THESIS ON INFORMATICS AND SYSTEM ENGINEERING C39

Pattern Discovery Using Seriation and Matrix Reordering: A Unified View, Extensions and an Application to Inventory Management

INNAR LIIV

TALLINN UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Informatics

Dissertation was accepted for the defence of the degree of Doctor of Philosophy in Engineering on July 15, 2008.

Supervisor:	Professor Dr. Rein Kuusik Department of Informatics Tallinn University of Technology
Opponents:	Docent Dr. Timo Honkela, Chief Research Scientist Laboratory of Computer and Information Science Helsinki University of Technology, Finland
	Professor Dr. Jaak Vilo Institute of Computer Science University of Tartu, Estonia

Defence of the thesis: August 29, 2008

Declaration:
Hereby I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology has not been submitted for any academic degree.

/ *Innar Liiv* /

ISSN 1406-4731
ISBN 978-9985-59-837-5

INFORMAATIKA JA SÜSTEEMITEHNIKA C39

Mustrite avastamine kasutades järjestamist ning maatriksi ümberkorrastamist: unifitseeritud vaade, edasiarendused ning rakendus ladude juhtimises

INNAR LIIV

Pattern Discovery Using Seriation and Matrix Reordering: A Unified View, Extensions and an Application to Inventory Management

Abstract

Seriation is an exploratory combinatorial data analysis technique to reorder objects into a sequence along a one-dimensional continuum so that it best reveals regularity and patterning among the whole series. In this thesis, we propose a unified view and an objective function for parameter-free seriation, based on Kolmogorov complexity and data compression.

Unsupervised learning, using seriation and matrix reordering, allows pattern discovery simultaneously at three information levels: local fragments of relationships, sets of organized local fragments of relationships and an overall structural pattern. It, therefore, combines, in a single result, and enhances the structural analysis abilities of popular unsupervised data mining methods, like clustering and association rules. We advocate that seriation should be put on a par with those standard data mining methods due to their lack of ability to analyse complex structures and to defocus from details to global relationships.

Seriation methods, however, are computationally much more expensive than traditional data mining techniques and, therefore, do not scale well in comparable scenarios. To remedy this situation, we propose a new algorithm for sparse binary matrix seriation. The algorithm provides a fair compromise between computational complexity and mere enumeration of direct and explicit relationships in the datasets. In addition, we propose a new approach for an expeditious implementation of seriation natively in databases with standard relational algebra and relational calculus in structured query language (SQL) without any use of external procedures or functions.

An application to inventory management is presented to provide the currently missing functionality in inventory management softwares for reclassifying and prioritizing items according to dependencies in customer behaviour. Therefore, besides the evaluation of the proposed seriation algorithm, a novel inventory classification solution is proposed to the problem using data mining and seriation.

Keywords: seriation, two-mode clustering, combinatorial data analysis, minimum description length principle, information visualization, data mining.

Acknowledgements

I am grateful to my supervisor, Professor Rein Kuusik, for introducing me to the topic of this thesis and for the support and advice during the years.

I also wish to thank Professor *emeritus* Leo Võhandu for his influential contributions in the field. His catching enthusiasm is always motivating us to think about BIG ideas and complex problems, surrounding everyday life.

I am thankful to Professor Tanel Tammet for his clear-cut suggestions for the refinement of the focus of the thesis. I wish to thank all my colleagues, partners and friends for their help and stimulating discussions over the years. I am honoured to have had several clients and employers throughout these years with similar data analysis and "number crunching" interests.

I wish to thank Associate Professor José Fernando Gonçalves for sharing his precollected cellular manufacturing problem datasets with me; and my student Tanel Pipar for reimplementing several algorithms by other authors. Over the years, I have had several interesting discussions and received invaluable technical support from Sven Petai, a good friend and a perfect system administrator, who fulfilled even the most obscure request concerning compiling legacy codes or installing obsolete compilers.

Sufficient financial resources were secured by the Doctoral School in ICT (Measure 1.1. of the Estonian NDP for the Implementation of the European SF) and the Estonian Information Technology Foundation. The managers of those funds are highly acknowledged for making the reimbursements practically free from the bureaucratic over-head from the students' perspective, which allowed complete concentration on research.

I would also like to express my gratitude to my family and relatives, especially my parents, who have always emphasized the importance of education and provided all the resources and infrastructure necessary to facilitate my studies and research. I find it most interesting that, while starting my studies at the university, I thought I pursued the education at a completely different area than my mother (linguistics) and father (psychology), but ended up of choosing to understand the behaviour of complex systems. It took me years to realize that languages are some of the most complex systems to understand, probably even more complex than the brain.

Above all, I wish to thank Kristi, for constantly encouraging me to work hard towards finishing the thesis, supporting me and staying with me, regardless all the long trips to conferences and the seemingly never-ending writing process of this thesis. And especially, for giving birth to the utmost precious little guy, Kristjan, whose heartfelt smile and laughter gave me most of the energy to finish this thesis.

Contents

1 Introduction

The world is constantly transforming the way it works and there is a permanent change in every field. From the perspective of a data analyst, there are several very challenging and interesting changes of understanding in progress: a paradigm shift from the hypothesis-driven data analysis to the exploratory data analysis; a shift from product-centered and technology-centered thinking to customer-centered thinking. Moreover, where and how we perform computing is changing in two seemingly conflicting directions: away from people and forwards them. From desktops to the internet and "clouds", and, at the same time, more and more computing power is distributed to users world wide with very powerful mobile phones. Computers are transforming to be more mobile; mobile phones and other devices are starting to resemble noticeably the classical meaning of computers, paving the way for ubiquitous computing.

However, one thing that remains constant amid change is the fundamental desire to make sense of things, to be able to categorize and arrange them into some meaningful order. This does not mean at all that putting together pieces of information to get the overall picture is trivial and the way to do it remains or should remain the same. On the contrary, the scientific community is forced by the above mentioned and other paradigm shifts to develop not only enhanced methods to analyse myriads of data, but to develop methodologies completely off the beaten track. In order to cope with the amount of information, it is also crucial to bring those methodologies closer to end-user consumption. Especially, when the way data is recorded is also changing – from professionally and carefully handpicked hypothesis-driven data acquirement to "secondary data analysis" – making sense of virtually everything ever recorded or logged for some other primary or exploratory purpose. Furthermore, in the past few years, those limits of data generation have been stretched again. People have started to actively participate and give more feedback to the system, in both an unstructured and a structured manner by using information tags, recommendations and personal intentions due to completely new incentives provided by the web 2.0 and online social networks.

This thesis is about the science of classification and taxonomization, but with a very specific and incisive focus on the problem of *seriation* – reordering and arranging objects in the order that reveals regularity and patterning in the best way among the whole series. A metaphorical example to illustrate the concept and help the reader to understand the difference and interplay of fundamental operations like ordering and grouping, could be the Mendeleev's periodic table of the chemical elements. It is a table with a similar aim – to illustrate relationships, periodic patterning and trends in the properties of the elements. So, in a way, this thesis is about a methodology that allows everybody to discover and create their own "Mendeleev's periodic table" for whatever sets of objects and attributes or other more complex systems and relations under investigation. Interestingly, the above example is more than a metaphor. Bertin (1981, p.53) has reported a pedagogical study of discovering and reconstructing the periodic table of chemical elements, using the reordering of rows and attributes in the table until regularity and patterning between the neighbouring elements is maximized.

The chapter will discuss the concept of seriation, its place in the data analysis paradigm and account for the author's interest in and enthusiasm about the subject. The chapter will also list the research problems under study and present the organization of the dissertation.

1.1 Seriation: setting the scene

Seriation has the longest roots among the disciplines of archaeology and anthropology, where, for the moment, it has reached also the maturest level of research. A recent monograph about seriation by O'Brien and Lyman (1999) includes an extended discussion of the terminology, and a consensual definition, allowing convergence of several seriation traditions, is agreed to be given by Marquardt (1978, p. 258):

> [Seriation is] a descriptive analytic technique, the purpose of which is to arrange comparable units in a single dimension (that is, along a line) such that the position of each unit reflects its similarity to other units.

O'Brien and Lyman point out and emphasize that "nowhere in Marquardt's definition is the term *time* mentioned", which perfectly coincides with our interpretation and focus. However, to make the definition more general and compatible with the rest of this thesis and set the scene for our own definition construction, we suggest to:

- understand and interpret the phrase of *comparable units* as *units from the same mode* according to Tucker's (1964) terminology, making it less ambiguous about whether it is allowed or not to arrange column-conditional and other explicitly non-comparable units along a continuum;
- give more emphasis to simultaneous pattern discovery at several information levels - from local patterns to global. It would make the definition compatible with the requirements set to such matrix permutations by Bertin (1981, p.12). He saw information as a relationship, which can exist among elements, subsets or sets and was convinced that "the eye perceives the three levels of informations spontaneously" (1981, p.181).

Considering that discussion, we are able to construct a definition of seriation to reflect our emphasis and focus of this thesis the best as follows:

Seriation is an exploratory data analysis technique to reorder objects into a sequence along a one-dimensional continuum so that it best reveals regularity and patterning among the whole series.

Higher-mode seriation can be simultaneously performed on more than one set of entities, however, entities from different sets are not mixed in the sequence and preserve a separate one-dimensional continuum.

For the best concept delivery purposes, examples in this section will only use binary values, which is also the scope limitation of proposed approaches and algorithms. However, we consider and discuss several common value types of data, where applicable. The scope is additionally limited to entity-to-entity and entity-to-attribute data tables, or using Tucker's (1964) terminology and Carroll-Arabie (1980) taxonomy, we are concentrating on two-way one-mode (NxN) and two-mode (NxM) data tables. It should be emphasizes that such a scope definition does not restrict the one-mode data table to be symmetric, or make entity-to-entity data table to be exclusively only one-mode, i.e. there can be relations between entities from different sets, making such a table a two-mode matrix.

Let us look at the following example of seriation along with the introduction of three most common forms of how data is presented throughout this thesis: a matrix, a double-entry table with labels and a color-coded graphical plot. We may often use the word *matrix* in the thesis interchangeably to refer to all of those forms.

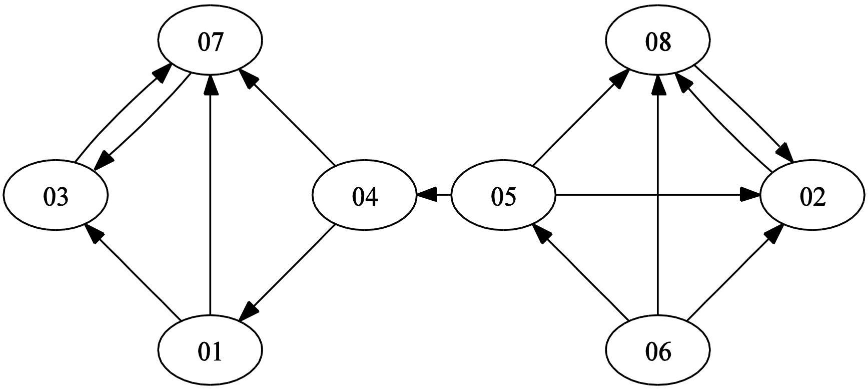


Figure 1.1 An example graph

An example dataset is first presented as a graph in Fig.1.1. We will first construct an asymmetric adjacency matrix that reflects the structure of such a directed graph:

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Another way to present the same structure is by using a double-entry table with node labels, where the positive elements have been shaded and formatted differently for better visual perception:

Table 1.1 Double-entry table of the example dataset

	01	02	03	04	05	06	07	08
01	**1**	0	**1**	0	0	0	**1**	0
02	0	**1**	0	0	0	0	0	**1**
03	0	0	**1**	0	0	0	**1**	0
04	**1**	0	0	**1**	0	0	**1**	0
05	0	**1**	0	**1**	**1**	0	0	**1**
06	0	**1**	0	0	**1**	**1**	0	**1**
07	0	0	**1**	0	0	0	**1**	0
08	0	**1**	0	0	0	0	0	**1**

Inspired by Czekanowksi (1909) and Bertin (1967), it is often reasonable to present the matrix with a graphical plot, where numerical values are color-coded. With binary data, the most typical way is to use filled cells to denote "ones" and empty cells, "zeros", respectively. Using such an approach, we can visualize the above structure as follows:

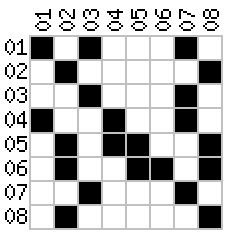


Figure 1.2 A graphical "Bertin" plot for the example dataset

From such plain matrices, tables and plots, it is still rather complicated to identify the underlying relationships in the data, find patterns and an overall trend. Objects in such an adjacency matrix are ordered in an arbitrary order, typically in the order of data acquisition/generation, or just sorted alphabetically by labels or names. Changing the order of rows and columns, therefore, does not change the structure: there are *n!* (or *n!•m!* in case of two-mode matrix) permutations of the same matrix that will explicitly reflect the identical structure of the system under observation. The goal of seriation is to find such a permutation, i.e. to reorder the objects from the same mode in a sequence so that it best reveals regularity and patterning among the whole series. This does not, by any means, exclude the chance that data acquisition or alphabetical ordering actually lead to structurally best ordering, but it should never be assumed a priori. We can look at this also from the perspective of a single element (cell), which position can be changed arbitrarily with the constraint that it must always be moved together with the whole row or column – making it somewhat similar to the classical game of Rubik's cube. An example of the seriation procedure is demonstrated on the Fig.1.3:

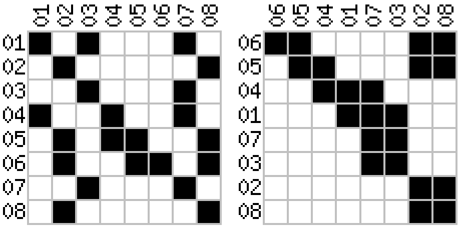


Figure 1.3 An example of the seriation procedure

Clearly, from the right plot of Fig.1.3, the underlying structure and relationships can be far more easily perceived. However, this is exactly where the challenge of this problem is hidden – how to develop algorithms to perform seriation without exhaustive search of all permutations and how to evaluate the goodness of the result. The new order for rows and columns on the right plot of Fig.1.3 was reached manually by the author with a highly subjective on-the-fly evaluation of the goodness using visual perception. Actually, this is exactly how it was done in the 60s and 70s by a research group directed by a French cartographer Jacques Bertin (1981, p. 47), who stated that, with assistants and mechanical devices, "it only takes three days to construct a matrix and three weeks to process and interpret it more deeply", which was hoped to get even more comfortable using computers. At the same time, several algorithms for automatic seriation already existed, but a quick propagation of such developments and results was restrained and muted by the barriers of different scientific traditions and disciplines.

An example of two valid alternative permutations found for the investigated dataset are presented in Fig. 1.4. Those results are achieved with algorithms called a bond energy algorithm (McCormick *et al.*, 1969; 1972) and "minus" technique (Mullat, 1976a,1976b,1977; Vyhandu, 1980,1989). The former optimizes an objective function and the latter, if we use the terminology proposed by Van Mechelen *et al.* (2004) for similar algorithms, does modeling at a procedural level – a specific heuristical strategy is followed and an overall loss or objective function to be optimized is not implied.

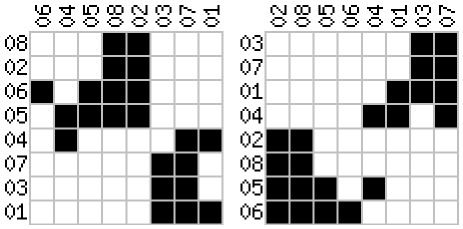


Figure 1.4 Alternative permutations for the same dataset

One might notice that both of those matrices (Fig. 1.4) have different orders for rows and columns. Although we are dealing with one-mode data, such a treatment is reasonable if the graph is directed and therefore the adjacency matrix is asymmetric. Finding only a single permutation is possible (as seen in Fig 1.3) in such a scenario, but it could result in less structured output due to the reordering restrictions and require some extra data processing (e.g. making the adjacency matrix temporarily symmetric for the duration of the seriation procedure).

Another challenging and focal question with the problem of seriation is defining and evaluating which permutation is the best. For the same example graph (Fig. 1.1) and dataset (Fig. 1.2) we already have proposed three (one on the right of Fig. 1.3 and two at Fig. 1.4) relatively good and subjectively interesting permutations. But which one of those opens up the natural inner structure, patterns, regularity and the overall trend the most? One could subjectively argue that the manual reordering result (Fig. 1.3, right) offers the best seamless structural transformation, or that the result of the minus technique (Fig. 1.4, left) illustrates clearly the decomposition of the system and identifies the bridging elements between the two groups. The consensual seriation goal of maximizing the similarity between neighbouring objects still leaves a lot of ambiguity and vagueness for the exact objective function formulation due to virtually hundreds of ways to define sameness and similarity. Chapter 3 will further discuss this specific issue, together with another practical view of measuring the goodness of seriation objectively using data compression.

Furthermore, it is important to make a distinction between seriation and clustering. With clustering, objects are assigned to groups, but with seriation, objects are assigned to a position within a sequence. Efficiency of clustering algorithms is broadly due to the property that objects are unordered in the emerged groups. If we think about the hierarchical clustering and dendrograms, then at every split of the tree the order of succeeding elements is chosen arbitrarily or according to the order of appearance in the data source. According to Brusco and Stahl (2005, p. 18), the number of all feasible partitions of n objects into K clusters is:

$$\frac{1}{K!}\sum_{k=0}^{K}(-1)^k\binom{K}{k}(K-k)^n \tag{1.1}$$

Table 1.2 Number of all possible clusterings and seriations

# / **objects** ***(n)***	# / **clusters (*K*)**	# / **Possible clusterings**	# / **Possible seriations**
10	3	9330	3628800
10	4	34105	3628800
10	5	42525	3628800
10	6	22827	3628800
20	3	580606446	2.43×10^{18}
20	4	45232115901	2.43×10^{18}
20	5	7.49×10^{11}	2.43×10^{18}
20	6	4.31×10^{12}	2.43×10^{18}
30	3	3.43×10^{13}	2.65×10^{32}
30	4	4.80×10^{16}	2.65×10^{32}
30	5	7.71×10^{18}	2.65×10^{32}
30	6	2.99×10^{20}	2.65×10^{32}
40	3	2.03×10^{18}	8.16×10^{47}
40	4	5.04×10^{22}	8.16×10^{47}
40	5	7.57×10^{25}	8.16×10^{47}
40	6	1.85×10^{28}	8.16×10^{47}
100	3	8.59×10^{46}	9.33×10^{157}
100	4	6.70×10^{58}	9.33×10^{157}
100	5	6.57×10^{67}	9.33×10^{157}
100	6	9.07×10^{74}	9.33×10^{157}

We have constructed Table 1.2 to illustrate the difference between exhaustive searches and validations for the best clustering (equation 1.1) and seriation (*n!*) solutions. For clustering, we have added K number of clusters to the table, which must also come as a parameter before the procedure is executed. For a two-mode and a higher-mode analysis the total number of solutions equals either the product or the sum of the search spaces of all modes, depending on the interplay between modes in the algorithm. At the time of writing this thesis, the author is not yet aware of any algorithms, which cannot be reduced to separate problems (e.g. *n!+m!* instead of *n!•m!*) mode-wise and would perform better according to any criteria. Separate modes are, therefore, mainly reordered sequentially (or in parallel), not simultaneously.

In other words, if the goal for an algorithm is to perform "pure" clustering, it would not be a reasonable and an effective strategy to perform some extra optimizations, especially if the added complexity makes it computationally much more expensive. A good example of such an additional procedure is called optimal leaf ordering (Bar-Joseph *et al.,* 2001), which is performed after a hierarchical clustering to maximize the similarity of adjacent objects. However, if such extra optimizations and goals are included in the analysis, then the result is already something in between a clustering and a seriation, or eventually a seriation.

Späth (1980, p.212) considered such matrix permutation approaches to have a great advantage in contrast to the cluster algorithms, because "no information of any kind is lost, and because the number of clusters does not have to be presumed; it is easily and naturally visible." Murtaugh (1989) referred to similar advantages calling such an approach a "non-destructive data analysis", emphasizing the essential property that no transformation of the data itself takes place, contrary to a classical step in clustering, where a two-mode matrix is converted into a one-mode similarity matrix. However, Hartigan (1972) demonstrated that it is also possible to perform a so-called "direct clustering of a data matrix", i.e. one does not necessarily need to convert a two-mode matrix into a one-mode similarity matrix to perform clustering. Proceeding from Hartigan's contribution and research (1972, 1975) towards two-mode clustering (for a recent review, see Van Mechelen *et al.* (2004)), a direct analysis of the data matrix does not perform as an effective distinguisher any more. Therefore, Bertin's (1981, p.6) proposed goal of "simplifying without destroying" is more preferable together with his three information levels (a relationship, which can exist among elements, subsets or sets), which all must be emphasized and concentrated. Bertin was convinced (1981, p.7) that simplification was "no more than regrouping similar things." We will show in Chapter 3 how we see such claim strongly connected to a form of Occam's razor and how the objective goal of seriation in general could be defined and described similarly to the hypothesis identification by the minimum description length principle (Rissanen, 1978; Grünwald, 2007). The simplicity of a hypothesis reaches a completely new niche meaning, if we look at it with the problem of seriation in mind. A thorough review and discussion of how such a perspective of simplicity is compatible with the philosophies of Epicurus, Occam, Bayes, with comments on Karl Popper's opinion on Occam's razor, is available in the Kolmogorov complexity monograph by Li and Vitanyi (1997, Chap.5).

1.2 Motivation

There are several distinct layers of motivation for this work. The most essential motivation is to develop better tools to augment human capability in making sense of all the data that is surrounding us and pouring in, and making sure it is not just passing by. Research on telescopes, microscopes and Röntgen's X-rays has augmented and amplified human vision. It is equally important to develop something analogous for data – tools that would amplify thinking, cognitive processing and perception.

Different traditions and disciplines struggle for the leading position for achieving or moving towards providing the best insights from the data. Seriation, which is the central approach to the goal in this thesis, currently positions itself somewhere in the middle of data mining, information visualization and social network analysis. All those areas share similar essential goals, but have minimal intersections in mainstream research progress. Prominent authors in the discipline of information visualization (Bederson and Shneiderman, 2003, p.351) have identified that the data mining community gives minimal attention to information visualization, but believe that "there are hopeful signs that the narrow bridge between data mining and information visualization will be expanded in the coming years." Shneiderman (2002) has pointed out that "most books on data mining have only a brief discussion of information visualization and vice versa " and that "the process of combining statistical methods with visualization tools will take some time because of the conflicting philosophies of the promoters". However, besides time, such progress requires systematic and coherent research on unified views, common taxonomies and algorithmic enhancements to work in familiar and comparable scenarios. Moreover, if the intersection of the research in data mining and information visualization communities is diaphanous, the case with systematic view on seriation is even worse. Dunham's (2003, p.145) introductory book on data mining is rather an exception of covering matrix permutation approach at all, regardless of a very localized view on the problem and slight misinterpretations of the original work by McCormick *et al.* (1972). Arabie and Hubert (1992, 1996) have also identified seriation approaches to be "orphaned in most overviews of data analysis" and therefore being constantly off the scene or reinvented.

The motivation to put seriation on a par with standard data mining techniques like clustering and association rules is due to the lack of their ability to analyse complex structures and to defocus from details to global relationships. On the contrary, those two problems are exactly what a social network analysis and information visualization are addressing from different perspectives. The concept of seriation can work towards attaching together advantages from both of those fields and expected results from association rule mining and clustering. A seriation result also contains the clustering of data with additional information about how one cluster is related to another, what the bridging objects are and what the transition of the objects is like inside the cluster. From the perspective of association rules, a result of seriation can also be interpreted as a chain of associations between objects, which is a non-redundant and optimal presentation of the possibly very lengthy list of association rules. Moreover, it also introduces context to those relationships.

There has been some recent work towards such goals and more complex structures, e.g. mining chains of relations (Afrati, Das, Gionis, Mannila, Mielikäinen, and Tsaparas, 2005), which clearly shows that such direction is promising. However, even if it is done by a strong and established data mining research group as the one above, it has remained strongly off the track from the mainstream data mining community and popular software packages and workbenches. Another encouragement and justification for the research direction is a recently organized workshop "From Local Patterns to Global Models" at a prestigious machine learning and knowledge discovery conference ECML/PKDD 2008 with a goal to bring together researchers for a discussion on constructing global models from fragmented knowledge of local patterns. Such initiatives and goals are also related to the problem of generalizing across domains, in the context highlighted by Domingos (2007) as one of the top problems for structured machine learning for the next ten years.

And finally, of course, the motivation that originates from the present unmet demand in the industry. The paradigm shift from product-centered thinking to customer-centered thinking has been gaining acceptance in marketing (Rust *et al.*, 2004). A good example, which previously would have gone unnoticed, is an undesired effect within customers' behaviour is called *Profitable Product Death Spiral* (Rust *et al.*, 2000, p.30), "in which decisions that seem to be increasing profitability alienate the customer by ignoring the effect of assortments of choices, eventually leading the firm to disaster". It is a situation, where a product being frequently bought, assembled or used together with some other product is disregarded and therefore may lead to customer retention for those who are accustomed of buying specific products in bundles. Tools to support the management of such an effect and behaviour at operational level are critical in order for the shift to be effective. However, current enterprise resource planning (ERP) and inventory management softwares lack the support and functionality of reclassifying and prioritizing items according to dependencies in customer behaviour. We have addressed that problem in Chapter 5 with a solution using data mining and seriation to bridge this specific gap between the strategy and execution in the industry.

1.3 Theoretical background

Main theoretical and methodological foundations and influencers of this work are:

- an exploratory data analysis paradigm introduced by Tukey (1977);
- the algorithmic approach and heuristics from the works of Vyhandu (1979,1980,1981,1989) and Mullat (1976a,1976b,1977);
- Bertin's (1981) philosophy for finding relationships within the data with the emphasis on the importance of graphics (Tufte, 1983) in that dialogue;
- Coombs' (1964) and Carroll-Arabie (1980) taxonomy of data;
- levels of optimization and models for two-mode clustering by Van Mechelen *et al.* (2004);
- Arabie's and Hubert's (1992) perspective of combinatorial data analysis;
- a practical look on Kolmogorov-Chaitin-Solomonoff complexity by Li and Vitanyi (1997) and the minimum description length principle (Rissanen, 1978; Grünwald, 2007).

We also think of a classification to be much more than a discriminant analysis of categorical data, as it is commonly considered in the machine learning and statistics community. Our perspective of it is rather compatible with the viewpoint of IFSC[1] community, with an in-depth treatment and discussion in the infamous monograph *Numerical Taxonomy* by Sokal and Sneath (1963). Recent examples of a similar systematic treatment for classification and data analysis include works by Mirkin (1996) and Arabie, Hubert, and De Soete (1996).

1.4 Research questions

A starting point for this thesis was a set of interesting algorithms that were developed and applied successfully to real-world problems in the local research team for the past 30 years. Due to strong traditions of classical statistics, the arguments for preference and superiority mainly coincided with the general discussion about the exploratory data analysis versus hypothesis-driven data analysis. No comparative studies had been published with the problems of the same class, nor had such a neighbourhood explicitly identified or defined.

There are two main research goals for the dissertation:

- To link the seriation research done in Tallinn University of Technology for the past 30 years with similar research done by other groups, which assumably has influence beyond the exact scope of this thesis;

[1] International Federation of Classification Societies, non-profit scientific organization founded in 1985, http://www.classification-society.org/

- To narrow the gap between seriation and data mining community, which requires unification, alignment of the starting points and goals with bridging developments of usable parameter-free algorithms to make it applicable in similar contexts and environments.

Main research questions to be answered in this thesis are the following:

- Where does seriation fit in the paradigm of exploratory data analysis and unsupervised learning? What is the immediate neighbourhood?
- What are the main gaps between seriation and data mining? How do the goals and practices coincide with traditional data mining techniques?
- Does the heuristic used in this research give reasonable and comparable results with other fields of seriation usage?
- How could we formalize the goal of seriation? What would be the compatible objective function to measure and evaluate the goodness of seriation?
- What has to be changed and modified within the seriation algorithms to make them more applicable in real-world scenarios?
- Are there any real-world problems where seriation algorithms with such enhancements could provide new ways to solve problems?

1.5 References to previously published work

Several chapters of this thesis are based on the author's previously published papers and presentations. Parts of Chapter 2 and 3 have been presented at the International Conference on Artificial Intelligence (Liiv, 2007a), held in Las Vegas, USA (2007). Section 4.2 is based on a paper presented at the International Conference on Artificial Intelligence, Knowledge Engineering and Databases, held in Corfu, Greece (Liiv *et al.*, 2007a), and subsequent extended version in a journal (Liiv *et al.*, 2007b). Chapter 5 is based on papers "Inventory classification enhancement with demand associations" (Liiv, 2006) and "Visualization and data mining method for inventory classification" (Liiv, 2007b), presented on the consequential years of the IEEE/INFORMS International Conference on Service Operations and Logistics, and Informatics held in Shanghai, China (2006) and Philadelphia, USA (2007). Some statements and observations of matrix reordering structural properties that are used in Chapter 3 were presented at the ACM International Conference on Artificial Intelligence and Law (Liiv *et al.*, 2007c). In the papers with several authors, the contributions of seriation and matrix reordering are from the author of this dissertation.

1.6 Organization of the dissertation

This chapter presented the general introduction to seriation and highlighted main motivations for undertaking the research presented in this dissertation. The rest of the dissertation is organized as follows.

A review of the related work is presented in **Chapter 2**, with the main emphasis on the motivation and the incentives in different disciplines to use seriation, with comments on the developments and examples from the perspective of the taxonomy developed by Carroll and Arabie (1980). The following disciplines are included in the review: archaeology and anthropology; cartography, graphics and information visualization; sociology and sociometry; psychology and psychometry; ecology; biology and bioinformatics; cellular manufacturing, and operations research. An overview of local research on seriation is presented after the discipline-wise review.

In **Chapter 3**, we will review related work on unification and evaluation measures and propose our own formulation and an objective function for parameter-free seriation, based on Kolmogorov complexity and data compression. A proposed measure is demonstrated on examples and evaluated with empirical experiments, using 35 problems from the literature to investigate the agreement with recognized measures from operations research. The main goal of Chapter 3 is to enhance repeatability, scrutiny and rigorous benchmarking ability for seriation.

New approaches and algorithms are presented in **Chapter 4**. In Section 4.2, we will demonstrate that it is possible to define and implement the conformity analysis algorithm with standard relational algebra and relational calculus in structured query language (SQL) without any use of external procedures or functions. In Section 4.3, a quick seriation algorithm for binary sparse datasets is presented, which exploits the symmetric property of sparse binary matrices to reduce significantly the number of steps to reach identical results of the "minus" technique algorithm (Vyhandu, 1981; Mullat 1976a). We have evaluated the properties of the algorithm and measured the execution time with two datasets further discussed in Chapter 5.

In the last part of the dissertation, **Chapter 5**, we suggest a different and more customer-centered approach for solving particular fallacies of the classical ABC analysis in inventories – using customer behaviour and demand associations for classification enhancement. We will discuss several aspects of well-known inventory classification strategies and propose a solution, using data mining and seriation approaches and algorithms presented in Chapter 4, to bridge this specific gap between customer-centric corporate strategies and the available functionality in current ERP systems and inventory management softwares. Experimental results for two warehouse datasets are included and analyzed, followed by the discussion.

Chapter 6 concludes the thesis with the summarized list of contributions of this dissertation and directions for future research.

2 Related work

2.1 Introduction

An interested reader of all related work on the topic should probably start from the *Organon* collection of the works by Aristotle (384 BC – 322 BC), especially the *Categories* (an interesting discussion from the perspective of classification and clustering has been published by Mirkin (1996, Section 1.1)) and the *Topics*. However, in order to keep the specific and incisive focus on the problem of seriation, we will draw the line at the works of Petrie (1899) and Czekanowski (1909). Those works represent a recognized and a systematic start of seriation and matrix permutation visualization, respectively.

Even within the area of seriation, this overview clearly cannot be an exhaustive one, but it should give a coherent view to the related work on the problem and not overlap existing reviews and survey papers. There is also a specific theme to follow – the main emphasis is given to the motivation and the incentive to use seriation, commenting on the developments and examples from the perspective of the taxonomy developed by Carroll and Arabie (1980) and making suggestions of minor modifications for implementation steps, where necessary, to make the approaches compatible with others and cross-applicable.

