THESIS ON INFORMATICS AND SYSTEM ENGINEERING C50

Formal Concepts in the Theory of Monotone Systems

ANTS TORIM

TALLINN UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Informatics

Dissertation was accepted for the defence of Doctor of Philosophy in Engineering on November 6, 2009.

Supervisor:	Professor Dr. Rein Kuusik Department of Informatics Tallinn University of Technology
Opponents:	Professor Dr. Sergei O. Kuznetsov Department of Applied Mathematics State University Higher School of Economics, Russia
	Professor Dr. Peeter Normak Department of Informatics Tallinn University, Estonia

Defence of the thesis: December 7, 2009

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.

/Ants Torim/

ISSN 1406-4731
ISBN 978-9985-59-951-8

INFORMAATIKA JA SÜSTEEMITEHNIKA C50

Formaalsed mõisted monotoonsete süsteemide teooria käsitluses

ANTS TORIM

Formal Concepts in the Theory of Monotone Systems

Abstract

Formal concept analysis and the theory of monotone systems are both well-established methods for data mining and knowledge discovery whose connections have so far not been well researched. This thesis explores such connections and proposes some new methods for knowledge discovery that combine the features of both approaches.

Formal concept analysis is based on the idea of a formal concept, that is characterized by its extent, a set of objects, and by its intent, a set of attributes these objects have in common. There is a mathematical relation between intent and an extent, requiring both of them to be locally maximal in certain sense.

Theory of monotone systems is based on the monotone weight functions and is often used for seriation: reordering of a data table to reveal the hidden structure. We show that such structure is a set of formal concepts.

This thesis is organized around the case study about the socio-economic data of the islands of Saaremaa and Hiiumaa. The case study had a goal of revealing the patterns of economic and social development.

We propose a conformity plot visualization method, that gave good results for our case study and redefinition of the problem of finding the best decision as the problem of finding the best formal concept chain and the discovery of its symmetry in regards to objects and attributes and propose some enhancements to the algorithm for finding the best concept chain.

Monotone systems methods and formal concept analysis have some difficulty in dealing with larger data tables. We propose a MONOCLE method for data mining and knowledge discovery that aims to remedy these difficulties. The result of a MONOCLE method is a list of formal concepts, sorted according to their importance.

We also propose a method for comparing these results for different contexts, like Hiiumaa and Saaremaa: entropy based similarity measure for evaluating the uniqueness of a formal concept.

Keywords: data mining, knowledge discovery, information visualization, monotone systems, formal concept analysis.

Acknowledgements

I would like to thank my supervisor, professor Rein Kuusik, for his great help and patience, professor emeritus Leo Võhandu for many discussions and good ideas, Karin Lindroos for fruitful collaboration, Innar Liiv for his help with the topic of seriation and professor Åke Sivertun from Linköping University for the data about dengue fever.

Contents

List of Figures

List of Tables

List of Theorems

Chapter 1

Introduction

The world we live in is complex, chaotic and noisy. The task of science is detecting hidden order and patterns behind all that chaos and noise. Sometimes these patterns take the form of absolute laws, sometimes they take the form of statistical and probabilistic rules and correlations. Sometimes the result is useful terminology. This thesis presents some automated methods for knowledge discovery whose results can form a basis for such a terminology as well as being a basis for probabilistic rules. Some methods presented here were invented by the author, some were not. They are influenced by the theories of formal concept analysis (FCA) [45] and monotone systems [26].

1.1 Theoretical background

FCA, as a field of study, began in 1982 with the article by Rudolf Wille [43]. It has mathematical foundations in lattice theory and the notion of a formal concept is based on the ideas from linguistics, semantics and philosophy. The formal concept is characterized by its extent - a set of objects - and by its intent - a set of attributes. All objects in concepts extent possess all the attributes in its intent and vice versa. In the terminology of semantics, extent and intent correspond to the extension and intension of a concept. To quote Wikipedia's article about extension: "...in linguistics, logic, mathematics, semantics, and semiotics, the extension of a concept, idea, or sign consists of the things to which it applies, in contrast with its comprehension or intension, which consists very roughly of the ideas, properties, or corresponding signs that are implied or suggested by the concept in question.". This symmetry between the set of attributes and the set of objects is a peculiar feature of a formal concept, in contrast to the clusters from cluster analysis, where we have clear set of objects but not a clear set of common attributes. Objects, attributes and relations between them in FCA form a formal context. Informally, it could be called a binary data table. Binary data tables are the objects of study in this thesis. This is not as restrictive as it may seem as any multi-valued data table can be transformed into a binary form, with a likely loss of some precision when it comes to real-valued attributes.

Another field of research that is concerned with the knowledge discovery from data tables (and sometimes from structures of a different nature) is the theory of monotone systems. It was developed somewhat earlier than FCA, in Tallinn University of Technology by Mullat in his 1976 article [26] and later by Võhandu and others. The theory of monotone systems is based on monotone weight functions, defined for elements of the monotone system, and different clever orderings based on those weight functions. A very common use for this theory is seriation - reordering of rows and columns in the data table to show the hidden patterns and structure. As we will see in the later chapters, these patterns correspond to formal concepts.

1.2 Case study

Big part of this thesis is based on the fruitful collaboration between the author and Karin Lindroos [23], [14], [35]. The collaboration started as an attempt to apply monotone system methods to the socio-economic data about the island of Hiiumaa and later its southern neighbor island of Saaremaa. To resolve the question of authorship: data was provided by Karin Lindroos, methods and tools were provided by the author, interpretation is mixed and sometimes joint work and Karins contribution here will be marked by references. Interpretation in the Chapter 5 is sole work of the author. Idea to apply the methods of FCA came as a result of some weaknesses of monotone systems methods that became apparent during that analysis: results of monotone systems methods required lots of interpretation without offering guidance for that task.

1.3 Research aims

Research aims of this thesis are following:

- To outline the connections between FCA and the theory of monotone systems.
- To present, for the first time, the theory of a monotone systems in the form compatible with the terminology of FCA.
- To develop new data mining methods based on such an unification and to improve some old ones.
- To apply those methods, along with more traditional ones, for our case study about Hiiumaa and Saaremaa.

These research aims are motivated by the obvious connections between FCA and the theory of monotone systems and by the fact that these two methods seem to complement each other well. Our case study provides an excellent test platform for those methods.

1.4 Organization of the thesis

In Chapter 2 we will describe the FCA and the theory of Monotone Systems. We will examine the implicit presence of the notion of formal concept in the theory of monotone systems and describe the theory of monotone systems in the terms of FCA. Both approaches have some weaknesses when dealing with large amounts of data - complex concept lattices in the case of FCA and complex, though rearranged, data table in the case of monotone systems approach.

Chapter 3 describes the application of theory of monotone systems to our case study. We propose the conformity plot visualization that allows us to visualize the clusters in data, according to two monotone weight function. We restate the monotone systems problem of finding the best decision in FCA terms as the problem of finding the best concept chain and describe some speed enhancements to algorithm for finding it.

In Chapter 4 we will present our MONOCLE method that combines both FCA and monotone systems and aims to mitigate their weaknesses by sorting formal concepts according to their monotone weight function (concepts "importance"), thus presenting information in the easily understandable form. Our aim is to find the concepts that cover a large area of the formal context and that do not overlap too much with each other. This is one of the main contributions of the thesis. We illustrate this method through case studies.

In Chapter 5 we introduce a measure for concepts uniqueness for one context as compared to another context. This measure allows for comparisons between different contexts, for example the comparison of the island of Hiiumaa with the island of Saaremaa. This can mean contexts with same attributes and different objects or contexts with same objects and different attributes. The aim of such a measure is to make the interpretation of the results of MONOCLE method easier, clearer and more objective. The measure combines ideas from FCA, monotone systems and information entropy.

