum of likelihood measures are the clusters that need to be united.

2.1.4. Likelihood measure In all the hierarchical algorithms, we used the squared Euclidean distance measure as the likelihood measure. This measure calculates the distance between two samples as the square root of the sums of all the squared distances between the properties.

As seen above, the algorithms and the likelihood measures differ in their definition of the task, i.e. the clusters are different and the distance of a sample from a cluster is measured differently. This results in the fact that the data set classification differs without obvious dependence between the applied algorithms. The analysis becomes even more complicated if the true classification is unknown and the researcher has no means of identifying the core of the correct classification and the samples that are difficult to classify.

2.2. Cluster analysis visualization

Currently, data sets are analysed according to the following method.

- The researcher selects the best classification algorithm based on his/her experience and knowledge of the data set.
- The researcher tunes the chosen classification algorithm by determining parameters such as the likelihood measure.
- The researcher applies the algorithm to the data set using one of the following options:
  - ○ predetermination of a fixed number of clusters to divide the data set into (supervised classification);
  - ○ deciding on the preferred number of clusters to classify the data set into based on the algorithm output (unsupervised classification).

Currently, the results can be displayed in numeric tables and in some cases, when hierarchical classification algorithms are applied, the researcher may use a *dendrogram*, which is a tree-like graph that presents the merger of clusters from the initial case, where each sample is a different cluster, to the total merger, where the whole data set is one cluster. The vertical lines in a dendrogram represent clusters that are joined, while the horizontal lines represent the likelihood coefficient for the merger. The shorter the horizontal line, the more likely the clusters will merge. An example for the use of a dendrogram can be seen in Figure 1. In this dendrogram, we see that samples 2 and 18 should probably belong to the same cluster according to the average linkage algorithm used, while samples 2 and 32 are less likely to merge. Actually, according to the dendrogram, these samples belong to the same cluster only when the whole data set is merged into a single cluster.

Though the dendrogram provides the researcher with some sort of visual representation, it is limited to only a subset of the algorithms used. Furthermore, the information in the dendrogram relates only to the used algorithm and does not compare or utilize additional algorithms. The information itself serves as a visual aid to joining clusters, but does not provide a good indication of inconsistent samples in the sense that their position in the data set spectrum according to the chosen properties is misleading, and likely to be wrongly classified. This is the only visual aid available to the researcher and it is only applicable to some algorithms.

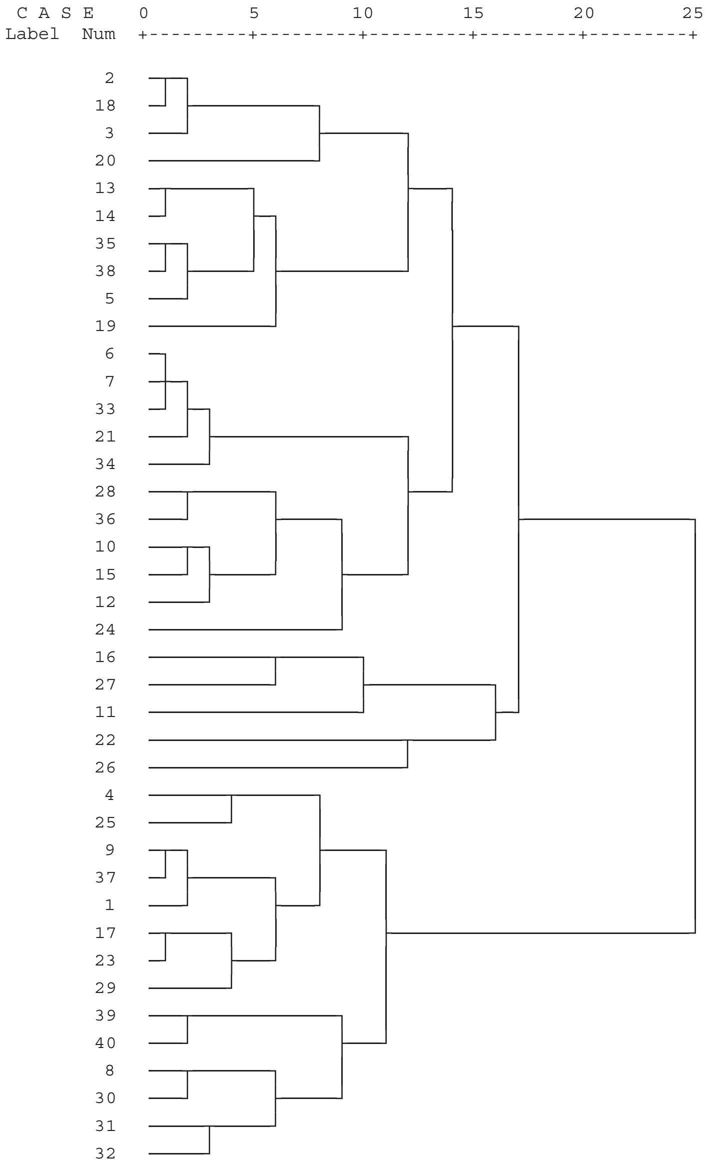


Figure 1: *Dendrogram of the user profiles data set using average linkage hierarchical classification.*

Furthermore, current methodologies are usually incapable of making comparisons between different algorithms and leave the decision-making, regarding which algorithm to choose, to the researcher. This leaves the researcher with very limited visual assistance and prohibits the researcher from having a full view of the relations between the samples and a comparison between the data set classifications based on the different available tools.

