Information Visualization and Exploration

Potential and Requirements

Johan Hagman

SSKKII / Dept of Linguistics
University of Göteborg
hagman@ling.gu.se

SSKKII Report #3

Center for Studies of
Language, Cognition, Interaction,
Semantics, Communication, and Information

Abstract

Starting with examples of a range of applications which would benefit from a powerful tool for information visualization and exploration, the report suggests three important modules of such a tool, vz. multi-dimensional analysis, visualization, and human-computer interaction. The importance of the integration of these three modules is especially emphasized. A short introduction to multi-dimensional analysis is given with the focus on cluster analysis. In connection to this, visualization is discussed and, briefly, human-computer interaction.

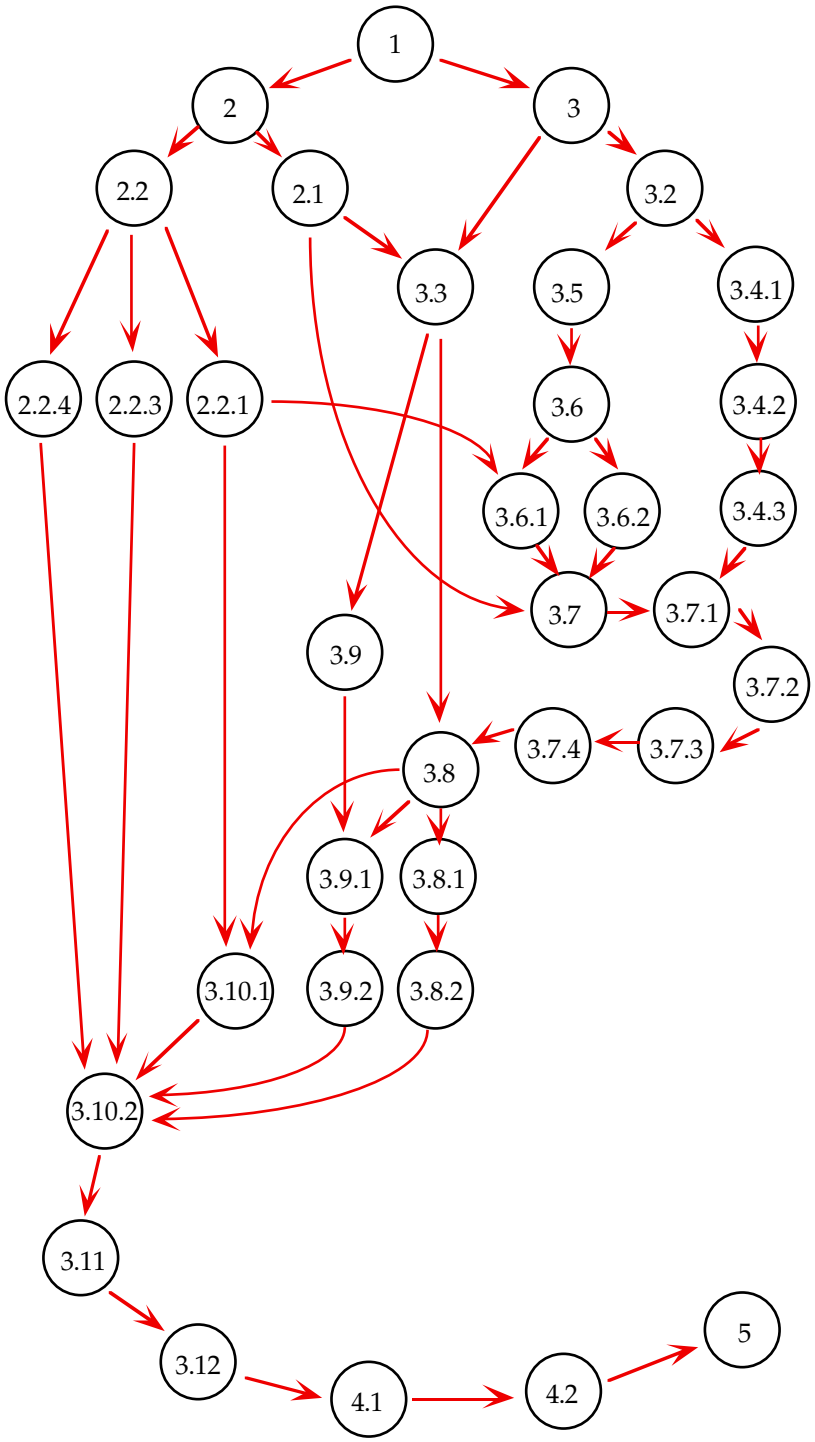
Keywords

information exploration, visualization, cluster analysis, human-computer interaction, multi-dimensional analysis, MDS, factor analysis, automatic document classification, networks, projections, maps, correlations

Contents

1	**Introduction**	**1**
2	**Potential of IVE**	**1**
2.1	A Promising Combination	
2.2	Applications I	2
2.2.1	Science	
2.2.2	Resource Detection and Mobilization	
2.2.3	Information Retrieval and Library Service	
2.2.4	Communication	3
3	**Requirements for IVE**	**4**
3.1	Data Formats	
3.2	Data Types and Scales	
3.3	Visualization Types	5
3.4	Data Preparation	7
3.4.1	Operationalization	
3.4.2	Normalization	8
3.4.3	Equalization	
3.5	Structuring and Classification of Data	9
3.6	Dimensions of the Data	
3.6.1	Using Known, Preexisting Classes or Dimensions	10
3.6.2	Generating "New" Dimensions	
	Factor Analysis	11
	Multidimensional Scaling	
3.7	Clustering Methods and Algorithms	13
3.7.1	Pattern Matrix	14
3.7.2	Proximity Matrix and Proximity Index	
3.7.3	Dendrograms	16
3.7.4	Hierachical clustering	18
3.7.5	Partitional clustering	
3.8	Projections	19
3.8.1	Linear Projections	
3.8.2	Nonlinear Projections	21
3.9	Visualization — More Examples	22
3.9.1	2D Projections	
3.9.2	3D and hD Projections	23
3.10	Exploratory Data Analysis	25
3.10.1	Methodology	
3.10.2	Applications II	26
	Science	
	Document clusters and I.R.	27
	Networks	29
3.11	Human-Computer Interaction	
3.12	Hardware	31
4	**Tools**	**31**
4.1	Types of Visualization Tools	
4.2	Putting it All Together: Integration	32
5	**Summary**	**33**
	References	**35**

The author has forced his holistic view of the material into a linear order. The internal reference structure of the chapters may be indicated as follows (a few minor chapters have been omitted):



1 Introduction

This report is divided into three main sections. The next section *Potential of IVE* presents three important components of information visualization and exploration (IVE) support, followed by some developing applications which could benefit from integration of these components into efficient IVE systems. The section *Requirements for IVE* then gives a more detailed description of two of these components, i.e. their disciplinary background and realization. Finally, in the section *Tools* we will comment on some existing types of auxiliaries for IVE.

Since the techniques dealt with here are new and maybe still unfamiliar to some readers, this report is styled as an introduction which also tries to gather some basic terminology of these techniques into one place.

2 Potential of IVE

When building systems to support visualization and exploration of large databases with complex numerical or textual information, there are three components that seem particularly important and whose integration would be very powerful. They are *cluster analysis*, *data visualization*, and *human-computer interaction*. The present study has focussed on the first two of these components, whereas the third one will be more briefly characterized.

2 .1 A Promising Combination

The term 'botryology' was suggested by [Good 77] for the discipline of cluster analysis. The term is derived from a Greek word meaning 'a cluster of grapes'. This emergent discipline is given the following presentation in [Dubes & Jain 80]:

> Cluster analysis is a product of the Computer Age. Without modern digital computation, only the most elementary clustering techniques could be realized for very small sample sizes and the impetus to fabricate a discipline of Cluster Analysis would not exist. The development of clustering methodology has been a truly interdisciplinary project. Research in several areas of the physical, life, and social sciences required the tools of exploratory data analysis that modern digital computation made possible. [p. 114]

Visualization, the second object for special attention in this report, is thus characterized in [Rheingans 92]:

> Commonly, a researcher wishes to explore a large set of data in order to develop an understanding of the structure and relationships within the data. She may have informal or incomplete hypotheses about that data that she wishes to develop further. This sort of exploratory process differs from more formal hypothesis testing in that specific beliefs about the precise meaning of the data have not yet been formed. Representing the data visually for this exploratory process is appealing because it allows viewers to harness the powerful processing capabilities of the human visual system. Some structures in the data, especially those involving complex spatial relationships and patterns, are easy to detect visually, but difficult to specify for computational detection. [p. 252]

Researchers are not the only ones who benefit from these techniques. The next section examplifies this.

2.2 Applications I

Here some applications are exemplified to give an idea about the usefulness of powerful tools for data exploration. They also serve as illustrations for the theoretical discussion in the subsequent sections which will introduce some terminology that will be useful when later continuing this theme in section 3.10.2, called *Applications II*.

2.2.1 Science

Meteorology, fluid dynamics, materials research, physical chemistry, drug design, medical science, biochemistry, biology, psychology, archaeological data analysis and reconstruction, geology, cartography, remote sensing, image processing ... the list could easily be made longer and not only specialized scientific disciplines with their particular needs must be met with well designed IVE techniques — there are also a lot of interdisciplinary activities in today's increasingly synthesizing research where many specialized fields again converge to share their knowledge and join forces. Just to mention one example of multidisciplinar collaboration, we may refer to the research on the origin of cultural humanity where archaelology, genetics, and linguistics collaborate to offer new, much elaborate suggestions of how to map the evolvement and movements of humanity and its cultural attributes, such as language, through space and time[1].

2.2.2 Resource Detection and Mobilization

Some of the most important measures of development of a society [UNDP 90-94] — once the basic needs of peace, food and housing are met — deal with the fields *information* and *communication*. Parameters used to esimate one or both of these fields are mean years of schooling, scientist density, densities of passenger cars, radio, TV, and telephones, and newspaper circulation. Resource detection and mobilization concerns both human skill and goods. Both employers and employees want the right person in the right place on the right time. The whole society depends on how efficiently this works. On a larger scale, both the society, its industries and single individuals are interested in matching offers and requests in trade and on the labour market within greater communities. Diplomats and people working with marketing and public relations are aware of correlations between cultural characteristics, politics, and consumption habits. A deepened insight in these correlations could be gained by the help of the kind of tools discussed here and be most valuable. Although the concepts *resource*, *information* and *communication* are partly overlapping, the following two sections will be associated to the latter two.

2.2.3 Information Retrieval and Library Service

The increase of information in various media is accelerating and this leads to an increased need of improved storage and retrieval routines. Putting aside the physical requirements emerging for the different media, we instead address the problem of organizing the information conceptually. Most

---

[1] A presentation of such work is found in [Renfrew 94]. Interesting further reading is found in [Renfrew 89] and [Ammerman & Cavalli-Sforza 84]. An earlier, suggestive study on the correlation between genetics and phonetics is presented in [Brosnahan 61] and FIGURES 1.1 and 1.2 have been borrowed from there. We see computerized tools for graphic IVE as important means for both facilitating interdisciplinary research and to make its result easier to mentally grasp by the public to which it is presented.

libraries of any considerable size have during the latest 10-15 years computerized their card systems to permit searches on other than the traditional fields, i.e. the writer, the title or the category of the library's classification system. This is already an improvement but it could be taken further, for instance by giving the library visitor a visual representation of (a relevant part of) the book collection with icons or other symbols suitably sorted and distrubuted over a computer screen — the greater the number of books, the more crowded the screen or the smaller the icons. The icons could be "clickable" for more detailed information to pop up. As we will see below in the section 3.10.2, computers also make it possible to classify and sort books and articles in new complementary and automatic ways.

Facilities of this kind do not have to be limited to large, public libraries but could be used even by the single individual for his private library or document files. In an interesting study presented in [Lin 92] a collection of context-indexed documents are fed into a self-organizing neural network and the result is graphically displayed on a 2D map showing the multidimensional relations between the documents in consideration. This map also works interactively as an interface for document retrieval systems.

2.2.4 Communication

For a society, communication is part of both its "neural" and "vascular" system, and its flow is one important indicator of the "state of health" of that society. As hinted above, communication is closely connected to human and material resource mobility and information.

For users of communication systems like computer networks, the common telephone net and in many sorts of daily interaction, including even chatting, there is a question seldom asked and even more seldom answered: "Do I constitute a node in a contact net and, if so, are there any other relevant nodes that I should know of?". An informative answer to this could consist of a graph which enables the questioner to "get the picture". In [Schwartz & Wood 93] over 50,000 users of 'email' on the 'Internet' (the worldwide correspondence network for university computers) were mapped according to who used to communicate with whom. Out of an enormous number of relations between persons and/or groups the result was reduced to an intelligible format (applying clustering techniques that will be discussed below) which showed isolated users, large contact centra, and certain "interest groups". One could also see which nodes seemed to be most related — knowingly or unknowingly — to any other given node.

A system which suggests contacts which the user is likely to be interested to make is useful for many applications. We mentioned this above and one intuitively typical application is the labor market where persons and skills are matched with the current request. In such an application, the user may be placed as an icon describing his "skill profile" in a scatter diagram, near other similar icons of people who offer similar things — or near "inverted" icons representing *requests* of these offers.

Other, maybe only indirectly "public" applications are tools for the optimizations of telephone nets on different scales and for traffic mapping and planning. The following section 3 will describe the modules required for

these tasks and towards its end (3.10.2) this discussion about applications will be continued.

3 Requirements for IVE

In this section we will discuss some modules and measures which seem necessary for the realization of an information visualization and exploration support that is powerful but still easy to understand and use.

3.1 Data Formats

As is the case with much modern software, the format of the import data is allowed to be at least of two kinds. Normally a simple, generic format is accepted like ASCII files with "raw" text for editors or with units of data separated by spaces, tabs or newlines for statistical programs. Simple are also files with pixel information for drawing programs. In addition to this, almost every program uses special formats both for data import and export which contains extra information, e.g. the result of manipulation of the data (e.g. font choices; calculations; and a picture's division in objects). Other reasons for using a special format can be the possibility to compress data in order to minimize storage space or to make the data unusable with other programs. Anyway, the possibility to load data in a generic format is an important feature which is often provided and, not at least, of the constructor's and vendor's interest. See [Brodlie & al. 92] for an exposé of data formats.