The thesis contains some formal definitions, theorems and proofs, but mathematics used here is not very advanced. Basic knowledge of discrete mathematics should be sufficient for understanding it.

1.5 Publications

Many results of this thesis have been published in international peer-reviewed journals. The problem and ehnhanced algorithm for finding the best concept chain from

Chapter 3 are described in “WSEAS Transactions on Information Science and applications” from year 2005 [34] and also mentioned in a more general article about monotone systems in the same publication from year 2006 [36]. The conformity plot visualization method from Chapter 3 is described in “Proceedings of IADIS International Conference on Applied Computing” from year 2006 [14]. The MONOCLE method from Chapter 4 is described in year 2008 Springer’s “Lecture Notes in Computer Science; Conceptual Structures: Knowledge Visualization and Reasoning” [35]. This article was awarded Tallinn University of Technology’s best article of a year prize in the field of social sciences. The results in Chapter 5 - measuring concepts importance for the different contexts - are latest and not yet published.

Chapter 2

Formal Concept Analysis and the Theory of Monotone Systems, State of the Art

2.1 Formal Concept Analysis

2.1.1 Introduction to FCA

Here we provide an introduction into formal concept analysis (FCA). A detailed exposition is given in “Formal Concept Analysis, Mathematical foundation” by Ganter and Wille [11] or “Formal Concept Analysis: Foundations and Applications” by Wille, Stumme and Ganter [45]. For the following definitions we use “Introduction to Lattices and Order” by Davey and Priestley [9].

Formal concept analysis (FCA) is a way of automatically deriving an ontology from a collection of objects and their properties. It was introduced by Rudolf Wille in year 1982 [43]. FCA has many applications in linguistics [28], text retrieval and mining [7], association rule mining [19], economics [44], software engineering [33] and so on.

In linguistics or philosophy, concept is characterized by its extension - the set of objects that the concept applies to - and by its intension - all attributes relevant to concept, its definition. Formal concept is defined in the formal context (informally, a binary data table) and is characterized correspondingly by its extent and intent that must be related in a certain way. Formal definitions follow.

Definition 1. *A **context** is a triple (G, M, I) where G and M are sets and* $I \subseteq G \times M$. *The elements of G and M are called objects and attributes respectively.*

We can say less formally that a context is a binary data table. Relation $(g, m) \in I$ means that object g has true value for an attribute m. That is element of a binary data table, determined by row g and column m has value 1. We use a shorthand gIm for this relation.

Definition 2. *For $A \subseteq G$ and $B \subseteq M$, define*

$$A' = \{m \in M \mid (\forall g \in A), gIm\} \ , \tag{2.1}$$

$$B' = \{g \in G \mid (\forall m \in B), gIm\} \ ; \tag{2.2}$$

so A' is the set of attributes common to all the objects in A and B' is the set of objects possessing the attributes in B.

Definition 3. *A **formal concept** is any pair (A, B) where $A \subseteq G$ and $B \subseteq M$, $A' = B$ and $B' = A$. The **extent** of the concept (A, B) is A while its **intent** is B.*

We can say less formally that a concept is a set of objects together with the attributes these objects have in common under the restriction that we cannot add an additional attribute without removing an object and we cannot add an additional object without removing an attribute. The special concept $\top$ has the extent G and the special concept $\bot$ has the intent M.

Algorithms for efficiently generating concepts from the context are described in [11], [45] and [9].

2.1.2 Concept lattices

Subset relations $A_1 \subseteq A_2$ and $B_2 \subseteq B_1$ define an order on the set of all formal concepts and it can be shown [9] that they form a complete lattice, known as the **concept lattice** of the context. Concept lattices are commonly visualized as line diagrams[1] where concepts are shown as nodes, and subset relations between their extents (and inverse subset relations between their intents) are shown by lines. More general concepts are drawn above less general concepts. The ordered set of formal concepts of (G, M, I) is denoted by $\mathfrak{B}(G, M, I)$.

Let us consider the example from Figure 2.1 which describes sizes of various watercourses. Object set G and attribute set M are abbreviated as follows: $G = \{$**C**hannel, **B**rook, **S**tream, **R**iver$\}$, $M = \{$**v**ery **s**mall, **s**mall, **l**arge, **v**ery **l**arge$\}$. The set of concepts for the context $\mathfrak{B}(G, M, I)$ is $\{x_1, x_2, x_3, x_4\}$ where $x_1 = (\{C, B\}, \{s\})$, $x_2 = (\{S, R\}, \{l\})$, $x_3 = (\{C\}, \{vs, s\})$ and $x_4 = (\{R\}, \{l, vl\})$. The corresponding concept lattice is then drawn. As the extent of x_1 contains that of x_3 and the extent of x_2 contains that of x_4 these concepts are connected with lines and the more general concepts are placed higher in the diagram.

It has been shown that concept lattices are isomorphic to complete lattices: every concept lattice is complete and every complete lattice is isomorphic to some concept

[1] Here we used GaLicia Platform [1], [37], also known as Hasse diagrams, for the generation of complex concept lattice diagrams.

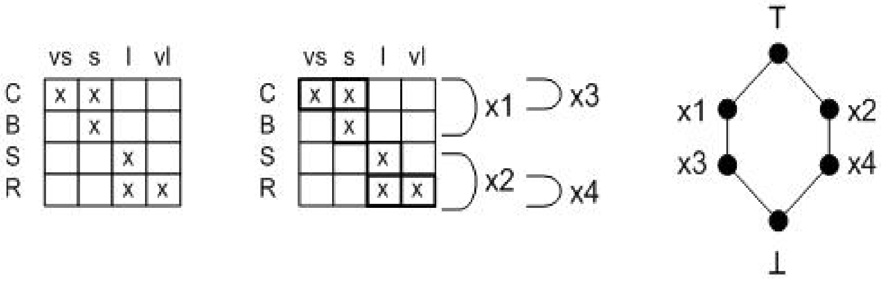


Figure 2.1: A context as a binary data table, same context with the concepts marked inside the table by borders and labeled outside the table by their extents and the corresponding concept lattice. Taken from Davey and Priestley [9]. There is no requirement that attributes and objects in the concepts should be adjacent, we use such data tables only for the ease of illustration.

lattice [12]. This result is known as the basic theorem on concept lattices. This result connects FCA to the lattice theory. Lattice theory in general is not covered in this thesis, interested reader is referred to the following books: Introduction to Lattices and Order by Davey and Priestley [9] and General Lattice Theory by Grätzer [13].

Let $H \subseteq P$ and $a \in P$. Then a is an upper bound of H if and only if $h \leq a$, for all $h \in H$. An upper bound a of H is the supremum of H, denoted as $a = \bigvee H$, if and only if for any upper bound b of H, we have $a \leq b$ [13]. Infimum $a = \bigwedge H$ is defined dually [13].

Let $Q \subseteq P$. Then Q is called supremum-dense (join-dense) in P if for every element $a \in P$ there is a subset A of Q such that $a = \bigvee A$. The dual of supremum-dense is infimum dense (meet dense) [9].