Where possible, related contributions are categorized by disciplines. This enables to highlight the domain-specific incentives, peculiarities and traits of character, which could possibly have other interesting interpretations across disciplines.

Most of the redundancy of research comes from calling the same thing with different names and calling different things with the same name. This also applies to seriation, therefore, we also try to identify the common terminology in different disciplines.

In addition, an overview sketch (Fig. 2.1) has been compiled to summarize the connectedness of related work in different disciplines. Relations between research groups and contributions are broadly defined between combinations of implicit and explicit references in the works, together with the author's subjective judgement of influences, similar approaches and descendence of methods. It is also a visual abstract of the insights found in the subsequent sections.

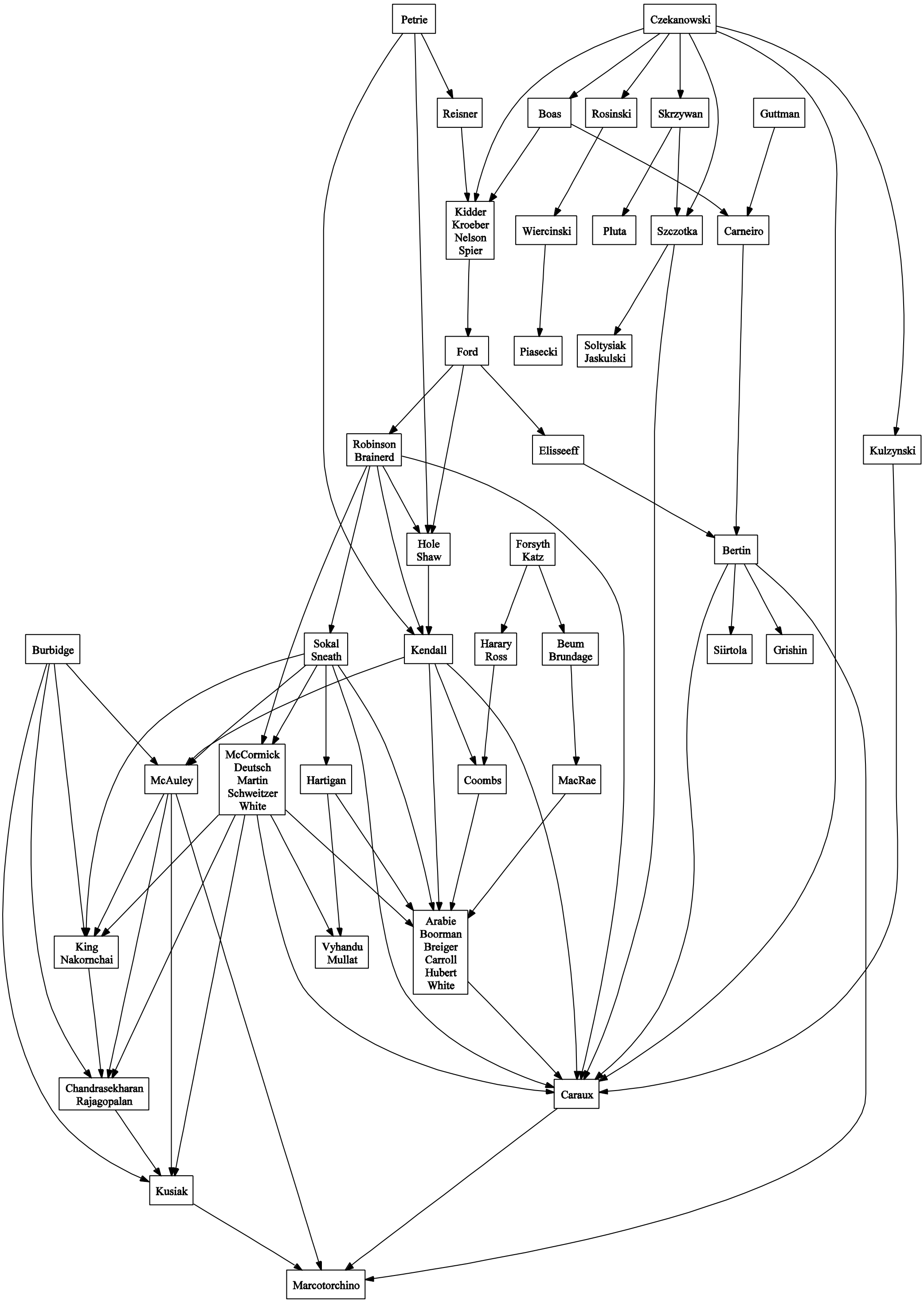


Figure 2.1 A visual abstract of the related work in different disciplines

2.2 Archaeology and anthropology

"The order of events and the relation of one country to another is the main essential of history."
Sir William Matthew Flinders Petrie (1899)

Regardless of the geographic differences in understanding and the exact classification of the fields of archaeology and anthropology, in this section we are interested in something they have in common - scientific approaches for understanding and reconstructing the past upon partial and incomplete information. Time and sequential arrangement of events, cultures and traditions play an important role in achieving that goal. There is a wide range of methods for dating in the field of archaeology. According to our focus, we are considering only *seriation*, which belongs to the branch of relative dating. Although the meaning of the word strongly associates with chronological ordering and time according to most definitions and amongst field practitioners, there exist several accepted and more general definitions that fit our focus better. According to the definition proposed by Marquardt (1978, p.258), seriation is "a descriptive analytic technique, the purpose of which is to arrange comparable units in a single dimension (that is, along a line) such that the position of each unit reflects its similarity to other units." O'Brien and Lyman (1999, p.60) emphasized that such a definition did not narrow down the range or characteristics of units to be seriated, nor did it mention the time as the only or preferred resulting continuum of the linear order attained.

As far as the author knows, the first systematic method for seriation was developed by an English Egyptologist W. M. Flinders Petrie (1899), who called it *sequence dating*. His approach was different from others for depending exclusively on the information and the similarity of findings versus professional human judgement of evolutionary and development complexity of artefacts. This type of distinction and classification is also supported by the seriation taxonomy presented by Lyman, Wolverton and O'Brien (1998), who called those two fundamentally different branches *similiary* and *evolutionary* seriation, respectively. They also distinguished between three types of similiary seriation – phyletic, frequency and occurrence. However, the goal and the structure to be found in those three distinctions coincided with the only difference in the types of the underlying data. Therefore, from the perspective of this thesis, we will discuss those methods interchangeably and not highlight such distinction.

Observations and methods presented by Petrie (1899) were not written down using classical mathematical notations, but are nevertheless recognized (Kendall, 1971) for being the first to clearly formulate the idea of sequencing objects on the basis of their incidence or abundance. Petrie examined about 900 graves, "representing the best selected graves from among over 4000" (Petrie, 1899) and assigned them *sequence dates* using mainly the characteristics of the found pottery. Hole and Shaw (1967, p.4) describe Petrie being able to "seriate the pottery chronologically by merely looking at the characteristics of the handles". However, there is another way to look at his data, which makes it more systematic. From the figure presented by Petrie (1899, p.301), we have compiled Table 2.1, showing the enumerations of the types of pottery that pass through into an adjacent stage. Such a transformation makes the results compatible with the current, generally acceptable seriation formats and would classify as a two-way one-mode data table.

Table 2.1 An enumeration of the types of pottery which pass through stages

Sequence dates	30	31-34	35-42	43-50	51-62	63-71	72-80
30	6	2	0	0	0	0	0
31-34	2	8	2	0	0	0	0
35-42	0	2	8	2	0	0	0
43-50	0	0	2	7	2	0	0
51-62	0	0	0	2	7	2	0
63-71	0	0	0	0	2	7	2
72-80	0	0	0	0	0	2	6

We are able to construct several matrices based on the data from Table 2.1, depending on the required input of our analysis. Matrices can either take into consideration the numerical values of enumerations (e.g. $A^{(1)}$ and $A^{(2)}$) or present only the occurrence or absence (e.g. $A^{(3)}$) of pottery forms passing through into an adjacent stage.

$$A^{(1)} = \begin{pmatrix} 6 & 2 & 0 & 0 & 0 & 0 & 0 \\ 2 & 8 & 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 8 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 7 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 7 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 7 & 2 \\ 0 & 0 & 0 & 0 & 0 & 2 & 6 \end{pmatrix}$$

$$A^{(2)} = \begin{pmatrix} \times & 2 & 0 & 0 & 0 & 0 & 0 \\ 2 & \times & 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & \times & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & \times & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & \times & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & \times & 2 \\ 0 & 0 & 0 & 0 & 0 & 2 & \times \end{pmatrix}$$

$$A^{(3)} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Petrie demonstrated using his visual representation that "it will be readily seen how impossible it would be to invert the order of any of these stages without breaking up the links between them" and stated that "the degradation of this type was the best clue to the order of the whole period". Those statements are also intuitively true for such a matrix representation, with the reservation that complete inversions of the order (e.g. flipping vertically or horizontally) or other operations that would not change the adjacent links will preserve the regularity and will be considered with equal regularity throughout this thesis.

The interpretation of the work done by Petrie will remain largely subjective, as he did not explicitly describe all the details in the papers and according to Ihm (2005), his notes and records were destroyed.

Petrie's work influenced several prominent American anthropologists and archaeologists like George Andrew Reisner, Alfred Vincent Kidder, Alfred Louis Kroeber, Nels Nelson, Leslie Spier, James A. Ford (several good reviews and a discussion of such a methodological lineage include Lyman *et al.* (1997, 1998), O'Brien *et al.* (1999) and Ihm (2005)), who applied, popularized and further developed the methodology to better suit the practical needs for relative dating.

The evaluation of seriation results remained primarily intuitive and subjective until Brainerd (1951) and Robinson (1951) proposed a desired final form for the matrix: the highest values in the matrix should be along the diagonal and monotonically decrease when moving away from the diagonal. The paper included a description of an agreement coefficient (basically a similarity coefficient customized for percentage calculations) and a manual procedure and guideline for reordering. Additionally, an external relative dating and archaeology-specific test for reordering validation was also introduced. The most influential contribution, however, was the mathematical property of the desired matrix form, which has remained popular along other authors and is often referred to as *Robinsonian matrix*, *Robinson Matrix* or *R-matrix*.

About a decade later, several algorithms for chronological ordering were proposed (Ascher and Ascher, 1963; Kuzara *et al.*, 1966), following a comprehensive monograph by Hole and Shaw (1967), covering and evaluating the state-of-the-art techniques for automatic seriation. Hole and Shaw also presented an efficient algorithm called the *permutation search*, which requires $n(n-1)/2+n^2$ evaluations instead of the exhaustive evaluation of all possible orderings.

Besides the algorithmic enhancements, several papers (Rowe, 1961; Dunnell, 1970) were published about the assumptions, requirements and conditions under which seriation results can be considered to be approximating the chronological order, not some other underlying regularity.

The approach of Brainerd and Robinson faced some immediate criticism by an anthropologist Lehmer (1951) for being too dependent on exact numbers of frequencies and for not taking into account the differences in the size of the collections. Dempsey and Baumhoff (1963) proposed a *contextual analysis* method to cope with such a problem, which would merely use the information, irrespective of whether specific type of artefact was present or absent. They justified their approach for being less sensitive to sampling variations and emphasized that "types that occur with low frequency may be among the best time-indicators [and] the presence of single specimens of certain types may be crucial in establishing chronologies." Their approach was classified as occurrence seriation by O'Brein *et al.* (1999), together with an extensive discussion on the differences between frequency and occurrence seriation. We consider the progress from frequency seriation to occurrence seriation favourable due to being directly compatible with our definition of seriation.

The dialogue between archaeologists and statisticians was pioneered by Kendall (1969a,1969b), who contributed several papers on the research of mathematical properties of the matrix-analysis used in archaeology. A similar cooperation between mathematicians, statisticians and archaeologists eventually led to a dedicated joint conference on *Mathematics in the Archaelogical and Historical Sciences*. Those proceedings published in a volume edited by Hodson, Kendall and Tautu (1971), serve to date as one of the most comprehensive collections on research done on archaeological seriation. The research mainly focused on one-mode two-way seriation methods, but there were also examples of two-mode two-way seriation (e.g. Spaulding (1971, p.7)) where artefacts and their variables were directly analyzed without transformation to a similarity matrix format.

Regardless of the classical retrospective look on matrix reordering techniques in archaeology and anthropology, there is another important branch of research that is seldom if ever mentioned in the context of previous methods. It is the work of Jan Czekanowski (1909) on matrix reordering and visualization, which is probably the first published work on one-mode data analysis that was based on the permutation of the rows and columns, complemented with color (pattern) coding for better visual perception. He did not have the goal of chronological ordering, but aimed to develop a differential diagnosis of the Neanderthal groups. Differential diagnosis as a term is mainly used in medicine as a systematic method to identify the disease based on an analysis of the clinical data. However, Czekanowski used it in a wider sense — as a systematic classification method to identify and describe groups and their formation in the data. He defined the (dis)similarity coefficient as the average difference of the characteristics of two individuals — the average of the absolute values of differences in characteristics. The results of the difference calculations helped to form a similarity matrix, where the elements/cells were shaded in five different (visual) patterns (as shown in Fig. 2.2). Czekanowski did not have any formal procedure for rearrangement of the elements in the matrix, therefore, probably, visual inspection and intuition was used as the size of the dataset was also considerably small.

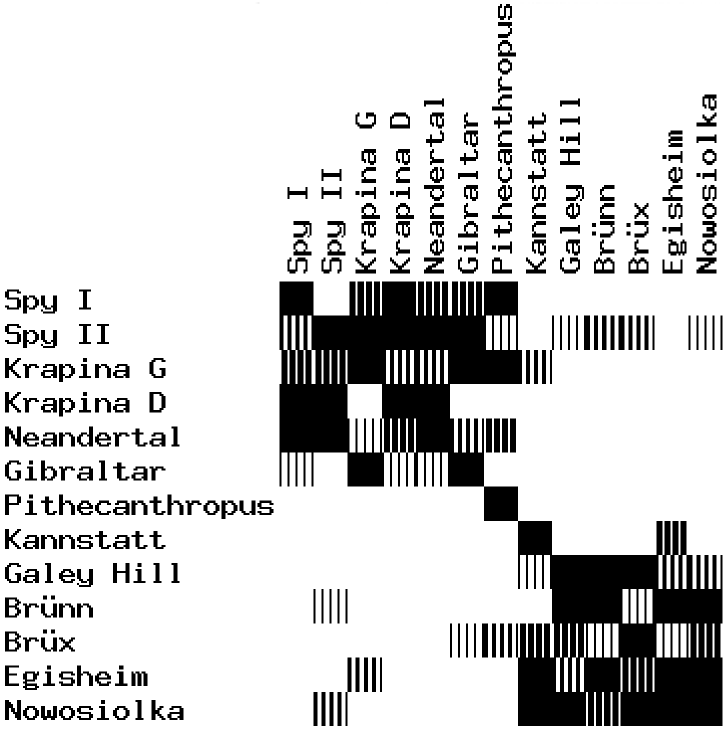


Figure 2.2 Czekanowski's (1909) diagram of differences and groups of skulls

Methods developed by Czekanowski have suffered partly for being isolated from the Western science. However, the method was widely used by Polish anthropologists Boleslaw Rosinski, Andrzej Wiercinski and Karol Piasecki (Sołtysiak, personal communication, March 12, 2007), who were disciples of Czekanowski's tradition and secured the methodological continuity. According to Sołtysiak (personal communication, March 12, 2007), rearrangement of the objects was done visually up to matrices with 50 or more objects and "first attempts to find a less intuitive ordering procedure were made in early 50s by Skrzywan (1952) and in the Wrocław school of math (so-called "Wrocław dendrit", a kind of graph accompanying the Czekanowski's diagram)". Decades later, Szczotka (1972) published a method and developed a computer program for that purpose and several applications to economics were reported by Pluta

(1980). Recent algorithmic advances on the research of Czekanowski's diagram include a genetic algorithm proposed by Soltysiak and Jaskulski (1999).

Another interesting methodological lineage exception in the field of anthropology was the work of Carneiro (1962), who performed a seriation of a two-way two-mode data table of nine societies and eight culture traits. He developed a scale analysis method for the study of cultural evolution, based on a renowned concept of Guttman scale, applied typically to statistical surveys. An example of the initial data used by Carneiro and the rearranged "scalogram" is presented on Tables 2.2 and 2.3, respectively.

Table 2.2 Initial data used by Carneiro

	Kuikuru	Anserma	Jivaro	Tupinamba	Inca	Sherente	Chibcha	Yahgan	Cumana
Social stratification	–	+	–	–	+	–	+	–	+
Pottery	+	+	+	+	+	–	+	–	+
Fermented beverages	–	+	+	+	+	–	+	–	+
Political state	–	–	–	–	+	–	+	–	–
Agriculture	+	+	+	+	+	+	+	–	+
Stone architecture	–	–	–	–	+	–	–	–	–
Smelting of metal ores	–	+	–	–	+	–	+	–	–
Loom weaving	–	+	+	–	+	–	+	–	+

Table 2.3 Rearranged Carneiro's "scalogram"

	Yahgan	Sherente	Kuikuru	Tupinamba	Jivaro	Cumana	Anserma	Chibcha	Inca
Stone architecture	–	–	–	–	–	–	–	–	+
Political state	–	–	–	–	–	–	–	+	+
Smelting of metal ores	–	–	–	–	–	–	+	+	+
Social stratification	–	–	–	–	–	+	+	+	+
Loom weaving	–	–	–	–	+	+	+	+	+
Fermented beverages	–	–	–	+	+	+	+	+	+
Pottery	–	–	+	+	+	+	+	+	+
Agriculture	–	+	+	+	+	+	+	+	+

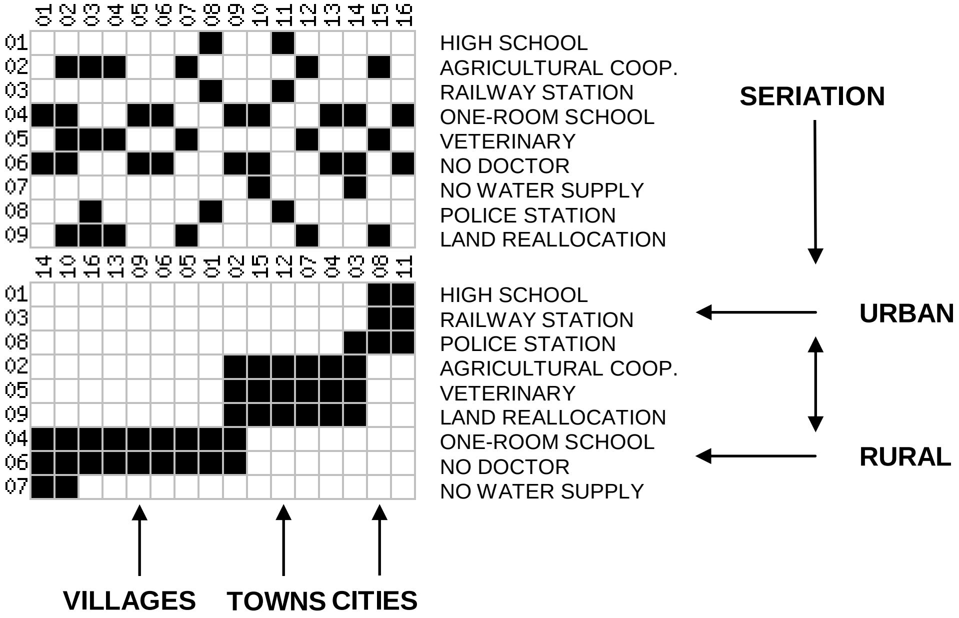
Using the rearrangement procedure proposed in Carneiro's paper, it is clear that, one does not need to evaluate all *n!•m!* permutations of the table. However, the solution is far from trivial in case of larger tables with noisy and missing data. An instructive discussion on non-perfect scales and unilinear evolution was included in the paper.

Carneiro's work serves as an interesting example of how fundamentally different methodological foundations can lead to methods with similar goals and results.

2.3 Cartography and graphics

From the perspective of this thesis, it would be hard to overestimate the importance of a monograph, *Semiology of Graphics*, published by a French cartographer, Jacques Bertin (1967). The main arguments and statements of the presented methodology are accompanied by fine-grained illustrative examples. Despite his main area of expertise, his goal was to propose a concept of a "reorderable matrix" (*matrice ordonnable*) as a convenient generic tool for analyzing different structures and systems. Reordering of the rows and columns of matrices was performed on two-mode data tables, with a strong emphasis on visualization and value encoding aspects. He stressed the importance of simultaneous availability of three information levels in every effective visual display of data, e.g. a classical Bertin's (1981, p.33) example of townships in Fig. 2.3. One should be able to find an immediate visual reply to:

- questions asked specifically about the details of data presented in rows and columns (e.g. Does township *'08'* have a railway station? Which townships have police stations?);
- local patterns found in the data (e.g. Where there is no water supply, there are no high schools);
- global patterns and trends found in the data (e.g. We are able to identify the transformation of rural areas to urban and what changes take place in the characteristics supporting such a transition).



	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	
01								■			■						HIGH SCHOOL
02		■	■	■			■					■			■		AGRICULTURAL COOP.
03								■			■						RAILWAY STATION
04	■	■			■	■			■	■			■	■		■	ONE-ROOM SCHOOL
05		■	■	■			■					■			■		VETERINARY
06	■	■			■	■			■	■			■	■		■	NO DOCTOR
07										■				■			NO WATER SUPPLY
08			■					■			■						POLICE STATION
09		■	■	■			■					■			■		LAND REALLOCATION

	14	10	16	13	09	06	05	01	02	15	12	07	04	03	08	11	
01															■	■	HIGH SCHOOL
03															■	■	RAILWAY STATION
08														■	■	■	POLICE STATION
02									■	■	■	■	■	■			AGRICULTURAL COOP.
05									■	■	■	■	■	■			VETERINARY
09									■	■	■	■	■	■			LAND REALLOCATION
04	■	■	■	■	■	■	■	■	■								ONE-ROOM SCHOOL
06	■	■	■	■	■	■	■	■	■								NO DOCTOR
07	■	■															NO WATER SUPPLY

Figure 2.3 Bertin's (1967, 1981) example of matrix reordering

Bertin had some influences from the works of Carneiro (Bertin, 1967, p.196) and Elisseeff's (1965) scalogram (Bertin, 1981, p.58), but the systematic principles he developed for matrices were far more advanced – categorical and continuous values of data were supported and the emphasis was to manipulate the matrices for maximizing the perception of regularity and relationships, regardless of what the final structural pattern would look like. However, the only thing that was not there was the mathematical treatment of matrix permutations and automatic procedure to evaluate and perform the matrix reordering. Bertin was prepared to believe that after defining the problem, i.e. composing the matrix, data could be processed by machine, but he himself was performing it manually using visual perception. One can find an overview of several mechanical tools and equipment to aid an analyst to perform those datasets from the two subsequent monographs (Bertin, 1967, 1981). Bertin (1981, p.31) considered the comfortable limits of the proposed graphic information-processing to be 120 x 120 with reorderable matrix, 500 x 100 with experimental equipment and 1000 x 30 with the "matrix-file" approach, where one dimension (mode) is fixed to be non-permutable.

Bertin's work is highly recognized within the communities of human-computer interaction (HCI) and information visualization, however, with the main emphasis not on reorderable matrices, but fundamentals of visual perception and graphic information processing. One of the most cited recent applications and enhancement of Bertin's ideas of reorderable matrices and especially the "matrix-file" approach, is the *Table Lens* (Rao & Card, 1994), which incorporated interactive elements for better usability together with the general focus+context mantra of the information visualization community.

Bertin (1981, p.15) was convinced that a set of pie charts is one of the most useless graphical constructions. However, Friendly *et al.* (2002, 2003) demonstrated that combining matrix cell shading with small pie charts to present symmetric correlation matrices can result an interesting visualization. The idea of the *correlogram* or *corrgram* was to use color and intensity of shading in the lower triangle of a symmetric matrix and circle symbols in the corresponding cells of the upper triangle.

Siirtola *et al.* have recently published, in the information visualization community, several discussions on the interaction (Siirtola, 1999, 2004; Siirtola & Mäkinen, 2005) and algorithmic (Mäkinen & Siirtola, 2000, 2005) aspects of Bertin's reorderable matrices and developed a tool for combining visualization of parallel coordinates with the reorderable matrix (Siirtola, 2003).

Chen *et al.* (2008) most recently published a *Handbook of Information Visualization*, with several chapters presenting discussions and examples about matrix reordering and visualization. It reflects, among others, his own contributions on the *generalized association plots* (Chen, 2002; Chen *et al.*, 2004), which was based on the idea of visualizing the two-way one-mode matrix after seriation, using different shading to represent the values of proximity.

2.4 Sociology and sociometry

> *"When you can measure what you are speaking of and express it in numbers you know that on which you are discoursing. But when you cannot measure it and express it in numbers, your knowledge is of a very meagre and unsatisfactory kind."*
> *Lord Kelvin*

If one has to draw a distinction between "hard" and "soft" science, sociology and social sciences in general are as often as not referred to as "soft" sciences. However, such judgement intuitively seems to be self-contradictory — it is a study of social relations, behaviour and its underlying structure that deals with answering difficult questions about one of the most complex systems. Lack of a decent mathematical apparatus for explaining different phenomena therefore itself is not a sufficient condition for such a classification.

One of the first influential attempts to introduce a rigorously measurable and new way of thinking was by Jacob L. Moreno with his classic *Who Shall Survive?* (1934[2]). It started a new branch in sociology now known as sociometry, which stressed the importance of quantitative and mathematical methods for understanding social relationships and catalyzed several works interesting in the context of the current thesis.

Forsyth and Katz (1946) were the first to introduce an approach of rearranging the rows and columns of the *sociomatrix* for a better presentation of the results of sociometric tests. There seems to be neither an obvious nor an implicit influence of previous works with rearranging the matrices and the motivation for method development seems to descent directly from Moreno's work on sociograms. Forsyth and Katz credited the sociogram to be clearly advantageous over verbal descriptions and relationship listings, but "confusing to the reader, especially if the number of subjects is large". Katz (1947) also argued that "the sociometric art has simply progressed to the point where pictorial representation[3] of relationships is not enough" and quantifications of the data should be seeked. It was hoped that the sociomatrix and the development of methods for analyzing the matrices would fulfil that gap.

The concept and construction of the sociomatrices (also interrelation matrices) was already an accepted research practice in sociometry. Jennings (1937) analyzed leadership and isolation structures, their variations and illustrated choices of preferences between individuals with an adjacency matrix. Dodd (1940) wrote a paper about interrelation matrices with a purpose "to apply

[2] Aside this chronologic fact, interested readers are directed to the revised edition which is a strongly enhanced version with more background information and available online free of charge (Moreno, 1953).

[3] Sociogram drawing was a manual process and there were still decades left until automatic graph drawing algorithms started to emerge in computer science, moreover across disciplines.

algebra to the data of inter-personal relation in order to increase both the precision and the generality of any analyses or syntheses of those data".

A sociomatrix is an asymmetric NxN one-mode two-way adjacency matrix reflecting the underlying structure of a directed or undirected graph, which is called a sociogram (see Fig. 2.4) in this context. According to Wasserman and Faust (1994), sociomatrix is the most common form for presenting social network data.

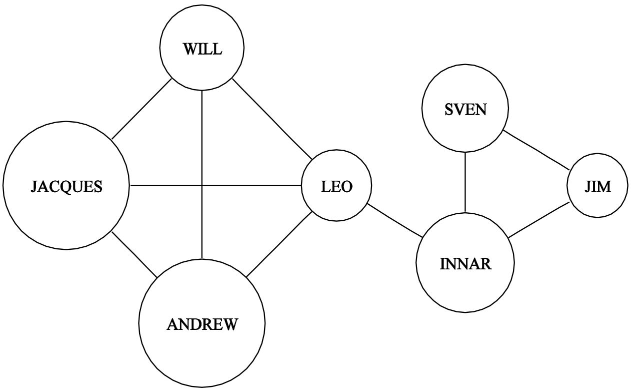


Figure 2.4 Sociogram with undirected relations

Essence of the method which Forsyth and Katz (1946) built upon the sociomatrix consisted of "re-arranging the rows and columns in a systematic manner to produce a new matrix which exhibits the group structure graphically in a standard form." We constructed a simple example to demonstrate the concordance between a sociogram (Fig. 2.5) and a corresponding sociomatrix before (Fig. 2.5, left) and after (Fig. 2.5, right) row and column permutation, where one can directly identify two distinct groups of people and a seamless transformation from one cluster to another. Instead of a binary sociomatrix with

	JIM	LEO	ANDREW	SVEN	WILL	JACQUES	INNAR
JIM	X	0	0	1	0	0	1
LEO	0	X	1	0	1	1	1
ANDREW	0	1	X	0	1	1	0
SVEN	1	0	0	X	0	0	1
WILL	0	1	1	0	X	1	0
JACQUES	0	1	1	0	1	X	0
INNAR	1	1	0	1	0	0	X

	JACQUES	ANDREW	WILL	LEO	INNAR	SVEN	JIM
JACQUES	X	1	1	1	0	0	0
ANDREW	1	X	1	1	0	0	0
WILL	1	1	X	1	0	0	0
LEO	1	1	1	X	1	0	0
INNAR	0	0	0	1	X	1	1
SVEN	0	0	0	0	1	X	1
JIM	0	0	0	0	1	1	X

Figure 2.5 Symmetric sociomatrix before (left) and after (right) permutation

undirected single relations, Forsyth and Katz (1946) proposed a matrix permutation (see Fig. 2.6 for reconstruction of their results) with multiple directional relations, denoting positive choices with "+" and negative choices or rejections with "–". However, those relations were mutually exclusive (otherwise tensors would have to be used instead of matrices), so they can be considered as different values for a single relation.

	LN	RL	HN	TA	WR	PC	HT	BU	SO	WT	SV	LP	ES	BS	ET	RA	WT	GU	HM	WN	LU	BR	JH	RG	CD
LN	X	+			+						+	+			—			—							
RL	+	X	+		+	+									—										
HN		+	X		+												—	+							
TA				X	+	+	+				+														
WR					X																			—	
PC		—		+	+	X	+	+							—							+	—	+	—
HT				—	+	+	X	+	+				+		+		—	—	—					+	+
BU				—	+	+	+	X	+	+	+				—							—			
SO					+		+	+	X	+			+		+										
WT								+	+	X	+								—			+			
SV								+			X		+	—			+								
LP											+	X	+												
ES							+				+	+	X	+	—	+									
BS				—							+	+	+	X											—
ET				—			+		+						X							+			—
RA		+			+								+		—	X									
WT			+				+				+						X				+				
GU																		X							
HM				—	+		+			+	+								X			—	—		—
WN			—															—		X	+				
LU											+									+	X	+			
BR	+					+				+												X	+		+
JH																			—			+	X	+	
RG					+	+	+				+												+	X	
CD					+		+					+										+			X

Figure 2.6 Shaded one-mode two-way asymmetric permutated Forsyth-Katz(1946) sociomatrix

Self-relations or "self-choices" for a particular relation are usually[4] (e.g. Jennings, 1937; Forsyth and Katz, 1946; Beum and Brundage, 1950; Moody, 2001) undefined, serve as an identifier of rows/columns (Katz, 1947) or, like in this case, marked with 'X' along the main diagonal of the sociomatrix. Such common practice is clearly not accidental – sociometry is, after all, a study focusing on *inter-human* relations. Wasserman and Faust (1994) point out that

[4] However, from the unification point of view, there seems to be no explicit reason why self-relations should be undefined or excluded from the analysis and we suggest to replace them with positive choices, at least during the reordering procedure.

"there are situations in which self-choices do make sense", but they are typically assumed to be undefined "since most methods ignore these elements" (Wasserman and Faust, 1994) or not to relate to themselves (Moody, 2001).