3. Research objectives

This study outlines a methodological process and indicates criteria for cluster analysis decision-making using a visual approach. It explains how to best utilize the known set of tools, mainly algorithms that allow us to build a decision support system. Using the suggested methodology, the decision support system is helpful when trying to decide on

- the preferred cluster analysis algorithm
- the preferred number of clusters to divide the data set into
- evaluating the classification properties
- identifying inconsistent samples.

As a visual aid, the process uses a clear visualization that encompasses the cluster analysis in a comprehensible, two-dimensional perspective. This view allows a comparison of the effect of the different methods on the data set, the effectiveness of the classification properties and the classification consistency of the individual samples.

The output includes a set of measures that quantifies the results and directs the researcher in making the decisions outlined above.

4. The decision methodology

4.1. The model concept

The suggested methodology presents the classification model from a clear, two-dimensional perspective, together with tools used for the analysis of this perspective.

4.1.1. Vote matrix The concept of the 'vote matrix' process recognizes that each algorithm represents a different view of the data set and its clusters, based on how the algorithm defines a cluster and how the algorithm measures the distance of a sample from a cluster. Therefore, each algorithm is given a 'vote' as to how it perceives the data set should be classified.

The suggested methodology is based on a vote matrix generated by the vote of each algorithm used in the process. Each row represents a sample and each column represents an algorithm and its vote for each sample about which cluster it should belong to, according to the algorithm's understanding of clusters and distances.

4.1.2. Heterogeneity meter This suggested methodology requires the association of clusters from different algorithms, since each algorithm divides the data set into different clusters. Although the number of clusters in each case remains the same for each algorithm, tools are required to associate the clusters of each algorithm, e.g. cluster number 2 according to algorithm A1 is the same as cluster number 3 according to algorithm A2. To achieve this correlation, we will calculate a measure called the 'heterogeneity meter' for each row, i.e. the collection of votes for a particular sample, and sum it up for all the samples.

The heterogeneity meter is calculated as follows:

$$H = \sum_{i=1}^{n} (N - M_i)^2$$

where H is the heterogeneity meter, N is the number of algorithms voting for the sample, M is the maximum number of similar votes according to a specific association received for one sample, i is the sample number and n is the total number of samples in the data set.

In order to find the best association, the heterogeneity meter needs to be minimized, i.e. the association that makes the votes for each sample as homogeneous as possible needs to be identified. The heterogeneity meter is then used to sort the voting matrix, giving the researcher a clear, two-dimensional perspective of the clusters and indicating how well each sample is associated with its designated cluster.

4.1.3. Properties evaluation Another outcome of this perspective is that it provides a clear comparison between the algorithms used to classify the samples, thereby allowing for easy identification of the effective algorithms.

When the real classification is known, an alternative measure for the cluster association is used for each algorithm, i.e. a column. Cluster association is carried out by comparing each sample association with the true association and maximizing the sum of correct associations for that algorithm. If there is a difference between the best association measured by comparing the association to the true classification and the best association measured using the heterogeneity meter, this implies the existence of a problem regarding the properties chosen to perform the cluster analysis.

4.1.4. Number of clusters decision When the data set's true classification is unknown, then there is also the issue of deciding how many clusters the data set should be classified into. The way to deal with this issue is by building the vote matrix for several suspected numbers of clusters and observing the result. The matrix view with the highest number of clusters that maintains a clear view of the classification should be chosen. Currently, the process of evaluating the classification quality with the different number of classes is done manually.

Future research should find a way to normalize the heterogeneity meter received by forcing the classification of multiple numbers of clusters to identify the best number of clusters.

4.2. Decision flow chart

The flow chart (Figure 2) describes the steps used to apply the suggested methodology.

Step 1 The researcher decides on the algorithms he/she will use for the data set analysis (in the case where the true classification is unknown, this step also includes making a decision about how many clusters the data set needs to be divided into).

Step 2 The researcher applies the chosen algorithms to the data set.

Step 3 The researcher builds the vote matrix based on the results of the chosen algorithms, when applied to the data set.

Step 4 The researcher calculates the heterogeneity meter for the vote matrix and associates the different classes of each algorithm to one another.

Step 5 If the true classification is known, the association of the clusters and the real classification are compared.

Step 6 The two associations are compared.

Step 7 If the two associations differ, then there is a problem with the properties used to perform the classification. This problem is indicated to the researcher.

Step 8 If the true classification is unknown, then the process of calculating a vote matrix for a different number of clusters is performed until all possible numbers of clusters decided upon in Step 1 have been covered.

Step 9 From all the generated vote matrices, the researcher chooses the vote matrix with the highest number of clusters that shows a clear classification.

Step 10 Looking at the vote matrix, the algorithm that is closest to the majority vote for each sample is chosen as the best algorithm.