Theorem 1 (The basic theorem on concept lattices [12])**.** *The concept lattice $\mathfrak{B}(G, M, I)$ is a complete lattice in which infimum and supremum are given by:*

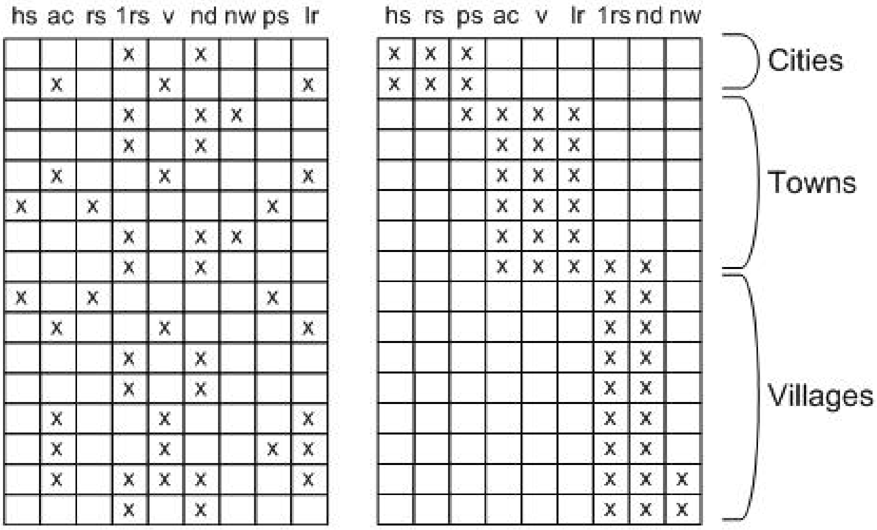
$$\bigwedge_{t \in T} \langle A_t, B_t \rangle = \left(\bigcap_{t \in T} A_t, \left(\bigcup_{t \in T} B_t \right)'' \right), \tag{2.3}$$

$$\bigvee_{t \in T} \langle A_t, B_t \rangle = \left(\left(\bigcup_{t \in T} A_t \right)'', \bigcap_{t \in T} B_t \right). \tag{2.4}$$

A complete lattice L is isomorphic to $\mathfrak{B}(G, M, I)$ if and only if there are maps $\tilde{\gamma} : G \to L$ and $\tilde{\mu} : M \to L$ such that $\tilde{\gamma}(G)$ is supremum-dense on L, $\tilde{\mu}(M)$ is infimum-dense in L and gIm is equivalent to $\tilde{\gamma}g \leq \tilde{\mu}m$ for all $g \in G$ and all $m \in M$. In particular $L \cong \mathfrak{B}(L, L, \leq)$.

2.1.3 Managing complexity

Not all formal concepts are equal. Some concepts have large extents and/or intents, others have small. Example in Figure 2.2 is taken from the work of French cartographer Jaques Bertin [5], and while not originally connected to FCA, the 3 concepts identified by Bertin are clearly formal concepts. This example is also closely connected to the theory of monotone systems as it is an example of seriation [21], a field where monotone system methods are widely used and it connects well to our case study about settlements in two islands. Attribute set M is abbreviated as follows: $M = \{$**h**igh **s**chool, **r**ailway **s**tation, **p**olice **s**tation, **a**gricultural **c**ooperative, **v**eterinary, **l**and **r**eallocation, **1** **r**oom **s**chool, **n**o **d**octor, **n**o **w**ater supply $\}$.



hs	ac	rs	1rs	v	nd	nw	ps	lr
			x		x			
	x			x				x
			x		x	x		
			x		x			
	x			x				x
x		x					x	
			x		x	x		
			x		x			
x		x					x	
	x			x				x
			x		x			
			x		x			
	x			x				x
	x			x			x	x
	x		x	x	x			x
			x		x			

hs	rs	ps	ac	v	lr	1rs	nd	nw
x	x	x						
x	x	x						
		x	x	x	x			
			x	x	x			
			x	x	x			
			x	x	x			
			x	x	x			
			x	x	x	x	x	
						x	x	
						x	x	
						x	x	
						x	x	
						x	x	
						x	x	
						x	x	x
						x	x	x

Cities
Towns
Villages

Figure 2.2: A context before and after reordering (seriation) and three important concepts (villages, towns , cities).

Concept lattices can become large for quite a small contexts. For example, a 488 × 234 sparse binary data table with economic data about settlements in Estonian island Saaremaa contained 1823 concepts. It is obvious that such a number of concepts is too large for the unaided human analysis. Several methods try to mitigate that problem. A full comparative review could be a topic for another article, here we give only a short review.

Blocks [11] introduce additional ones into the binary data table, generating bigger and fewer concepts. Our method sorts original concepts, without modifying them.

Nested line diagrams [11] summarize parallel lines and display them as just one line. Inner nodes contain sub-lattices. No concepts are removed, however, number of lines in the lattice is reduced.

Software tools such as **TOSCANA** [4] aim to manage complexity by allowing user to navigate through complex data sets with the help of nested line diagrams.

Iceberg view, described by Stumme et al. [32], is based on selecting only the concepts that have extent of certain minimum size k, that is, cover at least k objects. Connecting this method with our theme, it can be described as sorting concepts by size of their extent and selecting those above some suitable cut-off point. Size of extent is intuitive and easy-to-calculate weight function. It does, however, eliminate concepts with few objects and many attributes. For some types of data, in our case economic data of settlements, these concepts are of great importance as they represent, for example, important regional centers. Our method takes into account both extent and intent sizes. But before describing our method, we need to give some background into the theory of monotone systems.

One measure for concepts goodness is **stability index**, proposed by Kuznetsov [18]. Stability measures independence of hypotheses on particular pieces of data that can be random, similar to the way scientific hypotheses are evaluated. Informally, stability index for the formal concept (A, B) is correlated to the number of subsets of its extent $Y \subset A$ that leave its intent unchanged, that is $Y' = B$.

Closure operators, described for example by Bělohlávek and Vychodil [6], represent a class of operators that constrain the lattice; retained concepts are guaranteed to form a complete lattice. Iceberg view method belongs into this general class.

Role minimization problem and efficient solutions to it are described in the article by Ene, Horne, Milosavljevic, Rao, Schreiber and Tarjan [10]. Article is not originally framed in the FCA terms, however it is easy to relate the set of users to the set of objects G, the set of permissions to the set of attributes M and relations between them to set I. Role minimization problem of finding the minimal set of roles that will cover all relations then becomes the problem of finding the minimal set of formal concepts that covers all the relations in formal context.

2.2 Theory of Monotone Systems

2.2.1 Introduction to the Theory of Monotone Systems

The theory of monotone systems was developed in Tallinn University of Technology and introduced in 1976 in the article by Mullat [26]. Most of work about monotone systems has been carried out in Tallinn University of Technology, though there have been outside contributions like the work by Muchnik and Kuznetsov [17]. Monotone systems have been used in many application areas, this thesis tries to present he most central methods and some new enhancements.

A monotone system is a set of elements and a weight function. The weight function measures which elements are important for the system. Here we present the theory of monotone systems in a way compatible with the language of FCA. This presentation is one of the contributions of the thesis. The notation of FCA seems quite convenient for this task.

Definition 4. *A **monotone system** is a pair (W, w) where W is a finite set of elements, w(x, H) is a weight for element $x \in H$ for any $H \subseteq W$ and the co domain of w is a linearly ordered set. Following property of monotonicity should hold for all $x \in H$ and for all $y \in H$ where $x \neq y$:*

$$w(x, H) \geq w(x, H \setminus \{y\})\ . \tag{2.5}$$

That is, weights of the elements should decrease monotonically if any one element is removed from the system. There is a dual definition for monotonically increasing weights and a more general case where the removal of an element is replaced by an "operation" but for this article, these are not needed. Different weight functions and monotone systems algorithms are described in [36].