3.2 Data Types and Scales

When the information to be visualized and explored consists of other formats than text (which will be considered below) it is frequently expressed as some kind of numerical data. A categorization of data types and data scales is presented in [Anderberg 73] and summarized in [Jain & Dubes 88]. The question of which data category we are dealing with is very important for our choices of analysis and visualization technique. The information we have may contain data which describe a phenomenon with features that can be of one of the following three *types*: binary, discrete, or continuous. *Binary* features have exactly two possible values which could be 'yes'/'no' or '1'/'0'. A *discrete* feature has a finite number of values and may be compared to e.g. integers whereas other features are *continuous* like the real numbers. The data *scales* decide the relative significance of the values of a feature. There are two groups of scales: *qualitative* (nominal and ordinal) and *quantitative* (interval and ratio). Values of a *nominal* scale are merely names which do not say anything about how they shall be ordered. *Ordinal* scales add the possibility to order all values but say nothing about distances. *Interval* scales do refer to some unit of measurment and the values can be related to each other as well as to the interval's minimum and maximum in terms of distances. The values of a *ratio* scale are said to have ...

> an absolute meaning [which] implies that an absolute zero exists along with a unit of measurement, so the ratio between two numbers has meaning. [Jain & Dubes 88, p. 13]

... and the authors compare the interval temperature scales Celcius and Farenheit with Kelvin's scale of ratio type[2].

[2] A comment to that comparison is, however, that the definition of the concept "absolute zero" should exclude all the temperatures in which the liquidity of water actually *does* have an absolute zero (given a certain "air pressure"). As is known, Celcius found it convenient to divide the temperature interval in which water normally is liquid into 100 centigrades for every-day practical purposes and still e.g. weather

3.3 Visualization Types

In order to clearly define the types of data and visualizations of particular interest for this report, the classification of visualization techniques presented in [Brodlie & al. 92, p. 40ff] will be adopted. This classification of visualization techniques (and, implicitly, of the data) uses for any given entity (i.e. phenomenon described by the data) to visualize: (1) an index of its *functionality* and (2) the *dimensionality* of the domain over which the entity is defined. By functionality is meant that there is a known function which describes/gives all possible values over the domain; for instance, the entity 'illumination' at any point in a 3D space (except the space occupied by a possible light-giving source) is given by a function (of scalar type). The functionality index is identical to the function type and in addition to the scalar type, a vector type and a tensor type are also identified. The notation used for any entity E is ${E_n}^S$, ${E_n}^V$, and ${E_n}^T$ respectively for the different function types, where n is the index of the of the domain dimensionality. Sometimes, and especially in data exploration, there is no function identified which explains the data observed. The data just constitute a set of points and in that case the functionality index used is ${E_n}^P$. The domain dimensionality is further subclassified according to the nature of the domain and there are three cases identified (quoting [Brodlie & al. 92, p. 41]):

1. The entity is defined pointwise over a continuous domain — for example, in a terrain map, the height is defined separately for each point.

2. The entity is defined over regions of a continuous domain — for example, a chloropleth map showing the density of population in each country is of this type. To indicate this we use the notation $E_{[2]}$ to indicate a two-dimensional domain, with the entity defined over regions rather than at individual points.

3. The entity is defined over an enumerated set — for example, a chart showing the number of cars of each manufacturer sold in a particular year. To indicate this we use the notation $E_{\{1\}}$ to indicate a one-dimensional domain consisting of a set of enumerated values, i.e. car manufacturers.

The $E_{[n]}$ type is thus associated with *histograms*, where the items represent ranges (n=1) or regions (n≥2) of the domain which are naturally ordered.

The $E_{\{n\}}$ type is used for sets which are not necessarily intrinsically ordered (cf. nominal scales, above) and their elements are often represented with *bar charts*. In the example used in the definition above, we can think of bars associated with car manufacturers and their length are proportional to the number of cars sold.

This report will *not* discuss visualizations associated with the types E^V, $E_{\{n\}}$, or E_1. In other words: visualization involving 2D or 3D arrows, stream lines, particle tracks (which are all examples of vectors), or bar charts (referring to enumerable, not "naturally" ordered sets), or whatever could be visualized

---

forecasts are easier to understand when relating precipitation to that interval than to the distance from total subatomic paralysis (i.e. 0 Kelvin) — which however is practical in physics.

with one-dimensional *line graphs* will not be considered here and neither the E^T type. However, a visualization may be said to *include* nominal information, for example when points in a scatter plot are provided with labels referring by text or color to some nominal scale.

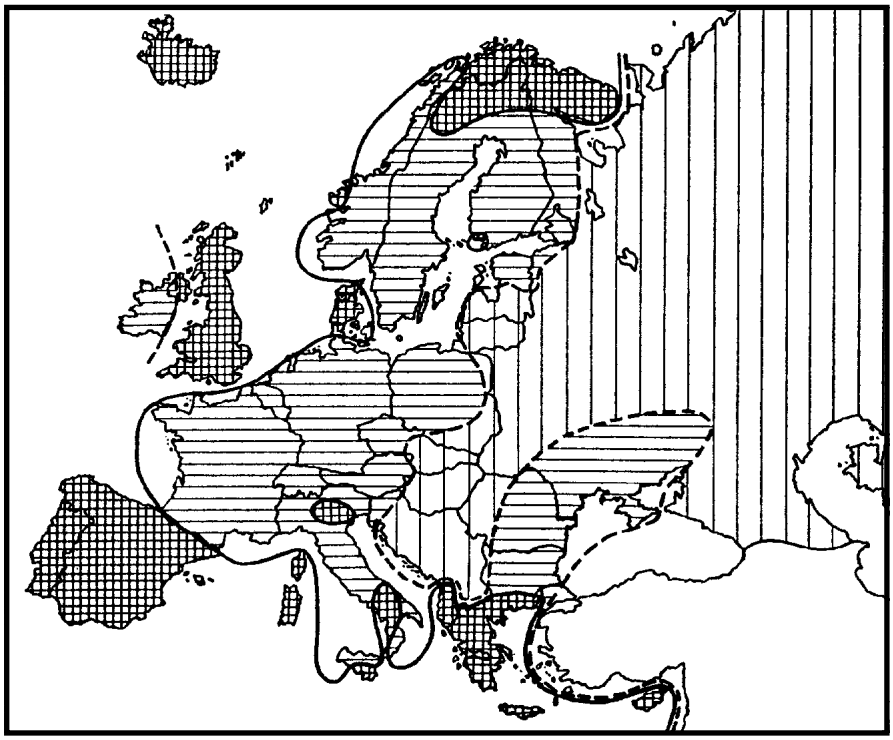


FIGURE 1.1 The present and past distribution of dental fricative in Europe

(from [Brosnahan 61])

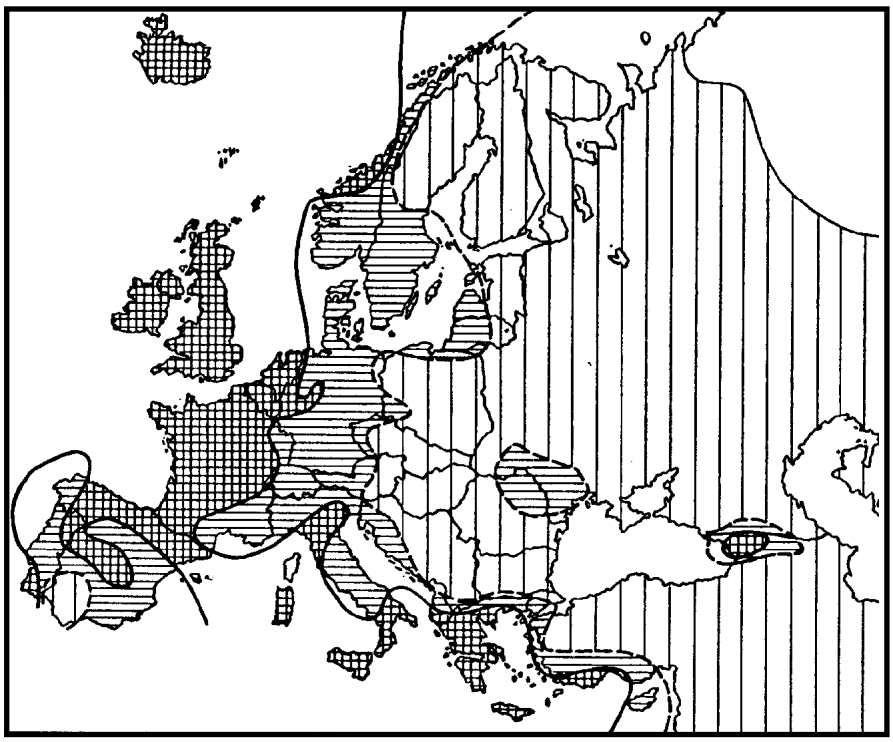


FIGURE 1.2 The distribution of blood group gene 0 in Europe

(from [Brosnahan 61], after [Mourant 54])

This report *will* focus on $E_{[2]}{}^S$ types (mainly geographical maps like those exemplified in FIGURES 1.1 and 1.2) and $E_2{}^S$ types (especially in combination with the outcome of cluster analysis). A discussion on the possibilities of E_3 types will also be done.

3.4 Data Preparation

3.4.1 Operationalization

If we by 'operationalization' mean to make data able to be processed mathematically, we have to convert qualitative features into quantitative ones. Above the binary data type was mentioned and when we come to this point variants like 1/0 (or 1/-1, or whatever is preferred) remain the only possible alternative. Nominal scales cannot be used but must be converted to a series of interval scales with binary values — for instance, the nominal scale {Talbot, Toyota, Triumph} in a data set describing cars must be converted to the three interval scales Talbot={0,1}, Toyota={0,1}, and Triumph={0,1}.

Here, in parallel, we will say something about the treatment of texts. For many purposes, the best way to visualize and explore text is to keep it just the way it is and simply read or browse it. In the applications that will be concerned in this report the texts are however many, so many that it is practically impossible to orient among them by looking at each of them. This is the situation, as was stated earlier, at libraries and not rarely in private collections of books and various documents. So, what we are primarily concerned with here is visualization and explorations of text *collections*.

To make a collection of text manageable, each single text must be represented by a reduced version of its original, informative content, it must be *indexed*. One way is to apply a predefined way of classifying the text using e.g. a standard bibliographical classification system which reduces the infinite number of text types to a finite number, albeit thousands, of nested subclasses. This classification is always done manually and that supposedly gives the most reliable result. However, we understand that this can be quite time-consuming and expensive a procedure and speculations about whether it is automizeable arise. The idea of subjecting every piece of text to automatic *semantic* parsing is utopic simply because there are no such systems but *syntactic* analysis have been carried out on large sets of documents to underlie a subsequent indexing procedure; [Fagan 87], [Smeaton & Sheridan 91], [Grefenstette 92]. The question is however wheteher the improvement in text retrieval effectiveness offered by syntactive indexing phrases as a supplement to word-based indexing is worth its price [Lewis & Croft 90]. Syntactic analyzers often contain some sort of morphological analyzer as a module which makes it possible to recognize all inflected variants of the same root. A morphological analyzer of reasonably high quality is normally fairly easy to find or develop and is probably worthwhile to apply in order to more accurately index a text.

The roughest and cheapest method to get a content "profile" of a text, is extracting and counting (possibly morphologically analyzed) words which are typical of that text. Such words are typically not highly frequent since that would not distinguish a text from another in any relevant way for what is normally intended by 'content'. There have been a number of studies on this in computational linguistics and AI, studies which all use list of words to ignore, or to filter out when a text is "threshed"; [Enser 85], [Lin 92], [Lewis 92]. The list have been called 'black list' or 'stop list', containing 'function words' or 'stop words', some of which (but far from all) coincide with the linguistically well-known concept of *syncategorematic* words. We must remember that these words are not that numerous when counted as

types in a text but they may constitute half the number of *tokens* in the text and this has implications for the performance of the comparing algorithms.

The result of each text analysis is a kind of list of the *n* most frequent terms (be they syntactic structures, morphological roots or 'graph words') along with their frequency. One can also think of a vector {0.0, 0.0, 0.0, 2.3, 0.0, 7.9, ...} where each position refers to a term and the value refers to the occurrence of that term in the text. Both absolute values and values relative to the text size can be used when calculating the similarity between texts of different sizes.

3.4.2 Normalization

A multivariate phenomenon that one would like to visualize and examine is often described with data of both different types and scales. Suppose, for instance, that we would like to make a survey of what cars are offered on the motor market. Let us imagine a list of 60 different car types where each car is described by a number of features from which we pick the following five: *acceleration*, *engine power*, *gas consumption*, *presence of airbag*, and *weight*. Some features are expressed in units that imply low numbers whereas other features' units imply high numbers. Acceleration (0-100 km/h, expressed in seconds) may, for instance, have a range of 7-15 whereas weight (expressed in kilos) may have a range from 500-2,500 and airbag has the binary interval 0-1. We can think of a table with 60 rows, one for each car, and as many columns as there are features. Now, this table can be hard to survey when each feature/column has a different scale, so therefore the columns can be *normalized* to be more easily comparable. Normalization means that the values of all features are recalculated to be scaled into a given range of values, e.g. 0.0-1.0, 0-10, 0-100, or whatever scale is preferred. For our example, we may prefer an integer scale from 0 to 10 where 10 could correspond to either the highest or the "best" (i.e. *lowest* in gas consumption) value of the original table. Then, we will end up with columns with values ranging from 0 to 10 and each car type will have a kind of "performance profile" expressed on its row. These profiles can then be summed and used to rank the cars and, when doing so, we easily see that all the features' possible values must be compatible. Raw numbers cannot be used since e.g. the values for acceleration and airbag would "drown" when fused with the one for weight.

As for text-derived data, it depends on the purpose of the comparison whether to normalize or not. When texts are compared by their *absolute* numbers, then e.g. a thick book that dedicates a whole chapter to the same subject as does a certain short article will be more strongly connected to this article than if *relative* numbers would have been used, which gives the importance of the terms in a given text related to the number of other terms in this text. More about how text profiles are compared will be discussed in section 3.10.2.