Moreno (1946) agreed that a sociogram and a sociomatrix both offered certain advantages and supplemented each other, but could not have a concurring opinion with the claim by Forsyth and Katz that their sociomatrix was superior and more objective in its presentation than the sociogram. He emphasized that "already pair relations are hard to find [from the sociomatrix], but when it comes to more complex structures as triangles, chain relations, and stars, the sociogram offers many advantages." Several of those shortcomings are perfectly justified criticism even today. However, a chain of relations is actually *the* most important thing that such matrix shuffling procedures bring out, which might even be difficult to detect on a sociogram with thousands of objects. Katz (1947) continued with the work of simultaneous reordering of the rows and columns of a sociomatrix with the accentuation of quantitative approaches and better mathematical formulation of the problem. A sociomatrix was formalized with bringing in the notation of zeros for indifference/no response and introducing permutation matrices. It seems there was, however, one contradiction. He used the permutation matrix and its transpose for multiplying the original matrix on the left and right, but even with N x N matrix, if it contains asymmetric relations (which is typical for sociometric tests), it would be far more reasonable to find two different permutation matrices to maximize the concentration of positive choices. Furthermore, considering the overall notations of this thesis, asymmetric sociomatrices should be taken as two-mode two-way matrices for direct compatibility reasons.

The first method for systematically rearranging a sociomatrix was presented half a decade later by Beum and Brundage (1950). By systematically, we consider methods which have single interpretations on every step of the procedure and do not depend on human visual perception or decision making. Forsyth and Katz (1946) also proposed a simple set of rules for iterative enhancement of the matrix reordering and approximate maximization of similarities, but included several abstract and intangible steps. Borgatta and Stolz (1963) implemented the Beum-Brundage procedure in FORTRAN-II, which handled matrices up to 145x145 variables and included several interesting additional features like de- emphasizing smaller values in a matrix.

Nowadays, quantitative approaches in sociology have advanced enormously, with a strong community and a vast number of contributions in the area of social network analysis. However, as far as the author is aware, a matrix reordering paradigm is not very commonly used. There is a reasonable amount of research originating from the seminal work on blockmodels (White, Boorman and Breiger, 1976), which includes alike matrix reordering procedures, but with the goal of structural aggregation. Consequently, for a blockmodeling community, acquiring the overall perfect seriation of objects is not important and even computationally inefficient.

2.5 Psychology and psychometry

It is quite hard to draw a rigid line between the research of psychometry and sociometry, as several authors have published in both fields and there has been significant cross-influence from both communities. However, for the moment, the community of psychometry has developed a compact and focused track of research on the problem of seriation with a strong consensus on common terminology and a general understanding of the problem.

Hubert (1974, 1976) was one of the early adaptors of seriation techniques in psychology, considering a subjects-by-item response matrix. He performed analyses on both, one-mode and two-mode matrices filled with zero-one and integer values and was using permutation procedures based on the algorithms and approaches developed for archaeological seriation.

Besides the archaeological background, Hubert's work was influenced by psychological scaling research carried out by Coombs (1964), who proposed the *parallelogram analysis* for searching the patterns in matrices. Coombs (1964, p.75) called the concept of reordering objects the "order *k/n*" analysis, which was a natural extension to the procedure, what he referred to as "pick *k/n*".

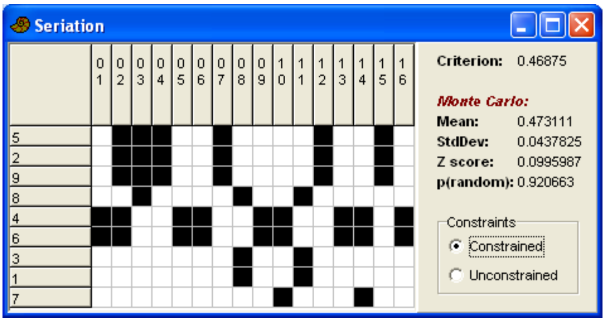
Having influences from the taxonomy of data developed by Coombs and terminology proposed by Tucker (1964), Carroll and Arabie (1980) proposed a taxonomy of data and models for multidimensional scaling, where the taxonomies of data and models were treated separately. To date, it can still be considered a *de facto* taxonomy to use with multidimensional scaling and related methods, e.g. seriation.

Comprehensive reviews and references for recent advances can be found from the subsequent monographs on combinatorial data analysis by Hubert *et al.* (2001) and Brusco *et al.* (2005) and from a structured overview of two-mode clustering (Van Mechelen *et al.*, 2004). It is interesting to observe that main contributions towards taxonomization of the methods of seriation and the methods related to seriation have come from scholars working in the area of psychology.

2.6 Ecology

Traditions of seriation (which is often referred to as *ordination*) and clustering methods in the disciplines of ecology have strong roots in and descendance from the works of the Polish botanist and politician Kulczynski (1927), who studied plant associations using the matrix coding and visualization approach developed by Czekanowski (1909). Kulczynski replaced the values of the upper triangle of a symmetric similarity matrix with different shadings and patterns of a cell (for a reprint of the diagram with a discussion of the ecological application, the reader is referred to (Legendre & Legendre, 1998, p.373)). This kind of approach for shading was somewhat different from the first visual coding proposed by Czekanowski, who did not preserve the initial similarity values and transformed the (dis)similarity matrix to an asymmetric form after recoding and shading the values.

In ecology, seriation was often considered the best practice to perform clustering without explicitly distinguishing between those two techniques. The application of seriation was also more far-spread than "classical" clustering techniques used in other disciplines. This may also be the reason why the tools for end-user to perform seriation had the highest representation in the packages developed for ecological studies. It is a significant sign of the maturity level of the discipline from the perspective of seriation methodology development and distinguishes it strongly from other fields. We have performed an illustrative (Fig. 2.7 & 2.8) experiment using the PAST (Hammer & Harper, 2005) software for data analysis, which was "originally aimed at paleontology but is now also popular in ecology and other fields" (Hammer, 2008), using the classical township dataset introduced in another field by Bertin (1967) to depict the universality and cross-applicability of seriation methods.



	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
5		■	■	■			■					■			■	
2		■	■	■			■					■			■	
9		■	■	■			■					■			■	
8			■					■			■					
4	■	■			■	■			■	■			■	■		■
6	■	■			■	■			■	■			■	■		■
3								■			■					
1								■			■					
7										■				■		

Criterion: 0.46875

Monte Carlo:

Mean: 0.473111

StdDev: 0.0437825

Z score: 0.0995987

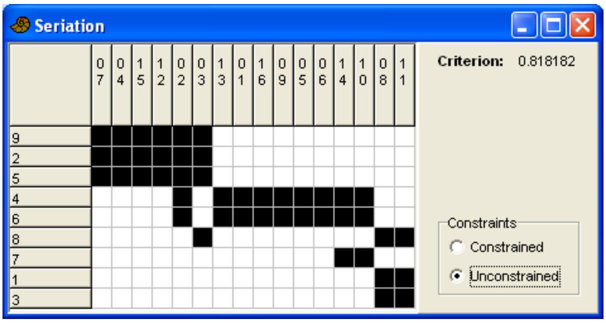
p(random): 0.920663

Constraints

● Constrained

○ Unconstrained

Figure 2.7 PAST results for row seriation (option: constrained).



	07	04	15	12	02	03	13	01	16	09	05	06	14	10	08	11
9	■	■	■	■	■	■										
2	■	■	■	■	■	■										
5	■	■	■	■	■	■										
4					■		■	■	■	■	■	■	■	■		
6					■		■	■	■	■	■	■	■	■		
8						■									■	■
7													■	■		
1															■	■
3															■	■

Criterion: 0.818182

Constraints

○ Constrained

● Unconstrained

Figure 2.8 PAST results for row and column seriation (option: unconstrained)

Besides the above-mentioned PAST software, seriation is also available in the clustering package Clustan (Legendre & Legendre, 1998, p.372) and described by the authors of the software package Primer-E (Plymouth Routines

In Multivariate Ecological Research) as "a simple re-ordering of columns (samples) and rows (species)" that can be an effective way to display "groupings or gradual changes in species composition" (Clarke & Warwick, 2001, p.7-2). Clarke and Warwick also introduced an interesting method for textual coding of the abundance values.

The abundance codes proposed by Clarke and Warwick (2001,p.7-2) are: "*1=1, 2=2, ..., 9=9, a=10-11, b=12-13, c=14-16, ..., s=100-116, ..., z=293-341, A=342-398, ..., G=1000-1165, ..., Y=8577-9999, Z≥10000 (For X≥4 this is the logarithmic scale int[15($\log_{10}$X)-5] assigned to 4-9,a-z,A-Z, omitting i,l,o,I,L,O)."* The result of this coding is an interesting visualization of a two-way two-mode matrix, using ASCII graphics instead of plotting the pixels, which is obviously a reasonable workaround for non-binary datasets on text-only displays and printers, but infeasible to apply with bigger matrices.

Legendre and Legendre (1998) have published a monograph about *numerical ecology*, which includes a comprehensive overview of data analysis methods in ecology, including discussions, examples and applications of seriation. Miklós *et al.* (2005) have recently published a paper about rearrangement of ecological data matrices, using Markov chain Monte Carlo simulation, which also includes a representative set of references to seriation methods in ecology.

2.7 Biology and bioinformatics

Seriation methods in biology have similar methodological roots to those discussed in the previous section covering the discipline of ecology. The paradigm of data analysis using the reordering of rows and columns was introduced to the community of biologists by the infamous monograph of *Numerical Taxonomy* by Sokal and Sneath (1963), which faced a lot of controversy for the strong statements and criticism against the traditional way of creating taxonomies in biology. Sokal and Sneath (1963, p.176) introduced matrix reordering techniques, using the name "differential shading of the similarity matrix", and referred to the result of the seriation procedure as "a *cleaned up* diagram". They saw the purpose of rearranging the rows and columns in the search for the "optimum structure in the system" and proposed a procedure suggested by Robinson (1951) to be suitable for this goal. It is interesting to observe, that, while the systematic approach followed the methodological lineage of Petrie (1899), the shading and general matrix visualization approach has rather a strong resemblance to the traditions of Polish scholars Czekanowski and Kulczynski. Their works were acknowledged and cited by Sokal and Sneath, yet, not in the context of matrix reordering but due to similarity coefficient contributions.

Recently, a related concept of *biclustering* (also *co-clustering* or *two-mode clustering*) has gained acceptance in experimental molecular biology, mainly, to cope with the latest developments in microarray and gene expression research. A typical dataset for reordering rows and columns is a two-way two-mode matrix with continuous data. In fact, matrix reordering techniques have been introduced to gene analysis decades ago (Mirkin and Rodin, 1984; Mirkin, 1996), but did not attract greater attention before the prominent contribution by Eisen *et al.* (1998), who proposed a visual display for genome-wide expression patterns by combining the dendrogram resulting from hierarchical clustering with the initial data matrix from DNA microarray hybridization. The data matrix was reordered using the order of the leaves in the clustering dendrogram acquired separately for both modes of the matrix. A decade later, the paper by Eisen *et al.* (1998) had accumulated well over 7000 citations, which gives a good impression of the influence of such an approach. Another important publication towards making data analysis and visualization of gene expression data popular was published by Cheng and Church (2000) who introduced the term "biclustering" to gene expression analysis and proposed a node-deletion algorithm to search for biclusters. Such mathematical treatment and introduction of two-mode clustering to the bioinformatics community attracted hundreds of follow-up articles, discussions and algorithms. However, from the overall picture of the seriation research, it seems that the community of bioinformatics has not yet established a general consensus concerning the goals and focal emphases of biclustering results. Several surveys and evaluations of biclustering methods have been published lately (Madeira & Oliveira, 2004; Tanay *et al.*, 2006; Prelić *et al.*, 2006), but there seems to be little work done towards taxonomization of the contributions and the present reviews rather serve as bibliographical lists with brief comments and hubs of references. The most important open question is whether the essential emphasis of the goals is on clustering (objects are assigned to groups) or on seriation (objects are optimally rearranged and assigned to a position within a sequence). If the goal is to perform clustering simultaneously (sequentially) over two sets of objects with the motivation of finding local clusters that could otherwise be left unnoticed, the community should strongly head towards collaboration and consolidation with diclique decomposition (Haralick, 1974) and formal concept analysis (Wille, 1992; Ganter & Wille, 1999) research. If the goal is to augment the human analyst to enable better visual perception of relationships within the data for better biological insight, the use of classical results of hierarchical clustering are not efficient in establishing a seriation of the rows and columns which introduces most of the regularity within the data. As illustrated in the introduction, hierarchical clustering and dendrograms choose the order of succeeding elements at every split of the tree arbitrarily or according to the order of appearance in the data source. However, there are "2^{n-1} linear orderings consistent with the structure of the tree" (Bar-Joseph *et al.*, 2001) generated by hierarchical clustering. An arbitrary selection from all possible orderings works as a strong heuristic, but most probably will not result in satisfactory results if the similarity between neighbouring elements and high overall regularity within the data matrix is important. To remedy this situation, several authors have proposed additional procedures to perform optimal leaf ordering of the dendrogram (Vandev & Tsvetanova, 1995, 1997; Bar-Joseph *et al.*, 2001, 2003), which is already a step away from clustering towards seriation. It is important to acknowledge that it is not mandatory to arrive at such results via hierarchical clustering using two steps. Caraux *et al.* (2005) have developed a software package *PermutMatrix*, where data analysis of gene expression profiles is performed using the methods and interpretation of seriation which we consider concurring with our definition of seriation. The reader is also referred to an earlier comprehensive overview of matrix reordering techniques published by Caraux (1984).

2.8 Group technology and cellular manufacturing

The machine-group formation problem and cellular manufacturing represents a community, applying block diagonal seriation with definitely the largest number of technical and algorithmic contributions towards a more optimal solution and formal definition. Machine-group formation is one of the essential steps in Burbidge's (1961) analytic "new approach" to production, which later became known as the *production flow analysis* (Burbidge, 1963). Production flow analysis is a manufacturing philosophy and technique for finding families of components and groups of machines. It was initially considered (McAuley, 1972) a technological enabler for group technology, but those terms were later often used interchangeably. Burbidge (1971) emphasized that it is "concerned solely with methods of manufacture, and does not consider the design features or shape of components at all". Although the general idea of product flow analysis to classify the components into product families was introduced already in the early 60s by Burbidge (1961) and independently by Mitrofanov (1959, 1966), the machine/part incidence matrix was first explicitly presented by Burbidge (1971). The results were obtained manually (Burbidge, 1977), the first attempt to develop a non-intuitive algorithm was by McAuley (1972), who also stated that "at present, as far as is known, the only way of finding the groups of machines and families of parts is to rearrange the rows and columns of the matrix, by hand, until the pattern [...] is obtained". McAuley's solution was influenced and based on the works of Sokal and Sneath (1963) and Kendall (1971). However, most algorithmic approaches started to appear after the rank order algorithm was proposed (King, 1980; King & Nakornchai, 1982), which worked directly on the initial matrix. This algorithm, among other similar approaches not requiring the conversion of two-mode matrix to one-mode matrix, is classified as *an array-based clustering method* within the cell formation research community.

Machines \ Parts	1	2	3	4	5	6
1	0	1	0	1	1	0
2	1	0	1	0	0	1
3	0	1	0	1	1	0
4	0	0	1	0	0	1

Machines \ Parts	3	6	1	4	2	5
4	1	1	0	0	0	0
2	1	1	1	0	0	0
3	0	0	0	1	1	1
1	0	0	0	1	1	1

Figure 2.9 McAuley's (1972) example of a machine/part matrix

The machine-group formation (or machine-part cell formation) problem is formulated as a binary part-machine incidence matrix *A*, where $a_{i,j}$=1 means that the machine *i* is required to process part *j* and $a_{i,j}$=0 otherwise. We have chosen a simple example (Fig. 2.9) from McAuley's (1972) paper to demonstrate a typical machine/part matrix and how groups emerge after reordering the rows and columns. A zero (often referred as *void*) element ($a_{4,1}$) within an emerged group depicts the reason why reordering and matrix display is used rather than clustering or any other classical partitioning method – groups often have irregular shapes and boundaries and the goal is not only to find groups but also minimize the void elements, exceptional elements (elements which lie outside the blocks on the diagonal) and bottleneck machines (elements which obstruct (the most) the decomposition into independent blocks and subsystems). This, however, makes the order within the cluster important and excludes, therefore, the possibility of using classical clustering methods. This kind of additional domain-specific structural properties might have other interesting semantic interpretations in other fields as well.

Grouping *efficiency* (Chandrasekharan & Rajagopalan, 1989) and grouping *efficacy* (Kumar & Chandrasekharan, 1990) are the two most frequently used measures to evaluate the quality of the cell formation solution and have had a strong impact on introducing rigorous benchmarking expectations to all new contributions in the field. They are based on measuring the quality of diagonalization by enumerating the number of void elements (zeros) in the block along the diagonal and exceptional elements (ones) away from the formed cells. Such a judgement, however, presupposes that we know (or have predicted) the number of two-mode clusters (there is a similar common problem of finding the correct *k* in the classical clustering paradigm as well) and have identified the cluster boundaries correctly. In addition to those pretentious assumptions, the measures are extremely sensitive to the format of the solution. For example, inversions (flipping horizontally or vertically) and rotations of the matrix, which essentially do not alter the structure in the matrix, could be considered equal solutions, can, however, have a fatal effect towards the ability of those measures to detect a good solution. Other, more universal measures that can find any structural pattern amenable to the dataset are, therefore, more favourable in our context (e.g. the measure of effectiveness proposed by McCormick *et al.* (1972), which will be discussed in the subsequent section).

The machine-part cell formation problem has been solved, using, among others, Hamiltonian path and other graph theoretic approaches (Askin & Chiu, 1990; Askin *et al.*,1991; Rajagopalan & Batra, 1975), integer programming (Gunasingh & Lashkari, 1989), fuzzy clustering (Chu & Hayya, 1991), evolutionary approaches (Hwang & Sun, 1996; Onwubolo & Mutigi, 2001;Goncalves & Resende, 2004), TSP (Balakrishnan, 1995), neural networks (Kusiak & Chung, 1991; Kaparthi & Suresh, 1992; Rao & Gu, 1993; Chu, 1997), branch-and-bound methods (Kusiak & Cheng, 1990; Kusiak, 1991;Kusiak *et al.*, 1993) and with simulated annealing (Venugopal & Narendan, 1992). However, it seems, unfortunately, that most of the algorithms are not known and acknowledged in the other fields applying two-mode clustering and other methods to analyze binary datasets. In addition, Park and Wemmerlöv (1994) have developed an artificial shop structure generator for cell formation research, which is also usable and well-applicable in all other fields doing research on seriation and two-mode clustering.

The reader is referred to dedicated surveys (Wemmerlöv & Hyer, 1989; Singh, 1993; Mosier *et al.*, 1997) and results of comparisons (Chu & Tsai, 1990; Miltenburg & Zhang, 1991; Kandiller, 1994) for a further analysis of the contributions in the field. A comprehensive overview and discussion of the research issues in cellular manufacturing is available in Wemmerlöv and Hyer (1987), where the applicability, justification and implementations of cellular manufacturing systems are being discussed.

2.9 Operations research

Operations research is an interdisciplinary branch of applied mathematics and other scientific methods for determining optimization strategies for the efficient management of organizations. Potential contributions of seriation and matrix reordering techniques originating from this discipline are, therefore, inherently more abstract and contain less domain-specific insights than the ones presented in previous sections. Such settings, on the other hand, enabled McCormick *et al.* (1969, 1972) to contribute, what retrospectively represents an important milestone towards making seriation methods universally applicable and less sensitive to structural pattern assumptions. McCormick *et al.* developed a seriation approach for matrix reordering called *bond energy algorithm* (BEA) to identify natural groups in complex data arrays. It was a nearest-neighbor sequential-selection suboptimal algorithm with the main intention (McCormick *et al.*, 1972) to assist "the analyst who wishes to begin understanding the interactions in a complex system." This algorithm can be considered a breakthrough for matrix reordering techniques. As far as the author of this dissertation is aware, no algorithms were published before 1969 that could perform such a universal reordering of the initial dataset for both one-mode (object-by-object or NxN) and two-mode (object-by-variable or NxM) datasets. One of the strongest properties of the BEA algorithm is not having any assumptions of the underlying structure and being less sensitive to noise in the data than its precedents, making the approach more practical in real-world scenarios.

McCormick *et al.* (1972) proposed a measure of effectiveness (ME) of an array as "the sum of bond strengths in the array, where the bond strength between two nearest-neighbor elements is defined as their product". For any non-negative two-mode matrix A, the ME is given by:

$$ME(A) = \frac{1}{2}\sum_{i=1}^{i=M}\sum_{j=1}^{j=N} a_{i,j}\left[a_{i,j+1} + a_{i,j-1} + a_{i+1,j} + a_{i-1,j}\right] \tag{2.3}$$

(with the convention $a_{0,j} = a_{M+1,j} = a_{i,0} = a_{i,N+1} = 0$)

As noted by McCormick *et al.* (1969, 1972) and Lenstra (1974), the given problem can be reduced into two separate optimizations (one for finding the order for columns, the other for rows; we have slightly modified the notation to make it more coherent with other measures in the next chapter):

Let $\Pi = \{\pi(1), \pi(2), \dots, \pi(M)\}$ denote all *M!* permutations of (1,2,…,*M*) and $\Phi = \{\phi(1), \phi(2), \dots, \phi(N)\}$ respectively over all *N!* permutations of (1,2,…,*N*) with the conventions $\pi(0) = \pi(M+1) = a_{i,0} = 0$ and $\phi(0) = \phi(N+1) = a_{0,j} = 0$:

$$\text{argmax}_{\Pi} \sum_{i=1}^{i=M}\sum_{j=1}^{j=N} a_{\pi(i),j}\left[a_{\pi(i-1),j} + a_{\pi(i+1),j}\right] \tag{2.4}$$

$$\text{argmax}_{\Phi} \sum_{i=1}^{i=M}\sum_{j=1}^{j=N} a_{i,\phi(j)}\left[a_{i,\phi(j-1)} + a_{i,\phi(j+1)}\right] \tag{2.5}$$

Lenstra (1974) pointed out that the bond energy algorithm is equivalent to the well-known traveling salesman problem (TSP), but, actually, the interpretation of the measure of effectiveness optimization as two traveling salesman problems was already shown earlier by the authors in the publicly available technical report (McCormick *et al.*, 1969, p.82). Climer and Zhang (2006) have recently presented an approach for converting the matrix reordering problem to one-mode TSP format with additional *k* dummy cities for cluster boundary detection. The solution provided by the TSP solver is used to rearrange the data matrix. They have reported better results according to the criteria of measure of effectiveness (ME) for the examples presented by McCormick *et al.* (1972), using the bond energy algorithm. However, several authors (Arabie & Hubert, 1990, Arabie *et al.*, 1990, Chu & Tsai, 1990) have revisited the original algorithm, investigated its properties in detail and found that the bond energy algorithm provides near-optimal results in different settings and is not trying to fit any specific structural pattern in the data, therefore sometimes outperforming even dedicated and less universal domain-specific algorithms.

In addition to the bond energy algorithm, another, less known algorithm was developed by the same group — the moment ordering algorithm (McCormick *et al.*, 1969; Deutsch & Martin, 1971). Deutsch and Martin consider the algorithm as a tool "for analyzing arrays of data whose underlying organization is known but which is hoped that there is a single underlying variable, according to which the rows and columns of the arrays can both be arranged in meaningful one-dimensional orders." Both of the algorithms provide seriation in the data, but the latter searches for a solution to position all the values along the diagonal.

2.10 Related local research

The algorithmic approach considered and enhancements presented in this thesis are based on the heuristics and approaches developed by Vyhandu (1979, 1980, 1981, 1989) and Mullat (1976a,1976b,1977). Mullat (1976a) proved that on every "monotone system", it is possible to define and maximize the function

$$m(S) = \min_{i \in S} m(i, S) \qquad (2.6)$$

and minimize

$$M(S) = \max_{i \notin S} m(i, S) \qquad (2.7)$$

in a polynomial number of calculations of the real-valued entity-to-set weight function $m(i,S)$, if the following conditions are satisfied[5]: S denote subsets of the system W, "for each i,S,S', $m(i,S) \leq m(i,S')$, whenever S is a subset of S'."

[5] We use the refined notations and compact presentation of the conditions presented by Mirkin and Muchnik (1996) for the monotone systems.

Some recent publications and monographs covering elements from this branch of heuristics and methods include Mirkin and Muchnik (1996), Mirkin (1996, Section 4.2.1) and Tyugu (2007, Section 3.5.4).

In this thesis, we are not interested in optimizing those criteria or finding that extremal subset *S*. However, the strategy for those polynomial steps for greedily optimizing the above mentioned functions and a suitable choice of the entity-to-set function are essential for the seriation methods considered.

The greedy approach, enabled by the properties of monotone systems, was applied to matrix reordering problems by Vyhandu (1980). Vyhandu's matrix reordering techniques can, similarly to McCormick's (1972) bond energy algorithm, reduce the problem into two separate tasks for reordering the rows and columns. Vyhandu (1980, 1989) demonstrated that specific entity-to-set weight measure called *conformity* (Vyhandu, 1981) is favourable for such task. Actually, sometimes reordering according only to initial conformity weights (establishing, therefore, the so-called *conformity scale* in the dataset) can produce satisfactory solutions, as discussed in Chapter 5.

Compared to other matrix seriation methods introduced in previous sections, this family of reordering methods has the following properties:

- a direct exploratory data analysis of two-mode two-way matrices without converting them to one-mode;
- support for categorical data, high and low numeric values are treated equally;
- the result is not restricted to one specific structural pattern in the output (e.g. block diagonal or checkerboard form);
- zeros are not considered as missing values, but used for structural balancing and normalization purposes;
- results can be interpreted as a typicality scale of objects along the continuum.

We will revisit and further discuss the conformity entity-to-set weight function and monotone systems approach on matrix seriation in Chapter 4.

3 Seriation: a unified view

3.1 Introduction

It is completely natural in the lifecycle of every method or approach to have a phase of rapid and intensive development of foundations, algorithms and applications (often in the opposite order!) and a phase for consolidation, unification and development of taxonomies. Unification will be iteratively followed by repeating the wave of development, which then either fits the existing model with or without modification of the acclaimed taxonomy or a separate branch or school is established if a considerable amount of scientists feel that the wings of the proliferous development would be clipped otherwise.

Having reached a mature development level of a specific problem, it is advantageous to benefit from the domino effect triggered by a breakthrough from another field. Moreover, not working toward finding or at least seeking for the unification of a problem is detrimental in the long run for all the scientists working on it along with all potential application areas.

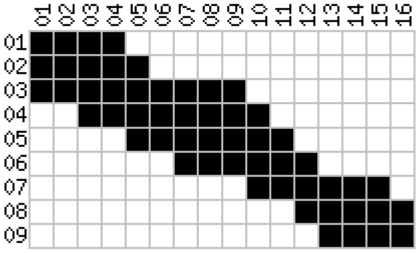
In addition to the wide range of different names representing variations of the basic underlying idea of seriation, the goal of such a process is mostly defined vaguely and ambiguously using natural language. Goals are often described with such rhetoric as *augmenting and maximizing the human visual perception of patterns and the overall trend* or *opening the inner structure of the system*, which is perfectly fine from the philosophical point of view, but impossible to benchmark systematically and contradicts with the best practice in science – repeatability.

The goal of this chapter is to a) enhance repeatability, scrutiny and rigorous benchmarking ability for seriation, and b) link all the subtasks, which should not be the core competence of researchers working with the problem, with the state of the art in other disciplines.

We will review related work on unification and evaluation measures and propose our own formulation and an objective function for seriation, based on Kolmogorov complexity and data compression. A proposed measure is demonstrated on examples and evaluated with empirical experiments, using 35 problems from literature to investigate the agreement with recognized measures from operations research. Empirical evaluations are followed by a discussion of the results and interesting observations.

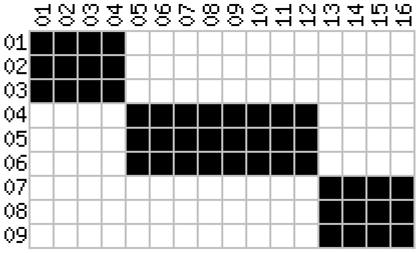
3.2 Related work on unification and evaluation measures

There are several commonly emerging and reported structural patterns from the seriation of matrices: unidimensional seriation (Fig. 3.1), block diagonal seriation (Fig. 3.2), block checkboard seriation (Fig. 3.3) and Pareto seriation (Fig. 3.4). Most preferably, algorithms to perform seriation should not be searching for any specific pattern, but for any regularity and inherent structure, which is amenable to the dataset. Moreover, different types of seriation results should not be restricted or limited to those presented here. Ideally, structures should, one day, have something similar to Sloan's (2003) encyclopedia of integer sequences, which is available online[6] for ad hoc querying. Research on blockmodeling (White *et al.*, 1976) is a potential key towards attaining such an objective with structures.



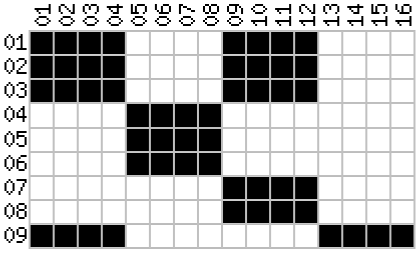
	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
01	■	■	■	■												
02	■	■	■	■	■											
03	■	■	■	■	■	■	■	■	■							
04			■	■	■	■	■	■	■	■						
05					■	■	■	■	■	■	■					
06							■	■	■	■	■	■				
07										■	■	■	■	■	■	
08												■	■	■	■	■
09													■	■	■	■

Figure 3.1 Type I: unidimensional seriation



	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
01	■	■	■	■												
02	■	■	■	■												
03	■	■	■	■												
04					■	■	■	■	■	■	■	■				
05					■	■	■	■	■	■	■	■				
06					■	■	■	■	■	■	■	■				
07													■	■	■	■
08													■	■	■	■
09													■	■	■	■

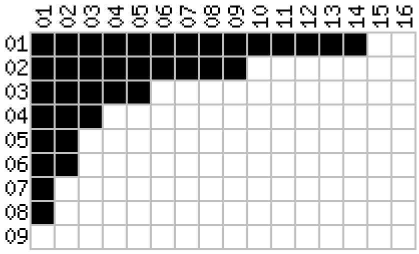
Figure 3.2 Type II: block diagonal seriation



	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
01	■	■	■	■					■	■	■	■				
02	■	■	■	■					■	■	■	■				
03	■	■	■	■					■	■	■	■				
04					■	■	■	■								
05					■	■	■	■								
06					■	■	■	■								
07									■	■	■	■				
08									■	■	■	■				
09	■	■	■	■									■	■	■	■

Figure 3.3 Type III: block checkboard seriation

[6] An On-Line Version of the Encyclopedia of Integer Sequences - http://www.research.att.com/~njas/sequences/



	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
01	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
02	■	■	■	■	■	■	■	■	■							
03	■	■	■	■	■											
04	■	■	■													
05	■	■														
06	■	■														
07	■															
08	■															
09																

Figure 3.4 Type IV: Pareto seriation

The former three types were considered already by McCormick *et al.* (1969), but we have not found any reports of the seriation structure presented in figure 3.4. We called it "Pareto seriation" (Liiv, 2007b) due to reflection of the 80/20 rule, which, in this context, seems to hold to the underlying hidden natural order as well – the majority of the relations and the structure in the dataset is concentrated to a relatively small percentage of the objects.

Although any formal investigation of the mathematical properties of seriation methods, process and structural forms can be considered a contribution towards generalization and unification, we will limit this discussion to only a selection of those authors who explicitly contributed towards a rigorous benchmarking of two-mode matrix permutations and are not restricted to any specific structural form. However, the reader is referred directly to a thorough evaluation of seriation algorithms and "matrix norms" in archaeology by Hole and Shaw (1967) and several publications considering mathematical properties and structural patterns best suited for one-mode matrices, using the terms like (*pre-)Petrie matrix*, *(pre-)Robinson matrix*, *anti-Robinson*, *anti-Robinsonian* (e.g. Kendall, 1969a,1969b and recently discussed: Hubert *et al.*, 2001, p.47; Brusco & Stahl, 2005, p.91), obviously, in honour of the seminal contributions of Petrie (1899) and Robinson (1951).