We want to measure the weight or "goodness" of subsystems of W. We use the weakest link principle and define the function F_{min} as:

$$F_{min}(H) = min\big(\{w(x, H) \mid \forall x \in H\}\big)\ . \tag{2.6}$$

We call the subsystems with the greatest value of F_{min} **kernels**.

Definition 5. *A subsystem $K \subseteq W$ is called the kernel of the system W if $F_{min}(K) \geq F_{min}(H)$ for any $H \subseteq W$.*

Minus technique means removing an element with the smallest weight from the monotone system and repeating this step until the system is empty. A minus technique sequence can therefore be found by a greedy algorithm, see [36]. Formal definition follows:

Definition 6. *We denote n-th element from the minus technique sequence for the system W by x_n. Let $H_1 = W$ and $H_n = (...((W \setminus \{x_1\}) \setminus \{x_2\})... \setminus \{x_{n-1}\})$.*

$$x_n = x \in H_n \text{ where } w(x, H_n) \leq w(y, H_n) \text{ for all } y \in H_n \tag{2.7}$$

The minus technique sorts the elements by their worth for the system. If we want to eliminate the k least interesting elements from the system we can apply minus technique and deal only with the set H_{k+1}. Thus we can use the minus technique to substitute arbitrary sized subset for the entire system. We can also use the kernels to suggest us good cut-off points. The following theorem deals with the relationship between the kernels and the minus technique.

Theorem 2 (Kernel as the global maximum)**.** [2] *Let $w(x_k, H_k) = F_{min}(H_k)$ be the maximal weight in the minus technique sequence $x_1, x_2, .., x_n$ for the monotone system W. That is,*

$$F_{min}(H_k) \geq F_{min}(H_i) \text{ for all } i \in \{1...n\} \ . \tag{2.8}$$

Then the subsystem H_k is a kernel for the system W.

Proof. For all $A \subseteq H_1$ where $x_1 \in A$ we know that $F_{min}(A) \leq F_{min}(H_1)$ because of the property of monotonicity from the Equation 2.5. Therefore, either H_1 is a kernel or there is some kernel $K \subseteq H_2$.

If we know that $K \subseteq H_i$ for $i \in \{1...n\}$ then for all $A \subseteq H_i$ where $x_i \in A$ we know that $F_{min}(A) \leq F_{min}(H_i)$. Therefore, either H_i is the kernel K, or $K \subseteq H_{i+1}$.

By induction, there is some kernel $K \in \{H_1, H_2, ..., H_k, ..., H_n\}$. As $w(x_k, H_k) = F_{min}(H_k)$ is the maximal weight in the minus technique sequence, H_k is the kernel for the system W. □

The kernel as the global maximum provides a good cut-off point in the minus technique sequence. For practical purposes we often want more cut-off points to study either smaller or larger subsystems. Therefore we will also introduce the notion of **local kernels** that correspond to local maxima in the minus technique sequence.

Definition 7. *Let sequence $H_1, H_2, ..., H_n$ be the sequence of subsets corresponding to the minus technique sequence $x_1, x_2, .., x_n$. Then $H_k \in \{H_1, H_2, ..., H_n\}$ is a local kernel if $F_{min}(H_{k-1}) \leq F_{min}(H_k) \geq F_{min}(H_{k+1})$.*

Figure 2.3 shows an example of simple graph-based monotone system before and after the removal of an element.

[2] This theorem was proven independently by A. Torim. Equivalent theorem, albeit with a longer proof, appeared in earlier work by Mullat [26].

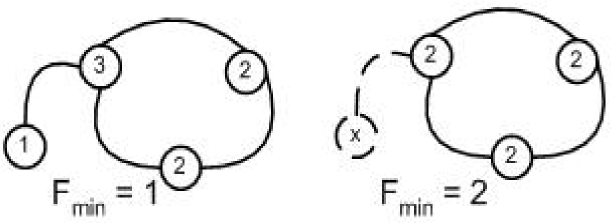
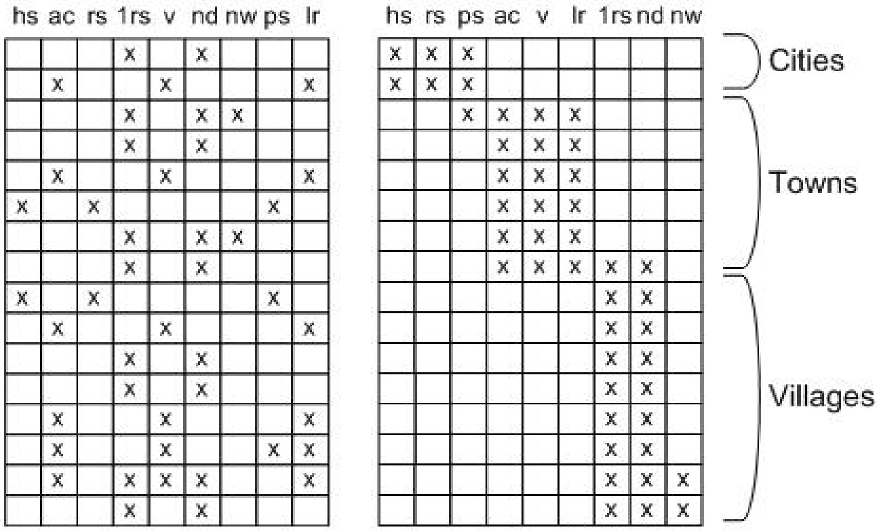


Figure 2.3: A monotone system whose elements are vertices of the graph and the weight for the element is the number of adjacent vertices. Weights are shown inside the vertex circles. In this example, after removal of the element with the smallest weight, we have the kernel.

2.2.2 Seriation

Methods of monotone systems are most often used for the purpose of seriation - reordering and arranging objects and attributes in the data table (or formal context) to reveal the regularity and patterning. Seriation has been reinvented in many different fields, a good overview of seriation is given in the PhD thesis by Innar Liiv [21]. A good example of seriation is depicted in Figure 2.4 by Jaques Bertin (1981) [5]. After reordering of initial data table, previously hidden patterns are revealed. We can also see that those hidden patterns - villages, towns, cities - are basically formal concepts. Attributes are abbreviated as follows:

- hs: high school
- ac: agricultural coop.
- rs: railway station
- 1rs: one-room school
- v: veterinary
- nd: no doctor
- nw: no water supply
- ps: police station
- lr: land reallocation.



hs	ac	rs	1rs	v	nd	nw	ps	lr
			x		x			
	x			x				x
			x		x	x		
			x		x			
	x			x				x
x		x					x	
			x		x	x		
			x		x			
x		x					x	
	x			x				x
			x		x			
			x		x			
	x			x				x
	x			x			x	x
	x		x	x	x			x
			x		x			

hs	rs	ps	ac	v	lr	1rs	nd	nw
x	x	x						
x	x	x						
		x	x	x	x			
			x	x	x			
			x	x	x			
			x	x	x			
			x	x	x			
			x	x	x	x	x	
						x	x	
						x	x	
						x	x	
						x	x	
						x	x	
						x	x	
						x	x	x
						x	x	x

Cities
Towns
Villages

Figure 2.4: A context before and after reordering (seriation) and three important concepts (villages, towns , cities).

Use of seriation goes back to the works by an English egyptologist W. M. F. Petrie (1899) [27] and Polish antrophologist Jan Czekanowski (1909) [8]. Figure 2.5 illustrates Czekanowski's work. We can see the results of seriation that seem to correspond to two large proto-concepts.

