Invited paper submitted to IEEE
Transactions on Pattern Analysis
and Machine Intelligence,
October 1981.

Concept-based Clustering versus Numerical Taxonomy

Ryszard S. Michalski
Robert E. Stepp

Department of Computer Science
University of Illinois at Urbana-Champaign

ABSTRACT

A "concept-based" method of clustering is described that, unlike the conventional methods, not only clusters objects, but also produces descriptions of the obtained clusters. The descriptions are conjunctive concepts constructed from relations on selected object attributes. The presented method and 18 numerical taxonomy methods were all applied to two different clustering problems, one--a simple made up problem, and the second--a complex practical problem taken from the area of plant pathology (a reconstruction of selected plant disease categories). In both experiments the majority of the numerical taxonomy methods (14 out of 18) were not able to produce a clustering matching a typical human solution, while the conjunctive conceptual clustering method did produce such a solution and, in addition, produced cluster descriptions closely related to human descriptions.

I. Introduction

Clustering is usually viewed as a process of partitioning a collection of objects (measurements, observations, etc.) into groups of similar objects, according to some measure of similarity. Such an approach to clustering raises two fundamental problems: "what should be the nature of the similarity used to cluster objects?" and "should the similarity between objects be the only principle for constructing clusters?" These questions are discussed in this paper, and given an answer that is substantially different from the one given by traditional techniques.

In the area of cluster analysis and the closely related field of numerical taxonomy, the similarity between objects is typically assumed to be

some proximity measure in a multi-dimensional space, spanned by a fixed set of attributes characterizing the objects. The clusters are then defined as collections of elements (points) of the space whose intra-cluster proximities are high, and inter-cluster proximities are low. Research in cluster analysis has been therefore, primarily concerned with devising various object proximity measures and developing efficient algorithms utilizing these measures. Surveys of these measures can be found in Sokal and Sneath [16], Anderberg [1], and Diday and Simon [3].

Such an approach to clustering has several important limitations. First, clusters determined as groups of objects that are "close" in a fixed, a priori assumed attribute space may lack any simple conceptual interpretation. One reason for this is that the similarity measures employed consider all attributes with equal importance and make no distinction between truly relevant attributes and those which may be less relevant. There is no mechanism for selecting and evaluating attributes in the process of generating clusters. Neither is there any way to produce conceptual descriptions of the clusters. Conventional clustering methods simply leave the problem of cluster interpretation to the data analyst. This is an important limitation because data analysts are typically interested not only in determining clusters but also in formulating some meaningful (conceptual) descriptions of them.

Second, traditional techniques do not take into consideration methods humans employ in clustering objects. Observations of how people cluster objects indicate that they tend to select one or a few relevant attributes (out of potentially very many attributes), and cluster objects on the basis of these selected attributes. Each cluster contains objects that are similar in the sense that they score similarly for these "important" attributes.

Different clusters are expected to have different values of these attributes. The description of such clusters can be formally expressed as logical conjunctions of relations on these object attributes. In short, people tend to cluster objects into categories characterized by non-intersecting conjunctive concepts.

This brings us to the third limitation of traditional methods: they do not take into consideration any concepts or linguistic constructs people use in describing object collections. Such concepts may be, e.g., a characterization of a configuration of objects such as ring-shaped, U-shaped, T-formation, etc., or a description of a cluster as a group of objects that are: "small and red, with either no spot or a blue spot."

The idea of clustering objects into categories described by conjunctive concepts (conjunctive coneptual clustering) and a methodology for its computer implementation was introduced in Michalski [11] and described in greater detail in Michalski and Stepp [13,14].

The purpose of this paper is to characterize and compare the conjunctive conceptual clustering method with techniques of numerical taxonomy. The paper describes briefly the program CLUSTER/PAF for conjunctive conceptual clustering and presents results of applying CLUSTER/PAF and a numerical taxonomy program NUMTAX (implementing 18 different techniques) to two clustering problems.

II. Specification of a clustering problem

Before clustering methods can be applied, a data analyst must specify certain components of the clustering problem. These components are:

• The set of objects to be clustered and their attributes

Typically, objects to be clustered come from an experimental study of some phenomenon and are described by a specific set of attributes (variables). The initial encoding of the attributes is dictated by the measurement devices used, or by an established convention. The attributes may be measured on different scales, such as nominal, ordinal, interval, ratio, and absolute. In a simple case, one only distinguishes qualitative attributes (the nominal scale) from quantitative attributes (the remaining scales). The initial measurements are subject to problem-dependent transformations, which may reduce the precision of the quantitative attributes or replace subranges of their values by qualitative properties (e.g., a numerical size may be replaced by characterizations such as "small size," "medium size," or "large size"). The attributes available are not always all relevant to the clustering problem. In conventional approaches, the selection of relevant attributes is treated as a separate preliminary step. In the conjunctive conceptual clustering method the selection of attributes is performed simultaneously with the formation of clusters. The method selects those attributes which, from the viewpoint of certain criteria, allow it to "simply" characterize the individual clusters.

• The principle for grouping objects into clusters

Objects are grouped together by a clustering method according to some principle. The traditional principle for grouping objects into clusters utilizes some measure of object similarity, usually a reciprocal of a distance measure. In conceptual clustering [11], objects are assembled into clusters that represent single concepts (linguistic terms or simple logical functions

defined on such terms). In the specific method called conjunctive conceptual clustering, described here, objects are grouped into clusters that are characterized by logical products of relations on selected object attributes, i.e., conjunctive concepts. These relations may also include disjunction of properties, but only if the disjunction involves values of the same attribute (the so-called internal disjunction [10]). Such conjunctive concepts seem to reflect very well the typical human characterization of object classes. A definition and an illustration of the above conjunctive concepts is given in the next section.

• The type of cluster structure

In clustering, a given set E of objects is divided into subsets of objects. Let E_1, E_2,...,E_k be subsets of E, each being one of k clusters and let α_i denote a description of cluster E_i. In general, a description α_i is satisfied not only by all observed objects in E_i, but also by some unobserved objects. Based on the relationships among the clusters or among the cluster descriptions three different types of inter-cluster structures are commonly distinguished in the literature:

- The partition structure: a set of clusters whose union is the set E, and whose descriptions are all disjoint[1] (this implies that the clusters themselves are disjoint),
- The overlapping structure: a set of clusters that includes at least one intersecting pair. When some descriptions intersect but corresponding clusters do not (i.e., the intersection of the descriptions contains only unobserved events), the structure is called weakly overlapping, otherwise it is called strongly overlapping.

---

[1]Descriptions are disjoint if there are no events (observed or unobserved) that satisfy more than one description.

- The hierarchical structure: the first level clusters represent a partition structure of the whole set E; clusters at a lower level are elements of partition structures of the corresponding clusters one level higher,

III. Cluster representation scheme

A cluster representation simply and generally characterizes objects in each cluster. Conjunctive conceptual clustering uses two cluster representation schemes: a single representative object selected from a cluster, called the seed of the cluster; and a conjunctive statement that describes objects in the cluster. This conjunctive statement, called a logical complex, is an expression in the variable valued logic system VL_1 (Michalski [6,8]).