Marcotorchino (1987) published a paper called "Block seriation problems: a unified approach", which, as the title suggests, was proposing a unified objective function for block seriation. Structural types of unidimensional seriation (Fig 3.1) and block seriation (Fig 3.2) were mentioned, however, the paper and contributions focused exclusively on block seriation[7]. Block seriation, especially with some exceptional and void elements, is less sensitive to the inherent structure in the data than unidimensional seriation (independent blocks can appear in different order if it otherwise satisfies equally the block diagonalization goals) and could be, therefore, considered quite universal in principle. Main blocks (clusters) would be positioned near the diagonal, and if the dataset would otherwise (preferably) be amenable to the block checkboard

[7] Marcotorcino (1991) published a subsequent paper on seriation problems with a wider span over different disciplines, however, did not lose the restriction of exclusive discussion of block seriation only.

form (Fig 3.3), the blocks would just be larger with more void elements. However, the new objective function proposed by Marcotorchino (1987), still required the *k* number of clusters (blocks) to be given as an external input, which makes it unpreferable from our perspective of parameter-free seriation measurement. For similar reasons we are not considering popular cellular manufacturing objective functions like grouping *efficiency* (Chandrasekharan & Rajagopalan, 1989) and grouping *efficacy* (Kumar & Chandrasekharan, 1990) due to their requirement to explicitly identify the cluster boundaries for exceptional element enumerations and, therefore, being very sensitive to structural form (e.g. a horizontal or a vertical flipping of a matrix in a relatively good block seriation form can have a significant negative influence).

There are, however, two robust approaches that are parameter-free and compatible with our restrictions – maximizing the similarity of adjacent objects and maximizing the similarity and "clumpness" of adjacent elements in the matrix. Those objectives remain to provide plenty of ambiguity in order to choose the best similarity coefficient or to define the meaning of neighbourhood and adjacent elements.

McCormick *et al.* (1969, 1972) suggested the use of four neighbouring elements[8] (also known as the von Neumann neighbourhood) and this approach has been revisited several times (Arabie & Hubert, 1990, Arabie *et al.*, 1990, Chu & Tsai, 1990) and proven to be very effective. Arabie and Hubert (1990) have also presented a unified approach to seriation problems, covering the three first types of seriation presented at the beginning of this section.

For measuring the similarity of adjacent objects, we have chosen to use the popular Hamming distance, which was presented by Verin and Grishin (1986) in the context of measuring the "quality of image smoothness" in a matrix.

Measure of effectiveness (McCormick *et al.*, 1972):

$$\operatorname{argmax}_{\Pi,\Phi} \sum_{i=1}^{i=M} \sum_{j=1}^{j=N} a_{\pi(i),\phi(j)} \left[a_{\pi(i),\phi(j+1)} + a_{\pi(i),\phi(j-1)} + a_{\pi(i+1),\phi(j)} + a_{\pi(i-1),\phi(j)} \right] \tag{3.1}$$

Verin and Grishin (1986) proposed a quality estimate of image smoothness:

$$\operatorname{argmin}_{\Pi,\Phi} \left[\left[\sum_{\mathrm{j}=1}^{\mathrm{L}-1} d(x_{\phi(j)}, x_{\phi(j+1)}) \right]^{-1} \times \left[\sum_{\mathrm{i}=1}^{\mathrm{N}-1} d(x_{\pi(i)}, x_{\pi(i+1)}) \right]^{-1} \right], \tag{3.2}$$

[8] Niermann (2005) has recently investigated the use of eight neighbouring elements (Moore neighbourhood).

which we have modified for convenience and better cross-dataset comparability as follows:

$$\mathrm{argmax}_{\Pi,\Phi}\left[1-\frac{\left[\sum_{\mathrm{j}=1}^{\mathrm{L}-1} d_H(x_{\phi(j)},x_{\phi(j+1)})\right]\times\left[\sum_{\mathrm{i}=1}^{\mathrm{N}-1} d_H(x_{\pi(i)},x_{\pi(i+1)})\right]}{[N\times(M-1)\times M\times(N-1)]}\right] \quad (3.3)$$

and refer to as the summarized (cumulative) Hamming distance of a matrix.

We argue that regardless of our definition of the similarity of adjacent objects or elements, it still comes down to detecting the general patternness and regularity in the matrix. Regularity, on the other hand, is something that is a central issue in a data compression community and, intuitively, it would be unreasonable to compete with the state-of-the-art regularity detection achieved in the field of data compression and, therefore, prudent to delegate that responsibility.

In the next section, we will propose a new measure for parameter-free seriation evaluation using data compression.

3.3 Seriation evaluation using data compression

Kolmogorov complexity is the length of the shortest effective description of an object. We suggest to look at the seriation evaluation, using data compression as a special case of Kolmogorov complexity of a string where it is allowed to "cheat" under specific restrictions to make regularity in the string more apparent and therefore more compressible. The most efficient "cheating procedure" is then what we are looking for. The use of Kolmogorov complexity and the minimum description length principle (Rissanen, 1978; Grünwald, 2007) is gaining acceptance and popularity in the data mining community (Keogh *et al.,* 2004; Dhillon *et al.*, 2003; Faloutsos & Megalooikonomou, 2007), but has not, as far as the author is aware, been used to measure the quality of seriation and matrix reordering methods.

Let us start with some preliminary definitions. Algorithmic "Kolmogorov" complexity definition by Hutter (2007) suits us the best. We say that program *p* is a description of a string *x* if *p* run on the universal Turing machine *U* output *x*, and write *U(p) = x* . The length *l* of the shortest description is denoted by

$$K(x) \coloneqq \min_{\mathrm{p}}\{l(p): U(p) = x\} \quad (3.4)$$

The previous definition explicitly states that *x* is a string, however, there can also be a broader interpretation of strings. Li and Vitanyi (1997, p.396) state that "the interpretation of strings as more complex combinatorial objects leads to a completely new set of properties and problems that have no direct counterpart in the *flatter* string world" using some predefined coding:

Definition. Each labeled graph *G = (V,E)* on *n* nodes *V* = {1,2,...,n} can be coded (up to automorphism) by a binary string *E(G)* of length *n(n - 1)/2*.

Hutter (2007) suggests clarifying intent of such interpretation by specifying some default coding $\langle\cdot\rangle$ (we will denote as $E(\bullet)$ for consistency reasons with the previous definition) for non-string objects and to define

$$K(object) \coloneqq K\big(E(object)\big) \tag{3.5}$$

Both of the evaluation measures introduced in the previous section are compatible with the most general definition of matrix seriation:

Definition. Seriation can be defined as a combinatorial optimization problem for minimizing a loss function $\boldsymbol{L}$ on a matrix **A** using permutation matrices **Π** and **Φ** for reordering the rows and columns in a way that maximizes the local and global patterns:

$$\mathrm{argmin}_{\Pi,\Phi} L(\Pi \mathrm{A} \Phi) \tag{3.6}$$

To set the scene for linking Kolmogorov complexity with seriation, let us also look at the alternative formulation of the minimum description length principle provided by Li and Vitanyi (1997, p.354):

Definition. Given a hypothesis space **H**, we want to select the hypothesis *H* such that the length of the shortest encoding of A together with hypotheses *H* is minimal.

In the context of seriation, we can similarly formulate the definition for a two-way one-mode seriation:

Definition. Given all N! permutations of A**,** we want to select the permutation matrix Π such that the length of the shortest encoding of matrix multiplication Π*A*Π is minimal.

In our specific case, the hypothesis space **H** is limited only to different permutations and, therefore, we are not restricting that the shortest encoding together with the chosen hypothesis has to be minimal. Two-way two-mode seriation can be defined analogously:

Definition. Given all N!•M! permutations of A, we want to select permutation matrices Π and Φ such that the length of the shortest encoding of matrix multiplication ΠAΦ is minimal.

We can look at this definition also as a combinatorial optimization problem of finding permutation matrices Π and Φ to minimize the following:

$$\mathrm{argmin}_{\Pi,\boldsymbol{\Phi}} K(\Pi\mathrm{A}\Phi) \tag{3.7}$$

However, as Kolmogorov complexity is incomputable, we will make an approximation using the length *l* of the result of an arbitrary compression algorithm:

$$\mathrm{argmin}_{\Pi,\boldsymbol{\Phi}} l\big(compress(\Pi\mathrm{A}\Phi)\big). \tag{3.8}$$

We will also choose a specific standard algorithm and a default encoding (see Fig. 3.5 for an illustration) of the data matrix:

$$\mathrm{argmin}_{\Pi,\boldsymbol{\Phi}} l\big(compress_{\mathbf{GZIP}} E(\Pi\mathrm{A}\Phi)\big) \tag{3.9}$$

McCormick *et al.* (1972) and others have demonstrated that regularity detection can be reduced to separate procedures for different modes of the matrix. This observation allows us to remedy the possible weakness with two-directional regularity identification of the standard compression algorithm by compressing the matrix together with its transposed matrix:

$$\mathrm{argmin}_{\Pi,\Phi} \min\Big(l\big(compress_{\mathbf{GZIP}} E(\Pi\mathrm{A}\Phi)\big), l\Big(compress_{\mathbf{GZIP}} E\big((\Pi\mathrm{A}\Phi)^{\mathbf{T}}\big)\Big)\Big). \tag{3.10}$$

This defines our loss function for evaluating the row and column permutations:

$$L(\mathrm{A}) = \min\Big(l\big(compress_{\mathbf{GZIP}} E(\Pi\mathrm{A}\Phi)\big), l\Big(compress_{\mathbf{GZIP}} E\big((\Pi\mathrm{A}\Phi)^{\mathbf{T}}\big)\Big)\Big), \tag{3.11}$$

which can be normalized for convenience, using a common space saving data compression ratio:

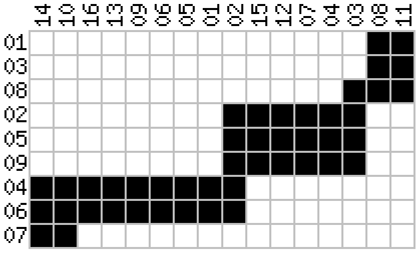
$$GZ(A) = 1 - \frac{\min\left(l\left(compress_{\mathbf{GZIP}}E(\Pi\mathrm{A}\Phi)\right), l\left(compress_{\mathbf{GZIP}}E\left((\Pi\mathrm{A}\Phi)^{\mathbf{T}}\right)\right)\right)}{l\left(E(\Pi\mathrm{A}\Phi)\right)} \tag{3.12}$$

Besides the formal notation, we will make a practical implementation example of the proposed measurement, using shell scripting and Unix piping to secure rapid repeatability and scrutiny for everyone with the access to some Unix-based operating system.

3.4 A practical example

We will make a practical example of implementing the proposed measure with a standard compression tool **`gzip`**, which is included by default under most and available for all Unix-based operating systems[9] (Mac OS X, Linux, FreeBSD, Solaris etc.). **`gzip`** is also freely available[10] for Microsoft Windows platforms and the following examples are easily repeatable there as well with possible minor modifications.

```
Matrix of the Bertin's Townships example:
```



```
Corresponding content of the file bertin_solution:
0000000000000011
0000000000000011
0000000000000111
0000000011111100
0000000011111100
0000000011111100
1111111110000000
1111111110000000
1100000000000000
```

Figure 3.5 Encoding example of the Bertin dataset

[9] We are using FreeBSD 6.2 with default gzip version 1.2.4, compilation options: DIRENT UTIME STDC_HEADERS HAVE_UNISTD_H ASMV

[10] Gzip version 1.2.4, available from http://www.gzip.org

First, we have to encode our example datasets into binary strings and textfiles as shown in Fig. 3.5. We will be measuring seriation quality with two datasets:

- the classical *Townships* example with the solution by Jacques Bertin (Bertin, 1981, p.33), which forms almost a reverse block diagonal form, but similarly real-world datasets, does not provide a completely pure decomposition to partitions;
- an example of unidimensional seriation, also known as "band diagonal, indicative of unidimensionality" (Arabie and Hubert, 1990) .

The easiest way to implement the "$compress_{\mathbf{GZIP}}E((\Pi\mathrm{A}\Phi)^{\mathbf{T}})$" part of the equation, (3.5) is to write a simple shell script in an arbitrary language for matrix transposition. We used PHP scripting language and the **TRANSPOSE_MATRIX** source code is presented in Fig. 3.6.

```
#!/usr/local/bin/php -q
<?PHP // TRANSPOSE_MATRIX
$D=file("/dev/stdin");
for ($i=0;$i<count($D);$i++) {
  $row=trim($D[$i]);
  for ($j=0;$j<strlen($row);$j++) { $A[$i][$j]=$row[$j]; }
}
for ($j=0;$j<count($A[0]);$j++) {
  for ($i=0;$i<count($A);$i++) { printf($A[$i][$j]); }
  echo "\n";
}
?>
```

Figure 3.6 Script for matrix transposition

Knowing that $\mathrm{A} = (\mathrm{A}^{\mathrm{T}})^{\mathrm{T}}$, it is easy to evaluate the correctness of matrix transposition script directly in a shell environment of the operating system[11] using standard **md5** hashing tool:

```
> cat bertin
0000000100100000
0111001000010010
0000000100100000
1100110011001101
0111001000010010
1100110011001101
0000000001000100
0010000100100000
0111001000010010
> cat bertin | md5
b650d20c6c224076c8b6baf69c61fcfc
> cat bertin | ./TRANSPOSE_MATRIX | ./TRANSPOSE_MATRIX | md5
b650d20c6c224076c8b6baf69c61fcfc
```

[11] Symbol '>' denotes the prompt in a shell and should not be used if the example is reconstructed.

Measuring the goodness of Bertin's initial and unordered dataset can be done by printing the content of the file *bertin* to **`gzip`** input using piping[12] and measuring the bytes using **`wc -c`**:

```
> cat bertin | gzip --best | wc -c
      57
> cat bertin | ./TRANSPOSE_MATRIX | gzip --best | wc -c
      62
```

Such output should be interpreted as follows: Matrix stored in *bertin* can be compressed to **57** bytes using **`gzip`** and the transposed matrix to **62** bytes, respectively. The lowest from those two is **57**, making it the goodness measure (equation 3.11) for that permutation of the matrix, which is the initial and unordered version of the matrix in the specific case.

Next, we will measure the compressibility of the matrix permutation solution provided by Bertin (1981, p.33), as a solution:

```
> cat bertin_solution
0000000000000011
0000000000000011
0000000000000111
0000000011111100
0000000011111100
0000000011111100
1111111110000000
1111111110000000
1100000000000000
> cat bertin_solution | gzip --best | wc -c
      49
> cat bertin_solution | ./TRANSPOSE_MATRIX | gzip --best | wc -c
      47
```

The evaluation of compressibility of that permutation gives us **49** bytes for the proposed solution matrix and **47** bytes for the matrix' transposition, resulting evaluation measure of **47** according to equation (3.11).

If we compare this solution with an alternative permutation *bertin.min*, a resulted permutation by "minus" technique using the same initial dataset, we get the following result:

[12] Piping the matrix into **`gzip`** input was chosen to minimize the gzip overhead concerning storing the file's name in headers, making the result therefore comparable regardless of the file name of input matrix.

```
> cat bertin.min
0000000011111100
0000000011111100
0000000011111100
0000000000000111
0000000000000011
0000000000000011
0000001100000000
1111111110000000
1111111110000000
> cat bertin.min | gzip --best | wc -c
      48
> cat bertin.min | ./TRANSPOSE_MATRIX | gzip --best | wc -c
      48
```

The evaluation measure for this permutation is **48** for the matrix and its transposition. By visual inspection, we can see that the alternative solution is also able to decompose the system into groups, however, not providing as seamless transformation as Bertin's manual solution, which also concords with the slightly better result attained with data compression.

Let us look at an example with a different structure. We have the unidimensional seriation example inherently similar to the first structural pattern presented in the beginning of Section 3.2, which is already subjectively in a perfect order and an alternative permutation provided by "minus" technique:

```
> cat uni_dim
1111000000000000
1111100000000000
1111111110000000
0011111111000000
0000111111100000
0000001111110000
0000000001111110
0000000000011111
0000000000001111
> cat uni_dim | gzip --best | wc -c
      63
> cat uni_dim | ./TRANSPOSE_MATRIX | gzip --best | wc -c
      59

> cat uni_dim.min
0000011110000000
0000111110000000
1111111110000000
1111111001000000
1111100001100000
1110000001110000
0000000001110111
0000000000001111
0000000000011111
```

```
> cat uni_dim.min | gzip --best | wc -c
     65
> cat uni_dim.min | ./TRANSPOSE_MATRIX | gzip --best | wc -c
     60
```

Again, we are able to observe that the data compression properties of subjective and manual ordering slightly outperform the automatic alternative.

3.5 Agreement with other evaluation measures

In this section, we perform experiments to measure the agreement of the proposed measure (GZ, equation 3.12) with the measure of effectiveness (ME, equation 3.1) and summarized (cumulative) hamming distance of the matrix (SH; equation 3.3). We are using 35 problems (see Table 3.1 for references) from the literature for the evaluation, which were recently chosen by Goncalves and Resende (2004) to benchmark their evolutionary algorithm for manufacturing cell formation. The complete evaluation of 35*N!*M! permutations would be infeasible, therefore, we are using the alternative permutations of eight seriation algorithms and the initial data matrix:

- ART – an algorithm based on the Carpenter-Grossberg network (Kaparthi and Suresh, 1992);
- BEA – the bond energy algorithm (McCormick *et al.*, 1969, 1972);
- CONF – conformity analysis (Vyhandu, 1981, 1989);
- MIN – conformity analysis with "minus" technique (Vyhandu, 1989; Mullat, 1976a);
- ROC2 – an enhanced rank order clustering (King & Nakornchai, 1982);
- MODROC – an algorithm by Chandrasekharan and Rajagopalan (1986b);
- PLUS – conformity analysis with "plus" technique (Vyhandu, 1989; Mullat, 1976a);
- ZODIAC – an algorithm by Chandrasekharan and Rajagopalan (1987);
- + MATRIX – the initial (unordered) matrix.

The evaluation results of the experiments (35 problems x 9 permutations x 3 measures) are presented in Table 3.2. We have calculated the agreement (%) between different evaluation measures (Table 3.3), using the following scheme:

a) If the best performing permutation of MEASURE1 is also the best performing[13] according to MEASURE2, then the agreement is 100%.

b) Otherwise, find the best permutation according to MEASURE1, its value according to MEASURE2 and divide it by the value of the best performing permutation according to MEASURE2.

Instead of drawing 35 tables of agreement coefficients, all results are combined in Table 3.3, along columns, in the form MEASURE1 vs MEASURE2.

[13] If several permutations have equal values, the one with the highest agreement is chosen.

Table 3.1 Benchmarking problems from literature

Problem	References	Size
1	King and Nakornchai (1982)	5 × 7
2	Waghodekar and Sahu (1984)	5 × 7
3	Seifoddini (1989)	5 × 18
4	Kusiak and Cho (1992)	6 × 8
5	Kusiak and Chow (1987)	7 × 11
6	Boctor (1991)	7 × 11
7	Seifoddini and Wolfe (1986)	8 × 12
8	Chanrasekharan and Rajagopalan (1986a,b)	8 × 20
9	Chanrasekharan and Rajagopalan (1986a,b)	8 × 20
10	Mosier and Taube (1985a)	10 × 10
11	Chan and Milner (1982)	10 × 15
12	Askin and Subramanian (1987)	14 × 24
13	Stanfel (1985)	14 × 24
14	McCormick *et al.* (1972)	16 × 24
15	Srinivasan *et al.* (1990)	16 × 30
16	King (1980)	16 × 43
17	Carrie (1973)	18 × 24
18	Mosier and Taube (1985b)	20 × 20
19	Kumar *et al.* (1986)	20 × 23
20	Carrie (1973)	20 × 35
21	Boe and Cheng (1991)	20 × 35
22	Chanrasekharan and Rajagopalan (1989)	24 × 40
23	Chanrasekharan and Rajagopalan (1989)	24 × 40
24	Chanrasekharan and Rajagopalan (1989)	24 × 40
25	Chanrasekharan and Rajagopalan (1989)	24 × 40
26	Chanrasekharan and Rajagopalan (1989)	24 × 40
27	Chanrasekharan and Rajagopalan (1989)	24 × 40
28	McCormick *et al.* (1972)	27 × 27
29	Carrie (1973)	28 × 46
30	Kumar and Vannelli (1987)	30 × 41
31	Stanfel (1985)	30 × 50
32	Stanfel (1985)	30 × 50
33	King and Nakornchai (1982)	36 × 90
34	McCormick *et al.* (1972)	37 × 53
35	Chanrasekharan and Rajagopalan (1987)	40 × 100

Table 3.2 Experiments (35 problems x 9 permutations x 3 measures)

	MATRIX			ART			BEA			CONF			MIN			MODROC			PLUS			ROC2			ZODIAC		
	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME	GZ	SH	ME
1	16.67	52.5	6	17.5	80.36	14	22.5	88.21	16	21.43	74.29	11	20	87.14	16	21.43	80.36	13	22.5	88.57	16	20	85.6	15	20	84.29	15
2	21.43	79.05	21	21.43	81.67	22	21.43	84.29	23	16.67	78.33	20	20	83.33	22	19.05	78.33	21	12.5	81.67	21	21.43	75.24	20	15	83.33	22
3	56.84	63.33	33	66.32	93.63	57	73.15	96.9	67	63.16	93.3	55	65.26	96.9	66	65.26	96.9	66	64.21	95.75	59	68.42	96.9	67	66.67	95.03	63
4	33.93	53.39	11	38.89	89.29	24	38.89	93.75	28	42.59	95.42	28	44.44	95.36	28	46.43	93.45	26	44.44	94.58	28	46.43	93.45	26	33.93	53.39	11
5	46.59	81.17	10	48.86	80.71	11	47.73	89.72	19	47.73	81.99	11	48.86	87.01	15	48.86	86.82	16	47.73	85.71	14	50	84.29	14	51.14	88.64	19
6	45.45	86.15	11	46.43	89.09	14	51.19	93.51	20	47.73	75.84	4	48.81	92.27	17	47.62	87.88	13	48.86	89.09	15	50	88.48	14	52.27	92.21	18
7	55.77	95.13	46	55.77	95.13	46	57.41	95.13	46	55.56	89.52	34	55.56	93.15	40	56.48	90.53	36	55.56	92.55	38	57.41	93.78	43	55.77	95.13	46
8	64.44	74.01	36	64.88	91.52	71	67.22	95.83	81	71.11	95.65	80	71.11	96.11	81	70	94.41	77	71.11	96	81	68.89	92.73	72	64.29	95.18	79
9	61.67	74.06	92	61.11	86.15	112	60.56	90.02	121	61.11	77.02	97	63.33	88.72	118	63.33	85.49	110	63.33	82.23	106	63.33	84.23	109	64.44	82.81	106
10	52.73	87.78	11	58.18	94.81	22	58.18	95.33	25	56.36	82.79	6	59.09	96.22	25	56.36	93.48	21	56.36	88.14	12	59.09	94.3	22	56.36	91.48	17
11	69.7	69.85	9	75.76	97.57	63	73.94	97.97	65	73.75	96.16	57	74.55	97.93	64	76.97	97.62	63	74.38	97.62	63	76.97	97.84	63	76.25	97.81	64
12	75.65	93.32	33	78.26	97.06	58	77.97	98.23	71	77.97	94.54	38	78.26	97.56	60	80	97.95	65	77.1	94.71	39	79.76	97.9	65	78.55	97.29	59
13	77.22	93.41	35	78.61	97.04	60	80	98.44	77	78	94.23	38	80.29	97.8	67	79.43	97.75	67	77.5	94.95	43	80.29	97.65	66	78.06	96.05	53
14	74.68	89.43	41	75.78	92.53	68	75.96	96.12	90	75.52	91.16	54	76.47	93.88	68	76.47	93.66	67	76.47	91.91	59	76.47	92.89	66	75.19	93.74	69
15	76.81	84.33	39	77.65	94.1	111	78.24	96.94	142	78.02	92.15	93	79.44	96.16	128	79.41	94.66	117	78.63	94.15	110	79.02	94.75	118	79.02	96.1	130
16	79.34	91.76	54	80.98	95.48	97	82.63	97.87	146	82.22	91.63	47	82.63	96.13	101	82.35	96.54	110	82.76	94.63	84	83.31	96.7	116	81.53	95.29	98
17	77.11	92.71	59	78.44	95.28	79	79.39	96.92	97	78	92.63	54	79.82	95.89	74	78	95.25	78	78.67	94.41	65	79.78	95.25	77	78.51	95.52	80
18	72.38	84.86	65	74.52	87.2	77	74.29	91.55	105	74.05	85.24	63	73.81	89.55	86	73.57	88.45	83	73.81	87.49	75	73.57	87.05	79	73.33	88.21	80
19	74.53	88.06	60	76.25	91.33	80	77.08	95.46	120	77.02	89.66	72	76.6	92.48	91	76.25	91.55	86	76.81	89.52	71	77.43	91.51	86	76.67	92.59	95
20	83.4	89.1	38	87.21	98.58	181	88.44	99.24	201	84.49	92.2	68	86.39	98.77	182	87.89	98.83	185	85.03	96.73	137	87.76	98.98	190	86.39	98.86	186
21	79.46	87.89	49	81.63	95.28	137	81.9	97.59	180	80.54	90.31	74	81.77	95.58	138	81.36	95.98	149	80.95	93.08	106	81.36	94.13	120	80.95	96.41	160
22	89.4	95	46	93.5	99.24	171	93.9	99.67	198	94.11	99.67	198	94.31	99.69	198	94.82	99.68	198	94	99.67	198	94.82	99.68	198	90.6	95.98	65
23	85.6	94.89	42	88.6	98.83	151	89.53	99.39	179	88.6	98.04	121	88.82	98.68	143	89.2	98.94	156	88.6	98.28	129	89.53	98.88	153	89	98.9	154
24	83.7	94.61	38	85.7	98.01	123	86	98.78	152	85.4	96.18	71	86.3	97.92	117	86.1	98.02	121	84.96	97.09	94	86.6	97.65	111	85.6	97.97	119
25	81.9	94.41	35	83	95.9	64	83.1	97.72	114	83.4	95.29	51	84.3	96.45	77	83.3	96.64	82	83.4	95.54	56	83.3	96.34	76	83.5	96.37	76
26	82	94.12	29	83	95.33	53	83.1	97.57	108	82.6	94.93	43	83.2	95.63	57	83.2	96.14	69	83.3	95.26	49	83	95.74	62	83.2	95.47	55
27	82.6	94.04	28	82.4	95.24	52	83	97.28	102	82.6	94.25	30	83.1	95.77	60	82.7	95.69	60	83	95.22	49	83	95.89	66	82.9	95.8	63
28	77.91	87.87	177	78.97	91.15	213	79.37	94.58	262	77.12	83.06	129	78.57	93.5	237	78.44	91.53	220	76.98	89.46	190	79.89	91.31	218	78.7	92.17	229
29	83.81	96.51	178	83.73	96.66	182	84.93	97.43	209	83.88	93.68	91	84.18	95.52	137	83.73	95.97	152	84.03	94.53	112	84.56	95.49	137	84.11	96.76	185
30	86.47	97.32	51	87.1	98.22	89	88.51	99.13	137	87.18	97.5	57	87.33	98.41	94	87.41	98.71	110	87.02	97.94	75	88.04	98.58	107	86.7	97.97	78
31	88.45	97.81	82	87.87	98.35	111	89.68	99.08	158	87.71	97.1	50	88.52	98.41	113	88.39	98.59	122	88.39	98.18	100	89.35	98.58	124	88.13	98.31	108
32	85.61	96.29	45	85.68	97.39	87	87.42	98.45	144	86.26	96.28	43	86.52	97.08	74	86.58	97.68	102	86.45	96.92	68	86.52	97.12	78	86.19	97.55	97
33	88.32	96.64	106	89.07	97.93	198	90.04	98.97	316	89.25	97.51	165	89.28	97.9	195	89.57	98.31	235	89.39	97.84	192	89.25	98.17	220	89.1	98.1	212
34	86.59	90.86	1269	87.59	96.03	1503	90.34	98.99	1726	87.54	93.29	1390	89.82	98.41	1658	88.83	97.21	1551	88.84	97.29	1586	90.27	97.76	1600	88.89	97.62	1617
35	91.44	97.2	159	92.65	99.24	483	93.83	99.74	619	93.12	98.55	341	93.44	99.41	521	93.98	99.61	575	93.17	99.07	437	94	99.21	468	93.93	99.62	578

Table 3.3 Agreement (%) between different evaluation measures

Prob.	GZ vs SH	SH vs GZ	GZ vs ME	ME vs GZ	SH vs ME	ME vs SH
1	100	100	100	100	100	100
2	100	100	100	100	100	100
3	100	100	100	100	100	100
4	97.94	91.73	92.86	95.71	100	100
5	98.8	93.33	100	100	100	100
6	98.61	97.93	90	97.93	100	100
7	100	100	100	100	100	100
8	100	100	100	100	100	100
9	91.99	93.98	87.6	93.98	100	100
10	100	100	100	100	100	100
11	99.87	96.06	96.92	96.06	100	100
12	99.71	97.46	91.55	97.46	100	100
13	99.35	99.64	87.01	99.64	100	100
14	97.67	99.33	75.56	99.33	100	100
15	99.2	98.49	90.14	98.49	100	100
16	98.8	99.18	79.45	99.18	100	100
17	98.94	99.46	76.29	99.46	100	100
18	95.25	99.69	73.33	99.69	100	100
19	95.86	99.55	71.67	99.55	100	100
20	100	100	100	100	100	100
21	100	100	100	100	100	100
22	99.99	99.46	100	100	100	100
23	100	100	100	100	100	100
24	98.86	99.31	73.03	99.31	100	100
25	98.7	98.58	67.54	98.58	100	100
26	97.63	99.76	45.37	99.76	100	100
27	98.45	99.88	58.82	99.88	100	100
28	96.54	99.35	83.21	99.35	100	100
29	100	100	100	100	100	100
30	100	100	100	100	100	100
31	100	100	100	100	100	100
32	100	100	100	100	100	100
33	100	100	100	100	100	100
34	100	100	100	100	100	100
35	99.47	99.82	75.61	99.82	100	100

3.6 Discussion

McCormick *et al.* (1969, 1972) focused on the maximization of "clumpness" of similar elements using matrix reordering. Bertin (1967, 1981), on the other hand, claimed the ultimate goal to be any structural pattern enabling the best (visual) perception of relationships, patterns and the overall structural trend. We interpret "clumpness", patterns and the overall structural trend as regularity and compression as the universal tool for identifying any structural pattern amenable to data by measuring the regularity of that specific configuration of the matrix. In other words, the less bytes it takes to represent the same information in the matrix, the better the permutation (configuration). There might be several underlying hidden variables or continuums to choose from (e.g. time, typicality). On such an occasion one should prefer the one which brings forth the highest regularity between objects and within all relationships and attributes, making it possible to achieve the best compression of the data. Such an approach is a preferable and a standardized way for exploiting regularity and, for example, insensitive to specific codings (e.g. inversion of the binary values is equally compressible).

In order to investigate the agreement between different measures, nine different permutations (initial matrix and seriation results of eight algorithms) were measured with three evaluation measures, introduced in previous sections (ME, SH, GZ), and the agreement coefficient was calculated according to the scheme presented in the previous section. Before investigating the suitability of the data compression measure (GZ), there are several noteworthy observations that can be summarized from the intermediate results (Table 3.2) of the experiments measuring nine permutations of the well-known 35 datasets:

- Out of two heuristic strategies presented by Mullat (1976a), the "minus" technique (MIN), which we chose as a base algorithm with the conformity entity-to-set function (Vyhandu, 1981) for the advancements presented in next chapters, outperformed the "plus" technique (PLUS) on the average with all three measures.
- The greedy approach of the "minus" technique comes with the cost of performing worse on data compression (GZ) than BEA and ROC2, but often manages to result equally good solutions and, if that is not the case, produces acceptable approximations.
- BEA performed better than other algorithms according to ME (closest algorithms to come were MODROC, MIN and ROC2) and SH (such a result concurs with several other evaluations and discussions, e.g. Chu & Tsai, 1990; Arabie & Hubert, 1990) but was slightly worse than ROC2, according to data compression (GZ).

We could see from the experiments that algorithms with different heuristics and complexities can perform sometimes equally and, at the same time, no approach or algorithm from the list can be regarded as a "panacea". Even if some algorithm would have performed best in every case from those limited choices, it would not provide adequate grounds to prove that no better permutation exists if the algorithm's heuristics are custom-tailored to shortcut the exhaustive search of *n!m!* permutations. Therefore, if the computational cost is not critical, the best strategy to maximize regularity in the matrix is by using ensemble approaches to apply a variety of algorithms and choose the one providing the best compression.