3.4.3 Equalization

Although the normalization has put all features within one range of possible values, the value *spread* within each feature may vary a lot. An example from [UNDP 91] could probably show this; life expectancy at birth varies among the 10 European states with the highest development index (1991) between 75.2 and 77.8 years with a mean value of 76.55 and the standard

population deviation (σ_n) is .89 which corresponds to 1% of the mean value. This could be compared with the data for suicides for the same countries which vary between 8.2 and 28.2 persons every 100,000 people, having the mean of 17.35 and the corresponding deviation 6.2 which equals to 36% of the mean value. If we displayed these two measures, normalized, by lining up the countries on the x-axis of a bar chart we would see that the "skyline" of the bars representing suicide rates would be much "rockier" than that of the life expectancy bars. So, although measures are normalized they seem comparable to different degrees depending on their spread or *internal variation*. This is the reason why an *equalization* sometimes is done. Then, the values are again recalculated taking account of the internal variation in order to make the features more comparable even in this respect. That process yields "skylines" or curves of a similar degree of "rockiness".

3.5 Structuring and Classification of Data

In order to conceptualize and maybe also visualize the phenomena our data describe, we try to "conquer" the data mentally by having an order found among them or "forced upon" them so that they become easy to survey (if possible, momentaneously) and easy to describe in terms of one or more parameters and the value distribution over these parameters. As stated in section 2.1, this not only enables the data items to be grouped and counted more easily but it also offers better possibilities to *understand why* the data is the way it is — one may detect trends, dependencies or correlations and this could lead to the formulation or refinement of *rules* or *formulae* which describe the data (and thereby supposedly the original phenomena). It is well known that data are preferred to be generalized to rules in knowledge bases and theories whenever possible, for the sake of expressive power and economy.

The usefulness of whatever structure or formal description of the data we may come up with is of course dependent on what order relation(s) we structured the data with from the beginning, i.e which dimensions or features are considered *cardinal*? An example: why do we usually find it more suitable to structure e.g. a set of cars according to a nominal or quasi-ordinal "dimension" like {sport cars, family cars, minitrucks, lorries, trailers, ...} than according to the color spectrum? In the following section 3.6 we will discuss cases where data are structured and visualized in cardinal or "principal" dimensions that were either obvious in the data set and easy to identify or that were "hidden" and had to be discovered through some analysis. In section 3.7 data items are not primarily structured according to *dimensions* but to their internal *relations*, especially the relation of *similarity* between them.

3.6 Dimensions of the Data

The most practised method to structure and visualize the data is by selecting a few of its most "important" dimensions and then make 2D or possibly 3D scatter diagrams. The following subsection 3.6.1 reminds of a familiar, straightforward procedure whereas 3.6.2 introduces two more sophisticated procedures whose principles will be contrasted with that of cluster analysis in section 3.7. The visualization techniques relevant to the content of 3.6.2 and 3.7 will be then characterized in section 3.8.

3.6.1 Using Known, Preexisting Classes or Dimensions

In case the number of dimensions is small the choice can be quite straightforward. And, the more "natural" the dimension seems to be the easier it is to choose one of them to structure the data in an early stage. An example: from geographic data about a set of cities could be chosen the longitude and latitute coordinates as the dimentions to structure the data with, on for example, a 2D plot. If we then choose e.g. the values for mean July temperature and relate them to the structure we got from the coordinates, we may find some correlations. But if we instead of the above-mentioned coordinates had chosen data on altitude and some index of the amount of open water surfaces within a given circle, then maybe we would get just as interesting correlations. If we do the same thing with the values for crime density we might find no special correlation at all (or very insignificant ones) when we compare them to the dimensiones just mentioned. Let us suppose that crime density instead is strongly correlated with population density in our data base over these cities. How shall we detect that? Answers will be proposed in 3.6.2 and 3.7.

Earlier was mentioned the multidisciplinary research on the human origin. There, the geographical longitude and latitude grids constitute a very straightforward choice of basic dimensions for data visualisation; it is most intuitive to show linguistic and genetic similarities with colored fields on world maps. Early human dispersal over the world continents formed both linguistic and genetic continua which, although some of them now are partly overlapped or completely covered, are detectable to a great extent still today. Many 2D continuous fields of the world surface in this way come to represent the current distribution of linguistic and genetic traits and some of them suggest the history (and, possibly, explanation) of this distribution. Probably there are other linguistic or genetic phenomena which have a much less continuous, scattered distribution as far as the world's inhabitable land masses are concerned — especially today with the present communication systems. Maybe in such case, suitable "dimensions" concerning profession or "general lifestyle" should be chosen to best visualize existing and emerging correlations.

3.6.2 Generating "New" Dimensions

Sometimes the elements in a data set are described with so many dimensions or features that it is impossible to conceptualize and to visualize both the data (as ordered by all features) or the features (as how they together structure the data). Below we shortly characterize two common ways of decreasing the number of features and thereby making them more manegable both mentally and representally. In these cases the features are compared as to how *interdependent* they seem to be, i.e. how similarly they seem to describe the data. As a result of such a comparison they can eventually be grouped together or substituted by other, fewer features which will suffice to describe most of the data variance.

Factor Analysis

This technique tries to reduce the number of dimensions or features which describe the phenomenon. It is thus characterized in [Jain & Dubes 88]:

> The goals of factor analysis ... are to create a model for a set of objects containing fewer and more pertinent features than did the original representation. Some 'underlying' or 'fundamental' factors are being sought ... . A great deal of effort has been devoted to minimizing the number of factors involved. ... A factor analysis resolves each original pattern into factors that may or may not have any interpretation in the original problem. [p. 260]

Factor analysis is naturally associated with *linear projections* (see section 3.8.1 for an illustration of both). This is an interesting technique but lies just outside the focus of this study. Therefore, we pass on to another technique which comes somewhat closer to the focus of our interest.

Multidimensional Scaling

In section 3.2 we encountered the term *ordinal* and in section 3.7.2 the term *proximity matrix* is explained for those who are not familiar with it. The two terms are used in [Jain & Dubes 88]'s description of this technique, often abbreviated 'MDS':

> Multidimensional scaling is a generic name for a body of procedures and algorithms that start with an ordinal proximity matrix and generate configurations of points in one, two or three dimensions. ... Multidimensional scaling translates an ordinal scale to a set of ratio scales and is an example of *ordination*. ... The literature of multidimensional scaling is large and diverse, with almost all theoretical developments and applications being in the behavioral and social sciences. [pp. 46-47]

An example of MDS output is shown in FIGURE 2 and is thus commented in [Forgas 82, pp. 151, 154], representing here the MDS literature:

> [The] perhaps most promising area of MDS techniques [is] the study of *social episodes*. The study of social situations and social episodes is a recent development in social psychology, which can be directly likenend to the major shift of orientation currently taking place ... . Initially, concern with the study of social episodes ... yielded little more than a spate of speculative, *ad hoc* taxonomies proposed by researches who often did not even consider the applicability of quantitative techniques to this difficult area. ... [Wish & al. 76] addressed the problem of which dimensions underlie the perceptions of dyadic relationships, an important component of the definition of an episode. Students rated dyadic relationships such as 'between causal relationships', 'between business rivals', 'between you and your class mates', etc., on a set of bipolar scales. Similarity judgments were extracted from these data and subjected to an ... [MDS] analysis, resulting in a four-dimensional solution defined by the equality, competitiveness, intensity and formality dimensions.

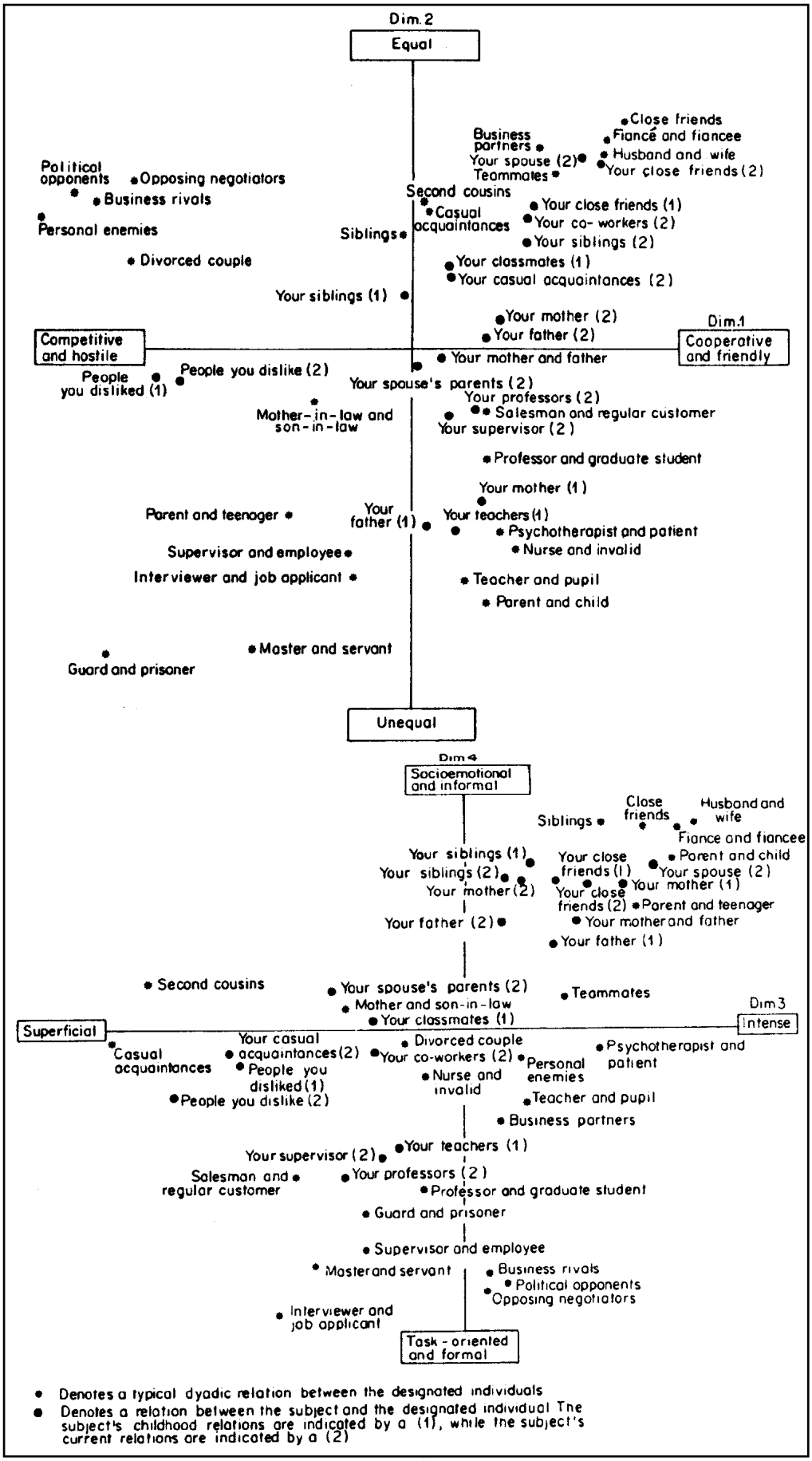


FIGURE 2 Dimensions no. 1, 2, 3, and 4 of the 4D MDS solution, showing the perception of 25 'typical' and 19 'own' interpersonal relations (from [Wish & al. 76]).

3.7 Clustering Methods and Algorithms

The general idea of MDS and factor analysis is to concentrate on the features or factors describing the phenomenon and to analyze their relationships in order to get suitable groups of them or to generate new, synthezised variants of them to re-represent the data. The result is, a managable number of dimensions (often 2 - 3), hopefully associable with intuitive labels, in which the data then can be plotted and so visualized.

Another idea is not to bother about the high number of describing features but just use them all in concentrating on getting suitable groups of data. If the two above-mentioned methods can be called "feature-focussed", then the technique of *cluster analysis* is "item-focussed".

Both ideas are ways of structuring data and obtaining descriptive features for them; MDS and factor analysis start with the features (obtained by analyzing the data) and then these are (re)applied to the data; cluster analysis starts with the items (as they are described by the features) and the result is an arrangement of the data which can also, if found interesting, make suggestions for improvement of the features (their internal hierarchy, interdependence, redundancy, &c). What is more, MDS and factor analysis often plot the data in space according to a limited and pre-established number of *dimensions* which are often labelled, whereas cluster analysis plot the data in space considering every single *relation* inbetween each possible pair of data items. Although the results between these two methods often are very similar, they differ in their interest of identifying and operating (meta)features.

Cluster analysis was already mentioned in section 2.1 and here is another couple of quotations from where it is described in the literature. In [Willett 88] and [Jain & Dubes 88], respectively, we read the following:

> Clustering involves the grouping of similar objects, and has been practiced, consciously or unconsciously, for many thousands of years. The development of computer technology has resulted in *cluster analysis*, a multivariate statistical technique that allows the production of classifications by purely automatic means, the grouping generally being achieved by means of the calculation of all the interobject similarities. Cluster analysis is often referred to as *classification* but this is potentionally misleading. Classification normally refers to the *assigment* of objects to predefined classes whereas cluster analysis requires the *identification* of these classes; thus clustering must precede classification in the analysis of a dataset. [p. 141]

> Cluster analysis is an important technique in the rapidly growing field known as exploratory data analysis and is being applied in a variety of engineering and scientific disciplines such as biology, psychology, medicine, marketing, computer vision, and remote sensing. Cluster analysis organizes data by abstracting underlying structure either as a grouping of individuals or as a hierarchy of groups. The representation can then be investigated to see if the data group according to preconceived ideas or to suggest new experiments. ... Cluster analysis is the formal study of algorithms and methods for grouping, or classifying, objects. An object is

described either by a set of measurements or by relationships between the object and other objects. Cluster analysis does not use category labels that tag objects with prior identifiers. [pp. xiii, 1]

3.7.1 Pattern Matrix

Let us continue our example from the motor market — we now suppose that each car is described with n features with normalized (and perhaps equalized) values in an n-place vector $\{\text{Value}_1, ..., \text{Value}_n\}$. The set of all cars described in this way is viewed as and called a *pattern matrix* where in our example there are 60 rows which define the 'pattern' for a car and each of the n columns denotes a feature. The 60 cars are then thought of as plotted out as points in an abstract n-dimensional space — the so-called *pattern space*. In the literature these points are also referred to as 'patterns' even if this may be misleading in that that term normally suggests the topological arrangement of objects. Here, thus, 'pattern' refers back to the one-dimensional row of feature values which were used to decide the position of the representing point in the pattern space. [Jain & Dubes 88] continue:

> A cluster can be visualized as a collection of patterns which are close to one another or which satisfy some spatial relationships. The task of a clustering algorithm is to identify such natural groupings in spaces of many dimensions. Although visual perception is limited to three dimensions, one must be careful not to think automatically of clustering problems as two- or three-dimensional. The real benefit of cluster analysis is to organize multidimensional data where visual perception fails. [p. 8]

For an illustration, see TABLE 3.1 which shows another pattern matrix. There are 15 occurrences each of the handwritten characters '8', 'O', and 'X' and the eight features describe how they extend over a 24x24 grid (explained in detail in [Jain & Dubes 88]). These so-called '8OX' data have been treated by both factor analysis (FIGURES 3.2 and 3.3) and cluster analysis (FIGURES 3.4 and 3.5).