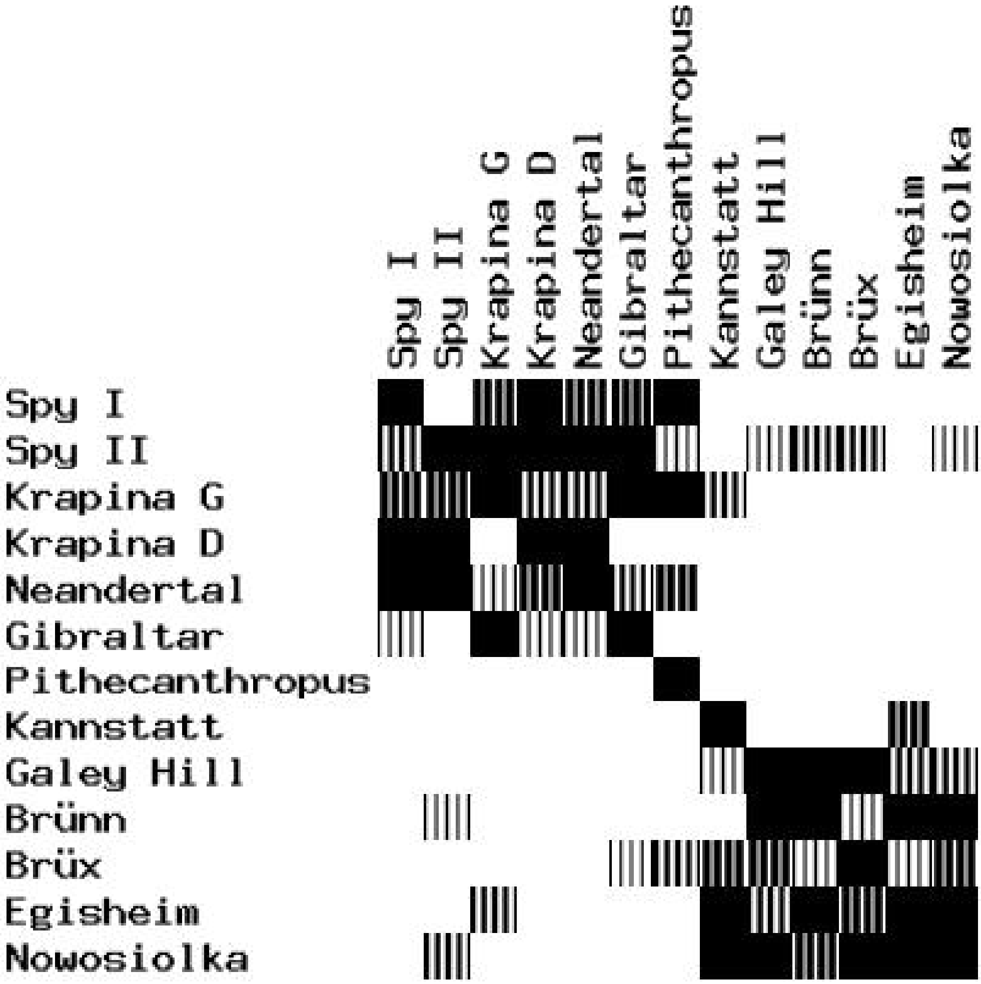


Figure 2.5: Czekanowski's [8] diagram of differences and groups of skulls. Taken from Liiv's PhD thesis [21]

How can we use the theory of monotone systems for seriation? For a formal context we can define a frequency based weight function for objects or dually for attributes that gives higher weight for an object with common attributes or an attribute with common objects as follows:

Definition 8. *For a context (G, M, I) we define a frequency based weight function $w_1(x, G)$ for any object $x \in G$:*

$$w_1(x, G) = \sum_{g \in G} \big|\{g\}' \cap \{x\}'\big|. \tag{2.9}$$

We define a weight function $w(y, M)$ for any attribute $y \in M$ dually:

$$w_1(y, M) = \sum_{m \in M} \big|\{m\}' \cap \{y\}'\big|. \tag{2.10}$$

Frequency based weight function is illustrated by the Figure 2.6 where relevant intersections $|\{g\}' \cap \{x\}'|$ and $|\{m\}' \cap \{y\}'|$ are grayed. We find that $w(x, G) = 3$ and $w(y, M) = 10$ for those contexts. Informally, weight $w_1(x, G)$ measures how big an area of the context (G, M, I) is covered by the attributes of the object x.

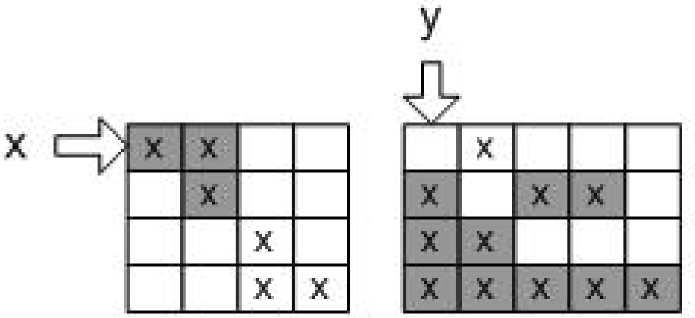
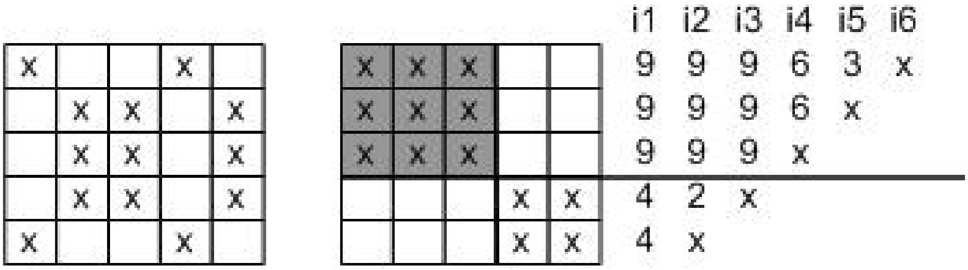


Figure 2.6: Weight calculation for object x and attribute y, context area that is summed in the weight function is gray.

Illustration of the seriation, using weight function from Definition 8 is given in Figure 2.7. It shows all the weights for all the iterations in a minus technique sequence. Weights for minus technique sequence are $4, 2, 9, 6, 3$ so objects corresponding to the weights $9, 6, 3$ form a kernel. As this context is symmetrical, weights for attributes are exactly the same as weights of objects. It is obvious how such a reordering brings out a regularity in the system and that this regularity, consisting of the 3x3 and 2x2 "‘squares"', is a set of two formal concepts.

Such a weight function is often generalized to the multivalued case, where one attribute can have several values (nominal scale) and their use for seriation are very common in the theory of monotone systems. Multivalued frequency based weight function w_* can be defined in terms of FCA by the use of conceptual scaling for multivalued contexts, where we create a new attribute for each attribute-value pair and apply the weight function w_1 for such single valued context to find the weights for objects. If we want to find the weights for attributes, then we need to create a new object for each object-value pair instead. Conceptual scaling in FCA is briefly explained in [12], with further references. Informally, weight $w_*(x, G)$ measures how typical object x is for the context (G, M, I) as it is the sum of frequencies of attribute values present in object x.

It is interesting to note that initial weights $9, 9, 9, 4, 4$, give here exact same ordering as the minus technique, with much less computational work. It does not hold in general, but often both orderings are reasonably close. Use of such a simple and fast ordering by a frequency based weight function is called a **conformity scale** and it was introduced by L. Võhandu in his works from 1979-1981 [38], [39], [40], [41].