The use of Kolmogorov complexity, the minimum description length principle (Rissanen, 1978; Grünwald, 2007) and a general information-theoretic approach has been reported and discussed to have a useful and practical application towards parameter-free data mining (Keogh *et al.*, 2004; Dhillon *et al.*, 2003; Faloutsos & Megalooikonomou, 2007). The prevailing agreement between different evaluation measures in Table 3.3 shows that data compression is suitable for seriation problems as well, enabling and facilitating the objective and rigorous measurement of different permutations of the same data matrix.

There is a perfect agreement between measures ME and SH, which should not be interpreted as a one-to-one relationship between the values, but as a complete agreement in choosing the "best" permutation. Instances of asymmetric low agreement between measures GZ and ME mean that the best permutation according to ME was amenable to a very good compression, but the permutation, which yielded the highest compression ratio did not perform very well according to ME. The cases of disagreement were manually re-evaluated and illustrate possible shortcomings of ME, as the best matrix permutation suggested by the objective GZ-measure can be subjectively considered an alternatively good solution. To convince the reader, we present an illustrative example from the experiments (see Fig. 3.7-3.15), where two permutations out of nine yielded good compression ratios, but were not so successful according to ME.

An important remark should be made towards the choice of a specific compression algorithm. For repeatability, scrutiny and rigorous benchmarking ability, it is essential to use a standard and popular algorithm, which would preferably be well-distributed and a default component in a wide range of operation systems. This is the reason for currently choosing `gzip`. Such a choice should be reconsidered if some significantly better-performing algorithm becomes as ubiquitous as `gzip` and, besides better compression of the same matrix, alternative permutations are recommended to be more compressible.

Finally, a clear remark about the scope and limitations of the presented approach is called for. The proposed practical implementation of the evaluation measure considers explicitly only two-way one-/two-mode binary matrices. The main reason for that is the observation that binary encoding is the most fundamental representation that intersects all the previous seriation research, enabling direct cross-discipline experiments. Categorical values can be represented using combinations of binary values or small alphabets to preserve the compatibility with the data compression approach, however, things get much more complicated with continuous data. In principle, the abstraction level of the equation 3.7 is high enough to consider any compressible regularity in the data. In practice, it would, however, require an additional discretization layer upon the encoding, which makes the final measurement overly sensitive to the specific discretization strategy and distribution.

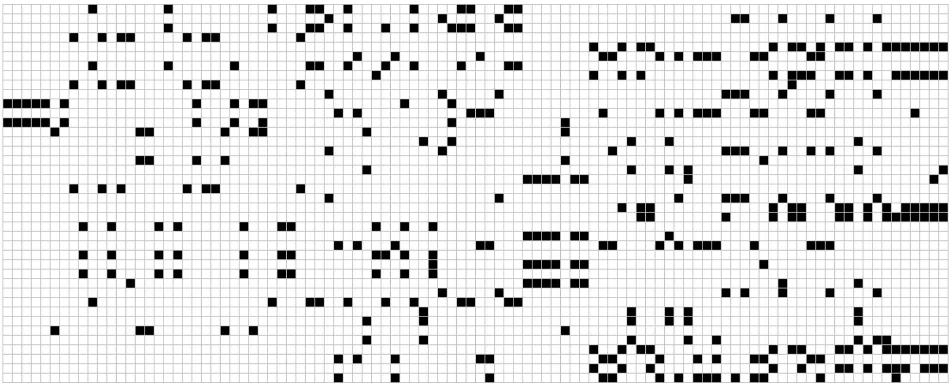


Figure 3.7 The initial matrix (Dataset=35 GZ=91.44 SH=97.2 ME=159)

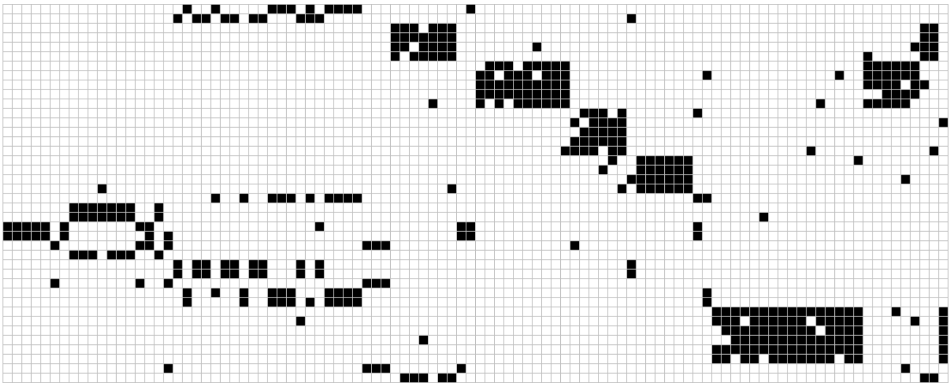


Figure 3.8 ART (Dataset=35 GZ=92.65 SH=99.24 ME=483)

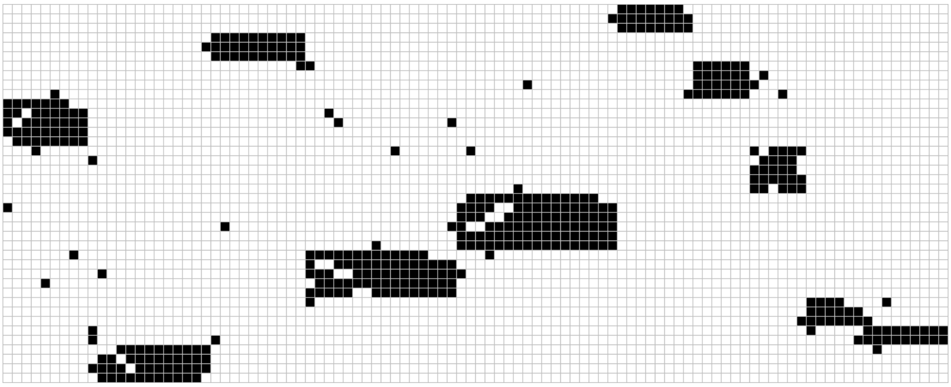


Figure 3.9 BEA (Dataset=35 GZ=93.83 SH=99.74 ME=619)

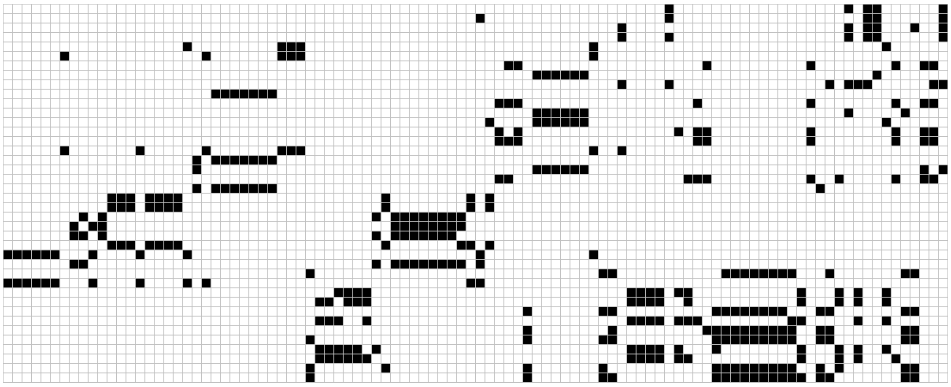


Figure 3.10 CONF (Dataset=35 GZ=93.12 SH=98.55 ME=341)

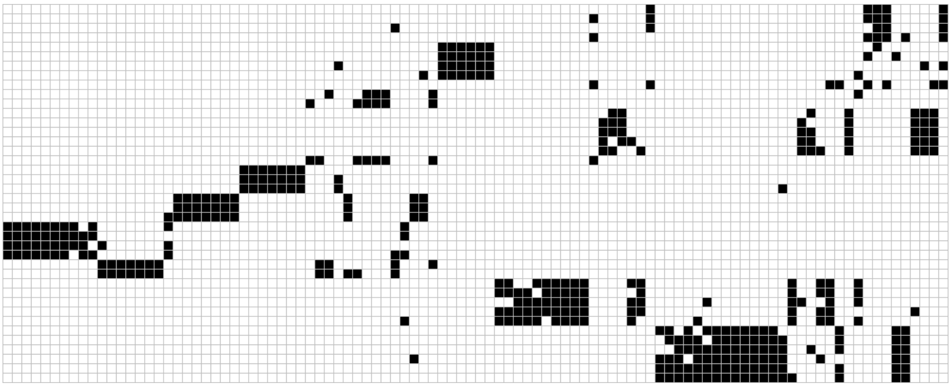


Figure 3.11 MIN (Dataset=35 GZ=93.44 SH=99.41 ME=521)

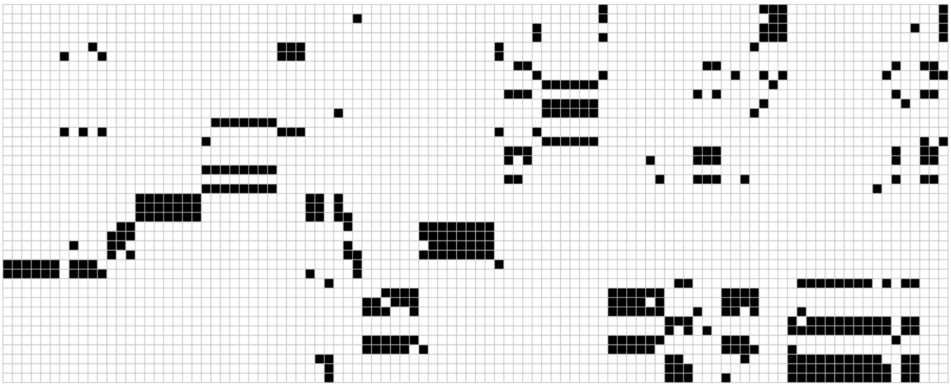


Figure 3.12 PLUS (Dataset=35 GZ=93.17 SH=99.07 ME=437)

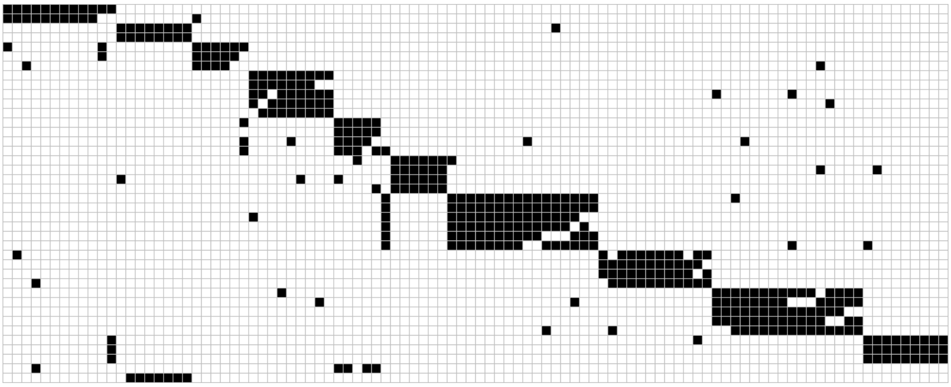


Figure 3.13 MODROC (Dataset=35 GZ=93.98 SH=99.61 ME=575)

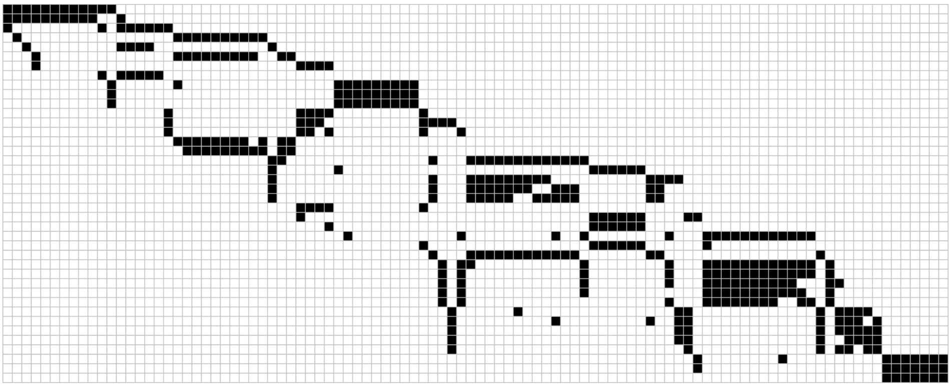


Figure 3.14 ROC2 (Dataset=35 GZ=94 SH=99.21 ME=468)

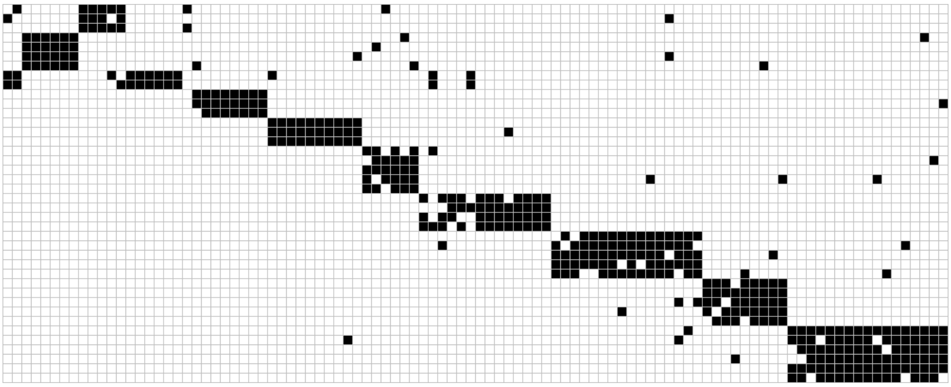


Figure 3.15 ZODIAC (Dataset=35 GZ=93.93 SH=99.62 ME=578)

4 Extensions: new approaches and algorithms

4.1 Introduction

New approaches and extensions presented in this chapter are based on the works of Vyhandu (1979,1980,1981,1989) and Mullat (1976a,1976b,1977). In Section 4.2, we will demonstrate that it is possible to define and implement the conformity analysis algorithm (Fig. 4.1 presents Vyhandu's conformity analysis – a procedure to perform seriation of two-mode matrices according to the entity-to-set weight function) with standard relational algebra and relational calculus in structured query language (SQL) without any use of procedures or functions.

Seriation of objects (rows):
1. Enumeration of the attribute values for each attribute;
2. Replacement of attribute values with the actual frequency of that value within that attribute;
3. Conformity for objects is calculated using the sum of attribute value frequencies;
4. Objects are reordered according to the conformity weights.

Seriation of attributes (columns):
1. Enumeration of the attribute values for each object;
2. Replacement of attribute values with the actual frequency of that value within that object;
3. Conformity for attributes is calculated using the sum of attribute value frequencies.
4. Attributes are reordered according to the conformity weights.

Figure 4.1 Conformity analysis for two-mode matrices

Vyhandu (1989) demonstrated that all conditions of monotone systems (Mullat, 1976a) hold and are satisfied, when the conformity weight function (Vyhandu, 1981) is chosen as the entity-to-set weight function for a monotone system. Using the combination of conformity analysis and the heuristics of monotone system procedures comes at the expense of additional computational complexity, but can significantly refine the results of seriation.

Mullat suggested two complementary strategies (a "plus" technique and a "minus" technique) for starting the search from either the most typical or unusual object. We have applied the "minus" technique (step-wise elimination of the most unusual object according to the entity-to-set weight function), because in sparse datasets, made out of mostly void elements, the most unusual object is actually the most typical object consisting of positive elements. In Section 4.3, a seriation algorithm for sparse binary datasets will be proposed.

4.2 Conformity analysis with structured query language

If we consider the steps in the algorithm introduced in Fig. 4.1, we can identify mostly enumeration, replacements and sorting. The goal of this section is to demonstrate that it is possible to delegate all the calculation steps to the database system and implement the algorithm with standard relational algebra and relational calculus in structured query language (SQL) without any use of external procedures or functions. One could make use of database capabilities, thereby leveraging on more than a decade of effort spent in making these systems robust, portable, scalable and concurrent. Also, it is possible to exploit the underlying SQL parallelization. Novelty from the perspective of database systems is step from sequential ordering towards consensual ordering. Such an approach would enable ordering rows not only sequentially according to the given columns (e.g. `ORDER BY A, B, C`), but simultaneously as an aggregated order with maximal agreement over (e.g. something like `ORDER BY A & B & C`) individual orders resulted by the columns. An example of possible corporate application is customer and market segmentation, which could then be done natively using structured query language. However, within our scope and focus, we concentrate on the "proof-of-concept" of performing seriation using the conformity analysis algorithm directly in database systems without procedures and external functions.

All necessary steps to reproduce the presented approach and implementation are described in detail to assure maximal repeatability and scrutiny. First, we will introduce the required data format and decompose the problem into six distinct saved queries (views), commenting on the results of those queries and correspondence to calculations and steps in the initial algorithm. Subsequently, a single complex query to perform all the subtasks will be presented, together with the comments on experimented platforms and versions of database systems.

The initial step is to transform the dataset used in this example (Table 4.1; *o – objects, a - attributes*) to a transactional format presented in the corresponding Table 4.2 (*o – objects, a – attributes, v – values).*

Table 4.1 Dataset for the SQL example

	a_1	a_2	a_3	a_4	a_5
o_1	1	0	0	0	0
o_2	0	1	0	1	1
o_3	0	1	0	1	1
o_4	1	1	0	1	0
o_5	0	0	1	0	1
o_6	0	1	1	1	1

Table 4.2 Dataset in transactional format

o	*a*	*v*
1	1	1
1	2	0
1	3	0
1	4	0
1	5	0
2	1	0
2	2	1
2	3	0
2	4	1
2	5	1
3	1	0
3	2	1
3	3	0
3	4	1
3	5	1
4	1	1
4	2	1
4	3	0
4	4	1
4	5	0
5	1	0
5	2	0
5	3	1
5	4	0
5	5	1
6	1	0
6	2	1
6	3	1
6	4	1
6	5	1

Binary data values are used for this example, but the approach is applicable also to categorical data values. Let us refer to this table as *DATA_TABLE* in the following SQL queries. The table structure and the initial data in data definition language (DDL), is presented in Fig. 4.2.

```
CREATE TABLE DATA_TABLE ( o int,  a int, v int );

INSERT INTO DATA_TABLE (o,a,v) VALUES ('1', '1', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('1', '2', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('1', '3', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('1', '4', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('1', '5', '0');

INSERT INTO DATA_TABLE (o,a,v) VALUES ('2', '1', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('2', '2', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('2', '3', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('2', '4', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('2', '5', '1');

INSERT INTO DATA_TABLE (o,a,v) VALUES ('3', '1', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('3', '2', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('3', '3', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('3', '4', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('3', '5', '1');

INSERT INTO DATA_TABLE (o,a,v) VALUES ('4', '1', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('4', '2', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('4', '3', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('4', '4', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('4', '5', '0');

INSERT INTO DATA_TABLE (o,a,v) VALUES ('5', '1', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('5', '2', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('5', '3', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('5', '4', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('5', '5', '1');

INSERT INTO DATA_TABLE (o,a,v) VALUES ('6', '1', '0');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('6', '2', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('6', '3', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('6', '4', '1');
INSERT INTO DATA_TABLE (o,a,v) VALUES ('6', '5', '1');
```

Figure 4.2 The table structure and the initial data

We are using Microsoft Access for the stepwise example, therefore the presented queries also perform as views or virtual tables composed of the results of the queries and can be referenced identically with tables.

For calculating the frequencies of the values within rows and columns, queries *frequency_h* and *frequency_v* will be used:

Query "frequency_h":

```
SELECT [o], [v], count(*) AS s FROM DATA_TABLE GROUP BY [o], [v];
```

Query "frequency_v":

```
SELECT [a], [v], count(*) AS s FROM DATA_TABLE GROUP BY [a], [v];
```

With the query *frequencies*, initial values are mapped to the results from the two previous queries, replacing values with their frequencies within the rows and columns.

Query "frequencies":

```
SELECT DATA_TABLE.o, DATA_TABLE.a, frequency_v.s AS vertical,
frequency_h.s AS horizontal
FROM frequency_h INNER JOIN (DATA_TABLE INNER JOIN frequency_v ON
(DATA_TABLE.v = frequency_v.v) AND (DATA_TABLE.a =
frequency_v.a)) ON (frequency_h.v = DATA_TABLE.v) AND
(frequency_h.o = DATA_TABLE.o);
```

Next, we will sum up the replaced values over rows and columns with queries *o_sum* and *a_sum*, respectively:

Query "o_sum":

```
SELECT [frequencies].[o],Sum([frequencies].[vertical]) AS o_sum
FROM frequencies
GROUP BY [frequencies].[o]
ORDER BY Sum([frequencies].[vertical]) DESC;
```

Query "a_sum":

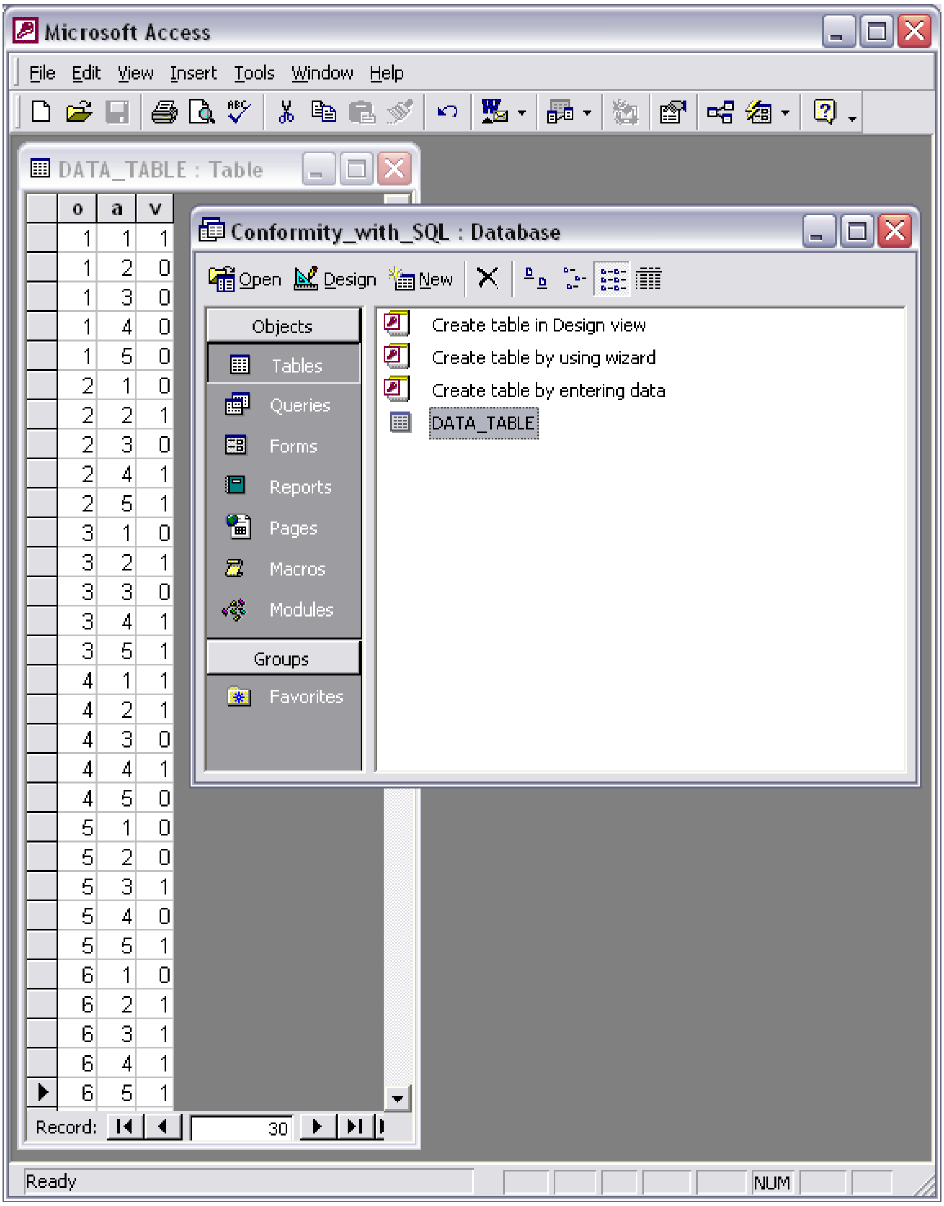
```
SELECT [frequencies].[a],Sum([frequencies].[horizontal]) AS a_sum
FROM frequencies
GROUP BY [frequencies].[a]
ORDER BY Sum([frequencies].[horizontal]) DESC;
```

We will compose a query *conformity_analysis* to combine the results again with the initial dataset and to reorder the results according to the conformity measure, which also performs as a seriation heuristic for the data.

Query "conformity_analysis":

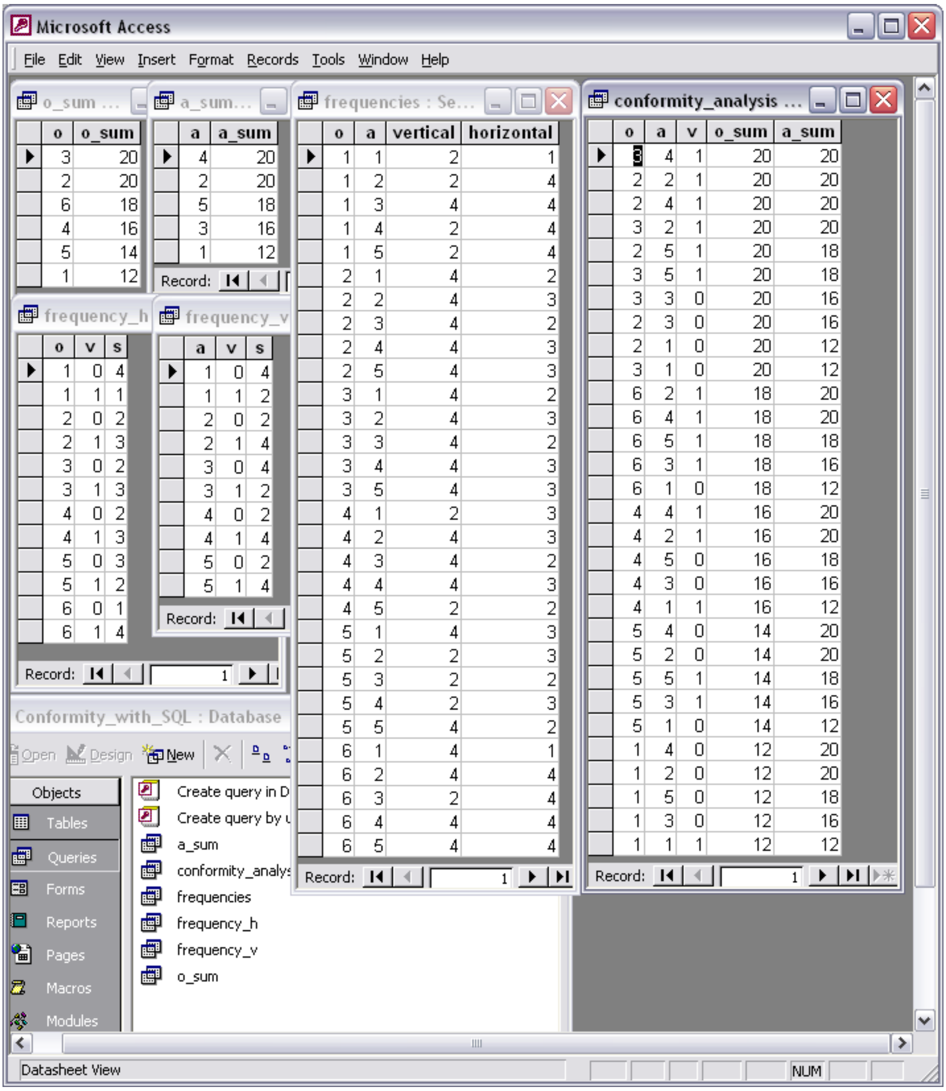
```
SELECT o_sum.o, a_sum.a, DATA_TABLE.v, o_sum.o_sum, a_sum.a_sum
FROM (o_sum INNER JOIN DATA_TABLE ON o_sum.o = DATA_TABLE.o)
INNER JOIN a_sum ON DATA_TABLE.a = a_sum.a
ORDER BY o_sum.o_sum DESC , a_sum.a_sum DESC;
```

Screenshots of the initial data (Table 4.2) will be presented in Fig. 4.3, together with the results of all the above queries in Fig. 4.4.



o	a	v
1	1	1
1	2	0
1	3	0
1	4	0
1	5	0
2	1	0
2	2	1
2	3	0
2	4	1
2	5	1
3	1	0
3	2	1
3	3	0
3	4	1
3	5	1
4	1	1
4	2	1
4	3	0
4	4	1
4	5	0
5	1	0
5	2	0
5	3	1
5	4	0
5	5	1
6	1	0
6	2	1
6	3	1
6	4	1
6	5	1

Figure 4.3 Screenshot of the implementation in Microsoft Access



o_sum ...

o	o_sum
3	20
2	20
6	18
4	16
5	14
1	12

a_sum...

a	a_sum
4	20
2	20
5	18
3	16
1	12

frequency_h

o	v	s
1	0	4
1	1	1
2	0	2
2	1	3
3	0	2
3	1	3
4	0	2
4	1	3
5	0	3
5	1	2
6	0	1
6	1	4

frequency_v

a	v	s
1	0	4
1	1	2
2	0	2
2	1	4
3	0	4
3	1	2
4	0	2
4	1	4
5	0	2
5	1	4

frequencies : Se...

o	a	vertical	horizontal
1	1	2	1
1	2	2	4
1	3	4	4
1	4	2	4
1	5	2	4
2	1	4	2
2	2	4	3
2	3	4	2
2	4	4	3
2	5	4	3
3	1	4	2
3	2	4	3
3	3	4	2
3	4	4	3
3	5	4	3
4	1	2	3
4	2	4	3
4	3	4	2
4	4	4	3
4	5	2	2
5	1	4	3
5	2	2	3
5	3	2	2
5	4	2	3
5	5	4	2
6	1	4	1
6	2	4	4
6	3	2	4
6	4	4	4
6	5	4	4

conformity_analysis ...

o	a	v	o_sum	a_sum
3	4	1	20	20
2	2	1	20	20
2	4	1	20	20
3	2	1	20	20
2	5	1	20	18
3	5	1	20	18
3	3	0	20	16
2	3	0	20	16
2	1	0	20	12
3	1	0	20	12
6	2	1	18	20
6	4	1	18	20
6	5	1	18	18
6	3	1	18	16
6	1	0	18	12
4	4	1	16	20
4	2	1	16	20
4	5	0	16	18
4	3	0	16	16
4	1	1	16	12
5	4	0	14	20
5	2	0	14	20
5	5	1	14	18
5	3	1	14	16
5	1	0	14	12
1	4	0	12	20
1	2	0	12	20
1	5	0	12	18
1	3	0	12	16
1	1	1	12	12

Figure 4.4 Screenshot of the results of all presented queries

Finally, we will combine all presented queries into a single complex SQL query, which is presented in Fig. 4.5. All queries previously performing separate calculation steps of the algorithm are rewritten as nested subqueries with multiple self-joins.

```
SELECT
tmp_o_sum.o, tmp_a_sum.a, DATA_TABLE.v, tmp_o_sum.tmp_o_sum as
o_sum, tmp_a_sum.tmp_a_sum as a_sum
FROM
((SELECT tmp_frequency.o, Sum(tmp_frequency.ver) AS tmp_o_sum
FROM
(SELECT DATA_TABLE.o, DATA_TABLE.a, tmp_freq_v.s AS ver,
tmp_freq_h.s AS hor FROM
(SELECT o, v, count(*) AS s FROM DATA_TABLE GROUP BY o, v)
tmp_freq_h
INNER JOIN
(DATA_TABLE INNER JOIN
(SELECT a, v, count(*) AS s FROM DATA_TABLE
GROUP BY a, v) tmp_freq_v ON (DATA_TABLE.v = tmp_freq_v.v) AND
(DATA_TABLE.a = tmp_freq_v.a)) ON (tmp_freq_h.v = DATA_TABLE.v)
AND (tmp_freq_h.o = DATA_TABLE.o)) tmp_frequency
GROUP BY tmp_frequency.o) tmp_o_sum
INNER JOIN DATA_TABLE ON tmp_o_sum.o = DATA_TABLE.o)
INNER JOIN
(SELECT tmp_frequency.a, Sum(tmp_frequency.hor) AS tmp_a_sum
FROM
(SELECT DATA_TABLE.o, DATA_TABLE.a, tmp_freq_v.s AS ver,
tmp_freq_h.s AS hor
FROM (SELECT o, v, count(*) AS s FROM DATA_TABLE GROUP BY o, v)
tmp_freq_h INNER JOIN (DATA_TABLE INNER JOIN
(SELECT a, v, count(*) AS s FROM DATA_TABLE GROUP BY a, v)
tmp_freq_v ON (DATA_TABLE.v = tmp_freq_v.v) AND (DATA_TABLE.a =
tmp_freq_v.a)) ON (tmp_freq_h.v = DATA_TABLE.v) AND (tmp_freq_h.o
= DATA_TABLE.o)) tmp_frequency
GROUP BY
tmp_frequency.a) tmp_a_sum ON DATA_TABLE.a = tmp_a_sum.a
ORDER BY
tmp_o_sum.tmp_o_sum DESC , tmp_a_sum.tmp_a_sum DESC;
```

Figure 4.5 Conformity analysis with structured query language

The query has been validated to produce correct results with the following database systems and respective versions:

- MySQL 4.1.1-alpha-standard;
- MySQL 5.1.23-rc;
- Microsoft SQL Server 2000;
- Microsoft SQL Server 2005;
- Microsoft Access 2000;
- Microsoft Access 2003;
- Microsoft Access 2007;
- PostgreSQL Database Server 8.1.0;
- Oracle 10g.