3.7.2 Proximity Matrix and Proximity Index

Another term belonging to cluster analysis and related topics is *proximity matrix*. A familiar proximity matrix is the table of interurban distances often attached to road maps. There, a set of cities are described in terms of relations inbetween themselves. The names of the rows in this special case of pattern matrix is normally identical to those of the features — i.e. the cities' names with the implicit addition of "Distance to ...". A table of road distances is a proximity matrix where the numbers express *dissimilarity* — here of geographical position. There are also proximity matrices where the *similarity* increases with the indices. This could, for instance, be the case for textual data, as will be seen below.

Let us again recall the table of what is offered on our motor market. The table has been duly normalized and equalized and all the 60 car types are described with features all of which have values ranging from (for the sake of simplicity, say the integer numbers) 1 through 10. Now, the similarity of any two cars is represented by their closeness in the corresponding pattern space but since this space has so many dimensions it is very difficult to mentally visualize and compare these "interpoint" relations in the space.

PM	1	2	3	4	5	6	7	8
'8	7	13	5	6	13	2	3	
'8	5	13	6	4	6	13	3	13
'8	9	10	6	6	8	10	2	3
'8	7	7	6	6	8	7	2	3
'8	8	7	6	6	8	7	2	0
'8	7	7	6	7	7	7	1	1
'8	6	10	7	8	8	9	4	4
'8	6	7	7	7	9	8	4	5
'8	5	5	5	12	10	7	2	3
'8	7	8	4	7	6	2	3	
'8	8	7	5	4	6	10	1	0
'8	6	10	5	2	6	8	1	2
'8	7	10	5	5	8	7	1	20
'8	7	10	6	6	6	8	3	3
'8	6	6	6	7	7	8	3	2
'O	7	7	5	7	6	3	4	6
'O	7	6	7	6	3	4	5	
'O	6	6	5	5	4	3	4	5
'O	8	8	7	6	5	6	5	5
'O	6	8	7	5	6	5	2	2
'O	7	7	7	7	6	6	3	2
'O	7	7	7	7	5	6	5	5
'O	8	7	7	8	7	5	6	4
'O	7	8	6	5	4	6	2	4
'O	9	7	7	6	8	6	4	3
'O	8	8	7	6	7	6	4	4
'O	7	6	6	5	3	2	5	4
'O	7	6	7	5	5	5	4	3
'O	0	8	7	6	6	7	2	4
'X	10	7	6	6	9	9	7	10
'X	10	4	4	4	8	8	3	10
'X	10	7	4	4	9	9	3	9
'X	7	7	6	5	10	10	10	8
'X	6	10	6	10	8	8	13	4
'X	8	10	7	10	9	8	11	4
'X	5	7	8	7	8	9	9	8
'X	5	6	7	7	9	9	9	7
'X	6	7	11	6	8	11	7	10
'X	6	12	10	4	8	11	8	3
'X	7	12	8	6	9	11	9	1
'X	11	8	7	10	11	10	6	9
'X	9	5	6	7	10	9	7	5
'X	10	5	6	4	9	9	6	11
'X	0	5	11	6	9	11	5	9

TABLE 3.1 Pattern matrix for the '8OX' data; eight features describe fifteen variants of each of the three characters '8', 'O', and 'X'

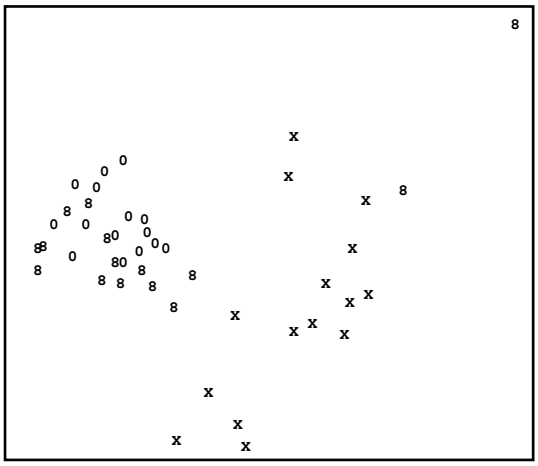


FIGURE 3.2 A 2D projection of the '8OX' data onthe first two principal components (from [Jain& Dubes 88]).

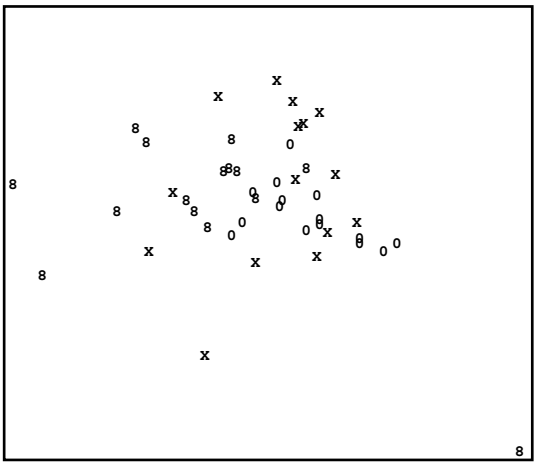


FIGURE 3.3 A 2D projection of the '8OX' data on the 3rd and 4th principal components [Jain& Dubes 88].

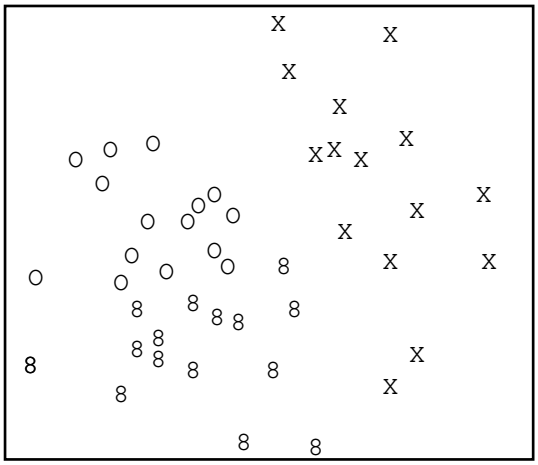


FIGURE 3.4 Euclidean projection of the '8OX' data by [Sammon 69] from [Jain& Dubes 88].

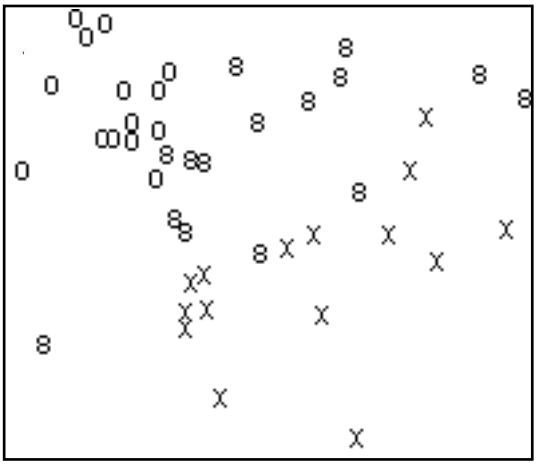


FIGURE 3.5 Manhattan projection of the '8OX' data by CAV, as presented in [Hagman 94].

We can however assume that every 6-dimensional car-car-relation can be expressed in terms of their distance in the pattern space; their *proximity index*. There are several ways of calculating a proximity index and [Jain & Dubes 88] give a good survey of some of these whereof it will suffice here to exemplify two of the so-called Minkowski metrics: the *Euclidean distance*, and the *Manhattan distance*[3]. In our example, we assume the following two patterns:

```
Car23=[4, 1, 9, 9, 2]     Car41=[3, 3, 10, 7, 7]
```

Then, the Euclidean distance would be:

3 The latter is also known as the *taxi cab* or c*ity block* distance.

$$\sqrt{((4-3)^2 + (1-3)^2 + (9-10)^2 + (9-7)^2 + (2-7)^2))} \approx 5.9$$

whereas[4] the Manhattan distance equals to:

$$|4-3| + |1-3| + |9-10| + |9-7| + |2-7| = 11$$

The final proximity matrix in our example will be a table with 60 x 60 indices but with the diagonal running from the upper left to the lower right corner having just zeroes (denoting distance null from any car to itself) and the indices in the upper right half will be repeated in the lower left half of the table. Thus, the table will contain (60x60-60)/2 = 1,770 proximity indices expressing dissimilarity as expressed by the chosen features.

Textual data is treated similarly. In such a pattern matrix each text is described with a (usually very large) number of features representing e.g. word roots. The pattern matrix may be normalized or not and a similar computation to obtain a proximity matrix is carried out as in the car example — the only difference being that this time the *similarity* is calculated; the values for the same features are better to add (or multiply) in this case. Suppose we have the following two patterns:

```
Text1={ 0.00, 23.12,  2.82,  51.40,  ...}
Text2={ 0.19, 11.95, 66.07,  62.21,  ...}
```

then, when adding, the proximity index could be:

	0	+ 11.95 +	2.82 +	51.40 +	...	(min)
or:	0	+ 23.12 +	66.07 +	62.21 +	...	(max)
or:	0	+ 35.07 +	68.89 +	113.61 +	...	(sum), (~mean)

... or whatever algorithm is preferred. Note the important possibility of assigning *weights* $w_1 ... w_n$ to the n features — maybe some word roots are considered to be of particularly heavy interest. The first alternative above then modifies to:

$$w_1*0 + w_2*11.95 + w_3*2.82 + w_4*51.40 + ...$$

This applies to both similarity and dissimilarity proximity matrices.

3.7.3 Dendrograms

Although a proximity matrix is less complex to survey than the underlying pattern matrix when comparing the patterns' internal relations, it will anyway soon grow harder as the number of patterns increases and it becomes quite impossible already when their number exceeds a handful.

The discussion which hitherto has dealt mainly with data processing will now also include graphic visualization of the result of this processing beyond proximity matrices. The first step is that of a *dendrogram* which, as its

[4] A note on the choice between Euclidean and Manhattan distances: In the first case the dimensions to which the features are associated are of the "same type" in the same way as the well-known *x*-, *y*-, and *z*-dimensions describing space are thought to be of the "same type". Philosophically then, each feature is associated with a unique dimension. In the second case each feature can be said to contribute to the distance within one single dimension which we could call "similarity". The notion of distance varies between different applications and that may influence the choice of representation and, to some extent, the final result.

greek name suggests, is formed like a tree. The single patterns or 'items' of the proximity matrix correspond to leaves and the more similar two patterns are, the closer they will sit together on the tree. FIGURES 4.2 and 5.3, below, exemplify this. TABLE 4.1 is interpreted as a proximity matrix where high values indicate a high degree of simularity between two consonants — simply because they proved to be more easily mixed up in an experiment.

p	—															
t	.229	—														
k	.432	.241	—													
f	.101	.057	.077	—												
T	.124	.079	.084	.423	—											
s	.052	.050	.063	.066	.157	—										
S	.038	.050	.047	.030	.048	.115	—									
b	.022	.013	.018	.046	.045	.024	.012	—								
d	.025	.022	.020	.025	.041	.031	.033	.058	—							
g	.013	.016	.030	.015	.039	.033	.021	.069	.342	—						
v	.016	.022	.020	.035	.040	.023	.020	.210	.059	.054	—					
D	.028	.016	.018	.032	.031	.026	.018	.145	.094	.120	.338	—				
z	.025	.023	.025	.018	.033	.035	.017	.055	.106	.139	.080	.161	—			
Z	.019	.017	.019	.007	.017	.022	.012	.027	.089	.125	.029	.033	.136	—		
m	.025	.022	.021	.016	.019	.017	.012	.038	.024	.032	.030	.034	.121	.016	—	
n	.017	.018	.020	.012	.018	.013	.011	.024	.032	.030	.022	.028	.016	.030	.151	—
	p	t	k	f	T	s	S	b	d	g	v	D	z	Z	m	n

TABLE 4.1 Symmetrized confusion matrix of consonants (from [Shephard 73]).