Step 11 Samples with a high heterogeneity meter are indicated as being inconsistent samples, which are difficult to classify based on the given properties.

5. Research environment

5.1. Tools

In this study, we examined the tools available in the SPSS version 13.0 for Windows. We used three types of classification algorithms: two-step, *k*-means and hierarchical classification algorithms with different methods as are available in the SPSS software.

We also used Microsoft Excel 2003 to perform the analysis and build the vote matrix.

5.2. The data sets

The suggested methodology was applied to two data sets as described below. The data sets represent the two cases that cluster analysis tries to evaluate: data sets with known classification and data sets with unknown classification.

5.2.1. Data sets with known classification In this test, we used the Fisher iris data set. This is a widely used data set that includes measures for the sepal length, sepal width, petal length and petal width in millimetres for 50 samples of three species of irises: *Iris Setosa*, *Iris Versicolor* and *Iris Virginica*.

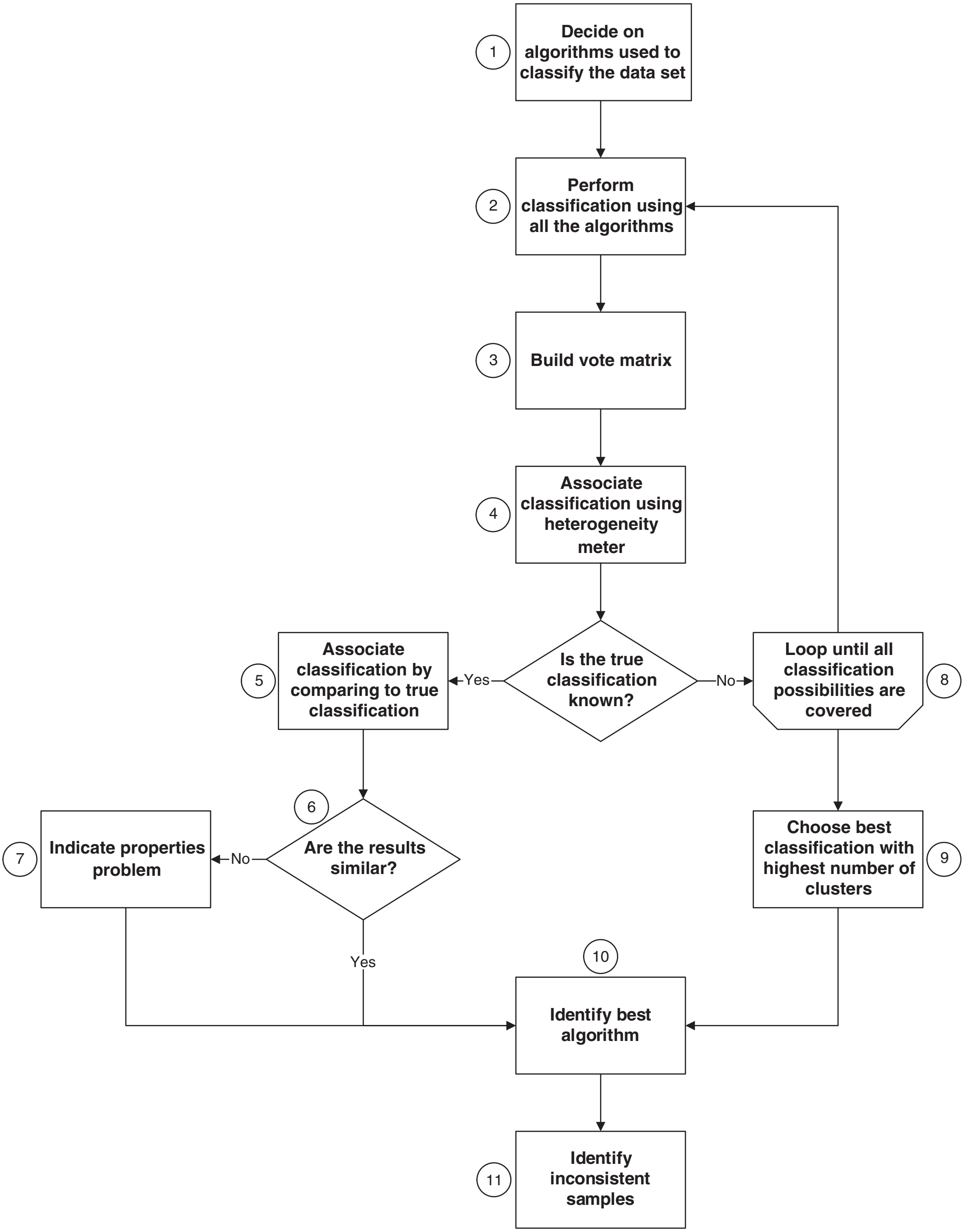

1
Decide on algorithms used to classify the data set
2
Perform classification using all the algorithms
3
Build vote matrix
4
Associate classification using heterogeneity meter
Is the true classification known?
Yes
No
5
Associate classification by comparing to true classification
8
Loop until all classification possibilities are covered
6
Are the results similar?
No
7
Indicate properties problem
Yes
9
Choose best classification with highest number of clusters
10
Identify best algorithm
11
Identify inconsistent samples

Figure 2: *The decision flow chart.*

5.2.2. Data sets with unknown classification We tested the methodology with a data set from a user profiles study that tries to relate users to a set of stereotypes based on sociological aspects and interests. The samples are defined using eight properties labelled A to H.