Suppose that $x_1, x_2, \ldots, x_n$ are variables selected to represent objects. We will assume that each variable, x_i, $i \in \{1,2,\ldots,n\}$, has an assigned domain, $D(x_i)$, that specifies all possible values the variable can take for any object in the collection to be clustered. The number of such values is given by d_i. The domains are assumed to be finite, and represented generally as $D(x_i) = \{0,1,2,\ldots,d_i-1\}$. We distinguish between nominal, linear, and structured variables, whose domains are unordered, linearly ordered, and tree-ordered sets, respectively. An example of a nominal variable is color or blood type; examples of linear variables are rank, size, or quantity of something; an example of a structured variable is shape, whose values may be triangle, rectangle, pentagon, ..., or polygon, which represents a more general concept (a parent node in the tree-structured domain). For simplicity, we assume here that variables are either nominal or linear.

The description space spanned by variables $x_1, x_2, \ldots, x_n$ is called the event space. Each point (event) in this space is a vector of specific values of variables $x_1, x_2, \ldots, x_n$. An event that is a description of some object in the collection to be clustered is called an observed event. Other events are called unobserved events.

A relational predicate (or selector) is defined as a form:

$$[x_i \ \# \ R_i]$$

where R_i (the reference) is a list of values from the domain of variable x_i,
$\#$ (the relation) is a relational operator = (equal) or $\neq$ (not equal).

A selector $[x_i = R_i]$ (or $[x_i \neq R_i]$) is satisfied if the value of x_i satisfies relation = ($\neq$) with any (all) values in the set R_i. In the set theoretic sense,

$[x_i = R_i]$ is equivalent to "value of $x_i \in \{R_i\}$" and
$[x_i \neq R_i]$ is equivalent to "value of $x_i \notin \{R_i\}$"

For example, the selector [length=small,medium] (value of length $\in$ {small,medium}) is satisfied whenever length has the value small or medium. The selector [length≠medium] is satisfied by any value of length except medium. The notation of a selector may be simplified by using the "or" operator for linking values of nominal variables on the list R_i, and using operators $<$ $>$ $\leq$ $\geq$ and the range operator ".." in selectors with linear variables, as illustrated by expression (i) below (the operator "or" denotes internal disjunction).

A logical product of selectors is called a logical complex (ℓ-complex). A set of objects that satisfy each selector in an ℓ-complex is called an s-complex (set-complex). Thus, an ℓ-complex can be viewed as a description of an s-complex. For example, the ℓ-complex:

(i) [height=tall][color=blue or red][length≥2][size≠medium][weight=2..5]

(the operation AND is implied by the concatenation of selectors) describes those objects that are tall, blue or red, with length ≥ 2, not medium size, and of weight 2 through 5. The set of all such objects constitutes the corresponding s-complex. The distinction between ℓ- and s- complexes is used to permit the application of logical or set-theoretic operators, respectively, whichever is more convenient. When this distinction is unimportant, the term complex will be used (without a prefix).

Not every collection of objects constitutes an s-complex, i.e., not every collection can be precisely described by an ℓ-complex. It is, however, possible to describe every collection of objects by an ℓ-complex, if the ℓ-complex is allowed to describe some additional objects (i.e., if it is permitted to be a generalized description of the collection). For example, events:

e_1: (blue, large, round)
e_2: (red, medium, round)

can be described by the complex:

[color=blue or red][size≥medium][shape=round]

This complex also covers the events:

e_3: (red, large, round)
e_4: (blue, medium, round)

which are distinct from e_1 and e_2. The number of such unobserved events contained in a complex is called the (absolute) sparseness of the complex.

IV. Conjunctive Conceptual Clustering

The general control structure of the conjunctive conceptual clustering algorithm can be viewed formally as a special case of the dyna clustering method (Diday et al. [2-5]). That method is a class of clusteri techniques

which find clusters iteratively by alternately applying a representation function and an allocation function (both explained below) while an evaluation criterion is monitored. The algorithm terminates when a specified number of cycles (consisting of one application of each function) are performed without yielding an improvement of the evaluation criterion. The distinguishing characteristic of conjunctive conceptual clustering is that the representation and allocation functions take specific unconventional forms. The following paragraphs present the conjunctive conceptual clustering algorithm in more detail.

The representation function: deriving descriptions from clusters

In the general formulation of dynamic clustering, the representation function derives a representation from given clusters. In our algorithm, this representation is a set of ℓ-complexes determined by a two step process:

1. Given k clusters, k representative events (seeds) $e_1, e_2, \ldots, e_k$ are determined. On the first iteration the seeds are selected randomly from the events to be clustered. On subsequent iterations, one seed is selected from the events in each cluster according to certain rules as described in section V.

2. Given seeds, a set of disjoint ℓ-complexes $\alpha_1, \alpha_2, \ldots, \alpha_k$, is derived such that
   (1) complex α_i covers (contains) seed e_i and possibly other events, but not other seeds,
   (2) the union of complexes covers the set to be clustered E, and
   (3) all k complexes considered as a group optimize the clustering evaluation criterion (called LEF, as defined below).

The allocation function: deriving clusters from descriptions

The allocation function performs an inverse of the representation function: given cluster descriptions, it determines events that satisfy each

description. Specifically, given k ℓ-complexes $\alpha_1, \alpha_2, \ldots, \alpha_k$ describing clusters, it forms a clustering $C^k = \{E_1, E_2, \ldots, E_k\}$ where the cluster E_i is the set of observed events satisfying α_i, $i=1,2,\ldots,k$.

The evaluation criterion

The evaluation criterion specifies the desired properties of a clustering (a collection of complexes representing individual clusters). The implemented method permits the user to maximize simultaneously one or more measures characterizing a clustering, such as:

- the fit between the clustering and the data,
- the total inter-cluster differences,
- the essential dimensionality (the number of attributes which singly distinguish between all clusters),
- the simplicity of cluster descriptions.

The fit between a clustering and the data is computed as the negative of the sum of sparsenesses of complexes defining individual clusters (i.e., the negative of the total number of unobserved events contained in the complexes). As the number of unobserved events in a complex decreases, the degree of overgeneralization of the complex decreases, which means that the fit between the observed events and the complex increases. Minimizing the sparseness is equivalent to maximizing the negative of the sparseness.

Inter-cluster difference is measured by the sum of the degrees of disjointness between every pair of complexes in the clustering. The degree of disjointness of a pair of complexes is the number of selectors in both complexes after removing pairs of selectors that involve the same variable and intersect. For example, the pair of complexes

- [color=red] [size=small or medium] [shape=circle]
- [color=blue] [size=medium or large]

has the degree of disjointness 3, because 2 of the 5 selectors intersect (intersecting selectors are underlined). Maximizing this criterion promotes clusters whose descriptions involve long sequences of different attribute values.

Essential (discriminative) dimensionality is defined as the number of variables that singly discriminate between all the clusters, i.e., which have different values in every cluster description (ℓ-complex). Single relations involving such variables are sufficient for distinguishing one cluster from the other clusters.

Simplicity of cluster descriptions is defined as the reciprocal of complexity, which is measured by counting the total number of selectors in all descriptions.

The above elementary criteria can be combined together into one general measure through the use of the Lexicographical Evaluation Functional with tolerances (LEF) [11]. The LEF is defined by a sequence of "criterion-tolerance" pairs (c_1,Δ_1), (c_2,Δ_2), ..., where c_i is a criterion (as described above) and Δ_i is a "tolerance threshold" ($\Delta \in [0..100\%]$). In the first step, all clusterings are evaluated on the first criterion, c_1, and those that score best or within the range defined by the threshold Δ_1 from the best are retained. Next, the retained clusterings are evaluated on criterion c_2 and trimmed similarly as above using Δ_2. This process continues until either the subset of retained clusterings is reduced to a singleton (the "best" clustering), or the sequence of criterion-tolerance pairs is exhausted. In

the latter case, the retained set contains clusterings that are considered to be equivalent with respect to the assumed evaluation criterion.