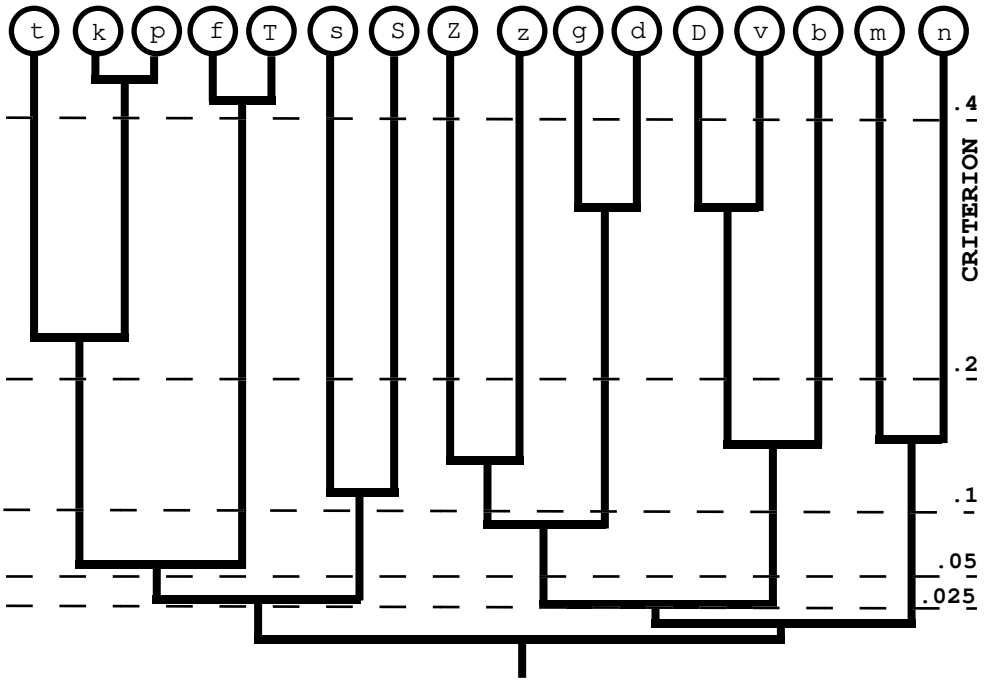


FIGURE 4.2 Hierarchical clustering analysis of the data in TABLE 4.1. Clusters at five levels are indicated by dashed horizontal lines (from [Shepard 73]).

There are different types of dendrograms and associated algorithms. A common trait is that they are based on *hierchical clustering*. Among the *exclusive* classification types which do not permit partly overlapping classes there are *extrinsic* and *intrinsic* subtypes of classification. The extrinsic types uses categories known a priori and tries to separate and label the studied ob-

jects in accordance with these. In our car example this would be to decide on car- (or pattern-) categories like 'French cars', 'Jeeps', or 'Luxury cars' first and then try to define and perhaps verify this a priori categorization in terms of their typical feature values. The intrinsic types of classification, on the other hand, only considers the feature values in all n dimensions when calculating the clusters but, nevertheless, the resulting clusters could then be compared to expected categories — to see if the latter are justifyable with respect to the supplied data and the obtained cluster configuration. What is more, the intrinsic types of classification are further divided into the "subsubtypes" of classification called *hiearchical* and *partitional*.

3.7.4 Hierachical clustering

The hierarchical types transform pattern matrices into a sequence of nested partitions. An *agglomerative* clustering algorithm starts with as many clusters as there are (underlying) patterns in the proximity matrix and each pattern represents a leaf of the tree-to-be. Then, the two "nearest" patterns are identified and their corresponding rows and columns are merged in the matrix and two leaves are joined by a node — a bifurcating twiglet in the tree — which from that moment will represent both of them (having feature values that are e.g. the mean values of its leaves) and so the procedure will continue until the whole proximity matrix is converted to a tree where all leaves are connected via nested chains of branches to the root. In the so-called *proximity dendrograms* the length of the branches reflects the distance between the two subtrees. FIGURES 4.2 and 5.3 are both examples of this. There are a variety of algorithms for clustering yielding different dendrograms but whose description would lead us too far into details for the purpose of this report. A *divisive* clustering algorithm works in the opposite order; it starts with the whole set of objects and divides them repeatedly into smaller subsets. We will come back to this topic under *Document clusters and I.R.* in section 3.10.2.

3.7.5 Partitional clustering

This type of clustering seems to be less intrinsic than hierachical clustering since although the method does not consider certain a priori defined categories when calculating the classes, it *does* consider an externally (user-) fed number of expected clusters. [Jain & Dubes 88] give the following description:

> The problem of partitional clustering can be formally stated as follows. Given n patterns in a d-dimensional metric space, determine a partition of the patterns into k groups, or clusters, so that the patterns in a cluster are more similar to each other than to paterns in different clusters. The value k may or may not be specified. [p. 90]

...but the dominant ideas about this method seems to expect the value k to actually be specified as it shows in the related literature. Since the philosophy of clustering is to "intervene" as little as possible, pre-establishing the number of clusters in a material one doesn't know at all seems quite brutal. However, [Clarkson 92], [Cutting & al. 92] and [Rose & al. 93] show situations where it is practical to permit this violation. To sum up, regarding comparison and choice of clustering algorithms, [Jain & Dubes 88] again, conclude:

> A theoretical comparison of clustering algorithms is not feasible because clustering algorithms are almost impossible to model mathematically in such a way that the models can be compared. Hierarchcal techniques are popular in biological, social, and behavioral sciences, whereas partitional techniques are used more frequently in engineering applications. ... There is no "best" clustering algorithm, so a user is advised to try several clustering algorithms on a given data set. [pp. 138, 142]

3.8 Projections

Sofar there has been little said about how the multivariate data are *projected* for visual examination of the result of the analysis. Before ending this section on structuring and classification of data, we now recollect the previous discussion on visualization to see how it relates to analyses of multivariate data. Part of the theory of multivariate analysis is actually dealing with projection of data. This is a special type of visualization — a matter which we will fully return to below.

Generally can be said that a projection maps a set of d-dimensional patterns onto a c-dimensional space where $c<d$. The following subsections will treat *linear projections* and *nonlinear projections* respectively. The value of c will be 2 in these sections but higher values are discussed further on, in section 3.9.2.

3.8.1 Linear Projections

In linear projections the x- and y-axes of the screen or paper at hand represent "dimensions" in the algorithm for multivariate analysis. Factor analysis, shortly characterized above, is related to cluster analysis but tries to reduce the number of features; [Jain & Dubes 88]:

> The type of linear projection used in practice is influenced by the availability of category information about the patterns in the form of labels on the patterns. If no category information is available, the eigenvector projection (also called the Karhunen-Loeve, or principal component method) is commonly used. [p. 26]

FIGURE 5.1 shows the steps of factor analysis where the final result is a linear projection. Another linear projection is shown in FIGURE 5.2 and its corresponding dendrogram in FIGURE 5.3.

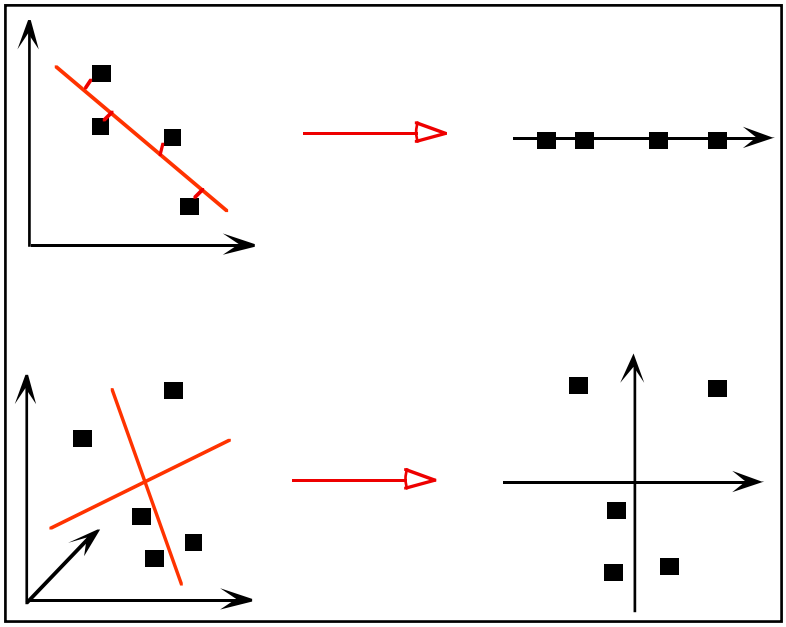


FIGURE 5.1 Illustration of a 2D⇒1D reduction and a 3D⇒2D reduction made possible by principal components analysis.

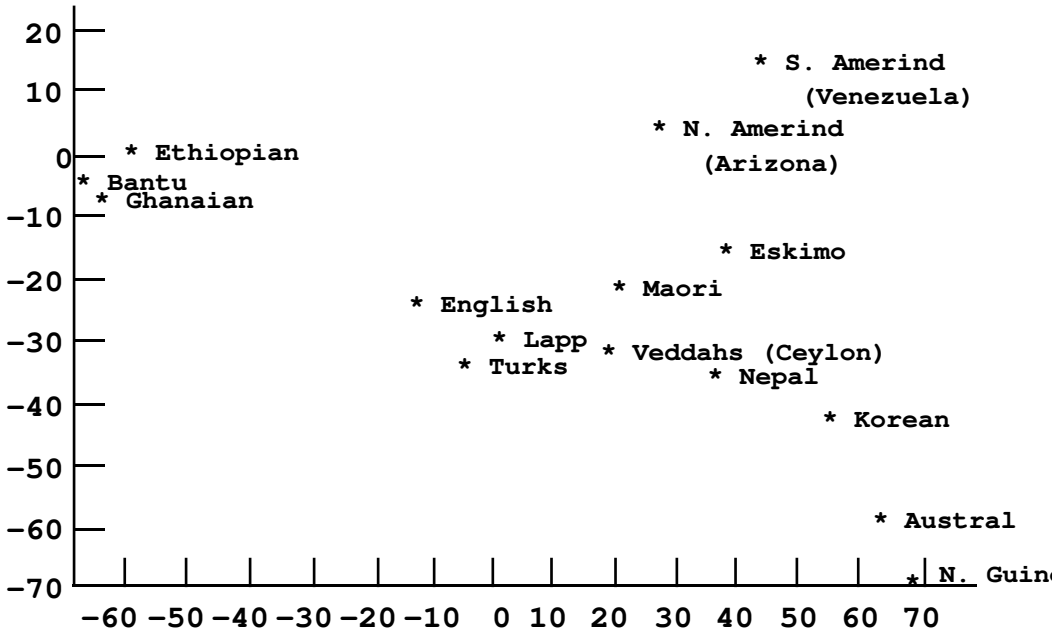


FIGURE 5.2 Chart of the first two principal components synthesizing frequencies of the five blood group systems A1A2B0, Rh, MNS, Fy, and Di (from [Cavalli-Sforza & Edwards 63]).

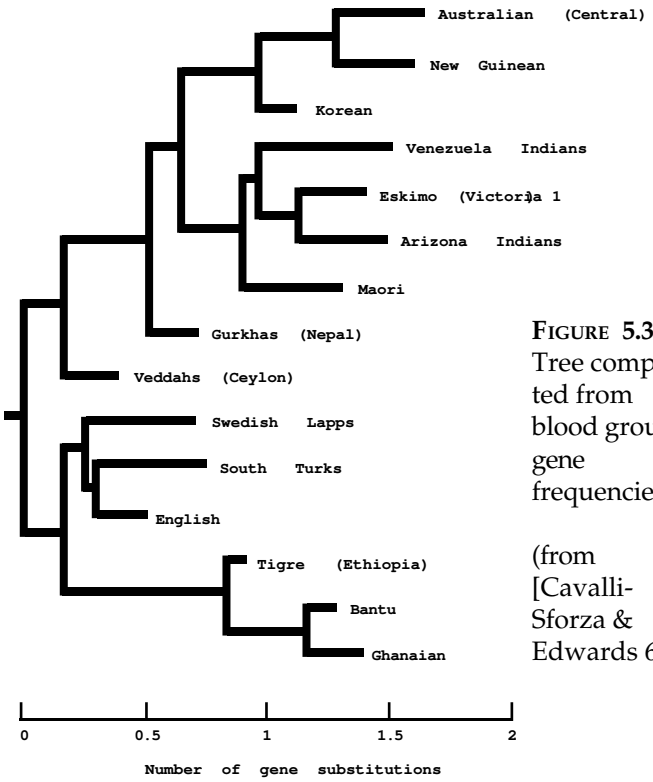


FIGURE 5.3 Tree computed from blood group gene frequencies

(from [Cavalli-Sforza & Edwards 63])

As linear projections, however, are connected to methods slightly outside the focus of this report, we pass on to the next section.

3.8.2 Nonlinear Projections

This very interesting class of projections is commented as follows by [Jain & Dubes 88]:

> The inability of linear projections to preserve "complex" data structures has made nonlinear projections more popular in recent years. By "complex data structures" we mean situations where the patterns lie on a curved surface. For example, the principal component method will not be able to obtain a good two-dimensional representation of a data set in which patterns lie along a helix in three dimensions. ... A change in the number of patterns requires recomputing the entire projection. Nonlinear projection algorithms are expensive to use, so several heuristics are employed to reduce the search time for the optimal solution. For eaxample, the best principal component projection could be used as the starting configuration for a nonlinear mapping algorithm. ... In the absence of category information, the goal is to project the patterns onto a low-dimensional space so as to retain as much structure as possible. For example, the goal could be to maintain all the interpattern distances. [p. 37]

Among the methods for nonlinear projection there is one group of particular interest here: the so-called *iterative* procedures. Their algorithms can be made to both cluster and visualize the data at the same time — like a moving picture. Methods similar to those proposed by [Sammon 69] start with a configuration of n points (representing the patterns) in a c-dimensional space and then move the points by a gradient descent procedure so as to minimize a measure of "stress" between the interpattern distances in the

feature space and interpoint distances in the configuration. Thus, the position of every point in the configuration is reconsidered, trying to approach similar points and withdraw from different ones. FIGURES 3.4 and 3.5 show two possible results of this method. This is a very complex ballet of repulsion and attraction which also requires quite a lot of processing time and memory. In spite of this, the method has been widely adopted and in recent years these problems have become far less limiting due to rapidly developing computer hardware. Further, since the nature of this problem is heavily parallel, impressive efficiency is gained when adapted algorithms run on parallel computers [Rasmussen & Willett 89].

3.9 Visualization — More Examples

As stated above, geographical maps and *scatter diagrams* belong to the visualization types $E_{[2]}{}^{S}$ and $E_{2}{}^{S}$, respectively. Projections on two dimensions seem to be the most common type of visualization and maybe this is due to ancient traditions of representing alphanumerical data on "flat" media such as clay tablets, rolls, and books — and most recently — computer screens. Media with a higher dimensionality in space has been comparatively little used for conveying information other than that contained in sculptures and other pieces of art. Probably this is explained by the difficulties and costs connected to manufacturing and distribution of 3D models en masse and the ease by which 3D phenomena are simulated and interpreted on the much more manageable 2D media.