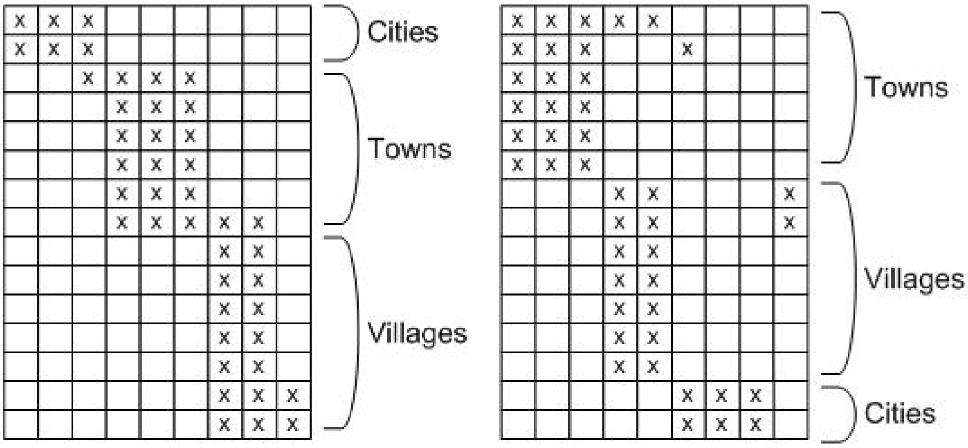
x			x	
	x	x		x
	x	x		x
	x	x		x
x			x	

					i1	i2	i3	i4	i5	i6
x	x	x			9	9	9	6	3	x
x	x	x			9	9	9	6	x	
x	x	x			9	9	9	x		
			x	x	4	2	x			
			x	x	4	x				

Figure 2.7: Seriation of 5x5 formal context. Objects and attributes of the formal context at the left are ordered randomly, at the right they are sorted according to the minus technique, using weight function from Definition 8. Weights of objects over the iterations $i_1, i_2, ..., i_6$ of the minus technique are shown, also 3x3 core is marked with lines and grayed relations.

If we apply minus tecnique with frequency based weight function to the objects and attributes from the Bertin plot from Figure 2.4 then we get an ordering shown in Figure 2.8. Main concepts are intact but logical order Cities-Towns-Villages has broken down.

McCormicks Bond Energy Algorithm (BEA) from 1972 [25] is an example of a non monotone systems seriation method. Comprehensive study of different seriation methods is given in I. Liiv's PhD thesis [21].



x	x	x							Cities
x	x	x							
		x	x	x	x				Towns
			x	x	x				
			x	x	x				
			x	x	x				
			x	x	x				
			x	x	x	x	x		
						x	x		Villages
						x	x		
						x	x		
						x	x		
						x	x		
						x	x	x	
						x	x	x	

x	x	x	x	x					Towns
x	x	x			x				
x	x	x							
x	x	x							
x	x	x							
x	x	x							
			x	x				x	Villages
			x	x				x	
			x	x					
			x	x					
			x	x					
			x	x					
			x	x					
					x	x	x		Cities
					x	x	x		

Figure 2.8: Original Bertin plot and Bertin plot after monotone systems seriation.

Chapter 3

Additions to the Theory of Monotone Systems

3.1 Case study: Hiiumaa and Saaremaa

3.1.1 Overview

We now apply monotone systems methods to social and economic data of two largest Estonian islands: Saaremaa and Hiiumaa [1]. Some of these results are published in an article by author, Karin Lindroos and L. Võhandu [14] and in the PhD thesis by Karin Lindroos [23]. For Hiiumaa $|G| = 184$ and $|M| = 226$; for Saaremaa $|G| = 488$ and $|M| = 234$. The attribute sets are mostly similar, however some attributes are present for only one island, hence some differences. Our attributes divide roughly as follows:

- attributes describing the population
- attributes describing the economic activity of companies
- attributes describing the private investment
- other nominal attributes, presence of some activity or a thing.

Data in our table is binary. Most attributes are binary by nature like existence of port or school. Each numerical attribute was replaced by several attributes that represent an interval. For example number of children in a village is represented by four binary attributes children$<$ 10, children10-50, children50-100, children$>$ 100 as shown in Table 3.1. Alternative encoding that would be more suitable with

[1] Saaremaa is the largest island (2,673 km^2) belonging to Estonia, Hiiumaa is the second largest (989 km^2). They are located in the Baltic Sea. The capital of Saaremaa is Kuressaare, which has about 15,000 inhabitants; the whole island has about 40,000 inhabitants. The capital of Hiiumaa is Kärdla, which has about 3,700 inhabitants; the whole island has about 10,000 inhabitants.

MONOCLE method, described in Chapter 4, is shown in Table 3.2, however, it is not used here. Ones in data table represent presence of certain feature or value located within interval. Data tables are sparse- for Hiiumaa only 4.7 % of values are ones.

Settlement	children <10	children 10-50	children 50-100	children >100	presence of children	...
Kärdla town	0	0	0	1	1	...
Emmaste village	0	0	1	0	1	...
Haldi village	1	0	0	0	1	...
...	...	...	...	...	...	...

Table 3.1: Fragment of the input. The table fragment shows the number of children in settlement.

Settlement	children >0	children >10	children >50	children >100	...
Kärdla town	1	1	1	1	...
Emmaste village	1	1	1	0	...
Haldi village	1	0	0	0	...
...	...	...	...	...	...

Table 3.2: Fragment of the input with alternative encoding - more ones, more children. The table fragment shows the number of children in settlement.

We apply our frequency based weight functions w_1 and w_* to these data. In our previous discussion it was easy to visualize the results of seriation as our example contexts were small. For this case study, the sorted context can be quite complex and confusing, as illustrated by the Figure 3.1 that depicts the context for Hiiumaa after seriation by the weight function w_1 and minus technique. There are some visible "‘concepts"’ or "‘clusters"’, like the thick black bar on the left and continuous black line in the middle corresponding to some bigger settlement. However, there are also lot of dots (ones in the data table, relations in the formal context) that don’t seem to have such a clear pattern - there seem to be some horizontal sparse ribbons of these dots, interleaved by some totally empty ribbons. It is also hard to extract the semantic information from such a picture as it is impractical to have a full attribute and object information in a picture depicting a context of such size. We can not see what attributes and objects those clusters represent. Tool support may provide some help here, one such tool is described in article by Innar Liiv [22]. We also see that while seriation visualizes well some concepts, some other concepts are broken up by such a seriation. For example, we have one big settlement that is clearly visible as a long continuous line, but there are several other big settlements that are depicted as broken lines by this ordering.

As shown in the example from Figure 3.2, finding an ordering that arranges all the concepts into continuous blocks is impossible for quite a simple contexts. In that example there is no ordering of attributes that would join all the concepts A, B and C.