V. The Algorithm PAF

This section will present briefly the actual clustering algorithm PAF implemented as the inner part of the conceptual clustering program CLUSTER/PAF [13,14]. The outer program invokes the inner portion in a sequence of iterative steps to determine the best number of clusters and then recursively repeats the whole process to construct the next level of the cluster hierarchy [14]. The algorithm proceeds as follows:

1. From the given collection of events E, k events (the initial seeds) are selected. The seeds may be chosen randomly or according to some criterion. (After this initial step, seeds are always selected according to certain rules, see step 5).

2. For each seed, a bounded star G(e|F,m) is determined, where e is the seed, F is the set of remaining seeds and m is an integer. Such a star is defined as a set of not more than m maximally general ℓ-complexes that cover seed e and do not cover any events in F. Such complexes have maximum sparseness among complexes satisfying the required conditions. When the the total number of such complexes exceeds m, the algorithm selects the m best complexes as determined by the evaluation criterion (LEF).

3. Each complex in every star is reduced (made maximally specific) by removing from selector references all values without which the complex still covers the same observed events.

4. From each reduced bounded star, one complex is selected such that the obtained complexes in the resulting collection are mutually disjoint, together cover all the data points, and optimize the given evaluation criterion. The search strategy used to find such a collection of complexes is based on the A* search algorithm, developed in the field of artificial intelligence (Nilsson [15]). The method searches through a tree structure of various choices of complexes by investigating at every step the most promising combination of complexes obtained so far. Details of this method are described in [13,15].

5. A new seed is selected from each set of observed events covered by one complex in the collection and a new iteration of the algorithm begins from step 2. Two seed selection techniques are used. Seeds may be either central events, having the maximum number of properties in common with other observed events in the complex, or they may be border events, having the minimum number of properties shared. Central events are chosen as new seeds as long as the clusterings improve with each iteration. When the improvement ceases, border events are selected.

6. The obtained clustering is evaluated using a LEF. The LEF is defined by the user in terms of evalutation criteria selected from: fit, inter-cluster differences, essential dimensionality, and simplicity. If this is the first iteration, the clustering is stored, otherwise it is stored only if it is better than the previously stored one. In this way the stored clustering is always the best one among all solutions generated so far. The algorithm terminates when a specified number of iterations does not produce a better clustering.

Figure 1 shows the flow diagram summarizing these steps.

In the actual implementation, the program stores not only the best (locally optimal) k-clustering, but also a user-specified number of alternative k-clusterings closest to the best one, as defined by the LEF. Along with the k-clusterings, the program provides ℓ-complexes describing individual clusters and their scores on the evaluation criteria in the LEF. A detailed explanation of the complete algorithm is given in Michalski and Stepp [13,14]. A proof that every object collection can be partitioned into an arbitrary number of conjunctive concepts is in Michalski [11].

It should be noted that different choices of the evaluation criterion (LEF) will usually lead to different, alternative solutions to the problem. An approach taken here is that the user is permitted to apply his own judgement and knowledge of the problem in specifying the LEF and can arrive at a particular choice by experimentation. In such experiments, the user looks for clusterings that, from the viewpoint of a given problem, form the most meaningful or interesting subcategories of the events. The obtained clusterings are judged on the basis of produced cluster descriptions. Because

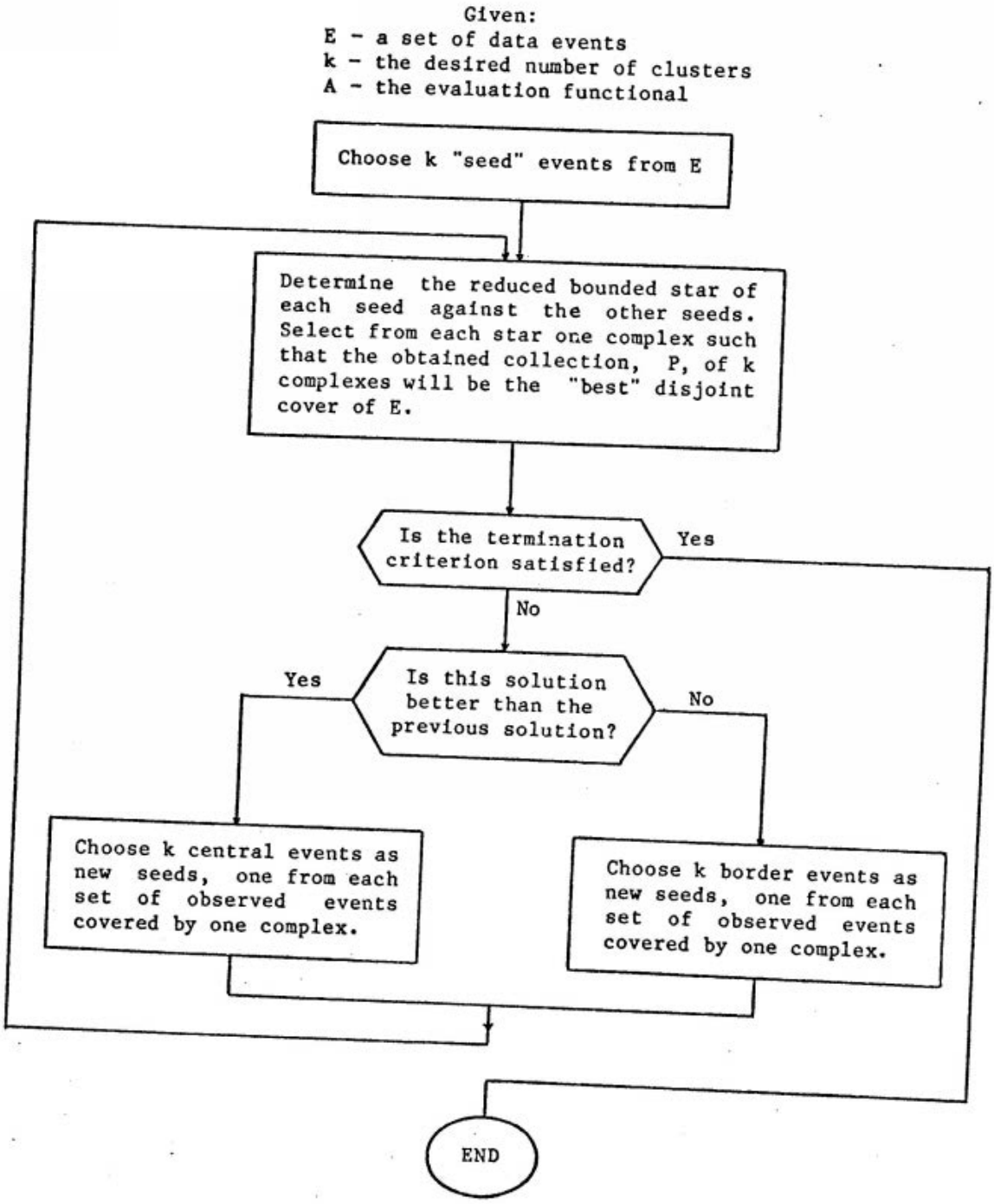


Figure 1. A flow diagram of the inner layer of algorithm PAF.

of the availability of these cluster descriptions, this method has a substantial edge over the numerical taxonomy methods, where a user has no means for comparing, at a conceptual level, clusterings obtained by applying different similarity measures.