A notable effort was needed for making the query compatible with the listed systems, as the development of the nested subquery functionality has been different for each of the systems. Further optimizations and shorter representations are possible within specific database systems.

An example of the result acquired using the single complex query with a different database system is presented in Fig. 4.6. It is possible to identify the correspondence of the elements' order and conformity weights with the standard results (Table 4.3) of conformity analysis.

```
+------+------+------+-------+-------+
| o    | a    | v    | o_sum | a_sum |
+------+------+------+-------+-------+
|    2 |    2 |    1 |    20 |    20 |
|    3 |    2 |    1 |    20 |    20 |
|    2 |    4 |    1 |    20 |    20 |
|    3 |    4 |    1 |    20 |    20 |
|    2 |    5 |    1 |    20 |    18 |
|    3 |    5 |    1 |    20 |    18 |
|    2 |    3 |    0 |    20 |    16 |
|    3 |    3 |    0 |    20 |    16 |
|    2 |    1 |    0 |    20 |    12 |
|    3 |    1 |    0 |    20 |    12 |
|    6 |    2 |    1 |    18 |    20 |
|    6 |    4 |    1 |    18 |    20 |
|    6 |    5 |    1 |    18 |    18 |
|    6 |    3 |    1 |    18 |    16 |
|    6 |    1 |    0 |    18 |    12 |
|    4 |    2 |    1 |    16 |    20 |
|    4 |    4 |    1 |    16 |    20 |
|    4 |    5 |    0 |    16 |    18 |
|    4 |    3 |    0 |    16 |    16 |
|    4 |    1 |    1 |    16 |    12 |
|    5 |    2 |    0 |    14 |    20 |
|    5 |    4 |    0 |    14 |    20 |
|    5 |    5 |    1 |    14 |    18 |
|    5 |    3 |    1 |    14 |    16 |
|    5 |    1 |    0 |    14 |    12 |
|    1 |    2 |    0 |    12 |    20 |
|    1 |    4 |    0 |    12 |    20 |
|    1 |    5 |    0 |    12 |    18 |
|    1 |    3 |    0 |    12 |    16 |
|    1 |    1 |    1 |    12 |    12 |
+------+------+------+-------+-------+
30 rows in set (0.02 sec)
```

Figure 4.6 Result of the presented query on MySQL

Table 4.3 Dataset after conformity analysis

	a_2	a_4	a_5	a_3	a_1	***conformity***
o_2	1	1	1	0	0	**20**
o_3	1	1	1	0	0	**20**
o_6	1	1	1	1	0	**18**
o_4	1	1	0	0	1	**16**
o_5	0	0	1	1	0	**14**
o_1	0	0	0	0	1	**12**
conformity	**20**	**20**	**18**	**16**	**12**	

It is also possible to define the presented approach as a structured query language view, allowing to overlook the general complexity of the query, e.g. to develop a conformity view of each dataset. Several industries need the measurement of usual and unusual behaviour in their application, and such an approach could reduce the time of preprocessing the data and allow concentration only on the problem itself.

4.3 Seriation algorithm for binary sparse datasets

The complexity of conformity weight calculation is *O*(*2n*) – one pass for the enumeration and another pass for frequency transformation and cumulative summarization. When we apply step-wise elimination of the minimal element (called the "minus" technique by Mullat (1976a)) to rank the elements according to the heuristics of a monotone system, the combined complexity is

$$O\left(\sum_{i=0}^{n} 2(n-i)\right) = O\big(n(n+1)\big). \tag{4.1}$$

Our goal is to reach identical results of the "minus" technique algorithm (Mullat 1976a) using the conformity weight (Vyhandu, 1981) in significantly fewer steps, exploiting the sparseness of the data and symmetric property of binary matrices. A lossy approach would be a trivial use of only the information of existing elements in every row, however that would reduce the cognizance of the structure explicitly only to the relations which exists in the dataset. White *et al.* (1976) argue that "the essential phenomenon portrayed in network imagery is the *absence* of connections between named individuals." We agree with that, because considering the non-existent connections allows balancing and normalizing the weights proportionally to the whole structure.

Another important refinement to the implementation is the use of linked lists of positive elements instead of matrices (arrays), which would, also, efficiently support the common data mining source format of transactions and items. However, a simple change of data structures itself would not help much due to our restriction of considering also the absent relations in the dataset. It could help to optimize memory usage, but our goal is to skip measuring the influence of all absent relations and, yet, reach identical results of the conformity weight as if those relations were enumerated and measured.

Theorem. We can calculate the conformity *Conf*(*r*) of a row *r* in a binary matrix *A* using only the positive elements with the following formula:

$$Conf(r) = 2 \cdot Conf_1(r) + ((N \cdot M) - e_1) - (e_r \cdot N) \tag{4.2}$$

Where

N – number of rows in a matrix A
M – number of columns in a matrix A
$Conf_1(r)$ – the conformity of a row r calculated only using the present positive elements
e_1 – number of positive elements (ones) in a binary matrix A
e_r – number of positive elements (ones) in a row r

Proof

Let us define the conformity *Conf*(*r*) of a row *r* using the Kronecker's delta notation:

$$\delta_{i,j} = \begin{cases} 1, if\ i = j \\ 0, if\ i \neq j \end{cases} \tag{4.3}$$

$$Conf(r) = \sum_{j=1}^{M}\sum_{i=1}^{N}\delta_{a_{r,j},a_{i,j}} \tag{4.4}$$

If we are dealing with a binary matrix, i.e. the following condition holds:

$$\sum_{j=1}^{M}\sum_{i=1}^{N}\delta_{0,a_{i,j}} + \sum_{j=1}^{M}\sum_{i=1}^{N}\delta_{1,a_{i,j}} = N \cdot M \tag{4.5}$$

and we can say that

$$Conf(r) = \sum_{j=1}^{M}\sum_{i=1}^{N}\delta_{a_{r,j},a_{i,j}} = \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},0}\right) + \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},1}\right). \tag{4.6}$$

And, therefore, we can split the conformity into two components:

$$Conf(r) = Conf_0(r) + Conf_1(r) \tag{4.7}$$

$$Conf_0(r) = \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},0}\right) \tag{4.8}$$

$$Conf_1(r) = \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},1}\right) \tag{4.9}$$

We will introduce the following additional notation:

e_0 – number of void elements (zeros) in a binary matrix A

Due to the symmetric property of a binary matrix, we can say that

$$e_0 = (N \cdot M) - e_1 \tag{4.10}$$

and

$$\sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},0}\right) = e_0 - \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{0,a_{i,j}} \cdot \delta_{a_{r,j},1}\right) \tag{4.11}$$

and

$$\sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{0,a_{i,j}} \cdot \delta_{a_{r,j},1}\right) = (e_r \cdot N) - \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},1}\right). \tag{4.12}$$

Therefore, we are able to construct the following equation:

$$Conf(r) = \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},1}\right) + \tag{4.13}$$

$$+\left(\left((N \cdot M) - e_1\right) - \left((e_r \cdot N) - \sum_{j=1}^{M}\sum_{i=1}^{N}\left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},1}\right)\right)\right),$$

which can be further simplified as:

$$Conf(r) =$$
$$2 \cdot \sum_{j=1}^{M} \sum_{i=1}^{N} \left(\delta_{a_{r,j},a_{i,j}} \cdot \delta_{a_{r,j},1}\right) + ((N \cdot M) - e_1) - (e_r \cdot N) \tag{4.14}$$

and

$$Conf(r) = 2 \cdot Conf_1(r) + ((N \cdot M) - e_1) - (e_r \cdot N).$$

Q.E.D.

A pseudocode of the seriation algorithm for sparse datasets (Liiv, 2007b) is presented in Fig. 4.7 with the corresponding input format for datasets in Fig. 4.8. We have implemented the algorithm in ANSI C with *hsearch* hash table search for a quick enumeration and with the following data structure (*dlllist* for rows and *itemlist* for the items in a row):

```
struct dllist {
    int number;
    long rowsum;
    int elements;
    struct itemlist *first;
    struct dllist *next;
    struct dllist *prev;
};

struct itemlist {
    char *name;
    struct itemlist *next;
};
```

For the seriation of the columns (items), it is reasonable to use a similar data structure (a doubly linked list *dllist* for items and a linked list of rows containing those items). A dataset transposing pre-processing could be another possibility, but we have managed to skip the overhead with a further refining of the data structure for column-wise seriation. A source code or executables (for any well- known operating system) of the implemented algorithm are available upon request for research and benchmarking purposes.

```
PROCEDURE CALCULATE_CONFORMITY()
// ALL variables except I;J;CONFORMITY are global variables
  FOR I=0 TO number of rows
    IF I NOT in EXCLUDED THEN
      FOR J=0 TO number of elements in row I
        CONFORMITY[I]=CONFORMITY[I]+FREQ[DATA[I][J]]
      NEXT J
      CONFORMITY[I]=2*CONFORMITY[I]+(B-A)-
      (number of not excluded rows)*(number of elements on row I)
    ENDIF
  NEXT I
  RETURN CONFORMITY
END PROCEDURE

A = total number of elements
B = number of items (SKU) * number of rows (transactions)
FREQ is frequency table array for all items (index: name, value:
frequency)
DATA is an array for transactions and items within it, read
directly from the input file

FOR I=0 TO number of rows
  FIND MINIMAL ELEMENT MIN_ELEMENT FROM RESULTS OF
CALCULATE_CONFORMITY()
  IF PREVIOUS.value < MIN_ELEMENT.value THEN
    PRINT "--- END OF GROUP ---"
  PRINT MIN_ELEMENT.name AND MIN_ELEMENT.value
  PREVIOUS.value=MIN_ELEMENT.value
  A = A - FREQ[MIN_ELEMENT.name]
  FOR J=0 TO number of elements in row MIN_ELEMENT.name
    FREQ[DATA[MIN_ELEMENT.name][J]]--
    A--
  NEXT J
  B = B - FREQ.Count()
  ADD MIN_ELEMENT.name to array EXCLUDED
NEXT I
```

Figure 4.7 Seriation algorithm for binary sparse datasets

```
Item1 Item4
Item2
Item3 Item4
Item3
Item2
Item1 Item4
Item1 Item2 Item3 Item4
Item3 Item4
Item1 Item2 Item3
Item2
```

Figure 4.8 Input format for datasets

The input format is compatible with the common input file format of frequent itemset mining and association rule mining (Agrawal *et al.*, 1993, 1994), which are among the most popular problems in data mining.

We have evaluated the properties of the algorithm and measured the execution time with two datasets further discussed in Section 5.5. Results are presented in Table 4.4 (time is measured[14] with the standard Unix command "time" and the output is piped directly to */dev/null*). We have also added to the table the measurements of the execution time for frequent closed itemset mining (Pasquier *et al.*, 1999; Zaki & Hsiao, 2002) with different support thresholds, using the award-winning LCM2.1 algorithm (Uno *et al.*, 2004). Measurements of the LCM algorithm are brought in only as a reference time of a state of the art implementation, solving another data mining problem and, therefore, even remotely, do not imply the superiority of the presented implementation over LCM.

Table 4.4 Experiments with datasets used in Chapter 5

Dataset name	**Inventory Dataset1**	**Inventory Dataset2**
Rows (transactions)	1465	1735
Columns (items)	234	1601
Density	4.55%	1.74%
Seriation of rows (transactions)	0.669s	1.657s
Seriation of rows divided by density[15]	14.703s	95.229s
Seriation of columns (items)	0.042s	1.126s
Seriation of columns divided by density	0.923s	64.713s
Frequent (closed) itemset mining (support=30%)	0.096s	37.953s
Frequent (closed) itemset mining (support=25%)	0.184s	1m30.147s
Frequent (closed) itemset mining (support=20%)	0.436s	4m0.584s
Frequent (closed) itemset mining (support=15%)	1.422s	11m48.137s
Frequent (closed) itemset mining (support=10%)	4.993s	29m29.860s

[14] Hardware used: CPU Intel Pentium 4 3.0GHz, 512MB RAM.

[15] Execution times for seriation of the rows and columns are divided by the density of the dataset to provide a broad approximation of execution times with comparable implementation on the same hardware without the modified conformity weight calculation scheme.

5 An application to inventory management

5.1 Introduction

Achieving effective inventory control is critical to help to ensure the success of manufacturing and distribution companies. A large number of stock-keeping units (SKUs) makes it unfeasible to manage items individually. Therefore, they are commonly grouped together and generic inventory stock control policies are applied to each group. The most common method for classifying and prioritizing items is the annual dollar usage ranking method (Dickie, 1951), which is based on the Pareto's Principle. Vilfredo Pareto was an Italian economist who made an observation that a preponderance of the wealth was concentrated in the hands of a relatively small percentage of the population (Pareto, 1971). In the context of inventory control, Pareto's Principle is important because it recognizes that all the individual items that comprise the total inventory are not of equal relative importance. It implies that effort, time, money, and other assets to be spent or used in the control of an inventory should be allocated among the items in proportion to their relative importance (Zimmerman, 1975).

The classical single criterion ABC inventory classification is simple, straightforward and practical. Regardless of advances in inventory management methodologies, according to Zhang *et al.* (2001), most of the companies are still using the basic single-criterion ranking method.

However, using only one criterion for decision making, may lead in some cases, to mismanaging the assets. Several other factors have been suggested (Flores & Whybark, 1986, 1987; Flores *et al.*, 1992) that may override dollar value: availability, criticality, scarcity, obsolescence, substitutability, lead time, average unit cost. From the business perspective, they are all necessary, but multi-criteria decisions pose completely different obstacles - besides investment justification, common understanding and trust in priority coefficients has to be introduced. One could resolve those issues by letting the inventory manager go back through all items and reclassify any that they felt were misclassified. A large number of stock-keeping units makes such an approach ineffective or even unfeasible.

Previous inventory classification methods share another common property – a product-centered approach for classification procedure. The situation of a product being frequently bought, assembled or used together with some other product is often disregarded. Ignoring such behaviour may lead to customer retention for those who are accustomed of buying specific products in bundles. Rust *et al.* (2000, p.30) described a similar effect as *Profitable Product Death Spiral*, "in which decisions that seem to be increasing profitability alienate the customer by ignoring the effect of assortments of choices, eventually leading the firm to disaster". Rust *et al.* suggest conducting focus-group interviews to determine those products that have interdependencies. To narrow this gap, we suggest new methods based on data mining and seriation, using the transaction history records available in most inventory management softwares. The seriation and visualization approach is related to group technology methods in manufacturing (i.e. machine-component cell formation, see Section 2.8 for a longer discussion), but the goal is rather to establish a two-mode typicality scale, not to form blocks near the diagonal.

We will discuss several aspects of well-known inventory classification strategies (in Section 5.2) and propose two efficient methods for demand association conflict detection (in Section 5.3), which are implementable both in single and multi-criteria classification environments. Experimental results for two warehouse datasets are included and analyzed, followed by the discussion.

5.2 Related work

The term "ABC Inventory Analysis" was first coined in the early 1950s by H. F. Dickie (1951) who gave an overview of the analysis in general and the results of the implementation in General Electric Company. Success stories in direct inventory reduction and turnover increasement were presented.

Zimmerman (1975) warned about using single criteria approaches to complex inventory problems and emphasized the common fallacy - misuse of a statistical technique. The current chapter and the methodology hopes to overcome exactly the specific focal problem brought out in Zimmerman's paper - some "C" items should be closely monitored regardless of their ABC classification. We will call such classification situations conflicts that must be reconciled by reclassification, rather than making exceptions in the system.

Another common classical ABC classification fallacy that was also brought out by Zimmerman, has become the main issue addressed in several papers (Flores & Whybark, 1986, 1987; Cohen & Ernst, 1988; Flores *et al.*, 1992; Güvenir, 1995; Güvenir & Erel, 1998; Zhang *et al.*, 2001; Partovi & Anandarajan, 2002; Lei *et al.*, 2005; Ramanathan, 2006) - distribution by value as the only criterion can lead to gross errors and mismanaging the assets.

Flores and Whybark (1986, 1987) suggested that multiple criteria ABC classification can provide a more comprehensive managerial approach, allowing consideration of other criteria such as lead time and criticality. They presented a joint criteria matrix procedure that could help the management to derive combined criteria (usually a combination from dollar value and criticality). Unfortunately, the method only works best with two criteria - if all criteria are important and need to be incorporated in the analysis, the task may become unmanageable (Flores & Whybark, 1986), if not impossible.

Saaty's Analytic Hierarchy Process (AHP) was used by Flores *et al.* (1992) to reduce multiple criteria to a univariate and consistent measure. AHP allows decision makers with a finite set of alternatives to combine multiple objectives (Saaty, 1977, 1980). Inventory management can include several criteria and reduce them to a single variable, using a linear combination of the variables. A clear drawback of the approach by Flores *et al.* (1992) is that more managerial time is needed to understand the process and to develop more information for each inventory item.

Neural networks and genetic algorithms (Güvenir, 1995; Güvenir & Erel, 1998; Partovi & Anandarajan, 2002) are very effective with an inventory classification when it comes to optimizing a set of parameters that represent the weights of criteria. Nevertheless, a possible limitation of such approaches is that they generate black box models - the structure of weights is never explained.

A contrasting unsupervised approach, referred to as the ORG method, was proposed by Cohen and Ernst (1988), who suggested clustering of the items based on 40 operational attributes of each item. They formulated the SKU-based control problem as an optimization problem where the objective is to obtain the minimum number of groups which satisfy both the operational performance (the penalty associated with the application of generic policies relative to individual-based policies) and constraints (a minimal level of statistical discrimination). Such an approach enables the generation of operations-related groups, which are based on the common properties and features of items, but it could fail to notice the non-product-based associations between items.

In this chapter we suggest two different and more customer-centered approaches for solving the problem in order to support the reclassification and prioritizing items according to the demand associations (dependencies in customer behaviour). We propose this approach and methods as an enhancement, not as a replacement to the existing ABC inventory classification.

5.3 Proposed methodology

We propose a two-step solution for the inventory classification enhancement using association rules and, depending upon the necessity, refinement of the solution using seriation. The initial step involves the use of an association rules framework (Agrawal *et al.*, 1993, 1994; also known as *market basket analysis*) for calculating the demand association criterion. Items which are frequently bought, assembled or used together should be applied with the same management policy and classified in the same class. The criterion is measured in the ordinal scale and can represent either a non-existent, a normal (from classical ABC analysis category "B" to "A" or "C" to "B") or a strong (from "C" to "A") recommendation for reclassification. In most cases, no recommendation is given, which allows better managerial concentration on special cases.

We provide a formal model for association rules framework with required restrictions. Let I = $i_1,i_2,\ldots,i_m$ be a set of binary attributes, called items. Let D be a set of transactions. Each transaction *t* is represented as a binary vector, with

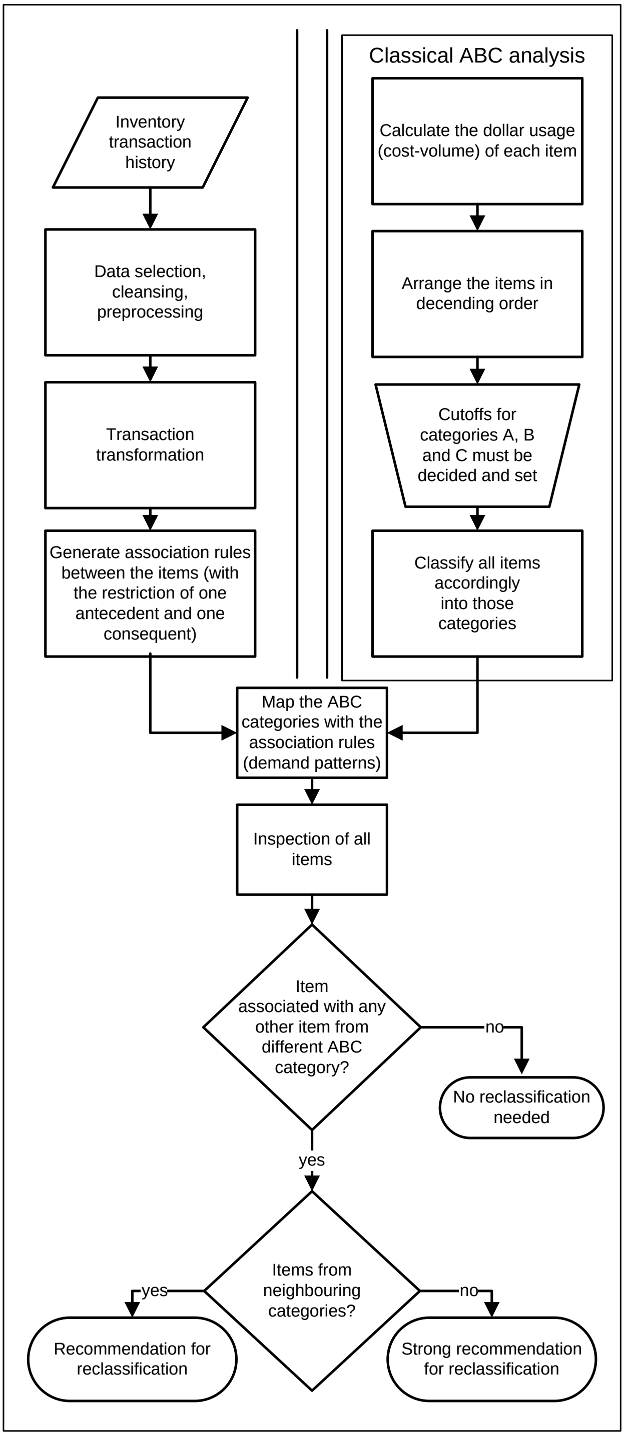

Classical ABC analysis
Inventory transaction history
Data selection, cleansing, preprocessing
Transaction transformation
Generate association rules between the items (with the restriction of one antecedent and one consequent)
Calculate the dollar usage (cost-volume) of each item
Arrange the items in decending order
Cutoffs for categories A, B and C must be decided and set
Classify all items accordingly into those categories
Map the ABC categories with the association rules (demand patterns)
Inspection of all items
Item associated with any other item from different ABC category?
no
No reclassification needed
yes
Items from neighbouring categories?
yes
Recommendation for reclassification
no
Strong recommendation for reclassification

Figure 5.1 Proposed method for demand association conflict identification within the initial ABC classification (Liiv, 2006)

t[k]=1 ↔ $i_k \in t$ if i_k was bought, assembled or used in transaction *t*. We also have annual dollar usage value for the item i_k.

By an association rule, we mean an implication of the form $i_h \rightarrow i_j$, where i_h is a single item from I, and i_j is a single item in I that is not i_h (i.e., the number of items as an antecedent and consequent is restricted to one). The *confidency* of a rule is the conditional probability that a randomly chosen transaction from D that matches i_h also matches i_j. As noted by Brin *et al.* (1997), the symbol → can be slightly misleading out of this context, since such a rule does not correspond to real implications and the confidence measure is merely an estimate of the conditional probability of i_j given i_h.

In this formulation, the problem of calculating the demand association criterion can be decomposed into three subproblems:

1. After data acquisition and pre-processing, generate all two-item association rules that have fractional transaction confidency above a certain threshold, which is based on managerial judgement.
2. Classify all items in I, using the annual dollar usage ranking method.
3. Calculate the demand association criterion for all items in I, using the following algorithm. The recommendation for reclassification for an item i_k is:
    - *non-existent* (or "0"), if no rules exist where the item is associated with an item from different annual dollar usage ranking ABC class;
    - *normal* (or "1"), if at least one rule exists where the item is associated with an item from a different annual dollar usage ranking ABC class - item from "B" associated with an item from "A" or item from "C" with an item from "B";
    - *strong* (or "2"), if at least one rule exists where the item is associated with an item from a different annual dollar usage ranking ABC class - item from "C" associated with an item from "A".

Hence, we are interested in rules where the antecedent and the consequent are from different ABC classes. However, depending on the situation, defining a proper threshold level for rule *confidency* can be a complex and time-consuming task for an inventory manager along with the large amount of demand-based association rules generated with such an approach. Moving towards industries with strongly interdependent products can make the list of reclassification recommendations economically unreasonable to implement and, therefore, a compromise solution of setting up an association chain out of seriation results is suggested.

Let Q denote the permutation of n items in I according to the result of the seriation procedure presented in Section 4.3. Then, demand association conflicts concerning i-th element can be detected with the following conditions (with the convention $1 \leq i \leq (n-1)$):

Strong conflict (recommendation for reclassification):
Class($Q(i$-1))=A *and* Class($Q(i)$)=C *and* Class($Q(i$+1))=A
Conflicts (recommendation for reclassification):
Class($Q(i$-1))=A *and* Class($Q(i)$)=B *and* Class($Q(i$+1))=A
Class($Q(i$-1))=B *and* Class($Q(i)$)=C *and* Class($Q(i$+1))=B
Class($Q(i$-1))=A *and* Class($Q(i)$)=C *and* Class($Q(i$+1))=B

In other words, if a product is classified to a lower priority class according to the classical ABC analysis, but, its position in the demand-based association chain is between products with higher priority (e.g. A, C, A), it should be considered for reclassification. Such demand association conflicts can be denoted using numerical values in ordinal scale for using as one objective in multi-criteria scenarios.

5.4 A numerical example

Let us look at the following numerical example. Table 5.1 shows four items referred to as i_1 to i_4. Each transaction t is represented as a binary vector, with $t[k]$=1 $\leftrightarrow$ $i_k \in t$ if i_k was bought, assembled or used in transaction t. The quantity of each item in the transaction history record is ignored, as we are concerned about the association. *DollarValue* of an item (in the last row) is the result of the classical ABC analysis, which is independently calculated from the binary transaction data. In most cases, an annual dollar-usage value can be extracted from the summary or ABC analysis reports, depending on the inventory management software.

Table 5.1 Transactions and Dollar-Usage values

	i_1	i_2	i_3	i_4
t_1	1	0	0	1
t_2	0	1	0	0
t_3	0	0	1	0
t_4	0	0	1	0
t_5	0	1	0	0
t_6	1	0	0	1
DollarValue	36	6	1	1

We can see that items i_1 and i_4 are twice (transactional rows 1 and 6) bought, assembled or used together, therefore the conditional probability that a randomly chosen transaction from Table 5.1 with i_1 also has i_4, is 100%. This indicates that no transactions exist where item i_1 did not co-occur with i_4.

According to annual dollar-usage ranking, item i_1 is classified as "A", i_2 as "B", i_3 and i_4 as "C". Despite that, we see a situation that we would call *a demand association conflict* within the initial classification. An item from category C (according to *DollarValue*) is always bought, assembled or used together (according to association rules) with the item from category A. Therefore, a strong recommendation for reclassification is given for i_4 under such circumstances.

The previous example illustrated the motivation for demand association approach in a single criterion environment. It is also possible to implement the recommendation as one objective in a multiple criteria classification environment based on Analytic Hierarchy Process (Saaty, 1977, 1980).

For the purposes discussed in the previous section, in some scenarios, it is preferable to refine the reclassification goal using an association chain established from the results of seriation. After applying the algorithm presented in Section 4.3 (Fig. 4.7; data has to be in the format presented in Fig. 4.8), we will get the following seriation result of columns for Table 5.1 (new order for items):

```
1. Element I3; Weight 12
2. Element I2; Weight 10
----- END OF GROUP -----
3. Element I4; Weight 12
4. Element I1; Weight 6
```

Secondly, when we apply the seriation algorithm to rows, we will get the following result (new order for transactions):

```
1. Element T6; Weight 12
2. Element T1; Weight 8
----- END OF GROUP -----
3. Element T5; Weight 12
4. Element T2; Weight 8
5. Element T4; Weight 8
6. Element T3; Weight 4
```

If we sort the matrix according to the new orders obtained, we get a permutated matrix presented in Table 5.2. Sometimes it is reasonable for efficiency purposes to calculate only the conformity weights (procedure `CALCULATE_CONFORMITY` in the algorithm in Fig. 4.7) for the objects instead of performing full step-by-step iterations (lines 12-24 in the algorithm; Fig. 4.7). For our numerical example, the conformity (typicality) weights would be the following:

- i_1 (16), i_4 (16), i_2 (12), i_3 (12);
- t_2 (14), t_3 (14), t_4 (14), t_5 (14), t_6 (12), t_1 (12).

One can see that rankings according to conformity weights and the order obtained by the whole algorithm are similar. If such results satisfy, it enables us to save a lot of computational time. However, plain conformity calculation provides neither comparable neighbouring similarity maximization nor the identification of cluster boundaries.

Table 5.2 Previous table (5.1) after reordering

	i_1	i_4	i_2	i_3	*Influence weight*
t_6	1	1	0	0	12
t_1	1	1	0	0	8
t_5	0	0	1	0	12
t_2	0	0	1	0	8
t_4	0	0	0	1	8
t_3	0	0	0	1	4
DollarValue	36	**1**	6	1	
Influence weight	6	12	10	12	

From this example we also notice an interesting property of the algorithms. Only two groups of transactions were detected in the dataset, although using visual investigation, we can identify three. Such a difference only emerges with very small datasets and it actually demonstrates that there are only two equally balanced groups with regard to the overall structural pattern. Influence weight of rows and columns (in Table 5.2) denote the weight of the conformity entity-to-set function at the time of the step-wise elimination of that object.

Another important property of such inventory classification methodology is that we also get the transaction segmentation (transition description from typical inventory transactions to untypical) and understanding of the inner structure of the item-transaction co-behaviour.

From the results of such ranking of the items, demand association conflicts are easily detectable. For example, in Table 5.2, the dollar value of an item i_4 is significantly lower than the neighbouring elements and, therefore, can be potentially misclassified when demand associations are not considered. Conflicts between classes are thus obtained from ABC-classification and *de facto* customer behaviour - products from different ABC classes that have strong interdependencies should be reclassified.

For feasibility and practicality purposes, instead of matrices with numerical values, dot plotting (black dot denoting value "1" in the matrix) is used for the matrix permutation visualization of larger datasets (e.g. results of the experiments in Fig. 5.3 and 5.4).

5.5 Experimental results

The aim of the experiments is to investigate how common demand association conflicts are in real world scenarios. Therefore, we first evaluate the proposed method based on association rules framework (Liiv, 2006) with respect to the initial classification results and enumerate demand associations between items from different categories. Secondly, demand association conflicts are detected using seriation (Liiv, 2007b). Corresponding reordered matrices are presented at the end of this section.

Two wholesale companies participated in the study, anonymized datasets are available upon request for benchmarking and research purposes (Liiv, 2006). The number of SKUs in the organizations were 234 (Dataset 1) and 1601 (Dataset 2), respectively. Data were gathered and prepared from the transaction history records in the inventory management software of each organization.