5.3. The prototype

In order to apply the suggested methodology and process, a prototype of the methodology was used. This prototype used the output from the SPSS software and imported it to an Excel worksheet. Using Excel macros and functions, we performed an automatic calculation of the heterogeneity meter and sorted the samples in the vote matrix according to it. The ordering of the rows in the final vote matrix, after being sorted according to their individual heterogeneity meter, was done manually to best reflect certain aspects such as classification of clusters or isolation of inconsistent examples.

6. Results

The process was applied to the two data sets mentioned above. The first data set is the Fisher iris data set, whose true classification is known. The second data set is the user profiles data set whose true classification is unknown and which was originally intended to be classified into four clusters.

6.1. The Fisher iris data set results

6.1.1. Vote matrix definitions For the Fisher iris data set, we chose to include the following algorithms in the vote matrix:

- two-step with log-likelihood as the distance measure – this algorithm is marked as M1;
- two-step with Euclidean as the distance measure – this algorithm is marked as M2;
- *k*-means – this algorithm is marked as M3.

The next seven classifications were performed using hierarchical classification with squared Euclidean as the distance measure:

- average linkage (within groups) – this algorithm is marked as M4;
- average linkage (between groups) – this algorithm is marked as M5;
- single linkage (nearest neighbour) – this algorithm is marked as M6;
- complete linkage (furthest neighbour) – this algorithm is marked as M7;
- centroid method – this algorithm is marked as M8;
- median method – this algorithm is marked as M9;
- ward method – this algorithm is marked as M10.

In addition, we added the following columns for reference:

- the sample number appears in the first column, marked as S;
- the known classification is marked as TR;
- the heterogeneity meter is marked as HM.

Greyscale color-coding of the different clusters was added for clarity.

6.1.2. Methodology application We know that the data set should be classified into three clusters. Therefore, we performed the test forcing the tools to merge the data set into three clusters. We performed these steps in applying the methodology:

1. building the raw vote matrix;
2. applying the measures:
   - ■ comparison to the true classification
   - ■ calculating the heterogeneity meter
3. associating the clusters with each other so that the resemblance to the true results was maximized and the heterogeneity meter was minimized;
4. reordering the vote matrix to give a clearer perspective of the data set and its classification according to the different algorithms.

6.1.3. Methodology implementation output The Fisher iris data set raw vote matrix (Figure 3) shows the greyscale-coded classification of the