VI. Example Problem I

The simple example problem described below is used to illustrate some of the differences between conjunctive conceptual clustering and methods of numerical taxonomy.

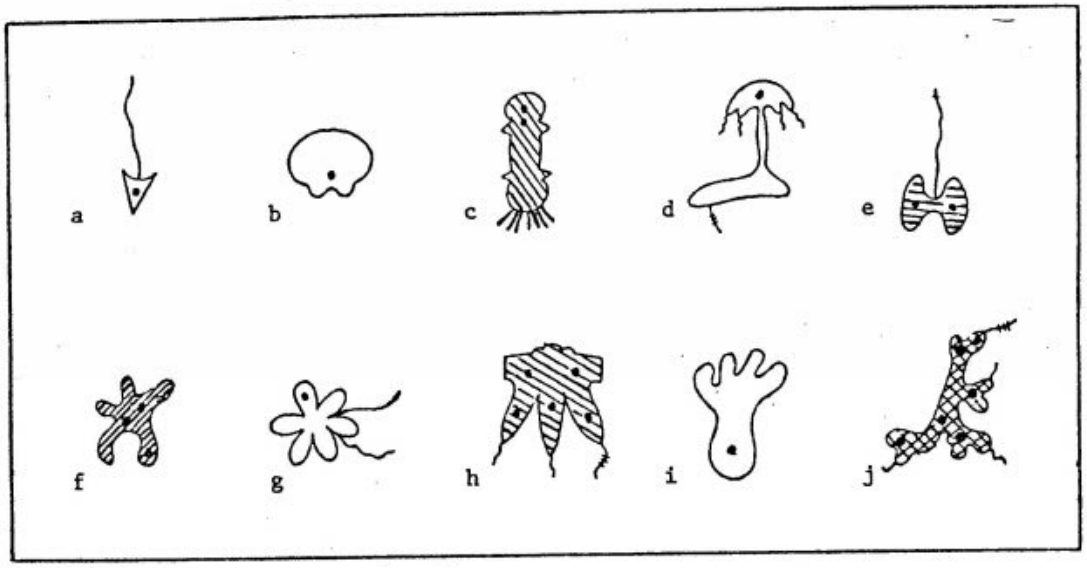


Figure 2. Microorganisms

The method-independent components of the problem (described in section II) are:

1. The set of objects to be clustered: "microorganisms" shown in Figure 2.

2,3. The variables selected for describing microorganisms and their domains:

Body parts
- 1 part
- 2 parts
- many parts

Body spots
- one spot
- many spots

Texture
- blank
- striped
- crosshatched

Tail type
- none
- single
- multiple

Figure 3 shows the descriptions of the microorganisms in terms of these variables.

4. The principle for grouping objects into clusters:

For numerical taxonomy:
18 different techniques are used, each being a combination made of one of three different similarity measures (product-moment correlation, simple matching coefficients, reciprocal Euclidean distance), one of three data transformations (none, normalizing variables into unit intervals, standardization), and one of two clustering schemes (average linkage, weighted average linkage).

For conjunctive conceptual clustering:
the technique described in section III.

5. The inter-class structure: the partition structure.

Two programs were applied to solve this problem:

1. NUMTAX, developed by Professor Selander at the University of Illinois, which implements the 18 techniques of numerical taxonomy mentioned above (described in Sokal and Sneath [16]),

2. CLUSTER/PAF, which implements conjunctive conceptual clustering.

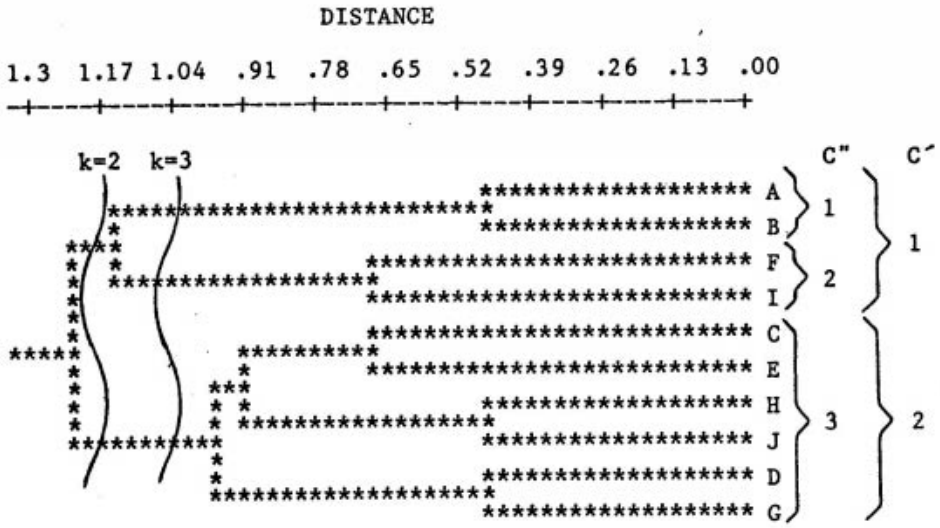
Results from NUMTAX for problem I

The numerical taxonomy program NUMTAX organizes the events into a hierarchy (a dendrogram) of clusters reflecting the numerical distances between consecutively larger clusters. The top level of the hierarchy

Micro-organism	Body parts	Body spots	Texture	Tail type
a	1	one	blank	single
b	1	one	blank	none
c	1	many	striped	multiple
d	2	one	blank	multiple
e	2	many	striped	single
f	many	many	striped	none
g	many	one	blank	multiple
h	many	many	striped	multiple
i	many	one	blank	none
j	many	many	crosshatched	multiple

Figure 3. Descriptions of microorganisms

represents the complete collection of events. The tips represent single events.



(The dendrogram was cut as indicated above to form the clusterings C´ and C" for k=2 and k=3, respectively)

Figure 4. Dendrogram produced by NUMTAX for microorganisms using average linkage and Euclidean distance on non-transformed data

Because dendrograms are constructed bottom-up, the entire dendrogram must always be generated. After this is done, the dendrogram may be cut apart at some level to produce clusters. In our experiment, 18 different dendrograms were obtained, one from each technique applied. One typical dendrogram is shown in Figure 4. Figure 5 shows two- and three-cluster solutions obtained from this dendrogram.