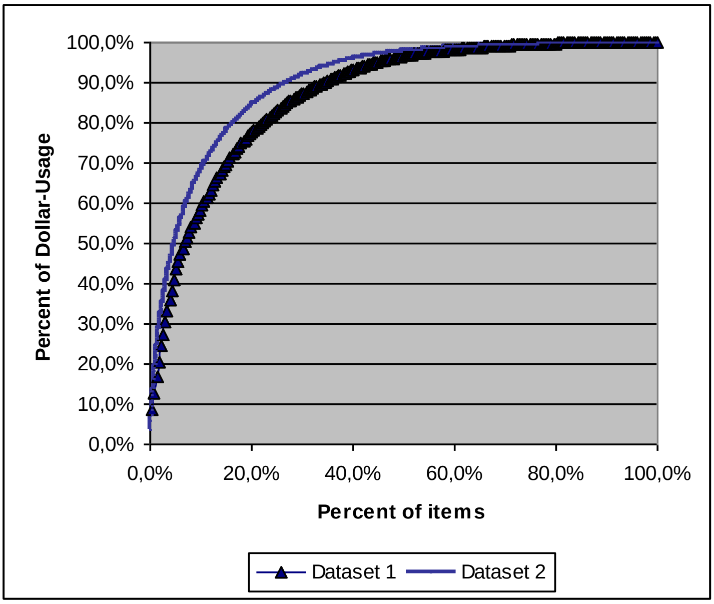


Figure 5.2 Distribution of dollar-usage values in datasets

Pre-processing activities included data selection, cleansing and transformation. The goal of pre-processing was to have two distinct input files for the method:

- Results of the classical ABC analysis (cut-offs for ABC categories and items in descending order with respect to dollar-usage values or any other chosen criterion).
- Transaction data in suitable format for the extraction process of association rules.

The following steps were performed for both datasets:

1. All the association rules were extracted from the transactions. For exploratory purposes different rule confidencies were tested (25%, 50%, 60%, 70%, 75%, 80%, 85%, 90%).
2. ABC categories for dollar-usage were developed, distributions of dollar-usage values for both datasets are shown in Fig. 5.2.
3. For both organizations, ABC categories were defined as 75%, 15%, and 10% of the dollar-usage, respectively.
4. We enumerated all the rules (for all tested confidencies) where the antecedent and the consequent were from different ABC categories.
5. Demand association criteria were calculated for all items and confidencies, which allows managers to perform a subjective evaluation in order to find the optimal confidency threshold.

Table 5.3 Dataset 1 and Enumeration results

	25 %	*50 %*	*60 %*	*70 %*	*75 %*	*80 %*	*85 %*	*90 %*
A→B	103	6	6	4	0	0	0	0
A→C	1	0	0	0	0	0	0	0
B→A	549	102	35	12	11	7	2	1
B→C	7	1	0	0	0	0	0	0
C→A	69	11	3	0	0	0	0	0
C→B	19	5	2	1	1	0	0	0

Table 5.4 Dataset 2 and Enumeration results

	25 %	*50 %*	*60 %*	*70 %*	*75 %*	*80 %*	*85 %*	*90 %*
A→B	1813	334	114	26	15	5	3	1
A→C	271	1	0	0	0	0	0	0
B→A	2629	629	239	95	65	26	15	6
B→C	396	16	2	1	1	1	0	0
C→A	436	98	30	14	6	1	1	0
C→B	177	61	24	8	3	0	0	0

The enumeration results for both organizations are shown in Tables 5.3 (Dataset 1) and 5.4 (Dataset 2), associations within the same category were not included. The values should be interpreted as numbers of demand association conflicts, relevant confidency level to be chosen depends on the managerial judgement. The results should illustrate the relative amount of ABC classification conflicts with the current prerequisites. Several items with strong recommendations for reclassification were found. Corresponding experiments with establishing an association chain using seriation gave the following results:

- Seriation of *Dataset1* (234 items) with plain conformity weights (procedure CALCULATE_CONFORMITY in the algorithm in Fig. 4.7) found **6** conflicts and **no** strong conflicts.
- Seriation of *Dataset1* (234 items) performing full step-by-step iterations of "minus" technique using conformity weights (lines 12-24 in the algorithm; Fig. 4.7) found **8** conflicts and **no** strong conflicts.
- Seriation of *Dataset2* (1601 items) with plain conformity weights found **40** conflicts and **16** strong conflicts.
- Seriation of *Dataset2* (1601 items) performing full step-by-step iterations of "minus" technique using conformity weights found **43** conflicts and **18** strong conflicts.

The amount of conflicts detected, using the compromise of establishing a single association chain instead of thousands of association rules, is substantially more feasible for managerial inspection. Seriation results of the datasets are visualized in Fig. 5.3 (matrix sorted according to *DollarValue* on the left, conformity analysis in the middle and "minus" technique on the right) and 5.4 (matrix sorted according to *DollarValue* on the top, conformity analysis in the middle and "minus" technique in the bottom position).

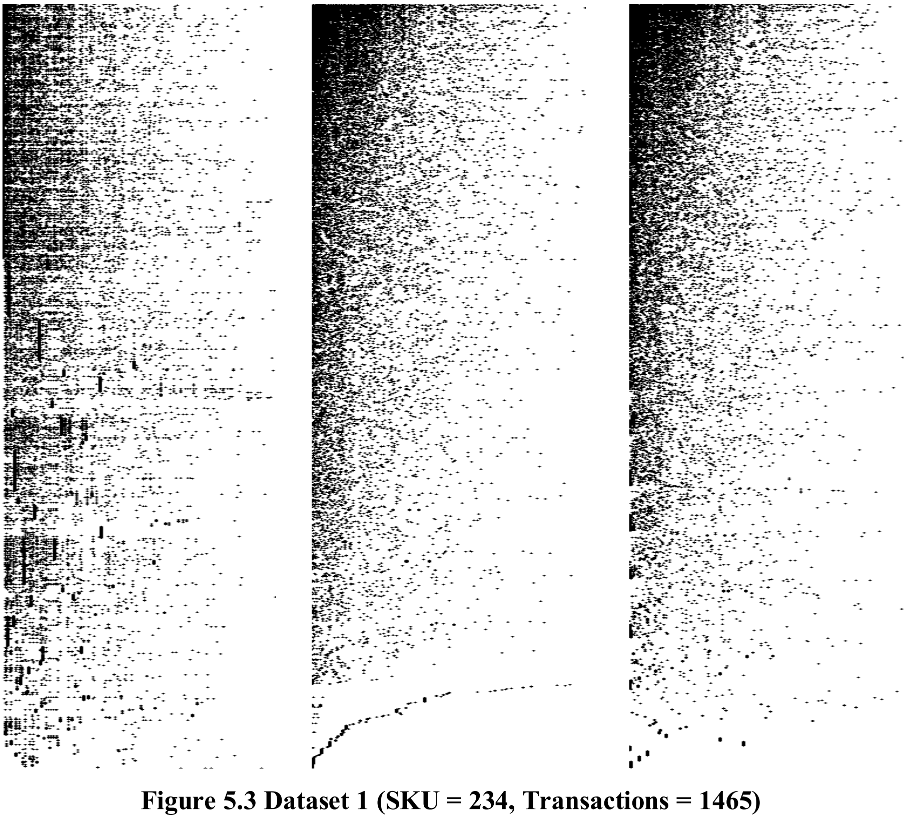


Figure 5.3 Dataset 1 (SKU = 234, Transactions = 1465)

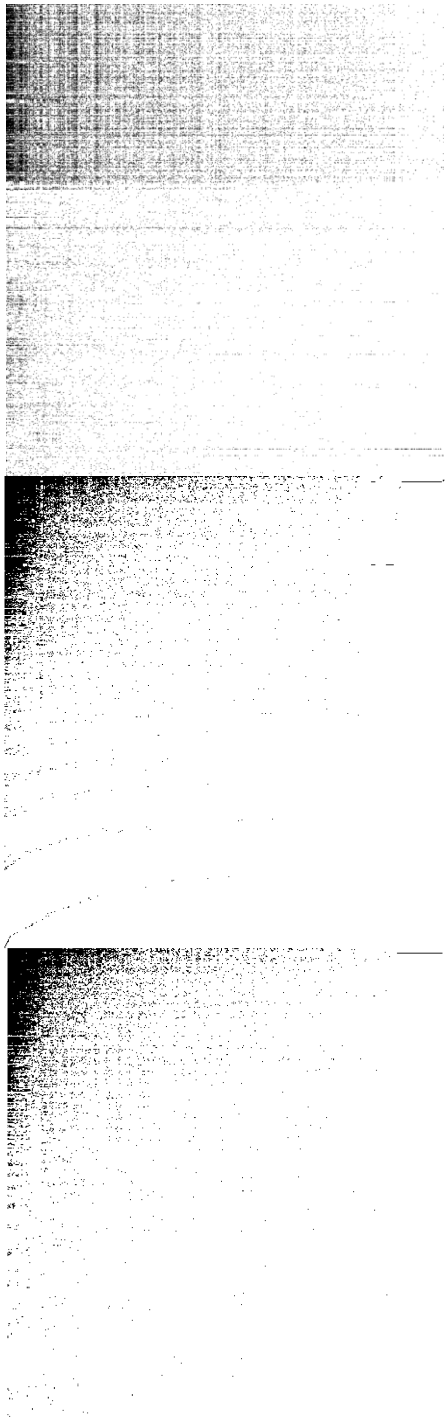


Figure 5.4 Dataset 2 (SKU = 1601, Transactions = 1735)

5.6 Discussion

We can observe from the experimental results in real-world scenarios that Pareto's principle (80/20) in inventories also holds to the underlying hidden natural order – the majority of the relations and structure in the dataset is concentrated to a relatively small percentage of objects. Such a structure of power law relationships does not succumb to any of the commonly reported structural patterns emerging after seriation of matrices (types 1-3 in Section 3.2, Fig. 3.1-3.3). Therefore, we made the distinction from other structural patterns and referred to it as a "Pareto seriation" (type 4 in Section 3.2, Fig. 3.4).

One can notice from both datasets that there is already a visually perceivable natural order in initial matrices before seriation. This is due to the matrix not being randomized similarly to toy-examples of reordering rows and columns, but already pre-ordered according to the dollar-usage values. In addition to more compact representation of the structure and detecting the dependency conflicts between products, both reordering results were more favourable as well according to the objective function presented in Chapter 3. If both reordering solutions are acceptable precision-wise, one should consider applying computationally less expensive plain conformity analysis, which is natively and directly implementable in all databases using the approach presented in Section 4.2.

Regardless of advances in inventory management methodologies, managers are accustomed to working with the simple and practical ABC inventory classification, although using single criteria approaches to complex inventory problems may lead to mismanaging the assets.

In this chapter, we suggested a different and more customer-centered approach for solving particular fallacies of the classical ABC analysis - using demand associations for classification enhancement. In addition to the annual dollar usage ranking method, items which are frequently bought, assembled or used together, should be applied to the same management policy and classified in the same category. The presented approach provides inventory managers a straightforward remedy to reduce dependency conflicts in the results of the classical ABC analysis. It is not suggested to replace the classical ABC-diagram (distributions of dollar-usage values), but to refine the results and use the exploratory data analysis benefits of seriation results to understand interrelations between products and customer behaviour - the transition of a typical transaction to an untypical. The advantage of the seriation-based method presented in this section is that no parameter configuration (e.g. setting a threshold) is needed, which is otherwise done with test-and-trial strategy to establish domain-specific rules of thumb.

The presented results with two warehouse datasets justify the demand association approach and illustrate the need for considering non-product-based associations between items.

6 Conclusions

Seriation is an exploratory combinatorial data analysis technique to reorder objects into a sequence along a one-dimensional continuum so that it best reveals regularity and patterning among the whole series. A representative variety of related work on seriation problems was highlighted in Chapter 2, where independent work in different disciplines has corroborated the advantages of understanding structural patterns in the system by reordering rows and columns in matrices.

Unsupervised learning, using seriation and matrix reordering, allows pattern discovery simultaneously at three information levels: local fragments of relationships, sets of organized local fragments of relationships and an overall structural pattern. It, therefore, combines, in a single result, and enhances the structural analysis abilities of popular unsupervised data mining methods, like clustering and association rules. We advocate that seriation should be put on a par with standard data mining techniques like clustering and association rules due to the lack of their ability to analyse complex structures and to defocus from details to global relationships. The concept of seriation can work towards attaching data mining together with the advantages in information visualization and social network analysis, which emphasize the importance of simultaneous consideration of global and local patterns.

Seriation problems, however, are computationally much more expensive and, therefore, do not scale well for present real-world problems. In this thesis, we have proposed an algorithm for sparse binary matrix seriation to remedy this challenge in comparable scenarios. The algorithm provides a fair compromise between computational complexity and mere enumeration of direct and explicit relationships in the datasets. We have also proposed a way for an expeditious implementation of the seriation approach natively in databases with standard relational algebra and relational calculus in structured query language (SQL) without any use of external procedures or functions.

An application to inventory management was presented to provide the currently missing functionality for reclassifying and prioritizing items according to dependencies in customer behaviour. Therefore, besides the evaluation of the proposed sparse binary matrix seriation algorithm, a novel solution was suggested to a domain-specific problem, using data mining and seriation.

The presented unified view for seriation is based on the certainty that it is advantageous to benefit from the domino effect triggered by a breakthrough from another field. Moreover, not working toward finding or at least seeking for links between related competences in different fields is detrimental in the long run to all the scientists working on it along with all potential application areas. Following this idea, we proposed the formulation and an objective function for parameter-free seriation, based on Kolmogorov complexity and data compression. Regularity is something that is a core competence in a data compression community and, therefore, it would not be reasonable to compete with the state-of-the-art regularity detection achieved in that field. This type of linkage is advantageous vice versa as well: all new contributions claiming to perform better regularity and pattern identification (e.g. for two-mode (n-mode) seriation results) can be presented and discussed in the data compression and minimum description length principle community. We have also investigated the agreement between compression-based and other evaluation measures for seriation. Those experiments also demonstrate that the heuristic approach used in this research gives reasonable and comparable results with other algorithmic approximation approaches.

In the future, reordering the matrices should be a ubiquitous and common practice for everybody inspecting any data table. According to Bertin's (1981) emphasis, a matrix or data table is never constructed conclusively, but reconstructed until all relations which lie within it can be perceived. However, seriation cannot be considered ubiquitously *usable*, until implemented and shipped as a standard tool in any spreadsheet and internet browser for enabling such analysis. Then one can say that seriation and matrix reordering is usable. That is the main future goal for seriation.

We hope that the contributions of this dissertation stimulate and augment the collaboration between scholars from different disciplines. Especially on the way of bringing this flavor of exploratory data analysis, seriation, closer to enduser consumption.

6.1 Contributions of the dissertation

Contributions of this dissertations can be summarized as follows:

- a new perspective to view seriation problems from a unified single perspective, using Kolmogorov complexity and data compression;
- a procedure-free implementation of conformity analysis with structured query language;
- a new seriation algorithm for binary sparse datasets, which was proved to be faster than the classical algorithm (Vyhandu, 1989; Mullat 1976a) in proportion to the sparsity;
- a novel application to inventory management and a solution to specific inventory classification problems using data mining and seriation;

6.2 Directions for future research

This thesis reflects what Chen (2006, p. xii) calls a *structure-centric* tradition. Clearly, a next generation seriation algorithm should, similarly to the trends in information visualization and social network analysis, be developed to govern all the dynamics of an underlying phenomenon. In other words, to be more *dynamics-centric* – to grasp change in several layers – the growth, evolution, and development (Chen, 2006, p. xii). It is not, however, impossible that one could establish it on the same foundations. A good example for that is Moreno's (1934, 1953) work in sociometry, what was also covered in the literature review chapter. His research on dynamics had a very special and interesting distinction – he was not just interested in change, but in the change toward positive scenarios. We are not sure if his theories on group therapy are explicitly applicable toward developing a discipline of "data therapy", but it is certainly an interesting avenue for research.

Practically every chapter of the thesis provides numerous problems which need to be further investigated. A non-exhaustive list of interesting directions for future research includes the following:

- algorithmic enhancements towards lower computational complexity – even with sparse datasets, subcubic complexity is infeasible with very large problems;
- automatic on-the-fly decision of heuristics and ensemble approaches;
- parameter-free seriation of column-conditional data with mixed data types and the evaluation using data compression;
- higher-way and higher-mode seriation using the presented approaches and algorithms;
- time series seriation;
- text mining with seriation;
- applying approaches and algorithms presented in this dissertation to bioinformatics and taxonomizing biclustering and coclustering methods, which currently evolve and transform at a rapid speed;
- information visualization and human-computer interaction (HCI) issues with very large dataset seriation;
- research on database systems to enable native ordering of the rows not only sequentially according to the given columns, but simultaneously as an aggregated order with maximal agreement over individual orders resulted by the columns.

References

Afrati, F. , Das, G. , Gionis, A. , Mannila, H. , Mielikäinen, T. , & Tsaparas, P. (2005). Mining chains of relations. In *The 5th IEEE International Conference on Data Mining, Houston, TX, USA, November 27-30, 2005* (pp.553-556).

Agrawal, R. , & Srikant, R. (1994). Fast Algorithms for Mining Association Rules. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94), Santiago de Chile, Chile, September 12-15, 1994* (pp.487-499).

Agrawal, R. , Imielinski, T. , & Swami, A. (1993). Mining Association Rules Between Sets of Items in Large Databases. In *Proceedings of the ACM SIGMOD Conference on Management of Data, Washington, D.C., May 26-28, 1993* (pp.207-216).

Arabie, P. , & Hubert, L. J. (1990). The Bond Energy Algorithm Revisited. *IEEE Transactions on Systems, Man, and Cybernetics*, *20*(1), 268-274.

Arabie, P. , & Hubert, L. J. (1992). Combinatorial Data Analysis. *Annual Review of Psychology, 43,* 169-203.

Arabie, P. , & Hubert, L. J. (1996). An overview of combinatorial data analysis. In P. Arabie, L. J. Hubert, & G. De Soete (Eds.) *Clustering and Classification*, River Edge, NJ: World Scientific, pp.5-63.

Arabie, P. , Hubert, L. J. , & De Soete, G. (Eds.) (1996). *Clustering and Classification*. River Edge, NJ: World Scientific.

Arabie, P. , Hubert, L. J. , & Schleutermann, S. (1990). Blockmodels from the bond energy approach. *Social Networks*, *12,* 99-126.

Ascher, M. , & Ascher, R. (1963). Chronological Ordering by Computer. *American Anthropologist, 65*(5), 1045-1052.

Askin, R. G. , & Chiu, K. S. (1990). A graph partitioning procedure for machine assignment and cell formation in group technology. *International Journal of Production Research*, *28*(8), 1555-1572.

Askin, R. G. , & Subramanian, S. P. (1987). A cost-based heuristic for group technology configuration. *International Journal of Production Research, 25*(1), 101-113.

Askin, R. G. , Cresswell, S. H. , & Goldberg, J. B. (1991). A Hamiltonian path approach to reordering the part-machine matrix for cellular manufacturing. *International Journal of Production Research, 29*(6), 1081-1100.

Balakrishnan, J. , & Jog, P. D. (1995). Manufacturing cell formation using similarity coefficients and a parallel genetic TSP algorithm formulation and comparison. *Math and Comput Model*, *21*(12), 61-73.

Bar-Joseph, Z. , Demaine, E. D. , Gifford, D. K. , Srebro, N. , Hamel, A. M. , & Jaakkola, T. S. (2003). K-ary Clustering with Optimal Leaf Ordering for Gene Expression Data. *Bioinformatics*, *19*(9), 1070-1078.

Bar-Joseph, Z. , Gifford, D. K. , & Jaakkola, T. S. (2001). Fast optimal leaf ordering for hierarchical clustering. *Bioinformatics*, *17*(S1), S22-S29.

Bederson, B. B. , & Shneiderman, B. (2003). *The Craft of Information Visualization: Readings and Reflections*. San Francisco, CA: Morgan Kaufmann.

Bertin, J. (1967). *Sémiologie graphique: les diagrammes, les réseaux, les cartes*. Paris: Mouton.

Bertin, J. (1981). *Graphics and Graphic Information Processing*. Berlin: Walter de Gruyter (Translated by W. J. Berg and P. Scott).

Beum, C. O. , & Brundage, E. G. (1950). A Method for Analyzing the Sociomatrix. *Sociometry*, *13*(2), 141-145.

Boctor, F. F. (1991). A linear formulation of the machine-part cell formation problem. *International Journal of Production Research*, *29*(2), 343-356.

Boe, J. W. , & Cheng, C. H. (1991). A close neighbor algorithm for designing cellular manufacturing systems. *International Journal of Production Research*, *29*(10), 2097-2116.

Borgatta, E. F. , & Stolz, W. (1963). A Note on a Computer Program for Rearrangement of Matrices. *Sociometry*, *26*(3), 391-392.

Brainerd, G. W. (1951). The Place of Chronological Ordering in Archaeological Analysis. *American Antiquity*, *16*(4), 301-313.

Brin, S. , Motwani, R. , & Silverstein, C. (1997). Beyond Market Baskets: Generalizing Association Rules to Correlations. In *Proceedings of The ACM SIGMOD International Conference on Management of Data, Tucson, AZ, USA, May 13-15, 1997* (pp.265-276).

Brusco, M. , & Stahl, S. (2005). *Branch-and-Bound Applications in Combinatorial Data Analysis (Statistics and Computing)*. New York: Springer.

Burbidge, J. L. (1961). The New Approach to Production. *Production Engineer*, *40*(12), 769-784.

Burbidge, J. L. (1963). Production flow analysis. *Production Engineer*, *42*(12), 742-752.

Burbidge, J. L. (1971). Production flow analysis. *Production Engineer, 50*(4), 139-152.

Burbidge, J. L. (1977). A manual method for production flow analysis. *Production Engineer, 56,* 34-38.

Caraux, G. (1984). Reorganisation et représentation visuelle d'une matrice de donnée numériques: un algorithme itératif. *Revue de Statistique Appliquée, 32*(4), 5-23.

Caraux, G. , & Pinloche, S. (2005). PermutMatrix: a graphical environment to arrange gene expression profiles in optimal linear order. *Bioinformatics, 21*(7), 1280-1281.

Carneiro, R. L. (1962). Scale Analysis as an Instrument for the Study of Cultural Evolution. *Southwestern Journal of Anthropology, 18*(2), 149-169.

Carrie, A. S. (1973). Numerical taxonomy applied to group technology and plant layout. *International Journal of Production Research, 11*(4), 399-416.

Carroll, J. D. , & Arabie, P. (1980). Multidimensional scaling. *Annual Review of Psychology, 31,* 607-649.

Chan, H. , & Milner, D. A. (1982). Direct clustering algorithm for group formation in cellular manufacturing. *Journal of Manufacturing Systems, 1*(1), 65-74.

Chandrasekharan, M. P. , & Rajagopalan, R. (1986a). An ideal seed non-hierarchical clustering algorithm for group technology. *International Journal of Production Research, 24*(2), 451-464.

Chandrasekharan, M. P. , & Rajagopalan, R. (1986b). MODROC: an extension of rank order clustering for group technology. *International Journal of Production Research, 24*(5), 1221-1233.

Chandrasekharan, M. P. , & Rajagopalan, R. (1987). ZODIAC – an algorithm for concurrent formation of part-families and machine-cells. *International Journal of Production Research, 25*(6), 835-850.

Chandrasekharan, M. P. , & Rajagopalan, R. (1989). Groupability: an analysis of the properties of binary data for group technology. *International Journal of Production Research, 27,* 1035-1052.

Chen, C. (2006). *Information Visualization: Beyond the Horizon*. London: Springer-Verlag.

Chen, C.-H.(2002). Generalized association plots: Information visualization via iteratively generated correlation matrices. *Statistica Sinica, 12,* 7-29.

Chen, C.-H., Hwu, H.-G., Jang, W.-J., Kao, C.-H., Tien, Y.-J., Tzeng, S. , & Wu, H.-M.(2004). Matrix Visualization and Information Mining. In *Proceedings in Computational Statistics 2004 (Compstat 2004), Prague, Czech Republic, August 23-27, 2004* (pp.85-100).

Chen, C.-H., Härdle, W. , & Unwin, A. (2008). *Handbooks of Computational Statistics: Data Visualization*. Heidelberg: Springer-Verlag.

Cheng, Y. , & Church, G. M. (2000). Biclustering of expression data. In *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology, La Jolla / San Diego, CA, USA, August 19-23, 2000* (pp.93-103).

Chu, C. (1997). An improved neural network for manufacturing cell formation. *Decision Support Systems, 20,* 279-295.

Chu, C. , & Hayya, J. C. (1991). A fuzzy clustering approach to manufacturing cell formation. *International Journal of Production Research, 29*(7), 1475-1487.

Chu, C. , & Tsai, M. (1990). A comparison of three array-based clustering techniques for manufacturing cell formation. *International Journal of Production Research*, *28*(8), 1417-1433.

Clarke, K. , & Warwick, R. (2001). *Change in Marine Communities: An Approach to Statistical Analysis and Interpretation, 2nd ed*. Plymouth, UK: Primer-E.

Climer, S. , & Zhang, W. (2006). Rearrangement Clustering: Pitfalls, Remedies, and Applications. *Journal of Machine Learning Research*, *7*, 919-943.

Cohen, M. , & Ernst, R. (1988). Multi-item classification and generic inventory stock control policies. *Production and Inventory Management Journal*, *29*(3), 6-8.

Coombs, C. H. (1964). *A Theory of Data*. New York: John Wiley and Sons.

Czekanowski, J. (1909). Zur Differentialdiagnose der Neandertalgruppe. *Korespondentblatt der Deutschen Gesellschaft für Anthropologie, Ethnologie und Urgeschichte*, *XL*(6/7), 44-47.

Dempsey, P. , & Baumhoff, M. (1963). The Statistical Use of Artifact Distributions to Establish Chronological Sequence. *American Antiquity*, *28*(4), 496-509.

Deutsch, S. B. , & Martin, J. J. (1971). An Ordering Algorithm for Analysis of Data Arrays. *Operations Research*, *19*(6), 1350-1362.

Dhillon, I. S. , Mallela, S. , & Modha, D. S. (2003). Information-Theoretic Co-Clustering. In *Proceedings of The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD-2003), Washington, DC, USA, August 24 - 27, 2003* (pp.89-98).

Dickie, H. F. (1951). ABC Inventory Analysis Shoots for Dollars Not Pennies. *Factory Management and Maintenance*, *109*(7), 92-94.

Dodd, S. C. (1940). The Interrelation Matrix. *Sociometry*, *3*(1), 91-101.

Domingos, P. (2007). Structured Machine Learning: Ten Problems for the Next Ten Years. In *Proceedings of Seventeenth International Conference on Inductive Logic Programming, Corvallis, OR, USA, June 19-21, 2007* (pp.1-4).

Dunham, M. H. (2003). *Data Mining: Introductory and Advanced Topics*. Upper Saddle River, NJ: Prentice Hall.

Dunnell, R. C. (1970). Seriation Method and Its Evaluation. *American Antiquity*, *35*(3), 305-319.

Eisen, M. B. , Spellman, P. T. , Brown, P. O. , & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences USA*, *95*(25), 14863-14868.

Elisseeff, V. (1965). Possibilités du scalogramme dans l'étude des bronzes chinois arhaiques. *Mathématiques et sciences humaines*, *11*, 1-10.

Faloutsos, C. , & Megalooikonomou, V. (2007). On data mining, compression, and Kolmogorov. *Data Mining and Knowledge Discovery*, *15*(1), 3-20.

Flores, B. E. , & Whybark, D. (1986). Multiple Criteria ABC Analysis. *International Journal of Operations and Production Management*, *6*(3), 38-46.

Flores, B. E. , & Whybark, D. (1987). Implementing multiple criteria ABC analysis. *Journal of Operations Management*, *7*(1/2), 79-84.

Flores, B. E. , Olson, D. L. , & Whybark, D. (1992). Management of Multicriteria Inventory Classification. *Mathematical and Computer Modelling*, *16*(12), 71-82.

Forsyth, E. , & Katz, L. (1946). A Matrix Approach to the Analysis of Sociometric Data: Preliminary Report. *Sociometry*, *9*(4), 340-347.

Friendly, M. (2002). Corrgrams: Exploratory displays for correlation matrices. *The American Statistician*, *56*, 316-324.

Friendly, M. , & Kwan, E. (2003). Effect ordering for data displays. *Computational Statistics & Data Analysis*, *43*, 509-539.

Ganter, B. , & Wille, R. (1999). *Formal Concept Analysis. Mathematical Foundations*. Berlin: Springer.

Goncalves, J. F. , & Resende, M. G. C. (2004). An evolutionary algorithm for manufacturing cell formation. *Computers and Industrial Engineering, 47*(2-3), 247-273.

Grünwald, P. D. (2007). *The Minimum Description Length Principle*. Cambridge, MA: MIT Press.

Gunasingh, R. K. , & Lashkari, R. S. (1989). Machine grouping problem in cellular manufacturing systems – an integer programming approach. *International Journal of Production Research, 27*(9), 1465-1473.

Güvenir, H. A. (1995). A Genetic Algorithm for Multicriteria Inventory Classification. In *Artificial Neural Nets and Genetic Algorithms, Proceedings of the International Conference, Ales, France, April 18-21, 1998* (pp.6-9).

Güvenir, H. A. , & Erel, E. (1998). Multicriteria Inventory Classification using a Genetic Algorithm. *European Journal of Operational Research, 105*(1), 29-37.

Hammer, Ø. (2008). *PAST - PAlaeontological STatistics*. Retrieved June 1, 2008, from http://folk.uio.no/ohammer/past/

Hammer, Ø. , & Harper, D. (2005). *Paleontological Data Analysis*. Oxford: Blackwell Publishing.

Haralick, R. M. (1974). The Diclique Representation and Decomposition of Binary Relations. *Journal of the ACM, 21*(3), 356-366.

Hartigan, J. A. (1972). Direct Clustering of a Data Matrix. *Journal of the American Statistical Association, 67*(337), 123-129.

Hartigan, J. A. (1975). *Clustering Algorithms*. New York: John Wiley and Sons.

Hodson, F. , Kendall, D. G. , & Tautu, P. (Eds.) (1971). *Mathematics in the Archaeological and Historical Sciences*. Edinburgh: Edinburgh University Press.

Hole, F. , & Shaw, M. (1967). *Computer Analysis of Chronological Seriation*. Houston: Rice University Studies.

Hubert, L. J. (1974). Problems of seriation using a subject by item response matrix. *Psychological Bulletin, 81*(12), 976-983.

Hubert, L. J. (1976). Seriation using asymmetric proximity measures. *The British Journal of Mathematical and Statistical Psychology, 29*, 32-52.

Hubert, L. J. , Arabie, P. , & Meulman, J. (2001). *Combinatorial data analysis: Optimization by dynamic programming*. Philadelphia: Society for Industrial and Applied Mathematics.

Hutter, M. (2007). Algorithmic information theory. *Scholarpedia, 2*(3), 2519-2519.

Hwang, H. , & Sun, J. U. (1996). A genetic-algorithm-based heuristic for the GT cell formation problem. *Computers and Industrial Engineering, 30*(4), 941-955.

Ihm, P. (2005). A contribution to the history of seriation in archaeology. In *Classification - the Ubiquitous Challenge, Proceedings of the 28th Annual Conference of the Gesellschaft für Klassifikation e.V., University of Dortmund, March 9–11, 2004* (pp.307-316).

Jennings, H. (1937). Structure of Leadership-Development and Sphere of Influence. *Sociometry, 1*(1/2), 99-143.

Kandiller, L. (1994). A Comparative study of cell formation in cellular manufacturing systems. *International Journal of Production Research, 32*(10), 2395-2429.

Kaparthi, S. , & Suresh, N. C. (1992). Machine-component cell formation in group technology: A neural network approach. *International Journal of Production Research, 30*(6), 1353-1367.

Katz, L. (1947). On the Matric Analysis of Sociometric Data. *Sociometry, 10*(3), 233-241.

Kendall, D. G. (1969a). Some Problems and Methods in Statistical Archaeology. *World Archaeology, 1*(1), 68-76.

Kendall, D. G. (1969b). Incidence matrices, interval graphs and seriation in archaeology. *Pacific Journal of Mathematics, 28*(3), 565-570.