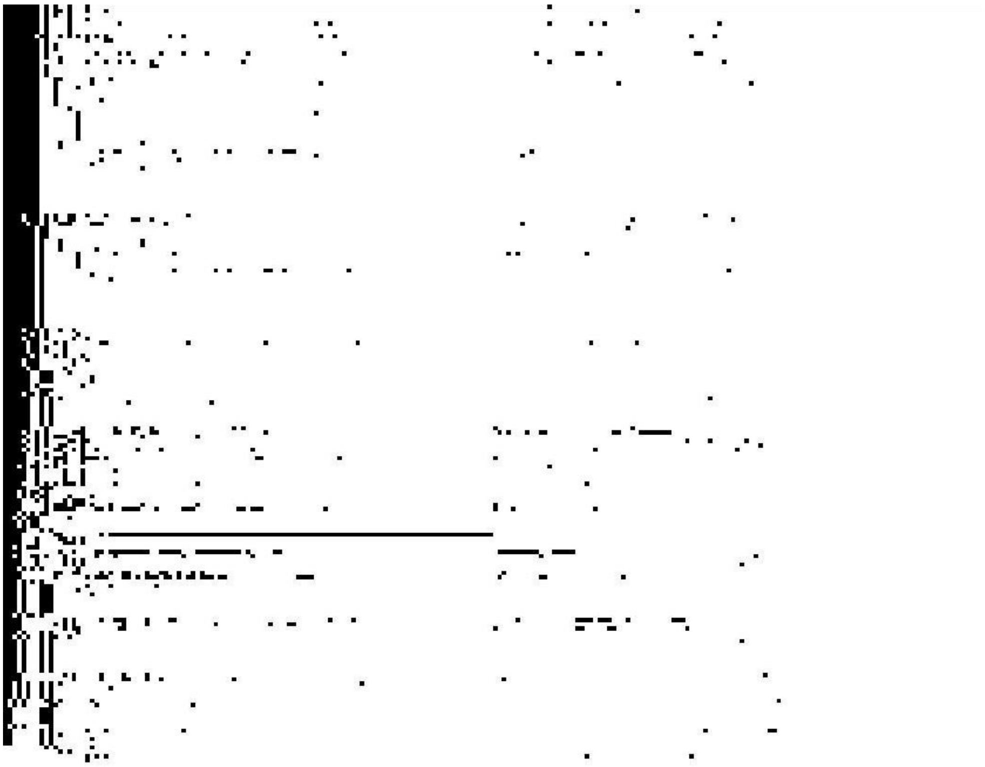


Figure 3.1: Sorted context for Hiiumaa, all attributes present, seriation by w_1 minus technique.

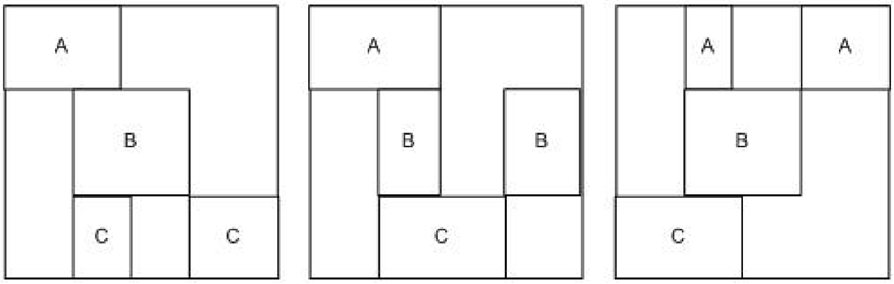


Figure 3.2: There is no seriation that would form continuous blocks for all three formal concepts A, B, C.

One common method to deal with bigger data tables is to plot the monotone weights as a graph, either sorted by the scale of conformism or as the weights $w(x_1)$, $w(x_2)$, ... , $w(x_n)$ in the minus technique sequence. Steep slopes in the scale of conformism plot and local maxima in the minus technique sequence plot provide a way to group the elements in the monotone system.

We will treat settlements - that is rows in the data table and objects in the formal context - as elements of the monotone system.

3.1.2 Scale of conformism

Figures 3.3, 3.4, 3.5 and 3.6 depict weights of objects for full contexts - scale of conformism. Objects are sorted according to their weights. Figures are given for combinations of Hiiumaa, Saaremaa and weight functions w_1, w_*.

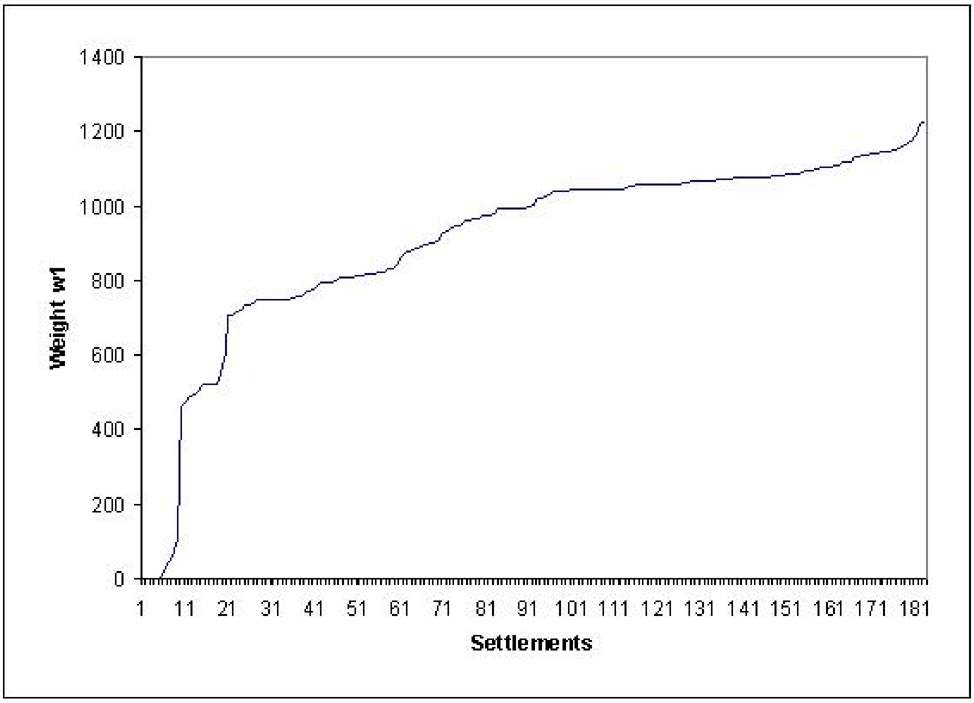


Figure 3.3: Hiiumaa, scale of conformism, w_1.

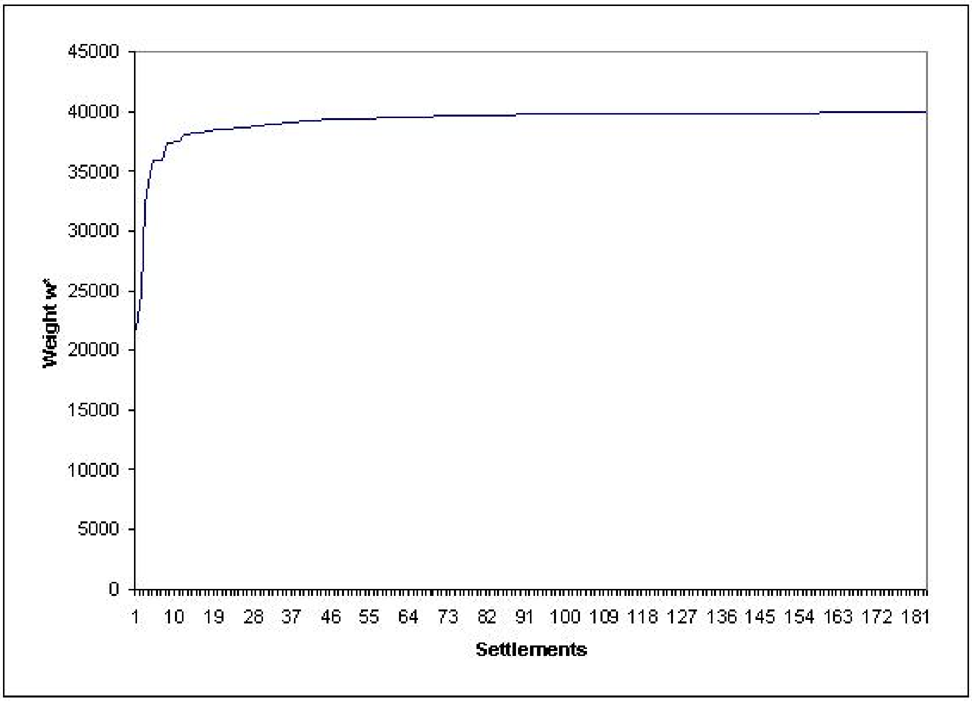


Figure 3.4: Hiiumaa, scale of conformism, w_*.