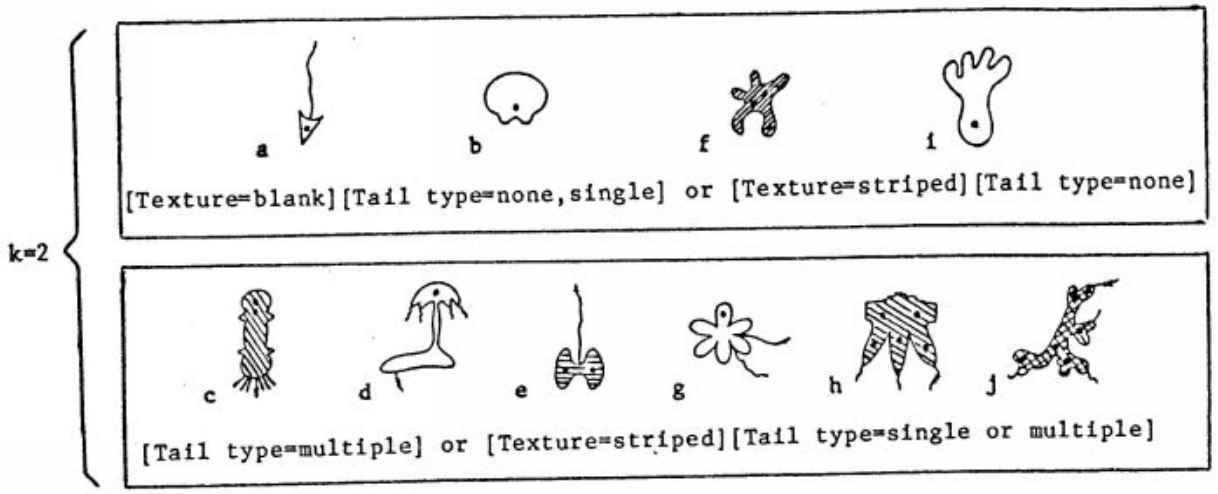
The clusters obtained from the dendrogram are not accompanied by any description. In order to determine descriptions of these clusters, they were presented to an inductive learning program, AQ11 (Michalski and Larson [9]), which formulated discriminant descriptions of each cluster (i.e., optimized descriptions sufficient to discriminate between the clusters) in the form of a logical disjunction of VL_1 complexes. The descriptions shown in Figure 5 are results from the AQ11 program.

Results from CLUSTER/PAF for problem I

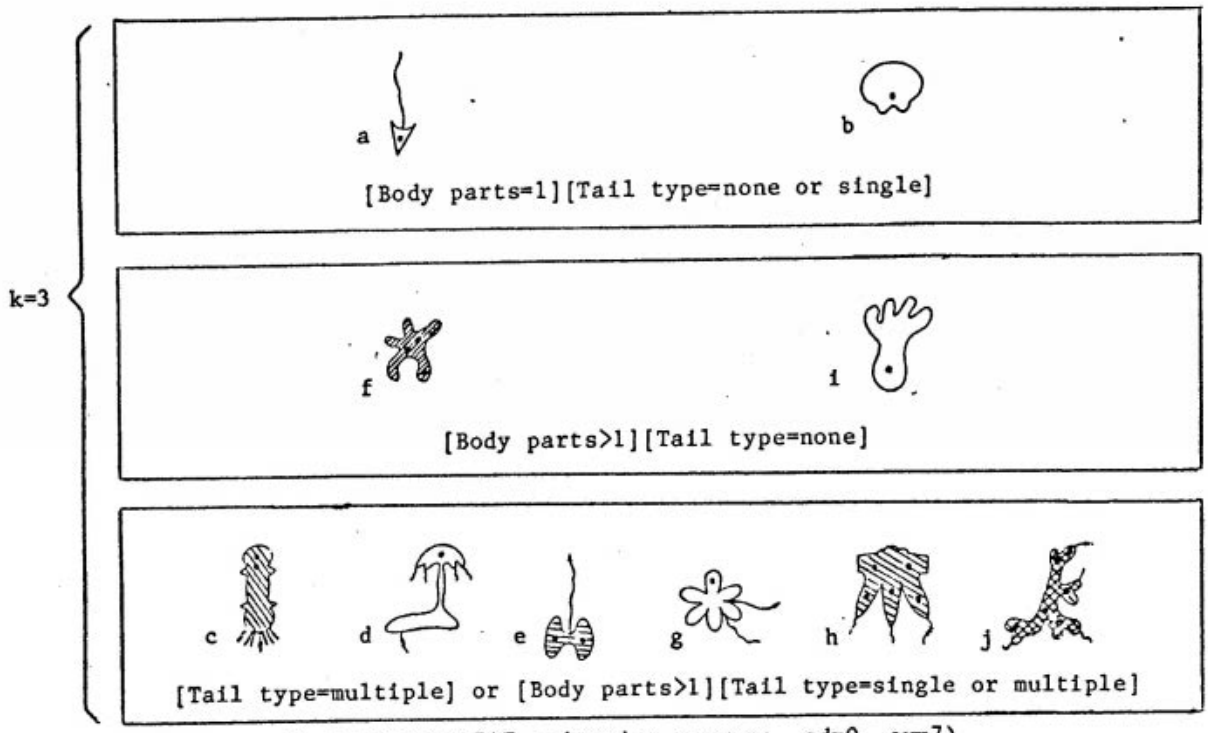
Program CLUSTER/PAF was run using the evaluation criterion: "maximize the essential dimensionality, then maximize the simplicity of cluster representations" (both with zero tolerance). The clusters and their descriptions obtained by CLUSTER/PAF with k=2 are shown in Figure 6a, and with k=3 in Figure 6b. The essential dimensionality and the complexity of each clustering are specified by parameters ed and cx, respectively.

Discussion of results

An experiment with human subjects solving this problem indicated that people catagorized objects using the objects' most noticeable properties. Most frequently given two-cluster solutions were:



(comparative PAF criterion scores: ed=0, cx=7)
a.



(comparative PAF criterion scores: ed=0, cx=7)
b.

Descriptions of clusters were produced by inductive program AQ11. To permit the comparison of the above descriptions to those obtained by PAF, the PAF criterion scores are shown above. These scores are:
ed - the essential dimensionality (the number of variables with different values in different clusters),
cx - the complexity (the total number of selectors).

Figure 5. Clusters obtained by NUMTAX using average linkage, Euclidean distance and raw (non-transformed) data, for k=2 and k=3