Kendall, D. G. (1971). Seriation from abundance matrices. In F. Hodson, D. G. Kendall, & P. Tautu (Eds.) *Mathematics in the Archaeological and Historical Sciences,* Chicago: Aldine-Atherton, pp.214-252.

Keogh, E. , Lonardi, S. , & Ratanamahatana, C. A. (2004). Towards Parameter-Free Data Mining. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, Seattle, WA, USA, August 22-25, 2004* (pp.206-215).

King, J. (1980). Machine-component group formation in production flow analysis: an approach using a rank order clustering algorithm. *International Journal of Production Research, 18*(2), 213-232.

King, J. , & Nakornchai, V. (1982). Machine-component group formation in group technology: Review and extension. *International Journal of Production Research*, *20*(2), 117-133.

Kulczynski, S. (1927). Die pflanzenassoziation der pieninen. *Bull. Internat. Acad. Polon. Sci. Lett., Classe Sci. Math. et Nat., Serie B Sci.Nat., Suppl.*, *2*, 57-203.

Kumar, K. R. , & Chandrasekharan, M. P. (1990). Group efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *International Journal of Production Research*, *28*(2), 233-243.

Kumar, K. R. , & Vannelli, A. (1987). Strategic subcontracting for efficient disaggregated manufacturing. *International Journal of Production Research*, *25*(12), 1715-1728.

Kumar, K. R. , Kusiak, A. , & Vannelli, A. (1986). Grouping of Parts and Components in Flexible Manufacturing Systems. *European Journal of Operational Research*, *24*, 387-397.

Kusiak, A. (1991). Branching algorithms for solving the group technology problem. *Journal of Manufacturing Systems*, *10*(4), 332-343.

Kusiak, A. , & Cheng, C. H. (1990). A branch-and-bound algorithm for solving the group technology problem. *Annual of Operations Research*, *26*, 415-431.

Kusiak, A. , & Cho, M. (1992). Similarity coefficient algorithms for solving the group technology problem. *International Journal of Production Research*, *30*(11), 2633-2646.

Kusiak, A. , & Chow, W. S. (1987). Efficient solving of the group technology problem. *Journal of Manufacturing Systems*, *6*(2), 117-124.

Kusiak, A. , & Chung, Y. K. (1991). GT/ART: using neural network to form machine cells. *Manufacturing Review*, *4*(4), 293-301.

Kusiak, A. , Boe, J. W. , & Cheng, C. H. (1993). Designing cellular manufacturing systems: branch-and-bound and A* approaches. *IIE Transactions*, *25*(4), 46-56.

Kuzara, R. S. , Mead, G. R. , & Dixon, K. A. (1966). Seriation of Anthropological Data: A Computer Program for Matrix-Ordering. *American Anthropologist*, *68*(6), 1442-1455.

Legendre, P. , & Legendre, L. (1998). *Numerical ecology*. Amsterdam: Elsevier.

Lehmer, D. J. (1951). Robinson's Coefficient of Agreement. A Critique. *American Antiquity*, *17*(2), 151-151.

Lei, Q. , Chen, J. , & Zhou, Q. (2005). Multiple Criteria Inventory Classification Based on Principal Components Analysis and Neural Network. *Lecture Notes in Computer Science*, *3498*, 1058-1063.

Lenstra, J. K. (1974). Clustering a Data Array and the Traveling-Salesman Problem. *Operations Research*, *22*(2), 413-414.

Li, M. , & Vitanyi, P. (1997). *An introduction to Kolmogorov complexity and its applications*. London: Springer Verlag.

Liiv, I. (2006). Inventory classification enhancement with demand associations. In *Proceedings of The 2006 IEEE International Conference on Service Operations and Logistics, and Informatics, Shanghai, China, June 21-23, 2006* (pp.18-22).

Liiv, I. (2007a). Czekanowski-Bertin Learning Paradigm: A Discussion. In *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, Las Vegas, NV, USA, June 25-28, 2007* (pp.90-95).

Liiv, I. (2007b). Visualization and data mining method for inventory classification. In *Proceedings of The 2007 IEEE International Conference on Service Operations and Logistics, and Informatics, Philadelphia, USA, August 27-29, 2007* (pp.472-477).

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007a). Conformity analysis with structured query language. In *Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, Corfu Island, Greece, February 16-19, 2007* (pp.187-189).

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007b). Analytical CRM with conformity analysis. *Transactions on Systems and Control*, *2*(2), 155-161.

Liiv, I. , Vedeshin, A. , & Täks, E. (2007c). Visualization and structure analysis of legislative acts: a case study on the law of obligations. In *The 11th ACM International Conference on Artificial Intelligence and Law (ICAIL), Proceedings of the Conference, Stanford University, CA, USA, June 4-8, 2007* (pp.189-190).

Lyman, R. L. , O'Brien, M. J. , & Dunnell, R. C. (1997). *The Rise and Fall of Culture History*. New York: Plenum Press.

Lyman, R. L. , Wolverton, S. , & O'Brien, M. J. (1998). Seriation, Superposition, and Interdigitation: A History of Americanist Graphic Depictions of Culture Change. *American Antiquity*, *63*(2), 239-261.

Madeira, S. C. , & Oliveira, A. L. (2004). Biclustering Algorithms for Biological Data Analysis: A Survey. *IEEE Transactions on Computational Biology and Bioinformatics*, *1*(1), 24-45.

Marcotorchino, F. (1987). Block seriation problems: A unified approach. Reply to the problem of H. Garcia and J. M. Proth (Applied Stochastic Models and Data Analysis, 1, (1), 25-34 (1985)). *Applied Stochastic Models and Data Analysis, 3*(2), 73-91.

Marcotorchino, F. (1991). Seriation problems: An overview. *Applied Stochastic Models and Data Analysis, 7*(2), 139-151.

Marquardt, W. H. (1978). Advances in Archaeological Seriation. In M. B. Schiffer (Ed.) *Advances in Archaeological Method and Theory*, New York: Academic Press, pp.257-314.

McAuley, J. (1972). Machine grouping for efficient production. *Production Engineer, 51*(2), 53-57.

McCormick, W. T. , Deutsch, S. B. , Martin, J. J. , & Schweitzer, P. J. (1969). *Identification of Data Structures and Relationships by Matrix Reordering Techniques (TR P-512)*. Arlington, Virginia: Institute for Defense Analyses.

McCormick, W. T. , Schweitzer, P. J. , & White, T. W. (1972). Problem Decomposition and Data Reorganization by a Clustering Technique. *Operations Research, 20*(5), 993-1009.

Miklos, I. , Somodi, I. , & Podani, J. (2005). Rearrangement of Ecological Data Matrices Via Markov Chain Monte Carlo Simulation. *Ecology, 86*(12), 3398-3410.

Miltenburg, J. , & Zhang, W. (1991). A comparative evaluation of nine well-known algorithms for solving the cell formation problem in group technology. *Journal of Operations Management*, *10*(1), 44-72.

Mirkin, B. G. (1996). *Mathematical Classification and Clustering (Nonconvex Optimization and Its Applications)*. Boston-Dordrecht: Kluwer Academic Press.

Mirkin, B. G. , & Muchnik, I. (1996). Clustering and multidimensional scaling in Russia (1960-1990): A review. In P. Arabie, L. J. Hubert, & G. De Soete (Eds.) *Clustering and Classification*, River Edge, NJ: World Scientific, pp.295-339.

Mirkin, B. G. , & Rodin, S. N. (1984). *Graphs and Genes*. Berlin: Springer-Verlag.

Mitrofanov, S. (1959). *Nauchnye osnovy gruppovoj tehnologii (in Russian)*. Leningrad: Lenizdat.

Mitrofanov, S. (1966). *The Scientific Principles of Group Technology*. London: National Lending Library for Science and Technology.

Moody, J. (2001). Peer influence groups: identifying dense clusters in large networks. *Social Networks*, *23*, 261-283.

Moreno, J. L. (1934). *Who Shall Survive? A New Approach to the Problem of Human Inter-relations*. New York: Beacon House.

Moreno, J. L. (1946). Sociogram and Sociomatrix: A Note to the Paper by Forsyth and Katz. *Sociometry, 9*(4), 348-349.

Moreno, J. L. (1953). *Who shall survive? Foundations of sociometry, group psychotherapy and sociodrama*. New York: Beacon House.

Mosier, C. T. , & Taube, L. (1985a). The facets of group technology and their impact on implementation. *Omega*, *13*(6), 381-391.

Mosier, C. T. , & Taube, L. (1985b). Weighted similarity measure heuristics for the group technology machine clustering problem. *Omega*, *13*(6), 577-583.

Mosier, C. T. , Yelle, J. , & Walker, G. (1997). Survey of Similarity Coefficient Based Methods as Applied to the Group Technology Configuration Problem. *Omega, International Journal of Management Science*, *25*(1), 65-79.

Mullat, J. E. (1976a). Extremal Subsystems of Monotonic Systems I. *Automation and Remote Control*, *37,* 758-766.

Mullat, J. E. (1976b). Extremal Subsystems of Monotonic Systems II. *Automation and Remote Control*, *37,* 1286-1294.

Mullat, J. E. (1977). Extremal Subsystems of Monotonic Systems III. *Automation and Remote Control*, *38*, 89-96.

Murtagh, F. (1989). Book Review: W. Gaul and M. Schader, Eds., Data, Expert Knowledge and Decisions, Heidelberg: Springer-Verlag, 1988, pp. viii + 380. *Journal of Classification*, *6*, 129-132.

Mäkinen, E. , & Siirtola, H. (2000). Reordering the Reorderable Matrix as an Algorithmic Problem. In *Proceedings of The Theory and Application of Diagrams: First International Conference, Diagrams 2000, Edinburgh, Scotland, UK, September 1-3, 2000* (pp.453-467).

Mäkinen, E. , & Siirtola, H. (2005). The barycenter heuristic and the reorderable matrix. *Informatica*, *29*(3), 357-363.

Niermann, S. (2005). Optimizing the Ordering of Tables With Evolutionary Computation. *The American Statistician*, *59*(1), 41-46.

O'Brien, M. J. , & Lyman, R. L. (1999). *Seriation, Stratigraphy, and Index Fossils: The Backbone of Archaeological Dating*. New York: Kluwer Academic/Plenum Publishers.

Onwubolo, G. C. , & Mutingi, M. (2001). A genetic algorithm approach to cellular manufacturing systems. *Computers and Industrial Engineering, 39*, 125-144.

Pareto, V. (1971). *Manual of Political Economy (English translation)*. New York: A.M. Kelley Publishers.

Park, Y.-T., & Wemmerlöv, U. (1994). A shop structure generator for cell formation research. *International Journal of Production Research, 32*(10), 2345-2360.

Partovi, F. Y. , & Anandarajan, M. (2002). Classifying inventory using an artificial neural network approach. *Computers & Industrial Engineering, 41*, 389-404.

Pasquier, N. , Bastide, Y. , Taouil, R. , & Lakhal, L. (1999). Discovering Frequent Closed Itemsets for Association Rules. In *Proceedings of the 7th International Conference on Database Theory (ICDT99), Jerusalem, Israel, January 10-12, 1999* (pp.398-416).

Petrie, W. M. F. (1899). Sequences in Prehistoric Remains. *The Journal of the Anthropological Institute of Great Britain and Ireland*, *29*(3/4), 295-301.

Pluta, W. (1980). *Sravnitel'nyi mnogomernyi analiz v ekonomicheskih issledovanijah (in Russian)*. Moskva: Statistika.

Prelic, A. , Bleuler, S. , Zimmermann, P. , Wille, A. , Bühlmann, P. , Gruissem, W. , Hennig, L. , Thiele, L. , & Zitzler, E. (2006). A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics*, *22*(9), 1122-1129.

Rajagopalan, R. , & Batra, J. L. (1975). Design of cellular production systems - a graph theoretic approach. *International Journal of Production Research, 13*(6), 567-579.

Ramanathan, R. (2006). ABC inventory classification with multiple-criteria using weighted linear optimization. *Computers & Operations Research, 33*(3), 695-700.

Rao, H. A. , & Gu, P. (1993). Design of cellular manufacturing systems: A neural network approach. *International Journal of Systems Automation: Research and Applications, 2*(4), 407-424.

Rao, R. , & Card, S. K. (1994). The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information. In *Proceedings of the ACM SIGCHI, Boston, MA, United States, April 24 - 28, 1994* (pp.318-322).

Rissanen, J. (1978). Modelling by shortest data description. *Automatica, 14*, 465-471.

Robinson, W. (1951). A Method for Chronologically Ordering Archaeological Deposits. *American Antiquity, 16*(4), 293-301.

Rowe, J. H. (1961). Stratigraphy and Seriation. *American Antiquity, 26*(3), 324-330.

Rust, R. T. , Lemon, K. N. , & Zeithaml, V. (2004). Return on Marketing: Using Customer Equity to Focus Marketing Strategy. *Journal of Marketing, 68*(1), 109-127.

Rust, R. T. , Zeithaml, V. , & Lemon, K. N. (2000). *Driving customer equity: how customer lifetime value is reshaping corporate strategy*. New York: The Free Press.

Saaty, T. L. (1977). A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology, 15*(3), 234-281.

Saaty, T. L. (1980). *The Analytical Hierarchy Process*. New York: McGraw-Hill.

Seifoddini, H. (1989). Single linkage versus average linkage clustering in machine cells formation applications. *Computers and Industrial Engineering, 16*(3), 419-426.

Seifoddini, H. , & Wolfe, P. M. (1986). Application of the similarity coefficient method in group technology. *IIE Transactions, 18*(3), 271-277.

Shneiderman, B. (2002). Inventing Discovery Tools: Combining Information Visualization with Data Mining. *Information Visualization, 1*(1), 5-12.

Siirtola, H. (1999). Interaction with the Reorderable Matrix. In *Proceedings of the International Conference on Information Visualization (IV'99), London, UK, July 14-16, 1999* (pp.272-277).

Siirtola, H. (2003). Combining Parallel Coordinates with the Reorderable Matrix. In *Proceedings of the conference on Coordinated and Multiple Views In Exploratory Visualization, London, UK, July 16-18, 2003* (pp.63-74).

Siirtola, H. (2004). Interactive Cluster Analysis. In *Proceedings of the 8th International Conference on Information Visualisation (IV'04), London, UK, July 14-16, 2004* (pp.471-476).

Siirtola, H. , & Mäkinen, E. (2005). Constructing and reconstructing the reorderable matrix. *Information Visualization, 4*(1), 32-48.

Singh, N. (1993). Design of cellular manufacturing systems: An invited review. *European Journal of Operational Research, 69*, 284-291.

Skrzywan, W. (1952). Metoda grupowania na podstawie tablic prof. Czekanowskiego. *Przeglàd Antropologiczny, 18*, 583-599.

Sloane, N. J. A. (2003). An On-Line Version of the Encyclopedia of Integer Sequences. *Notices of the American Mathematical Society*, *50*, 912-915.

Sokal, R. R. , & Sneath, P. H. A. (1963). *Principles of numerical taxonomy*. San Francisco: W. H. Freeman.

Soltysiak, A. , & Jaskulski, P. (1999). Czekanowski's Diagram. A Method of Multidimensional Clustering. In *New Techniques for Old Times. CAA 98. Computer Applications and Quantitative Methods in Archaeology. Proceedings of the 26th Conference, Barcelona, Spain, March 25-28, 1998* (pp.175-184).

Spaulding, A. C. (1971). Some elements of quantitative archaeology. In F. Hodson, D. G. Kendall, & P. Tautu (Eds.) *Mathematics in the Archaeological and Historical Sciences*, Chicago: Aldine-Atherton, pp.3-16.

Späth, H. (1980). *Cluster Analysis Algorithms for Data Reduction and Classication of Objects*. Chichester, UK: Ellis Horwood.

Srinivasan, G. , Narendran, T. T. , & Mahadevan, B. (1990). An assignment model for part-families problem in group technology. *International Journal of Production Research*, *28*(1), 145-152.

Stanfel, L. E. (1985). Machine clustering for economic production. *Engineering Costs and Production Economics*, *9*, 73-81.

Szczotka, F. A. (1972). On a method of ordering and clustering of objects. *Zastosowania Mathemetyki*, *13*, 23-34.

Tanay, A. , Sharan, R. , & Shamir, R. (2006). Biclustering algorithms: a survey. In S. Aluru (Ed.) *Handbook of Computation Molecular Biology*, Boca Raton, FL: CRC Press, pp.26:1-26:17.

Tucker, L. R. (1964). The extension of factor analysis to three-dimensional matrices. In H. Gulliksen, & N. Frederiksen (Eds.) *Contributions to mathematical psychology*, New York: Holt, Rinehart and Winston, pp.110-127.

Tufte, E. (1983). *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press.

Tukey, J. W. (1977). *Exploratory Data Analysis*. Reading, MA: Addison-Wesley.

Tyugu, E. (2007). *Algorithms and Architectures of Artificial Intelligence*. Amsterdam: IOS Press.

Uno, T. , Kiyomi, M. , & Arimura, H. (2004). LCM ver. 2: Efficient Mining Algorithms for Frequent/Closed/Maximal Itemsets. In *FIMI '04, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining*

Implementations, Brighton, UK, November 1, 2004. Retrieved June 1, 2008, from http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-126/uno.pdf

Van Mechelen, I. , Bock, H.-H., & De Boeck, P. (2004). Two-mode clustering methods: a structured overview. *Statistical Methods in Medical Research*, *13,* 363-394.

Vandev, D. L. , & Tsvetanova, Y. G. (1995). Perfect chains and single linkage clustering algorithm. In *Proceedings of Statistical Data Analysis (SDA-95), Varna, Bulgaria, September 23–28, 1995* (pp.99-107).

Vandev, D. L. , & Tsvetanova, Y. G. (1997). Ordering of Hierarchical Classifications. In *Colloquy on Mathematical Modelling in the fields of Food Technology*, Berlin: Humbold University, pp.111-122.

Venugopal, V. , & Narendran, T. T. (1992). Cell formation in manufacturing systems through simulated annealing: an experimental evaluation. *European Journal of Operational Research*, *63*, 409-422.

Verin, L. L. , & Grishin, V. G. (1986). Algorithm for interactive forming matrix data representation and estimation of its efficiency. *Pattern Recognition Letters*, *4*, 193-200.

Vyhandu, L. (1979). Rapid Data Analysis Methods (in Russian). *Transactions of Tallinn University of Technology*, *464*, 21-39.

Vyhandu, L. (1980). Some Methods to Order Objects and Variables in Data Systems (in Russian). *Transactions of Tallinn University of Technology*, *482*, 43-50.

Vyhandu, L. (1981). Fast methods for data processing (in Russian). In *Computer Systems: Computer methods for revealing regularities*, Novosibirsk: Institute of Mathematics Press, pp.20-29.

Vyhandu, L. (1989). Fast Methods in Exploratory Data Analysis. *Transactions of Tallinn University of Technology*, *705*, 3-13.

Waghodekar, P. H. , & Sahu, S. (1984). Machine-component cell formation in group technology: MACE. *International Journal of Production Research*, *22*(6), 937-948.

Wasserman, S. , & Faust, K. (1994). *Social network analysis*. Cambridge: Cambridge University Press.

Wemmerlöv, U. , & Hyer, N. L. (1987). Research issues in cellular manufacturing. *International Journal of Production Research*, *25*(3), 413-431.

Wemmerlöv, U. , & Hyer, N. L. (1989). Cellular manufacturing in the U.S. industry: a survey of users. *International Journal of Production Research*, *27*(9), 1511-1530.

White, H. , Boorman, S. A. , & Breiger, R. (1976). Social structure from multiple networks. I. Blockmodels of roles and positions. *American Journal of Sociology*, *81*, 730-790.

Wille, R. (1992). Concept lattices and conceptual knowledge systems. *Computers & Mathematics with Applications*, *23*, 493-515.

Zaki, M. J. , & Hsiao, C.-J.(2002). CHARM: An Efficient Algorithm for Closed Itemset Mining. In *Proceedings of the Second SIAM International Conference on Data Mining, Arlington,VA, USA,April 11-13, 2002*. Retrieved June 1, 2008, from http://www.siam.org/meetings/sdm02/proceedings/sdm02-27.pdf

Zhang, R. Q. , Hopp, W. J. , & Supatgiat, C. (2001). Spreadsheet Implementable Inventory Control for a Distribution Center. *Journal of Heuristics*, *7*, 185-203.

Zimmerman, G. W. (1975). The ABC's of Vilfredo Pareto. *Production and Inventory Management*, *16*(3), 1-9.

Mustrite avastamine kasutades järjestamist ning maatriksi ümberkorrastamist: unifitseeritud vaade, edasiarendused ning rakendus ladude juhtimises

Lühikokkuvõte

Järjestamine (ingl.k. *seriation*) on avastuslik kombinatoorne andmeanalüüs objektide seadmiseks järjendisse mööda ühemõõtmelist kontinuumi selliselt, et see paljastaks kogu seerias maksimaalselt regulaarsust ja mustreid. Käesolevas töös pakutakse välja unifitseeritud vaade ja sihifunktsioon parameetrite vabale järjestamisele, kasutades Kolmogorovi keerukust ning andmete pakkimist.

Suunamata õppimine kasutades järjestamist ning maatriksi ümberkorrastamist võimaldab mustreid avastada korraga kolmel informatsiooni tasemel: fragmendid kohalikest seostest, kogus omavahel seostatud fragmente kohalikest seostest ning üldine struktuurne muster. Selline lähenemine ühendab oma tulemustes ja parendab populaarsete suunamata andmekaevandamise meetodite, nagu klasterdamine ja assotsiatsioonireeglite leidmine, struktuurse analüüsi võimeid. Töö väidab, et järjestamine peaks olema asetatud samale tasemele nimetatud standardsete andmekaevandamise meetoditega viimaste nõrkuse tõttu analüüsida keerukamaid struktuure ning detailidest eemale fokuseerida globaalsetele mustritele.

Järjestamine on paraku traditsioonilistest andmekaevandamise tehnikatest arvutuslikult palju kallim ettevõtmine ning ei skaleeru rahuldavalt sarnastes stsenaariumites. Selle olukorra lahendamiseks on välja pakutud uus algoritm hõredate binaarsete maatriksite järjestamiseks. Algoritm pakub sobiva kompromissi arvutusliku keerukuse ning otseste seoste pealiskaudse loendamise vahel. Peale selle on välja pakutud uus lähenemine järjestamise koheseks rakendamiseks andmebaasis SQL-keele standardse relatsioonilise algebra ja arvutuse abil ilma väliste protseduuride ja funktsioonideta.

Töös esitatakse käsitletud lähenemise rakendus ladude juhtimises, et pakkuda lahendus hetkel puuduvale funktsionaalsusele laojuhtimise tarkvarades kaupade ümberklassifitseerimiseks ning prioritiseerimiseks vastavalt seostele klientide käitumises. Seega, peale väljapakutud algoritmi omaduste kontrollimise ja hindamise, pakutakse välja ka uudne lahendus ladude klassifitseerimiseks kasutades andmekaevandamist ning järjestamist.

Võtmesõnad: järjestamine, kahemõõtmeline klasterdamine, kombinatoorne andmeanalüüs, minimaalse kirjelduse pikkuse printsiip, informatsiooni visualiseerimine, andmekaevandamine.

Publications by the author

Liiv, I. (2006). Inventory classification enhancement with demand associations. In *Proceedings of The 2006 IEEE International Conference on Service Operations and Logistics, and Informatics, Shanghai, China, June 21-23, 2006* (pp.18-22).

Liiv, I. (2007a). Czekanowski-Bertin Learning Paradigm: A Discussion. In *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, Las Vegas, NV, USA, June 25-28, 2007* (pp.90-95).

Liiv, I. (2007b). Visualization and data mining method for inventory classification. In *Proceedings of The 2007 IEEE International Conference on Service Operations and Logistics, and Informatics, Philadelphia, USA, August 27-29, 2007* (pp.472-477).

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007a). Conformity analysis with structured query language. In *Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, Corfu Island, Greece, February 16-19, 2007* (pp.187-189).

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007b). Analytical CRM with conformity analysis. *Transactions on Systems and Control*, *2*(2), 155-161.

Liiv, I. , Vedeshin, A. , & Täks, E. (2007c). Visualization and structure analysis of legislative acts: a case study on the law of obligations. In *The 11th ACM International Conference on Artificial Intelligence and Law (ICAIL), Proceedings of the Conference, Stanford University, CA, USA, June 4-8, 2007* (pp.189-190).

Curriculum Vitae (in Estonian)

1. Isikuandmed
   Ees- ja perekonnanimi: Innar Liiv
   Sünniaeg ja -koht: 26.03.1982, Tallinn
   Kodakondsus: Eesti

2. Kontaktandmed
   Aadress: Raja 15, 12618 Tallinn
   Telefon: +3725200552
   E-posti aadress: innar.liiv@ttu.ee

3. Hariduskäik

Õppeasutus (nimetus lõpetamise ajal)	Lõpetamise aeg	Haridus (eriala/kraad)
Tallinna Tehnikaülikool	2004	M.Sc. (informaatika)
Tallinna Tehnikaülikool	2004	B.Sc. (informaatika)

4. Keelteoskus (alg-, kesk- või kõrgtase)

Keel	Tase
Eesti keel	Kõrgtase
Inglise keel	kõrgtase
Vene keel	Algtase
Soome keel	Algtase
Prantsuse keel	Algtase

5. Teenistuskäik

Töötamise aeg	Tööandja nimetus	Ametikoht
September 2005 - ...	Tallinna Tehnikaülikool	assistent
September 2004 – September 2005	Tallinna Tehnikaülikool	erakorraline assistent
September 2003 – Juuni 2004	Tallinna Tehnikaülikool	tunnitasuline õppejõud
September 2005 - ...	AS Koolibri	arendusjuht
November 2002 - Märts 2005	Vertical Tarkvara OÜ	juhatuse esimees

6. Teadustegevus

Kuusik, R., Liiv, I, & Lind, G. (2005). An Efficient Method for Post Analysis of Patterns. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria, February 14-16, 2005* (pp.101–105).

Liiv, I. (2005). Mining and Visualizing Power in Social Network Analysis. In *Proceedings of the 2005 International Conference on Artificial Intelligence ICAI 2005, Las Vegas, NV , June 27-30, 2005* (pp.754 - 759).

Liiv, I. (2006). Inventory classification enhancement with demand associations. In *Proceedings of The 2006 IEEE International Conference on Service Operations and Logistics, and Informatics, Shanghai, China, June 21-23, 2006* (pp.18-22).

Aps, R., Kell, L. T., Lassen, H., & Liiv, I. (2007). Negotiation framework for baltic fisheries management: striking the balance of interest. *ICES Journal of Marine Science, 64*(4), 858–861.

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007). Conformity analysis with structured query language. In *Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, Corfu Island, Greece, February 16-19, 2007* (pp.187-189).

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007). Analytical CRM with conformity analysis. *Transactions on Systems and Control, 2*(2), 155-161.

Liiv, I. , Vedeshin, A. , & Täks, E. (2007). Visualization and structure analysis of legislative acts: a case study on the law of obligations. In *The 11th ACM International Conference on Artificial Intelligence and Law (ICAIL), Proceedings of the Conference, Stanford University, CA, USA, June 4-8, 2007* (pp.189-190).

Liiv, I. (2007). Czekanowski-Bertin Learning Paradigm: A Discussion. In *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, Las Vegas, NV, USA, June 25-28, 2007* (pp.90-95).

Kirt, T., Liiv, I., & Vainik, E. (2007). Self-organizing map, matrix reordering and multidimensional scaling as alternative and complementary methods in a semantic study. In *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, Las Vegas, NV, USA, June 25-28, 2007* (pp. 385–390).

Liiv, I. (2007). Visualization and data mining method for inventory classification. In *Proceedings of The 2007 IEEE International Conference on Service Operations and Logistics, and Informatics, Philadelphia, USA, August 27-29, 2007* (pp.472-477).

7. Kaitstud lõputööd

"Assotsiatsioonireeglite leidmine kasutades monotoonsete süsteemide teooriat"

M.Sc. (Informaatika), Tallinna Tehnikaülikool, 2004

B.Sc. (Informaatika), Tallinna Tehnikaülikool, 2004

8. Teadustöö põhisuunad

andmeanalüüs, andmekaevandamine, prognoosmudelid, operatsioonianalüüs, visualiseerimine, sotsiaalvõrgustike analüüs, tehisintellekt

9. Teised uurimisprojektid

OILECO projekti teadur ("Ökoloogiliste väärtuste integreerimine õlilekke otsustusprotsessi Soome lahes"; http://hykotka.helsinki.fi/oileco/)

Curriculum Vitae

1. Personal data

   Name: Innar Liiv
   Date and place of birth: 26 March 1982, Tallinn
   Citizenship: Estonian

2. Contact information

   Address: 15 Raja Street, Tallinn 12618
   Phone: +3725200552
   E-mail: innar.liiv@ttu.ee

3. Education

Educational institution	Graduation year	Education (field of study/degree)
Tallinn University of Technology	2004	M.Sc. (Informatics)
Tallinn University of Technology	2004	B.Sc. (Informatics)

4. Language skills (basic, intermediate or high level)

Language	Level
Estonian	High level
English	High level
Russian	Basic
Finnish	Basic
French	Basic

5. Professional Employment

Period	Organisation	Position
September 2005 - ...	Tallinn University of Technology	Research and Teaching Assistant
September 2004 – September 2005	Tallinn University of Technology	Research and Teaching Assistant (extraordinary)
September 2003 – June 2004	Tallinn University of Technology	Teaching Assistant (contractual)
September 2005 - ...	Koolibri Publishers Ltd	Manager of Corporate Development
November 2002 - March 2005	Vertical Software Ltd	CEO

6. Scientific work

Kuusik, R., Liiv, I, & Lind, G. (2005). An Efficient Method for Post Analysis of Patterns. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria, February 14-16, 2005* (pp.101–105).

Liiv, I. (2005). Mining and Visualizing Power in Social Network Analysis. In *Proceedings of the 2005 International Conference on Artificial Intelligence ICAI 2005, Las Vegas, NV , June 27-30, 2005* (pp.754 - 759).

Liiv, I. (2006). Inventory classification enhancement with demand associations. In *Proceedings of The 2006 IEEE International Conference on Service Operations and Logistics, and Informatics, Shanghai, China, June 21-23, 2006* (pp.18-22).

Aps, R., Kell, L. T., Lassen, H., & Liiv, I. (2007). Negotiation framework for baltic fisheries management: striking the balance of interest. *ICES Journal of Marine Science, 64*(4), 858–861.

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007). Conformity analysis with structured query language. In *Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, Corfu Island, Greece, February 16-19, 2007* (pp.187-189).

Liiv, I. , Kuusik, R. , & Vyhandu, L. (2007). Analytical CRM with conformity analysis. *Transactions on Systems and Control, 2*(2), 155-161.

Liiv, I. , Vedeshin, A. , & Täks, E. (2007). Visualization and structure analysis of legislative acts: a case study on the law of obligations. In *The 11th ACM International Conference on Artificial Intelligence and Law (ICAIL), Proceedings of the Conference, Stanford University, CA, USA, June 4-8, 2007* (pp.189-190).

Liiv, I. (2007). Czekanowski-Bertin Learning Paradigm: A Discussion. In *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, Las Vegas, NV, USA, June 25-28, 2007* (pp.90-95).

Kirt, T., Liiv, I., & Vainik, E. (2007). Self-organizing map, matrix reordering and multidimensional scaling as alternative and complementary methods in a semantic study. In *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, Las Vegas, NV, USA, June 25-28, 2007* (pp. 385–390).

Liiv, I. (2007). Visualization and data mining method for inventory classification. In *Proceedings of The 2007 IEEE International Conference on Service Operations and Logistics, and Informatics, Philadelphia, USA, August 27-29, 2007* (pp.472-477).

7. Defended theses

"Assotsiatsioonireeglite leidmine kasutades monotoonsete süsteemide teooriat"

("Mining Association Rules Using the Theory of Monotone Systems")

M.Sc. (Informatics), Tallinn University of Technology, 2004

B.Sc. (Informatics), Tallinn University of Technology, 2004

8. Research Interests

data analysis, data mining, prediction models (churn, fraud, lifetime value), operations research, information visualization, social network analysis, artificial intelligence

9. Other research projects

Researcher in OILECO project ("Integrating ecological values in the decision making process on oil spill combating in the Gulf of Finland"; http://hykotka.helsinki.fi/oileco/)