S	TR	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	HM
1	1	3	3	1	1	1	1	1	1	1	1	4
2	3	1	1	2	2	2	2	2	2	2	2	4
3	2	2	1	3	2	2	2	2	3	2	3	16
4	3	1	1	2	2	2	2	2	2	2	2	4
5	3	2	1	3	2	2	2	2	3	2	3	16
6	1	3	3	1	1	1	1	1	1	1	1	4
7	3	1	1	2	2	2	2	2	2	2	2	4
8	2	2	1	3	2	2	2	2	3	2	3	16
9	2	2	1	3	2	2	2	2	3	2	3	16
10	1	3	3	1	1	1	1	1	1	1	1	4
11	2	2	1	3	2	2	2	2	3	2	3	16
12	2	2	1	3	2	2	2	2	3	2	3	16
13	3	1	1	2	2	2	2	2	2	2	2	4
14	2	2	1	3	2	3	2	3	3	3	3	16
15	3	1	1	2	2	2	2	2	2	2	2	4
16	3	2	1	3	2	2	2	2	3	2	3	16
17	3	1	1	2	2	2	2	2	2	2	2	4
18	1	3	3	1	1	1	1	1	1	1	1	4
19	2	2	1	3	2	3	2	3	3	3	3	16
20	3	1	1	2	2	2	2	2	2	2	2	4
21	3	1	2	2	3	2	3	2	2	2	2	9
22	2	1	1	3	2	2	2	2	3	2	3	25
23	3	1	2	2	2	2	2	2	2	2	2	1
24	3	1	2	2	3	2	2	2	2	2	2	4
25	3	2	1	3	2	3	2	3	3	3	3	16
26	1	3	3	1	1	1	1	1	1	1	1	4
27	3	1	1	2	2	2	2	2	2	2	2	4
28	2	1	1	3	2	2	2	2	3	2	3	25
29	2	2	1	3	2	2	2	2	3	2	3	16
30	2	2	1	3	2	3	2	3	3	2	3	25
31	1	3	3	1	1	1	1	1	1	1	1	4
32	3	2	1	3	2	2	2	2	3	2	3	16
33	2	2	1	3	2	3	2	3	3	3	3	16
34	3	2	1	3	2	2	2	2	3	2	3	16
35	3	1	1	2	2	2	2	2	2	2	2	4
36	1	3	3	1	1	1	1	1	1	1	1	4
37	1	3	3	1	1	1	1	1	1	1	1	4
38	2	2	1	3	2	3	2	3	3	2	3	25
39	3	1	2	2	3	2	3	2	2	2	2	9
40	1	3	3	1	1	1	1	1	1	1	1	4
41	3	1	1	2	2	2	2	2	2	2	2	4
42	1	3	3	1	1	1	1	1	1	1	1	4
43	2	2	1	3	2	3	2	3	3	3	3	16
44	1	3	3	1	1	1	1	1	1	1	1	4
45	3	1	1	2	3	2	2	2	2	2	2	9
46	3	2	1	3	2	2	2	2	3	2	3	16
47	1	3	3	1	1	1	1	1	1	1	1	4
48	2	2	1	3	2	3	2	3	3	3	3	16
49	2	2	1	3	2	3	2	3	3	3	3	16
50	3	1	2	2	3	2	2	2	2	2	2	4
51	1	3	3	1	1	1	1	1	1	1	1	4
52	1	3	3	1	1	1	1	1	1	1	1	4
53	3	2	1	3	2	2	2	2	3	2	3	16
54	1	3	3	1	1	1	1	1	1	1	1	4
55	1	3	3	1	1	1	1	1	1	1	1	4
56	1	3	3	1	1	1	1	1	1	1	1	4
57	3	2	1	2	2	2	2	2	2	2	3	4
58	3	1	1	2	2	2	2	2	2	2	2	4
59	1	3	3	1	1	1	1	1	1	1	1	4
60	1	3	3	1	1	1	1	1	1	1	1	4
61	1	3	3	1	1	1	1	1	1	1	1	4
62	2	2	1	3	2	2	2	2	3	2	3	16
63	3	2	1	3	2	2	2	2	3	2	3	16
64	1	3	3	1	1	1	1	1	1	1	1	4
65	1	3	3	1	1	1	1	1	1	1	1	4
66	2	2	1	3	2	3	2	3	3	2	3	25
67	2	2	1	3	2	3	2	3	3	3	3	16
68	1	3	3	1	1	1	1	1	1	1	1	4
69	1	3	3	1	1	1	1	1	1	1	1	4
70	2	2	1	3	2	2	2	2	3	2	3	16
71	2	2	1	3	2	3	2	3	3	3	3	16
72	1	3	3	1	1	1	1	1	1	1	1	4
73	1	3	3	1	1	1	1	1	1	1	1	4
74	3	1	2	2	3	2	2	2	2	2	2	4
75	3	1	1	2	2	2	2	2	2	2	2	4
76	3	2	1	3	2	2	2	2	3	2	3	16
77	2	2	1	3	2	3	2	3	3	3	3	16
78	3	1	1	2	3	2	2	2	2	2	2	9
79	1	3	3	1	1	1	1	1	1	1	1	4
80	1	3	3	1	1	1	1	1	1	1	1	4
81	3	1	1	2	3	2	2	2	2	2	2	9
82	3	1	1	2	2	2	2	2	2	2	2	4
83	3	2	1	3	2	2	2	2	3	2	3	16
84	3	1	1	2	2	2	2	2	2	2	2	4
85	2	2	1	3	2	3	2	3	3	3	3	16
86	2	2	1	3	2	3	2	3	3	3	3	16
87	2	2	1	3	2	3	2	3	3	3	3	16
88	1	3	3	1	1	1	1	1	1	1	1	4
89	1	3	3	1	1	1	1	1	1	1	1	4
90	3	1	1	2	3	2	2	2	2	2	2	9
91	3	2	1	3	2	2	2	2	3	2	3	16
92	1	3	3	1	1	1	1	1	1	1	1	4
93	2	2	1	3	2	2	2	2	3	2	3	16
94	2	2	1	3	2	3	2	3	3	3	3	16
95	2	2	1	3	2	2	2	2	3	2	3	16
96	1	3	3	1	1	1	1	1	1	1	1	4
97	1	3	3	1	1	1	1	1	1	1	1	4
98	2	2	1	3	2	3	2	3	3	3	3	16
99	2	2	1	3	2	3	2	3	3	3	3	16
100	2	2	1	3	2	3	2	3	3	3	3	16
101	1	3	3	1	1	1	1	1	1	1	1	4
102	1	3	3	1	1	1	1	1	1	1	1	4
103	3	1	1	2	2	2	2	2	2	2	2	4
104	2	2	1	3	2	2	2	2	3	2	3	16
105	3	1	1	2	3	2	2	2	2	2	2	9
106	2	2	1	3	2	3	2	3	3	2	3	25
107	1	3	3	1	1	1	1	1	1	1	1	4
108	1	3	3	1	1	1	1	1	1	1	1	4
109	3	2	1	3	2	2	2	2	3	2	3	16
110	2	2	1	3	2	3	2	3	3	3	3	16
111	3	1	1	2	3	2	2	2	2	2	2	9
112	3	1	1	2	2	2	2	2	2	2	2	4
113	1	3	3	1	1	1	1	1	1	1	1	4
114	2	1	1	3	2	2	2	2	3	2	3	25
115	2	2	1	3	2	2	2	2	3	2	3	16
116	1	3	3	1	1	1	1	1	1	1	1	4
117	2	2	1	3	2	3	2	3	3	3	3	16
118	2	2	1	3	2	2	2	2	3	2	3	16
119	2	2	1	3	2	2	2	2	3	2	3	16
120	2	2	1	3	2	2	2	2	3	2	3	16
121	2	2	1	3	2	3	2	3	3	3	3	16
122	2	2	1	3	2	3	2	3	3	3	3	16
123	3	1	1	2	2	2	2	2	2	2	2	4
124	3	1	1	2	2	2	2	2	2	2	2	4
125	1	3	3	1	1	1	1	1	1	1	1	4
126	1	3	3	1	1	1	1	1	1	1	1	4
127	3	1	2	2	3	2	2	2	2	2	2	4
128	3	1	1	2	2	2	2	2	2	2	2	4
129	2	2	1	3	2	2	2	2	3	2	3	16
130	2	2	1	3	2	3	2	3	3	3	3	16
131	2	1	1	2	2	2	2	2	3	2	3	16
132	3	2	1	2	2	2	2	2	2	2	2	1
133	3	1	1	2	2	2	2	2	2	2	2	4
134	2	2	1	3	2	3	2	3	3	3	3	16
135	1	3	3	1	1	1	1	1	1	1	1	4
136	1	3	3	1	1	1	1	1	1	1	1	4
137	1	3	3	1	1	1	1	1	1	1	1	4
138	3	2	1	3	2	2	2	2	3	2	3	16
139	1	3	3	1	1	1	1	1	1	1	1	4
140	1	3	3	1	1	1	1	1	1	1	1	4
141	2	2	1	3	2	3	2	3	3	3	3	16
142	2	2	1	3	2	2	2	2	3	2	3	16
143	2	2	1	3	2	2	2	2	3	2	3	16
144	1	3	3	1	1	1	1	1	1	1	1	4
145	1	3	3	1	1	1	1	1	1	1	1	4
146	1	3	3	1	1	1	1	1	1	1	1	4
147	3	2	1	3	2	2	2	2	3	2	3	16
148	2	1	1	2	2	2	2	2	3	2	2	9
149	3	1	1	2	2	2	2	2	2	2	2	4
150	1	3	3	1	1	1	1	1	1	1	1	4
		45	0	66	112	74	102	74	64	78	66	**1458**