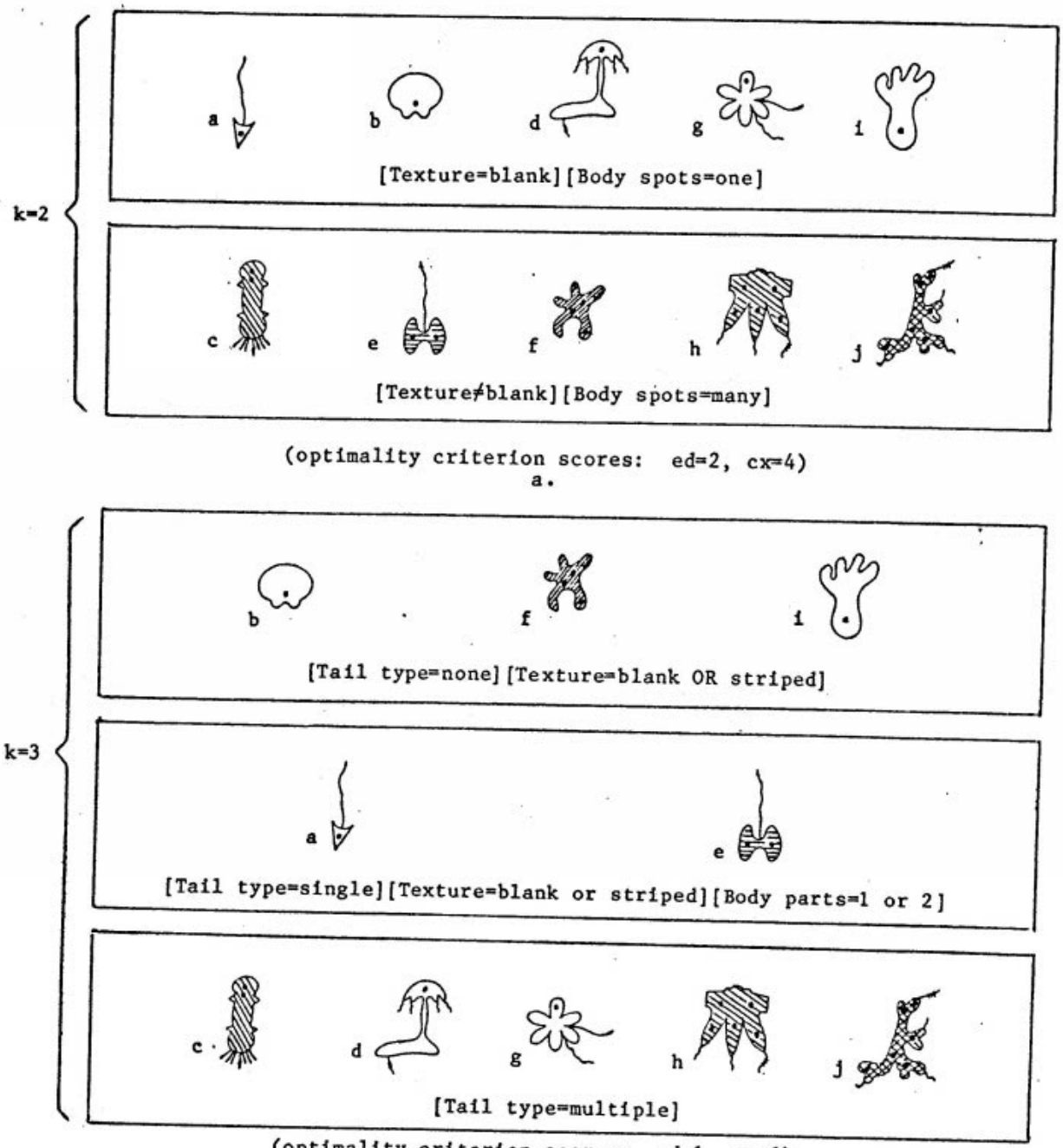


Figure 6. Clusters and cluster descriptions obtained by PAF for k=2 and k=3, using as the optimality criterion: "maximize the essential dimensionality, then minimize the complexity of cluster representations"

(1) [Texture=blank] vs. [Texture≠blank], and
(2) [Body spots=one] vs. [Body spots=many]

and the most frequently given three-cluster solution was:

(3) [Tail type=none] vs. [Tail type=single] vs. [Tail type=multiple]

When compared with the above solutions, the clusterings produced by NUMTAX seem rather arbitrary: the descriptions of single clusters (determined by program AQ11) involve disjunction in several cases, and are relatively complex. The descriptions produced by CLUSTER/PAF, however, correspond well to human solutions. The program found that clusterings (1) and (2) are in fact identical. These human descriptions can be obtained directly from the descriptions generated by CLUSTER/PAF by removing from PAF descriptions the conditions unnecessary for discriminating between the clusters. (In non-trivial problems this reduction is performed by applying inductive program AQ11). For k=3, CLUSTER/PAF found, in addition to the solution shown in Figure 6b, the alternative solution:

(1) [Body parts=1][Texture=blank or striped]
(2) [Body parts=2][Texture=blank or striped][Tail type=single or multiple]
(3) [Body parts=many][Tail type=none or multiple]

Of the 18 dendrograms generated by NUMTAX, only 4 (those involving either normalized or standardized data, Euclidean distance, and either average or weighted average linkage) yielded a partitioning of data that matched the human solution (and CLUSTER/PAF's solution, Figure 6a). Thus, in our experiments, numerical taxonomy methods produced clusters that in the majority of cases seemed to be rather inadequate from the viewpoint of human interpretation. This can be explained by noting that program NUMTAX is not equipped with any knowledge of human conjunctive concepts (or any other concepts) and therefore cannot knowingly produce clusters corresponding to such concepts.

VII. Example problem II

CLUSTER/PAF has been tried on several practical clustering problems. One such problem was to cluster data describing 47 cases of soybean disease each characterized by the 35 multi-valued variables shown in Figure 7.

• Time of occurrence	(7)	• Leaf mildew growth	(2)
• Plant stand	(3)	• Condition of stem	(3)
• Precipitation	(2)	• Presence of lodging	(4)
• Temperature	(4)	• Stem cankers	(3)
• Occurrence of hail	(3)	• Canker lesion color	(3)
• Number of years crop repeated	(2)	• Fruiting bodies on stem	(2)
• Damaged area	(3)	• External decay of stem	(3)
• Severity	(3)	• Mycelium on stem	(2)
• Seed treatment	(2)	• Internal discoloration of stem	(3)
• Seed germination	(2)	• Sclerotia internal or external	(2)
• Plant height	(4)	• Condition of fruit pods	(4)
• Condition of leaves	(2)	• Fruit spots	(3)
• Leaf spots	(2)	• Condition of seed	(3)
• Leaf spots margin	(2)	• Seed mold growth	(4)
• Leaf spot size	(5)	• Seed discoloration	(2)
• Shotholing/shreading	(2)	• Seed size	(2)
• Leaf malformation	(2)	• Seed shriveling	(2)
• Condition of roots	(3)		

(the numbers in parenthesis indicate the sizes of variable domains)

Figure 7. Multi-valued variables used to describe cases of soybean disease

These 47 cases were drawn from 4 populations--each population representing one soybean disease:

D1 - diaporthe stem canker
D2 - charcoal rot
D3 - rhizoctonia root rot
D4 - phytophthora rot

Therefore, ideally, a clustering method should partition these cases into four groups corresponding to the actual diseases. To test for this, we have applied CLUSTER/PAF and the above-mentioned 18 numerical taxonomy techniques to cluster these cases.