3.9.1 2D Projections

On geographical maps the x- and y -axes are almost exclusively related to (although not necessarily aligned with) the earth's longitude and latitude coordinates. The phenomenon visualized on maps are thereby related to the geography and the way it looks is compared with the way other things look on the map — e.g. areas of certain nature types or political or cultural units or features like those visualized in FIGURES 1.1 and 1.2. The purpose of projecting data of a phenomenon onto a map can simply be to show from which places it originates without trying to *explain* the data by the geographical features on the map.

An example of this is a town map where vacant houses and flats are marked (see FIGURE 6). Such a map has the very practical purpose of relating points to a geographical surrounding not to explain *why* but *how* things are related in that 2D space. On a scatter diagram with labelled axes the positions of the points are also explained *why* they have had such a distribution by referring to the chosen dimensions represented by the axes and the values upon these. On scatter diagrams with no labelled axes the points are not related to any external dimensions but to their own geometrical configuration. The former of these two types of scatter diagrams could be used for the outcome of a factor- or principal component analysis, as shown in FIGURES 5.2, 2, 3.2, and 3.3, whereas the latter could be used for the outcome of a nonlinear graphical cluster analysis as shown in Figures 3.4 and 3.5.

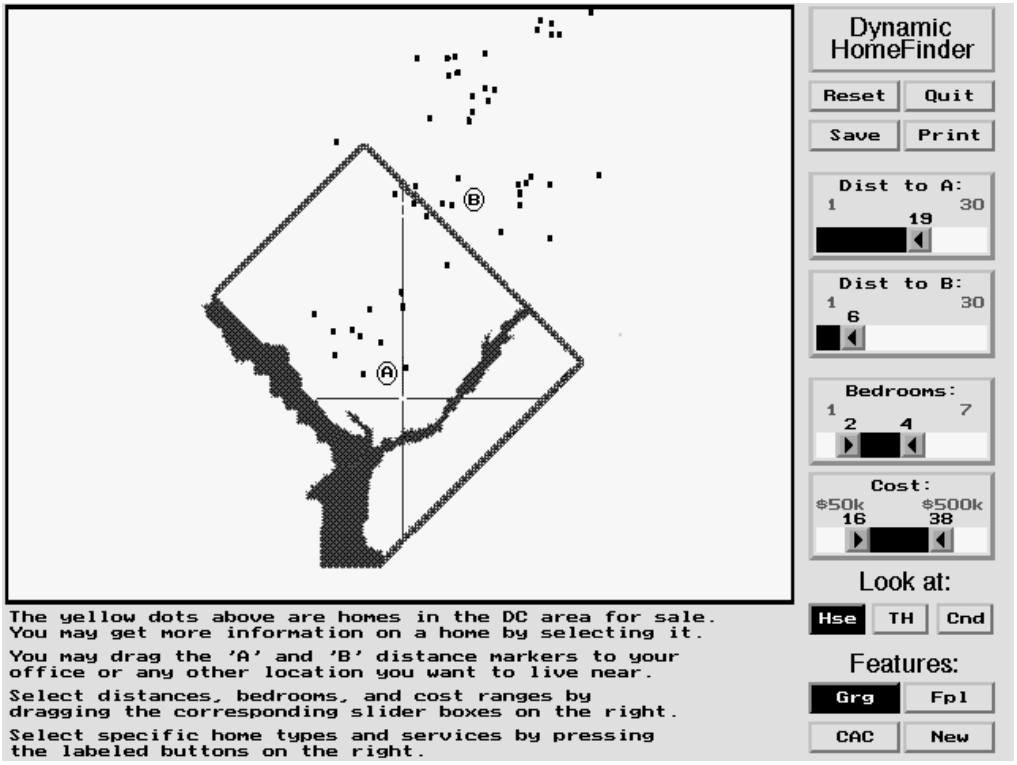


FIGURE 6 In the Dynamic HomeFinder query system each point satisfies the query described by the sliders for Location, Cost, number of Bedrooms, home type (House, TownHouse, or Condominium), and buttons (Garage, Fireplace, Central Air Conditioning, or New Construction). The points of light can be selected to generate a detailed description. (From [Ahlberg & Shneiderman 94])

3.9.2 3D and hD Projections

In [Follin 92a, 92b] the essentials of visualization and in particular the 3D-projecting type are presented. In addition they give an introductory information about computer graphics:

> In a 3D visualization, data are represented as colored objects ... in a three-dimensional space. ... This image can be put on photographic film, paper or videotape if the necessary peripherals are present. Animations (sequences of images, usually put on a videotape) are commonly used to show the evolution of data as time or some other variable changes. ... One of the goals of 3D visualization is to produce realistic images. Data should appear as though they are actual objects, existing at the desired location, illuminated by light sources, and viewed by the user. Adequate realism is essential so that the viewer will perceive the spatial relationships among the objects. [Follin 92a, p. 338]

The impression of depth can thus be accomplished by perspective, lightening and movement. Two or more 2D images in time renders a 3D phenomenon. The sofar poor and staccato performance of statistical packages is still far from what we would call an *animation*. A note from [Brodlie & al. 92] could suggest an improvement of this:

> In traditional animation, a large number of images, or cells, are hand-drawn and then filmed one frame at a time on a special movie camera. With the advent of computer animation, animators are only required to draw keyframes. The computer is then

> instructed to interpolate to find the frames between the key-frames. In scientific visualization, a similar process is often used, whereby the user specifies the situation for key frames in the sequence to be generated and the computer interpolates between them. [p. 69]

Another technique is to produce two slightly different pictures of the data space as if they were views from two angles close to each other. This is to imitate the 'parallax' of human vision — the unique view each eye receives because of its physical separation from the other. By looking simultaneously at the two pictures — by crossing one's eyes, looking parallelly instead of focussing, or by using some gadget — a *stereo view* is obtained. In this case, two 2D images in space renders a 3D phenomenon.

The more dimensions we can utilize when projecting a phenomenon of higher dimensionality, the better — or are there limits as to what can be easily interpreted and understood? So-called 'volumetric data' are 3D and per se not trivial to visualize on a 2D medium. Nevertheless, [Foley & Lane 91] propose:

> Although there are many methods for visualizing volumetric data, most of these methods can only render one set of volumetric data at a time. By rendering several volumetric data sets simultaneously, relationships among the data sets can be visualized, which may be difficult or impossible to visualize when the data sets are rendered individually. [p. 218]

The authors present their method and results on color plates which are beautiful to look at but may be hard for uninitiates to understand. Assuming that visualizing several sets of volumetric data on paper is equivalent to a 4D⇒2D reduction, then, provided that the viewer knows the original data well, he may enjoy the mere presence of a fourth dimension on a flat surface as much as he enjoys new insights of the phenomena.

The final statement above also goes for the invention presented in [Alpern & Carter 91]:

> A hyperbox is a 2-dimensional picture that corresponds to an n-dimensional box. It has a face for every pair of dimensions of the original box. It can be used to represent multidimensional arrays and to display simultaneously all 2-dimensional orthographic projections of n-dimensional data sets. It can also be used to help organize the search for a collection of dimension-reducing transformations that preserve interesting properties of the data. [p. 133]

Suggestive creations that however require knowledge of the data and maybe some training in order to be utilized optimally.

In [Young & Rheingans 91] a tool for high-dimensional data visualization is presented which projects data of any dimensionality onto a 3D space rendered very well on a 2D computer screen. The projections show the patterns as point-like objects in space and the interpretation is supported by possible dimension axes, grids, colors representing data categories, brightness cueing

or dimming to reinforce the impression of close and remote points. The "cloud" of points can be moved and spun interactively by the user along with its axes in space and the performance is very high. An unnecessary drawback of the version whose output was exemplified in this article were the clumsy letters (used for the labels of the points and axes) which somewhat impaired the overall impression.

Visualization in a *real* 3D space is also possible although the required technique of course goes beyond the ordinary computer screens (and budgets). In [R. D. Williams & al. 92] a prototype similar to 3D hologram displays is introduced for data projection in a 3D space:

> The need for true 3D volumetric displays derives from the fact that the human visual system has evolved to perceive and comprehend the world in three dimensions. The Texas Instruments OmniView™ display technology provides the capability to display computer graphics images in true 3D. Viewers can see the display from all angles and change views simply by walking around the display volume. The OmniView™ device uses lasers of three different colors to project images on a moving surface sweeping through the 3D cylindrical display volume. [p. 99]

This advanced technology operating with *voxels* (the *vol*umetric counterpart to pi*xels*) offers unique possibilites of direct team perception of real-time rendering of volumetric or multivariate data.

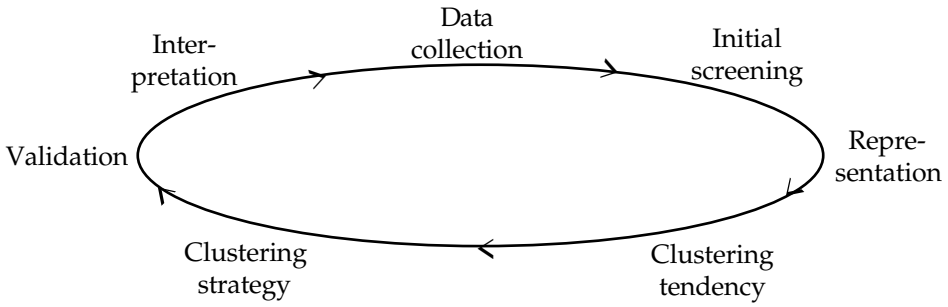
The recent development and spread of data helmets providing a *cyberspace* often accompanied by *data gloves* is another, probably cheaper alternative of high-performance 3D visualisation although it may not yet permit the image to be updated correctly as the user moves his head. It works with stereopsis as described above and the user sees his own hand represented by the data glove as a hand or a mechanical tool in the *virtual reality*. If this is to be experienced simultaneously by more persons, the images and positions of the hands would have to be initially well calibrated and then a massive synchronical updating of everybody's hand movements and data manipulation would have to be carried out. We are likely to see very interesting high-tech hybrids in the field of 3D projection methods for data visualization and exploration in the near future.

3.10 Exploratory Data Analysis

In this section we continue the discussion about applications where multivariate or high-dimensional data sets must be made intelligible by means of e.g. cluster analysis and advanced visualization techniques.

3.10.1 Methodology

Exploratory data analysis for scientific research in general is a process and [Jain & Dubes 88, pp. 135-137] see it as a loop where cluster analysis is one of its central components. Of course, the user's notion of the data and some intuition of a theory that describes them is fundamental. This exploratory loop has the following seven steps:



Data collection (as hinted above, already influenced by a notion or theory of the phenomenon) asks for a well-chosen methodology, patience and due precision. Later findings from data analysis may suggest ways to improve the data collection.

Initial screening partly coincides with what was treated in section 3.4.1. Normalization and equivalation might reveal redundant (exactly covariant) features that can be removed to ease the complexity of the data.

Representation includes the choices of proximity index, projection, and, if found suitable, an MDS run of the data. The outcome of this step is a pattern matrix or a proximity matrix.

Clustering tendency raises the question whether the data clusters at all or if they seem to be randomly scattered on the projection. Do the features suggest cluster analysis or are other types of analysis more relevant?

Clustering strategy contains the chioce between hierarchical or partitional procedures. The choice of parameters, algorithm and presentation of results are considered here. Shall both patterns and features be clustered?

Validation
"The prudent and enlightened validation of clustering results is the essential step that changes a qualitative analysis into hard evidence" — quotes part of the comment on this step [Jain & Dubes 88, p. 137].

Interpretation is the step where one tries to integrate the result with what was known before and where conclusions are drawn, discoveries are made and new ideas come up.

3.10.2 Applications II

Now as we are equipped with an adequate terminology after having passed the more theoretical parts separating the first section on applications (2.2) from this one, we will continue the discussion about what is currently done in the broad field of data exploration and visualization. The idea is to read about applications for cluster analysis and similar techniques with methods for visualization in mind and vice versa. This interbreeding of disciplines complements specialization and together they lead us to new discoveries.

Science

It is worthwhile again to quote [Jain & Dubes 88] which give a comprehensive description of application areas for cluster analysis:

> Clustering techniques were first developed in biology and zoology to group similar animals and plants to construct taxonomies. The need to organize vast amounts of data into "meaningful" groups, clusters, categories, partitions, or classes in several scientific

> disciplines has made clustering a valuable tool in data analysis. A variety of entities or objects have been clustered, including mental diseases, land-use patterns, rock samples, finger prints, training methods, stars, consumers, prose, and images. In several of these applications it is not extremely important to identify the exact number of clusters or the correct membership of each pattern into a cluster. Often it is enough to group the objects in a reliable and parsimonious manner so that the underlying physical, biological, or evolutionary process can be understood or learned. [p. 240]

In the first section on applications above some instances of multidisciplinar collaboration were exemplified. They dealt with the remotest known traits of man and his language. Intuitive dimensions to refer the collected data to have been the earth's north-south and west-east axes. Innumerable maps showing the covariance or complementary distribution of two or more genetic and/or linguistic traits are encountered in the literature and they often give quite a good "picture" of the phenomena. In some cases this research can however still be improved — e.g. by supporting the researcher in the phase of selecting features to group together and visualize. Every serious researcher is aware of the fact that an obtained cluster of features does *not* contain any information about possible internal *causal* relationships among them. As expected, causal relationships are easier to individuate inside each discipline but in the examples referred to, the distribution of some linguistic and genetic traits are interpreted as an indication of common causes. Common causes of this kind are mostly connected to mass migration but also to the mobility of minor populations such as commercial contacts along the coasts of the continents. In [Brosnahan 61] though (fragmentally illustrated by the FIGURES 1.1 and 1.2), the distribution of some features did inspire speculations about a few genetic traits *causing* linguistic traits[5].

Document clusters and I.R.