Figure 3: *Fisher iris data set raw vote matrix.*

first data set after classifying it into three clusters using ten classification methods. Unfortunately, the classification output did not always maintain the same cluster number. We can see, for example, that cluster number 3 in M1 (two-step with log-likelihood) refers to the same cluster as cluster number 1 in M3 (*k*-means). We can also see that there is very little likelihood between the results from algorithm M2 and the true classification.

Looking at the heterogeneity meter (in bold at the lower left), we see that it is very high, implying that there is a problem in associating the clusters with the different methods.

The *Fisher iris data set vote matrix after cluster association* (Figure 4) shows the results when implementing the next step in the suggested methodology on the results presented in Figure 3. In this step, we associate the different classifications so that the heterogeneity meter is minimized and the comparison with the true results is high. The fact that the heterogeneity meter has been minimized with the same association that maximizes the comparison to the true results indicates that the properties selection for the classification of the data set is good.

The *Fisher iris data set vote matrix after ordering* (Figure 5) shows the results of implementing the next step in the suggested methodology, ordering the samples according to their similarity. In this case, we sorted the samples according to the heterogeneity meter. The outcome was the samples ordered according to the consistency of the different algorithms regarding the clusters they belong to. Next we ordered the samples manually so that samples with similar classifications would be next to each other. The outcome gives a good visual representation regarding which sample was voted for which cluster. In some cases, it can be seen that clusters that really belong to a certain cluster are voted to a different one. A good example for such a case is sample number 25, which was wrongly voted by all the algorithms.

This is a two-dimensional presentation of the fact that this sample is located deep in the domain of a different cluster. Such a presentation would have been very difficult to present using conventional means since we have four properties. This would have required the performance of a four-dimensional perspective to present the sample distribution.

Another outcome of this presentation is that we can easily compare the different algorithms regarding this data set. It is obvious from viewing the vote matrix and the similarity to the true results that algorithms M3 (*k*-means), M8 (centroid) and M10 (ward) performed well with this data set, while algorithm M2 (two-step with Euclidean distance) and M6 (single linkage) performed poorly.

6.2. The user profiles data set results

6.2.1. Vote matrix definitions For the user profiles data set, we chose to include the following algorithms in the vote matrix:

- two-step with log-likelihood as the distance measure – this algorithm is marked as M1;
- *k*-means – this algorithm is marked as M3.

The next seven classifications were performed using hierarchical classification with squared Euclidean as the distance measure:

- average linkage (within groups) – this algorithm is marked as M4;
- average linkage (between groups) – this algorithm is marked as M5;
- single linkage (nearest neighbour) – this algorithm is marked as M6;
- complete linkage (furthest neighbour) – this algorithm is marked as M7;
- centroid method – this algorithm is marked as M8;
- median method – this algorithm is marked as M9;
- ward method – this algorithm is marked as M10.