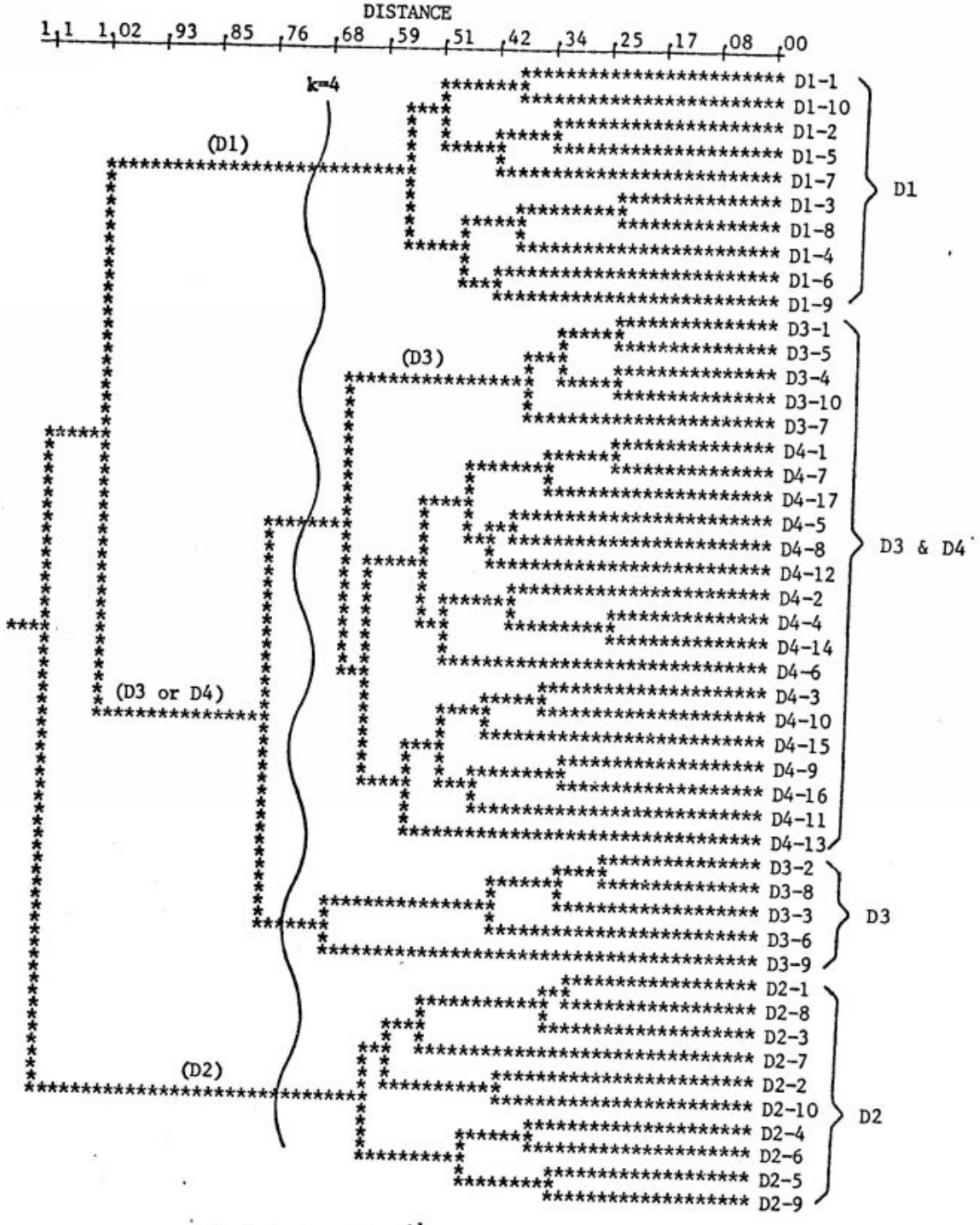
Results from NUMTAX for example II

Figure 8 shows a typical dendrogram produced by the program NUMTAX (18 such dendrograms were obtained, one from each technique). As we see, this denrogram separates correctly cases of diseases D1 and D2, however cases of diseases D3 and D4 are somewhat intermixed. For k=4 (Figure 8) the cluster denoted D3 & D4 contains cases of both diseases D3 and D4. Of the 18 dendrograms obtained, only 4 (those involving standardized data, product-moment correlation or simple matching scores, and average or weighted average linkage) precisely reconstructed the correct classification of the cases. The output from NUMTAX does not provide any description of the clusters formed.

Results from Conceptual Clustering for problem II

The program CLUSTER/PAF was applied to this problem with "maximizing the fit" as the evaluation criterion (LEF). CLUSTER/PAF partitioned the disease cases into four disease categories and described the clusters in terms of the characteristics (symptoms) of each disease, expressed in the form of a conjunctive statement. The produced disease categories corresponded exactly to actual soybean diseases and the descriptions produced by CLUSTER/PAF agreed well with the symptoms indicated by plant pathologists for these diseases (see Figure 9).

Figure 9 presents the complete ℓ-complex for one cluster (one disease category). The middle column contains the values for the 25 variables CLUSTER/PAF used to describe one cluster. The right-hand column of Figure 9 presents values of variables used by an expert plant pathologist to describe the same disease for diagnosis. The description of the disease determined by CLUSTER/PAF contains all the symptoms of the disease specified by the plant



(Di-j denotes the j^{th} case of disease Di)

Figure 8. Dendrogram of cases of soybean diseases D1, D2, D3, D4 using average linkage and Euclidean distance on non-transformed data

Variable	Value determined by PAF	Value determined by plant pathologist
Time of occurrence	July to October	August to September
Precipitation	above normal	normal or above normal
Temperature	normal	normal or above normal
No. yrs. crop repeated	several years	several years
Stem cankers	above second node	above second node
Canker lesion color	brown or n.a.	brown
Fruiting bodies	present	present
Condition of fruit pods	normal	normal
Plant stand	normal	not present in expert descriptions
Damaged ares	scattered areas or low areas	
Severity	potentially severe or severs	
Seed treatment	none or fungicide	
Plant height	abnormal	
Condition of leaves	abnormal	
Leaf spots	absent	
Shotholing/shreading	absent	
Leaf malformation	absent	
Leaf mildew growth	absent	
Condition of stem	abnormal	
Mycelium on stem	absent	
Condition of roots	normal	
External decay of stem	firm and dry	
Sclerotia int. or ext.	absent	
Int. discolor. of stem	none	
Condition of seed	normal	

Figure 9. The description for one cluster (the disease diaporthe stem canker) obtained by CLUSTER/PAF (variables having values in the left column) and as described by a plant pathologist (variables having values in the right column)

pathologist (the values of "Time of occurrence," "Precipitation," and "Canker lesion color" determined by CLUSTER/PAF are supersets of the values mentioned by the plant pathologist). The description produced by CLUSTER/PAF also involves many variables which the plant pathologist did not mention.

The logical statements produced by CLUSTER/PAF can be quite complex and for maximum comprehensibility, shorter cluster descriptions may be prefered. One way to create shorter descriptions is to apply an inductive program (e.g., AQ11 as described by Michalski and Chilausky [12]) to the obtained clustering.

Another way to handle this problem is to further analyze the output generated by CLUSTER/PAF and identify various categories of variables. The variables may be divided into those which take the same value for each cluster (common characteristics) and those which take different values for one or more clusters (discriminant characteristics). The variables which are discriminant characteristics can be further subdivided into those which take different values in all clusters (complete discriminant characteristics) and those which take different values in some (2 to k-1) clusters (partial discriminant characteristics). The complete discriminant characteristics ("key variables") are those variables which by themselves uniquely identify all clusters. The number of such key variables is the essential dimensionality of a clustering, defined in section III. In some problems complete discriminant variables are not present. In those cases, logical conjunctions of partial discriminant variables uniquely identify the clusters.

The clusters of soybean disease cases produced by CLUSTER/PAF do not have any complete discriminant characteristics. Figure 10 shows the common characteristics found by CLUSTER/PAF and a table of the values of the partial discriminant characteristics for each cluster. This table was derived from the descriptions produced by CLUSTER/PAF.

The measure of the total sparseness of the solution can be used as a heuristic to judge the best number of clusters to form. Data from the clustering of soybean disease cases for k=2 through k=6 are summarized in Figure 11. As k increases, the sparseness always decreases because data events are partitioned into smaller complexes which fit the data better. On the other hand, increasing k is undesireable as it raises the complexity of the clustering. A measure that reflects this trade-off is

Variable	Cluster 1 (diaporthe stem canker)	Cluster 2 (charcoal rot)	Cluster 3 (rhizoctonia root rot)	Cluster 4 (phytophthora rot)
Plant stand	normal	normal	irrelevant	less than normal
Precipitation	above normal	below normal	above normal	normal or above
Temperature	normal	normal or above	below normal	normal or below
Damaged areas	scattered or low areas	whole fields, upland areas	low areas	whole fields, low areas
Stem cankers	above 2nd node	absent	below soil line	below or slightly above soil
Canker lesion color	brown	tan	brown	dark br. or black
Fruiting bodies on stem	present	absent	absent	absent
External decay of stem	firm and dry	absent	firm and dry	absent or firm and dry
Int. discolor. of stem	none	black	none	none
Sclerotia int. or ext.	absent	present	absent	absent
Condition of fruit pods	normal	normal	few or none	irrelevant
Condition of roots	normal	normal	normal or rotted	rotted

Figure 10. Partial discriminant characteristics for clusters of soybean disease cases produced by CLUSTER/PAF

S = sparseness • (k-β), where β is a parameter which balances the influence of k vs. the sparseness. The results shown in Figure 11 are for β=1. In our experiment there was a strong correlation between the cpu time used and the parameter S. This fact may indicate that the algorithm operates more efficiently when the number of clusters formed agrees with some "natural" organization of the data.