As for automatic document classification and retrieval support, briefly commented above in the first section on applications, [Willett 88] gives a comprehensive summary of the research in the field. Two main justifications for document clustering are referred to: (1) it is a way of improving the *efficiency* of best match searching in information retreival systems — the documents can be grouped and ranked in probability order before they are individually matched against the query, and (2) the *effectiveness* can be improved in that a query may be matched against entities on other levels, i.e. whole groups of documents (by looking at their common "mean profile") and thus not only against single documents. Two diverse types of calculating the interdocumentary distances are given; (1) the one based on *citations* where the number shared relatively or absolutely by two documents is a measure of similarity degree — applied especially in studies of sociology of science; and (2) a proximity index based on shared *index terms*, *keywords*, or *thesaural terms* that give an indication of the content of the documents and

---

[5] The theory outlined there may be considered daring, especially by people who still feel genetics a taboo in connection with human sciences. This is probably an overreaction to the gruesome ideas and deeds of WWII where studies of this kind were made with unsane *values* and *purposes*. The serious scientist builds his theory carefully but if relevant data is not collected because of *taboos* the final result can never be claimed to be complete. This is a note on methodology; the phrase "Statistics can be lying!" is frequently uttered (often by people who do not *want* to study statistical material and draw their own conclusions. A reply to this is "Yes, indeed, but you can lie with *any* method or action. The question of *intent* is not answered by its *means*."

which has been discussed above. The two clustering methods adopted are the nonhierarchic and hierarchic ones. The first one tries to separate the documents into a (often user-supplied) number of well separated clusters. The relation between these clusters is not considered and that together with the (also often user-supplied) values of min/max cluster sizes and 'threshold values' for document-cluster similarity which are used to control the processing increases the risk of arbitrariness of this method. Recalling the differences between *agglomerative* and *divisive* strategies described above, let us see what this choice implies in document clustering [Willett 88, p. 144]:

> Divisive clustering methods generally result in *monothetic* classifications in which all of the documents in a given cluster must contain certain terms if they are to belong to it; such clustering methods are of less use for applications in document retrieval than *polythetic* classifications, where each document in a cluster has some, or many, terms in common with each of the other documents in that cluster but where there are no specific terms required for cluster membership.

...and he continues by stating (quite expectedly) that...

> Polythetic hierarchic classifications are normally generated using the agglomerative methods, which are by far the most popular type of clustering procedure. They were developed primarily for applications in the life sciences, where the hierarchic classifications could be compared with traditional biological taxonomies in which specimens are grouped into species and species grouped into genera.

Finally, once the tree of documents is built and we would like to search it, we formulate a query — either by listing keywords or by feeding a document that we have read and found interesting and whose closest neighbours we are looking for. The query can the be matched along the tree either *top-down*, starting from the root, or *bottom-up*, i.e. starting from the leaves. The top-down method risks missing the right path if the matching key terms of the documents "drown" among all the other terms of the documents belonging to the same branch at the trunk of the tree. Obviously this risk increases if we have been too permissive as to which words are allowed to represent the contents of the document (section 3.4.1). A bottom-up search, on the other hand, will be very expensive and almost invalidates the whole idea of clustering the documents at all. A smart solution to this is to compare the query to the *bottom level clusters* and procede upwards until the desired size of cluster or the minimum degree of match is reached.

In [Enser 85] monograph titles, tables of contents, and back-of-the-book indices were used as a basis for a content profile of each document. In that study it was concluded that at least some of the automatic classifications seemed to be superior to the conventional library classifications.

We conclude by quoting [Chalmers & Chitson 92], who report on their appealing research:

> Bead is a prototype system for the graphically-based exploration of information. In this system, articles in a bibliography are represented by particles in 3[D]-space. By using physically-based modelling techniques to take advantage of fast methods for the approximation of potential fields, we represent the relationships between articles by their relative spatial positions. Inter-particle forces tend to make similar articles move closer to one another and dissimilar ones move apart. The result is a 3D scene which can be used to visualize patterns in the high-D information space. [p. 330]

Networks

A third type of application could be generally called "network design." The networks we have in mind are all more or less electronical. Above, a study of the computer correspondence network was mentioned and we can think of similar studies where communication — the flow of information or people/resources — not only are made in order to locate interesting objects and access them but where the intention is that of optimizing the very structure of the medium by which the network is realized. If the network does not yet exist, an excellent way of planning its coming structure and dimensions is to study the points which it will connect and the characteristics and desiderata of their relations. One example of this (although not always electrical) is traffic planning. Another is how to best design a net for telecommunication. A third example concerns the architecture of both hardware and software in computers — modules that will communicate most intensely should be situated and linked accordingly.

Cluster analysis and *neural networks* (see [McClelland 88] or [Ritter & Kohonen 89]) have a most intriguing relation. The two techniques are briefly compared in [Hagman 94] and this comparison[6] will probably be deepened in the future.

3.11 Human-Computer Interaction

We have described exploratory data analysis both for scientific research and the layman's search for information and resources as a cyclic process where cluster analysis could be one component. It is relevant to visualize certain steps of this process and the visualization should enhance the feeling of a smooth human-computer interaction. Important issues of visualization are discussed quite thoroughly in [Brodlie & al. 92] where they are presented in two parts. The first part considers the user and is subdivided in cognitive, perceptual, and organizational issues, plus how to make the user feel comfortable. The second part deals among other things, with system flexibility and dialogue management. The conclusions of human-computer interface include the following three suggestions [p. 210]:

---

[6] For those who "can stand the discomfort", or, alternatively: "enjoys the peculiarity" of the fact that although both these techniques *work* in many existing applications, the theories explaining their performance and capabilities are still in the process of construction. In both cases we must relax our *analyst* view and adopt a *synthesist* one when we approach the "holistic" or "parallel" nature of phenomena such as the human brain (and psyche), and self-organizing systems. One of the greatest difficulties in taking this step is to overcome one's *attachment* to old, familiar ways of thinking. We all know that science evolves (and so it does rapidly) and sometimes it is presented with phenomena of a higher complexity. With the arrival of computers (per se a result of extraordinary science synthesis) we are now able to examine very complex phenomena which were previously separated from our reasoning mind by its pure complexity. In history, the arrival of e.g. writing systems is another example of instrumental enforcement for our reasoning capacity. Closely related to these instruments are their theoretical counterparts and not seldom there is no sharp border between them.

> Designers of visualization tools should take greater account of research in cognitive science when devising frameworks and interface concepts for their systems. Representation and manipulation of multi-dimensional data is one specific area where this could be beneficial.
>
> Visualization systems can change fundamentally the way that problems are solved and information is communicated. Organizations need to plan consciously for the introduction of such systems.
>
> New styles of interaction are important and attention should be paid in these. For example, direct manipulation of user models is one area where visualization systems may be fundamentally different from traditional graphics or windows system appoaches. Studies are required to find what kinds of interaction are most useful for visualization, and to devise software systems which can support these.

New styles of interaction are constantly presented and discussed. The so-called *dynamic queries*, presented in [Ahlberg & al. 92], let the user explore a database which is filtered in accordance with the interactive query parameters, rapidly adjustable with sliders or buttons. In [Ahlberg & Shneiderman 94] these concepts are combined with 'starfield displays' which are 2D projections of the that part of the interrogated database satisfying the current query conditions. The projections are structured in two dimensions and are constantly updated as the user modifies the query with the slides or buttons. Each data item is represented by an object in the field and is "clickable" for information. Recent research [Ahlberg & Truvé 94] aims at automatically analyzing the feature types and value ranges of an imported (or interfacing) database in order to adjust the slide ranges or choose buttons or other widgets. For related work on visual querying, see also [Keim & Lum 92], [Clarkson 92], and [Spoerri 93].

The importance of user-controlled modification of the visualization and that the user sees a smoothly changing picture as a feedback of the parameter settings is stressed in [Young & Rheingans 91]. There, the difference between *inter*action and mere action is suggested by the following definitions:

> We consider only *dynamic* movement, movement which is smooth in real time and which is controlled by a data analyst through graphic, high-interaction, immedeately effective actions. Here, the computer creates only one frame in the sequence before interrogating the analyst to see how the next frame in the sequence should be produced, with the creation-interrogation cycle occurring several types per second. Dynamic movement is in contrast to *animated* movement, in which the computer creates a series of frames and then presents them in sequence to the viewer who passively views the "movie." [p. 99]

A last comment here from [Brodlie & al. 92] leads us into the next section:

> Potentially the complexity of current and projected systems can result in incredibly complex user interfaces. We need to ensure that the potentially conflicting aims of power and ease of use are elegantly accommodated. Some visualization techniques become effectively unusable if the system is unable to react within a certain time. These approaches will require orders of magnitude increase in computational power before their true potential can be realized. [p. 115]

3.12 Hardware

The purpose of this section is merely to be a memorandum of the hardware requirements than a description of them. This report is dealing with real-time database consultation, complex cluster analysis, parallel programming, connection machines, visualization with smoothly moving images, interactive user interfaces, and (still today, quite) extraordinary externals such as data helmets and -gloves — not to mention large, laser-beaming cylinders for true 3D volume displays. Currently the computer performance/price ratio (on the PC market) is increasing by a factor of 4 every 18 months and that rate will be required to last yet for some time in order to realize some of the ideas explicitly and implicitly communicated in this report.

4 Tools

In addition to the tools mentioned above, especially in section 3.9.2, this section will examplify some types of tools, available on the market or existing just as prototypes which try to meet one or more of the requirements for data exploration addressed here. At first, an attempt was made to list a number of tools along with their description. This was succefully done in [Brodlie & al. 92] but since the number of software products for the purposes discussed in this report increases and their features change so rapidly, this work already today risks to be out of date.

4.1 Types of Visualization Tools

The technical report [Lantz 92] documents a study "to survey existing graphics packages for their ability to do 'bread-and-butter' graphics — i.e. to do the unglamorous kinds of visual data exploration which are the most crucial in day-to-day scientific research" [p. 1]. Thirteen different commercial and public-domain packages were examined. This study states that "scientific visualization is a rapidly evolving speciality with new products entering the market at a tremendous rate" but also that "tools for visualizing complex and multi-dimensional data would be extremely useful to large numbers of scientists, particularly if the tools were easy to use and readily available."

As for visualization, [Brodlie & al. 92, pp. 176-179] divide visualization systems in three main categories:

> *Graphics Libraries & Presentation Packages* This is the traditional method for creating new ways to view and analyse data. The libraries interface directly to graphics hardware or provide graphics functionality in software. The user has to supply nearly all the components to support the application: the main program, the user interface, data handling and geometry mapping.

Turnkey Visualization Systems For the systems in this category, the user provides the data and instructions to the main program. The system supplies the main program and rendering and usually has an attractive user interface. The user does not have to program these packages and can obtain results very quickly. However in general the packages cannot be modified or extended and therefore often only provide a part of the solution a user requires, hence the use of the term "turnkey".

Application Builders These offers a series of modules linked by interfaces which are connected interactively at runtime. They combine features from the previous categories by providing "turnkey" solutions for individual parts of the program and the flexibility to customise the final solution adopted. The supported modules can be replaced by user-written modules as required, providing they conform to the data input/output interface conventions. ... Applications are generally constructed by a mouse-driven interface, manipulating icons on the screens and linking them with data paths. Once the required modules have been connected and built the program can be executed.

Technologies that underly visualization, such as hardware, graphics software, user interface toolkits, image tools, and database management tools are considered in their study as well as the more "internal" features of representative software products, described with a set of parameters. Important parameters are cost, availability, procedural library, import/export formats, data models, and visualization techniques. The study does not specify the procedural library modules in the otherwise extensive and detailed list which is presentated along with the statement "ideally a visualization system accepts multidimensional data, but at present we have to relax this requirement for inclusion." [Brodlie & al 92., p. 175]

The application builders are inviting. Though, most of them cannot yet run on personal computers. More about visual programming and visualization software techniques is found in e.g. [C. J. Williams & al. 92], [Schroeder & al. 92], and [Kochhar & al. 91].

4.2 Putting it All Together: Integration

As noted in section 3.9.1, data exploration is a process involving many steps, where calculation and visualization are two. It would of course be easier and more time-saving to execute the steps of this process if all the required modules associated to these steps were integrated in one system. Among the attempts to develop systems of this kind, e.g. [Lang & al. 91] and [Smith & al. 91] can be mentioned. Such a system has to meet a number of very requiring conditions concerning data imput/export format, the management of data, interactivity, supports for user-supplied modules, and visualization techniques. These requirements concern both hardware and software. Not seldom the system will work against a database and as the latter often is relational, intermediate data representation is needed in order to handle multidimensional data or hierarchical data structures.

A distinction may however be made between *scientific* and *public* data exploration. In the first case, not only the data are explored but also the *meth-*

ods by which they are explored. In situations of public data exploration (as intended here) the exploration method could be stable; job searching in employment offices, computerized housing consultation, and information retrieval support for libraries and other collections of texts could be carried out sufficiently by modifying e.g. a handful of slides associated to an underlying database and a 2D scatter plot — everything designed by the creators of the system.

This report is actually written during the first stage of a project[7] aiming at the integration of moduled for cluster analysis, powerful visualization, and easy-to-use human-computer interfaces. The first and part of the second of these modules are further described in [Hagman 94]. This also applies to FIGURE 7 which is attached here just as an illustration of a (still simple) integrated clustering and visualizing system

5 Summary

This report started by giving an orientation of some types of applications which largely benefit from tools supporting the combination of cluster analysis, visualization, and easy-to-use human-computer interaction and then discussed some of these modules more or less in detail in order to explain, inspire (and maybe even provoke) the reader. Although computer-supported data exploration and visualization is quite a new area, especially in the world outside universities and a few industries, its every-day need is likely to stimulate continued, fast development both in methods and the number of different applications in the very near future.

Acknowledgements

This report was written with the support of NUTEK, grant no. 53-93-2760 which finances the research project *Information Exploration and Visualization trough the Cluster Analyzer & Visualizer and Dynamic Queries*. I would like to thank especially the other members of this project group, Christopher Ahlberg, Jens Allwood, and Staffan Truvé, for valuable comments on previous versions of this report.

[7] See *Acknowledgements*, below.

Page 34 is written after page 39.

References

Ahlberg, C., C. Williamson, and B. Shneiderman (1992)
"Dynamic Queries for Information Exploration: An Implementation and Evaluation",
Proceedings ACM CHI'92: Human Factors in ComputationalSystems

Ahlberg, C., B. Shneiderman (1994)
"Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays", To be published in *Proceedings ACM CHI'94: Human Factors in Computational Systems*.