In addition, we added the following columns for reference:

- the sample number appears in the first column, marked as S;
- the heterogeneity meter is marked as HM.

Greyscale color-coding of the different clusters was added for clarity.

S	TR	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	HM
1	1	3	3	1	1	1	1	1	1	1	1	0
2	3	1	1	2	2	2	2	2	2	2	2	9
3	2	2	1	3	2	2	2	2	3	2	3	9
4	3	1	1	2	2	2	2	2	2	2	2	9
5	3	2	1	3	2	2	2	2	3	2	3	9
6	1	3	3	1	1	1	1	1	1	1	1	0
7	3	1	1	2	2	2	2	2	2	2	2	9
8	2	2	1	3	2	2	2	2	3	2	3	9
9	2	2	1	3	2	2	2	2	3	2	3	9
10	1	3	3	1	1	1	1	1	1	1	1	0
11	2	2	1	3	2	2	2	2	3	2	3	9
12	2	2	1	3	2	2	2	2	3	2	3	9
13	3	1	1	2	2	2	2	2	2	2	2	9
14	2	2	1	3	2	3	2	3	3	3	3	0
15	3	1	1	2	2	2	2	2	2	2	2	9
16	3	2	1	3	2	2	2	2	3	2	3	9
17	3	1	1	2	2	2	2	2	2	2	2	9
18	1	3	3	1	1	1	1	1	1	1	1	0
19	2	2	1	3	2	3	2	3	3	3	3	0
20	3	1	1	2	2	2	2	2	2	2	2	9
21	3	1	2	2	3	2	3	2	2	2	2	0
22	2	1	1	3	2	2	2	2	3	2	3	16
23	3	1	2	2	2	2	2	2	2	2	2	4
24	3	1	2	2	3	2	2	2	2	2	2	1
25	3	2	1	3	2	3	2	3	3	3	3	0
26	1	3	3	1	1	1	1	1	1	1	1	0
27	3	1	1	2	2	2	2	2	2	2	2	9
28	2	1	1	3	2	2	2	2	3	2	3	16
29	2	2	1	3	2	2	2	2	3	2	3	9
30	2	2	1	3	2	3	2	3	3	2	3	1
31	1	3	3	1	1	1	1	1	1	1	1	0
32	3	2	1	3	2	2	2	2	3	2	3	9
33	2	2	1	3	2	3	2	3	3	3	3	0
34	3	2	1	3	2	2	2	2	3	2	3	9
35	3	1	1	2	2	2	2	2	2	2	2	9
36	1	3	3	1	1	1	1	1	1	1	1	0
37	1	3	3	1	1	1	1	1	1	1	1	0
38	2	2	1	3	2	3	2	3	3	2	3	1
39	3	1	2	2	3	2	3	2	2	2	2	0
40	1	3	3	1	1	1	1	1	1	1	1	0
41	3	1	1	2	2	2	2	2	2	2	2	9
42	1	3	3	1	1	1	1	1	1	1	1	0
43	2	2	1	3	2	3	2	3	3	3	3	0
44	1	3	3	1	1	1	1	1	1	1	1	0
45	3	1	1	2	3	2	2	2	2	2	2	4
46	3	2	1	3	2	2	2	2	3	2	3	9
47	1	3	3	1	1	1	1	1	1	1	1	0
48	2	2	1	3	2	3	2	3	3	3	3	0
49	2	2	1	3	2	3	2	3	3	3	3	0
50	3	1	2	2	3	2	2	2	2	2	2	1
51	1	3	3	1	1	1	1	1	1	1	1	0
52	1	3	3	1	1	1	1	1	1	1	1	0
53	3	2	1	3	2	2	2	2	3	2	3	9
54	1	3	3	1	1	1	1	1	1	1	1	0
55	1	3	3	1	1	1	1	1	1	1	1	0
56	1	3	3	1	1	1	1	1	1	1	1	0
57	3	2	1	2	2	2	2	2	2	2	3	25
58	3	1	1	2	2	2	2	2	2	2	2	9
59	1	3	3	1	1	1	1	1	1	1	1	0
60	1	3	3	1	1	1	1	1	1	1	1	0
61	1	3	3	1	1	1	1	1	1	1	1	0
62	2	2	1	3	2	2	2	2	3	2	3	9
63	3	2	1	3	2	2	2	2	3	2	3	9
64	1	3	3	1	1	1	1	1	1	1	1	0
65	1	3	3	1	1	1	1	1	1	1	1	0
66	2	2	1	3	2	3	2	3	3	2	3	1
67	2	2	1	3	2	3	2	3	3	3	3	0
68	1	3	3	1	1	1	1	1	1	1	1	0
69	1	3	3	1	1	1	1	1	1	1	1	0
70	2	2	1	3	2	2	2	2	3	2	3	9
71	2	2	1	3	2	3	2	3	3	3	3	0
72	1	3	3	1	1	1	1	1	1	1	1	0
73	1	3	3	1	1	1	1	1	1	1	1	0