Number of clusters	Sparseness (x 10^6)	Parameter S (fit vs. complexity)	Cpu time used (on Cyber 175)
2	15.0	15.0	10 sec
3	0.5	1.0	23
4	0.03	0.09	21
5	0.10	0.4	44
6	0.02	0.1	40

Figure 11. A summary of evaluation criterion scores for soybean disease clusterings for k=2 to 6

Another application of CLUSTER/PAF to a problem in the area of musicology is described in [14].

VIII. Conclusion

A method of conceptual clustering was discussed that produces clusters together with their descriptions in the form of conjunctive statements closely "fitting" the clusters. The important difference between this method and traditional clustering methods is that it does not use a similarity measure (in the usual sense) for forming clusters. Clusters are simply defined as groups of objects whose descriptions are disjoint logical products of relations on object attributes, optimizing a predefined global criterion. Experiments performed so far have shown that the method produces clusters that tend to match well solutions most satisfactory for people. Similar

experiments with numerical taxonomy methods resulted in clusters that were less satisfactory in this regard.

From the viewpoint of traditional clustering methods conceptual clustering can be interpreted as an approach that also uses a measure of object "similarity," but of a quite differend kind. This new kind of "similarity" takes into consideration not only the distance between objects (as in conventional clustering methods), but also their relationship to other objects and, most importantly, their relationship to some predetermined concepts (here, conjunctive concepts).

The price of using such a more complex similarity measure is the significantly greater computational complexity of the method. For example, each dendrogram produced by NUMTAX for example I required about 60 milliseconds of processor time on a CYBER 175, while clusterings produced by CLUSTER/PAF for the same example required 1.5 to 4 seconds of processor time. (The above comparison is not totally appropriate because NUMTAX produces only clusters, while CLUSTER/PAF produces both clusters and their descriptions.) The greater computational complexity is not necessarily a significant disadvantage of the method. If the results are indeed useful and practical, then the computational cost is of little relevance, especially now when the prices of computer technology are declining. Experience shows that researchers using presently available clustering techniques are most concerned not with the amount of computational time expended but with the difficulty of interpreting the results of the analysis.

Another important characteristic of the method (and a limitation or advantage depending on the problem at hand) is that it is specifically

oriented toward clustering problems using nominal or ordinal variables. It should be noted, however, that the method can also handle other types of variables, if they are properly quantized.

Concluding, the presented method of conjunctive conceptual clustering adds a new dimension to research in cluster analysis, and seems to have the potential to be a useful new tool for researchers analyzing data.

IX. Acknowledgements

The authors wish to thank Professor Richard Selander, University of Illinois Department of Genetics, for providing the numerical taxonomy program NUMTAX used in the comparative analysis of clustering methods. Partial support of this research was provided by the National Science Foundation grant No. MCS-79-06614.

REFERENCES

[1] Anderberg, M. R., Cluster analysis for applications, Academic Press, New York, 1973.

[2] Diday E., Govaert, G., Apprentissage et mesures de ressemblances adaptatives, Computer Oriented Learning Processes, Nato Advanced Study Institute, series E, No. 14, 1976.

[3] Diday, E. and Simon, J. C., Clustering analysis, Communication and Cybernetics 10, Springer-Verlag, Berlin, Heidelberg, New York, pp. 47-92, 1976.

[4] Diday, E. Problems of clustering and recent advances, 11th Congress of Statistics, Oslo, Norway, 1978.

[5] Diday, E., Govaert, G., Lechevallier, Y., Sidi, J., Clustering in pattern recognition, Fifth International Conference on Pattern Recognition, Miami Beach Florida, December 1-4, pp. 424-429, 1980.

[6] Michalski, R. S., VARIABLE-VALUED LOGIC: System VL_1, Proceedings of the 1974 Intern. Symp. on Multiple-Valued Logic, West Virginia University, Morgantown, West Virginia, May 29-31, pp. 323-346, 1974.

[7] Michalski, R. S., Synthesis of optimal and quasi-optimal variable-valued logic formulas, Proceedings of the 1975 Intern. Symp. on Multiple-Valued Logic, Bloomington, Indiana, May 13-16, pp. 76-87, 1975.

[8] Michalski, R. S., Variable-valued logic and its applications to pattern recognition and machine learning, in: Rine, D. (ed.), Multiple-valued Logic and Computer Science, North-Holland, pp. 506-534, 1975.

[9] Michalski, R. S. and Larson, J.B., Selection of most representative training examples and incremental generation of VL_1 hypotheses: the underlying methodology and the description of programs ESEL and AQ11, Report No. 867, Department of Computer Science, University of Illinois, Urbana, Illinois, 1978.

[10] Michalski, R. S., Pattern recognition as rule-guided inductive inference, Pattern Analysis and Machine Intelligence, Vol. 2, No. 4, pp. 349-361, 1980.

[11] Michalski, R. S., KNOWLEDGE ACQUISITION THROUGH CONCEPTUAL CLUSTERING: A theoretical framework and an algorithm for partitioning data into conjunctive concepts, A Special Issue on Knowledge Acquisition And Induction, International Journal of Policy Analysis and Information Systems, Vol. 4, No. 3, pp. 219-244, 1980.

[12] Michalski, R. S., Chilausky, R. L., Learning by being told and learning from examples: an experimental comparison of the two methods of knowledge acquisition in the context of developing an expert system for soybean disease diagnosis, International Journal of Policy Analysis and Information Systems, Vol. 4, No. 2, pp. 125-161, 1980.

[13] Michalski, R. S., Stepp, R., Revealing conceptual structure in data by inductive inference, in Machine Intelligence 10, eds. J. E. Hayes, D. Michie, Y.-H. Pao, Ellis Horwood, Chichester, Halsted Press (John Wiley), New York, 1981.

[14] Michalski, R. S., Stepp, R., An application of AI techniques to structuring objects into an optimal conceptual hierarchy, Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver, Canada, August 24-28, pp. 460-465, 1981.

[15] Nilsson, Nils, T., Principles of artificial intelligence, Tioga Publishing Company, Palo Alto, Calif., 1980.

[16] Sokal, R. R., Sneath, P. H., Principles of numerical taxonomy W. H. Freeman, San Francisco, 1963.

[17] Stepp, R., Learning without negative examples via variable-valued logic characterizations: the uniclass inductive program AQ7UNI, Department of Computer Science, Report 982, University of Illinois, Urbana, Illinois, July 1979.

[18] Watanabe, S., Knowing and Guessing; a quantitative study of inference and information, Wiley, New York, 1969.

[19] Winston, P. H., Artificial Intelligence, Addison-Wesley Publishing Company, Reading Mass., 1977.

[20] Zagoruiko, N. G., Lbov, G. S., Algorithms of pattern recognition in a package of applied programs, 4th Int. Conf. on Pattern Recognition, Kyoto, Japan, p. 1100, 1978.