Ahlberg, C., S. Truvé, 1994. "Exploring Terra Incognita in the Design Space of Query Devices", submitted UIST'94

Alpern, B. & L. Carter (1991)
"The Hyperbox" in *Visualization '91* (pp. 133-139)
Conference Proceedings IEEE Computer Society Press

Ammerman, A. J. and L. L. Cavalli-Sforza (1984)
The Neolithic Transition and the Genetics of Populations in Europe
Princeton University Press

Anderberg M. R. (1973)
Cluster Analysis for Applications, Academic Press, Inc., New York

Bartell, B. T., G. W. Cottrell, R. K. Belew (1992)
"Latent Semantic Indexing is an Optima Special Case of Multidimensional Scaling", in *SIGIR '92, Proceedings from the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* ed. by N. Belkin, P. Ingwersen, and A. Mark Pejtersen IEEE Computer Society Press (pp. 161-167)

Brodlie, K.W, L. A Carpenter, R. A. Earnshaw, J. R. Gallop, R. J. Hubbold, A. M. Mumford, C. D. Osland, P. Quarendon (1992)
Scientific Visualization - Techniques and Applications, Springer-Verlag

Brosnahan, L. F. (1961)
The sounds of Language W. Heffer & Sons Ltd, Cambridge, England

Carmichael, J. W., A. George, and R. S. Julius (1968)
"Finding Natural Clusters" in *Systematic Zoology*, Vol. 17 (pp. 144-150)

Cavalli-Sforza, L. L. and A. W. F. Edwards (1963)
"Analysis of Human Evolution" in *Genetics Today, Proceedings of the XI International Congress of Genetics*, (ed.) S. J. Geerts, Pergamon Press (pp. 923-933)

Chalmers & Chitson (1992)
"Bead: Explorations in Information Visualization"
in *SIGIR '92, Proceedings from the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* ed. by N. Belkin, P. Ingwersen, and A. Mark Pejtersen IEEE Computer Society Press (pp. 330-337)

Clarkson, M. A. (1992)
"The Information Theater" in *BYTE*, Nov.'92 (pp. 145-152)

Cutting, D. R., D. R. Karger, J. O. Pedersen, and J. W. Tukey (1992)
"Scatter/Gather: A Cluster-based Approach yo Browsing Large Document Collections", in *SIGIR '92, Proceedings from the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* ed. by N. Belkin, P. Ingwersen, and A. Mark Pejtersen IEEE Computer Society Press (pp. 318-329)

Dubes, R. and A. K. Jain (1980)
"Clustering Methodologies in Exploratory Data Analysis", in *Advances in Computers* Vol. 19 ed. by M. C. Yovits, Academic Press Inc.

Edwards, A. W. F. and L. L. Cavalli-Sforza (1966)
"A Method for Cluster Analysis"
in *Biometrics*, Vol. 21, No. 2 (pp. 362-375)

Enser, P. G. B. (1985)
"Automatic classification of book material represented by back-of-the-book index" in *Journal of Documentation* No. 41 (pp. 135-155)

Fagan, J. (1987)
Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods, PhD thesis, Dept of Computer Science, Cornell University.

Foley, T. A. and D. A. Lane (1991)
"Multi-Valued Volumetric Visualization", in *Visualization '91*
(pp. 218-225), Conference Proceedings IEEE Computer Society Press

Follin, S. E. (1992a)
"The Essentials of Scientific Visualization: Basic Tools and an Example" in *Social Science Computer Review* 10:3 '92 (pp. 337-354), Duke University Press

Follin, S. E. (1992b)
"The Essentials of Scientific Visualization: Basic Techniques and Common Problems" in *Social Science Computer Review* 10:4 '92 (pp. 337-354), Duke University Press

Forgas, J.P. (1982)
"Multidimensional Scaling: a discovery method in social psychology" in *Key Texts in Multidimensional Scaling*
P. M. Davies and A. P. M. Coxon (eds), Heinemann Educational Books

Furnas, G., S. Deerwester, S. Dumais, T. Landauer, R. Harshman, L. Streeter, and K. Lochbaum (1988)
"Information Retrieval Using a Singular Value Decomposition Model of Latent Semantic Structure"
Proceedings of 11th International Conference on Research and Development in Information Retrieval (pp.465-480)

Good, I. J. (1977)
"The botryology of botryology", in *Classification and Clustering*,
J. Van Ryzin (ed.), Academic Press, Inc., New York (pp. 73-94)

Grefenstette (1992)
"Use of Syntactic Context to Produce Term Association Lists for Text Retrieval", in *SIGIR '92, Proceedings from the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* ed. by N. Belkin, P. Ingwersen, and A. Mark Pejtersen IEEE Computer Society Press (pp. 89-97)

Hagman, J. (1994)
The Cluster Analyzer & Visualizer — a Prototype for Data Exploration SSKKII Technical Report #5, University of Göteborg / Chalmers University of Technology

Hofstede, G. (1991)
Cultures and Organizations, McGraw-Hill Book Company

Jain, A. K. and Dubes, R. C. (1988)
Algorithms for Clustering Data
Prentice Hall Advanced Reference Series

Keim, D. A., and V. Lum (1992)
"Visual Query Specification in a Multimedia Database System"
in *Visualization '92* (pp.194-201), Conference Proceedings,
IEEE Computer Society Press

Kochhar, S., M. Friedell, and M. LaPolla (1991)
"Cooperative, Computer-Aided Design of Scientific Visualization"
in *Visualization '91*, (pp. 306-313), Conference Proceedings,
IEEE Computer Society Press

Lang, U., R. Lang, and R. Rühle (1991)
"Integration of Visualization and Scientific Calculation in a Software System" in *Visualization '91*, (pp. 268-274), Conference Proceedings,
IEEE Computer Society Press

Lantz, S. R. (1992)
An Evaluation of Interactive, X-window-capable Software for Scientific Visualization, Technical Report CTC92TR101,
Cornell Theory Center, Cornell University

Lewis, D. and W. Croft (1990)
"Term Clustering of Syntactic Phrases", *SIGIR-90*, (pp. 384-404)

Lewis (1992)
"An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task", in *SIGIR '92, Proceedings from the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* ed. by N. Belkin, P. Ingwersen, and A. Mark Pejtersen IEEE Computer Society Press, (pp. 37-50)

Lin, X. (1992)
"Visualization for the Document Space"
in *Visualization '92*, Conference Proceedings,
IEEE Computer Society Press, (pp. 274-281)

Mourant, A. E., (1954)
Distribution of the Human Blood Groups, Oxford

Rasmussen, E. M. and P. Willett (1989)
"Database Processing Using a Highly Parallel Array Processor Prospects for Intelligent Retrieval" in *Informatics* 10 (ed. K. P. Jones) Proceedings of a conference jointly sponsored by Aslib, the Aslib Informatics Group, and the Information Retrieval Specialist Group of the British Computer Society, King's College, Cambridge

Renfrew, C. (1989)
"The Origins of Indo-European Languages"
in *Scientific American*, October '89 (pp. 82-90)

Renfrew, C. (1994)
"World Linguistic Diversity"
in *Scientific American*, January '94 (pp. 104-110)

Rheingans, P. (1992)
"Color, Change, and Control for Quantitative Data Display"
in *Visualization '92* (pp.252-259) Conference Proceedings,
IEEE Computer Society Press

Rose, D. E., R. Mander, T. Oren, D. B. Ponceleón,
Gitta Salomon, and Y. Y. Wong (1993)
"Content Awarness in a File System Interface: Implementing the 'Pile' Metaphor for Organizing Information", ACM-SIGIR'93-6 (pp.260-269)

Ritter, H. and T. Kohonen (1989)
"Self-Organizing Semantic Maps",
in *Biological Cybernetics*, 61 (pp.241-254)

Sammon, J. W. (1969)
"A nonlinear mapping for data structure analysis"
IEEE Transactions on Computers C 18, (pp.401-409)

Schroeder, W. J., W. E. Lorensen, G. D. Montanaro, and C. R. Volpe (1992)
"VISAGE: An Object-Oriented Scientific Visualization System"
in *Visualization '92* (pp.219-226) Conference Proceedings,
IEEE Computer Society Press

Schwartz, M. F., C. M. Wood (1993)
"Discovering Shared Interests Using Graph Analysis"
in *Communications of the ACM*, Aug.'93, Vol. 36, No.8 (pp. 78-89)

Shepard, R. N. (1972 /-73?)
"Psychological Representation of Speech Sounds",
in *Human communication: A unified view*,
(eds. E. E. David and P. B. Denes), (pp. 67-113)

Smeaton, A. and P. Sheridan (1991)
"Using morpho-syntactic analysis in phrase matching",
in *RIAO-91*, (pp. 414-429)

Smith, S., G. Grinstein, and R. D. Bergeron (1991)
"Interactive Data Exploration with a Supercomputer"
in *Visualization '91*, (pp. 248-254) Conference Proceedings,
IEEE Computer Society Press

Spoerri, A. (1993)
Visual Information Retrieval Tools, Technical Report-93-2, Center for Educational Computing Initiatives, MIT.

UNDP (1990, 91, 92, 93, 94)
The United Nations Development Programme (annual reports), Oxford University Press

Willett, P. (1988)
"Recent Trends in Hierarchic Document Clustering: A Critical Review" *Information Processing and Management* 24/5 (pp. 577-598)

Willett, P. (1990)
"Data Structures and File Organization"
Proceedings from the European Summer School in Information Retrieval, Organized. by GLIR ($\subset$ AICA), held in Bressanone, Italy

Williams, C, J. Rasure, and C. Hansen (1992)
"The State of the Art of Visual Languages for Visualization"
in *Visualization '92* (pp.202-209) Conference Proceedings, IEEE Computer Society Press

Williams, R. D., F. L. Wefer and T. E. Clifton (1992)
"Direct Volumetric Visualization" in *Visualization '92* (pp. 99-106), Conference Proceedings, IEEE Computer Society Press

Wish, M., M. Deutsch and S. J. Kaplan (1976)
"Perceived dimensions of interpersonal relations"
in *Journal of Personality and Social Psychology* 33, (pp. 409-420)

Young F. W. and P. Rheingans (1991)
"Visualizing in High-Dimensional Multivariate Data"
in *IBM Journal of Research and Development*, Vol. 35, No. 1/2, January/March '91

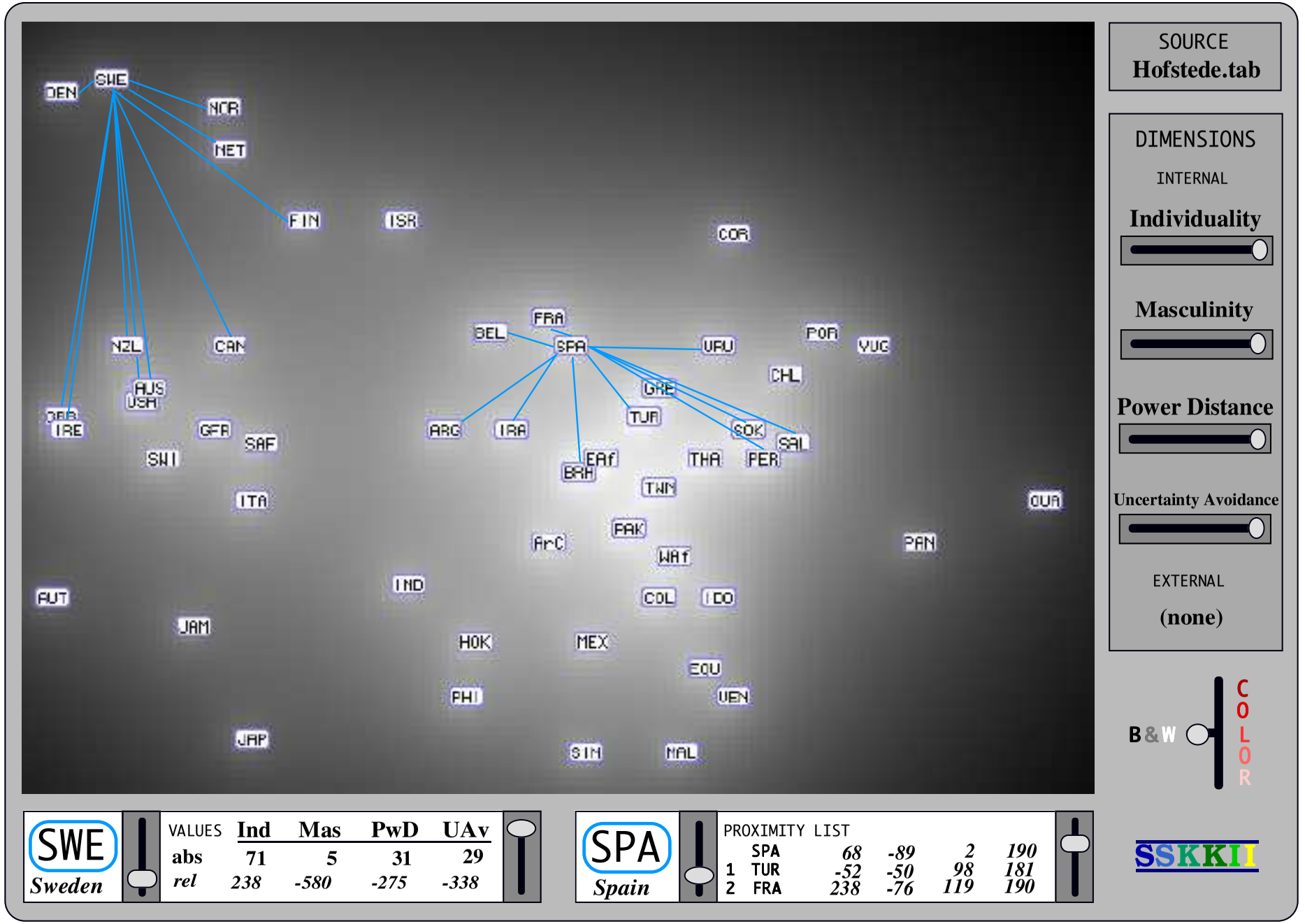


FIGURE 7 Interface of a proposed tool for information visualization and exploration. The cluster configuration is a product of [Hofstede 91] and [Hagman 94].