74	3	1	2	2	3	2	2	2	2	2	2	1
75	3	1	1	2	2	2	2	2	2	2	2	9
76	3	2	1	3	2	2	2	2	3	2	3	9
77	2	2	1	3	2	3	2	3	3	3	3	0
78	3	1	1	2	3	2	2	2	2	2	2	4
79	1	3	3	1	1	1	1	1	1	1	1	0
80	1	3	3	1	1	1	1	1	1	1	1	0
81	3	1	1	2	3	2	2	2	2	2	2	4
82	3	1	1	2	2	2	2	2	2	2	2	9
83	3	2	1	3	2	2	2	2	3	2	3	9
84	3	1	1	2	2	2	2	2	2	2	2	9
85	2	2	1	3	2	3	2	3	3	3	3	0
86	2	2	1	3	2	3	2	3	3	3	3	0
87	2	2	1	3	2	3	2	3	3	3	3	0
88	1	3	3	1	1	1	1	1	1	1	1	0
89	1	3	3	1	1	1	1	1	1	1	1	0
90	3	1	1	2	3	2	2	2	2	2	2	4
91	3	2	1	3	2	2	2	2	3	2	3	9
92	1	3	3	1	1	1	1	1	1	1	1	0
93	2	2	1	3	2	2	2	2	3	2	3	9
94	2	2	1	3	2	3	2	3	3	3	3	0
95	2	2	1	3	2	2	2	2	3	2	3	9
96	1	3	3	1	1	1	1	1	1	1	1	0
97	1	3	3	1	1	1	1	1	1	1	1	0
98	2	2	1	3	2	3	2	3	3	3	3	0
99	2	2	1	3	2	3	2	3	3	3	3	0
100	2	2	1	3	2	3	2	3	3	3	3	0
101	1	3	3	1	1	1	1	1	1	1	1	0
102	1	3	3	1	1	1	1	1	1	1	1	0
103	3	1	1	2	2	2	2	2	2	2	2	9
104	2	2	1	3	2	2	2	2	3	2	3	9
105	3	1	1	2	3	2	2	2	2	2	2	4
106	2	2	1	3	2	3	2	3	3	2	3	1
107	1	3	3	1	1	1	1	1	1	1	1	0
108	1	3	3	1	1	1	1	1	1	1	1	0
109	3	2	1	3	2	2	2	2	3	2	3	9
110	2	2	1	3	2	3	2	3	3	3	3	0
111	3	1	1	2	3	2	2	2	2	2	2	4
112	3	1	1	2	2	2	2	2	2	2	2	9
113	1	3	3	1	1	1	1	1	1	1	1	0
114	2	1	1	3	2	2	2	2	3	2	3	16
115	2	2	1	3	2	2	2	2	3	2	3	9
116	1	3	3	1	1	1	1	1	1	1	1	0
117	2	2	1	3	2	3	2	3	3	3	3	0
118	2	2	1	3	2	2	2	2	3	2	3	9
119	2	2	1	3	2	2	2	2	3	2	3	9
120	2	2	1	3	2	2	2	2	3	2	3	9
121	2	2	1	3	2	3	2	3	3	3	3	0
122	2	2	1	3	2	3	2	3	3	3	3	0
123	3	1	1	2	2	2	2	2	2	2	2	9
124	3	1	1	2	2	2	2	2	2	2	2	9
125	1	3	3	1	1	1	1	1	1	1	1	0
126	1	3	3	1	1	1	1	1	1	1	1	0
127	3	1	2	2	3	2	2	2	2	2	2	1
128	3	1	1	2	2	2	2	2	2	2	2	9
129	2	2	1	3	2	2	2	2	3	2	3	9
130	2	2	1	3	2	3	2	3	3	3	3	0
131	2	1	1	2	2	2	2	2	3	2	3	25
132	3	2	1	2	2	2	2	2	2	2	2	16
133	3	1	1	2	2	2	2	2	2	2	2	9
134	2	2	1	3	2	3	2	3	3	3	3	0
135	1	3	3	1	1	1	1	1	1	1	1	0
136	1	3	3	1	1	1	1	1	1	1	1	0
137	1	3	3	1	1	1	1	1	1	1	1	0
138	3	2	1	3	2	2	2	2	3	2	3	9
139	1	3	3	1	1	1	1	1	1	1	1	0
140	1	3	3	1	1	1	1	1	1	1	1	0
141	2	2	1	3	2	3	2	3	3	3	3	0
142	2	2	1	3	2	2	2	2	3	2	3	9
143	2	2	1	3	2	2	2	2	3	2	3	9
144	1	3	3	1	1	1	1	1	1	1	1	0
145	1	3	3	1	1	1	1	1	1	1	1	0
146	1	3	3	1	1	1	1	1	1	1	1	0
147	3	2	1	3	2	2	2	2	3	2	3	9
148	2	1	1	2	2	2	2	2	3	2	2	16
149	3	1	1	2	2	2	2	2	2	2	2	9
150	1	3	3	1	1	1	1	1	1	1	1	0
		129	107	134	112	126	102	126	136	122	134	**634**

Figure 4: *Fisher iris data set vote matrix after cluster association.*