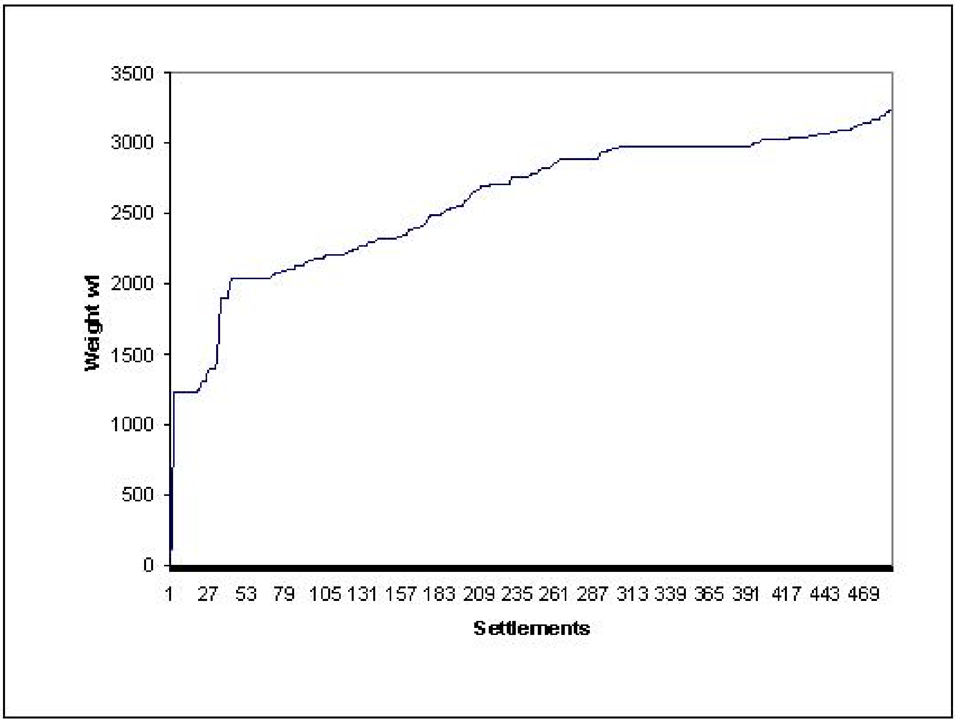


Figure 3.5: Saaremaa, scale of conformism, w_1.

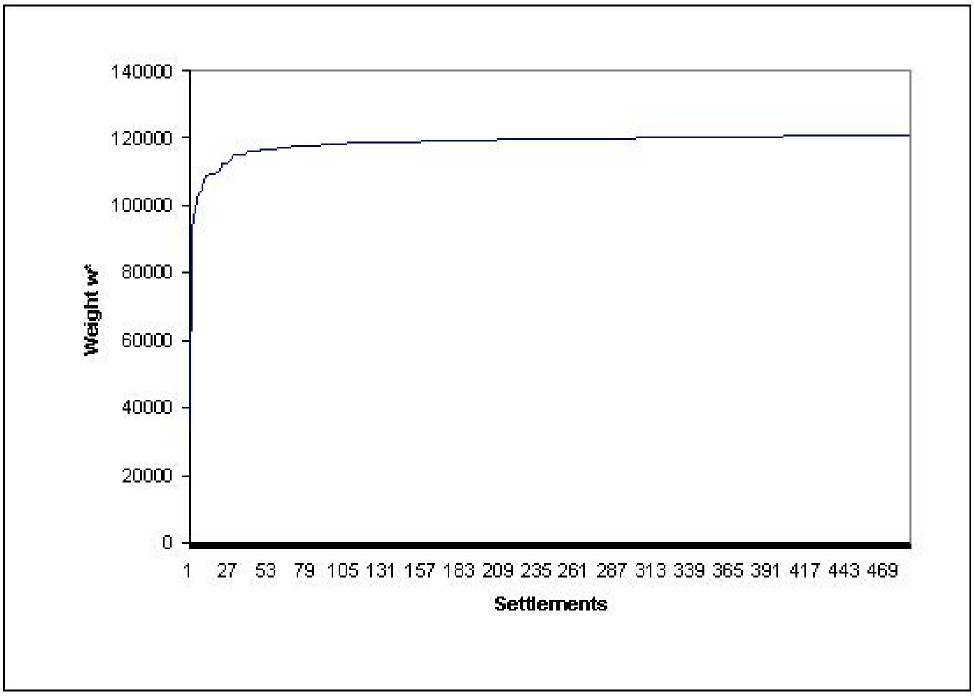


Figure 3.6: Saaremaa, scale of conformism, w_*.

We can see some steep slopes in those figures and that there are about 10-20 very atypical objects according to the weight function w_* as shown by the steep climb in the range 0-20 for both Hiiumaa and Saaremaa. There seem to be several of steep slopes for weight function w_1. These plots don't show what kind of attributes objects in those groups have in common. Some sort of interactive tool might be of help here, another way is to complement those plots with sequential tables describing groupings in more detail. One such table, generated quite tediously by hand, is shown below as Table 3.3.

Table 3.3 identifies those atypical objects mentioned before as the regional centers of island. Most typical settlements - objects with largest weights - are small settlements with the population 10-50 with no economic activity. Drawing such tables for both weight functions and producing a combined interpretation would be somewhat tedious. Next section describes a visualization that shows combined picture for both weight functions: w_1 and w_*.

Settlement	w_*	Attributes present	Comment
Kärdla	21659	109	Capital of the island
Käina	24733	91	Regional center
Emmaste	32221	91	Regional center
Kõrgessaare	34849	34	Regional center
Kassari	35795	30	
Nõmme	35947	27	
Männamaa	36017	27	
38 settlements	...	...	...
Nõmmerga	39375	0	Settlement without attributes
Sülluste	39375	0	-”-
Tiharu	39375	0	-”-
Viitasoo	39375	0	-”-
Leerimetsa	39375	0	-”-
112 settlements	...	...	...
Kuusiku	40005	8	population 10-50, children 1-10, workers 1-10, elderly 1-10
Kõmmuselja	40005	8	-”-
Kleemu	40005	8	-”-
Laheküla	40005	8	-”-
Mäeltse	40005	8	-”-
Pärnselja	40005	8	-”-
Heigi	40005	8	-”-
Heiste	40005	8	-”-
Kidaste	40005	8	-”-
Lilbi	40005	8	-”-
Poama	40005	8	-”-
Laartsa	40029	8	population 10-50, children 1-10, workers 10-50, elderly 1-10
Lepiku	40029	8	-”-
Ulja	40029	8	-”-
Aadma	40029	8	-”-
Jõeküla	40029	8	-”-
Ühtri	40029	8	-”-
Otste	40029	8	-”-
Kalgi	40029	8	-”-
Pilpaküla	40029	8	-”-
Sakla	40029	8	-”-

Table 3.3: Data table showing the settlements sorted according the scale of conformism and weight function w_* for Hiiumaa.

3.2 Conformity plot

Conformity plot visualization for binary data tables was developed by the author of this thesis and was published in 2006 [14] and in 2007 [15]. It is basically a scatter plot for objects where axes correspond to weight functions w_1 and w_*. As both weight functions generate somewhat different groupings, such scatter plot allows us to visualize clusters of objects - elements of monotone system - that would not be obvious if we had considered each weight function separately.

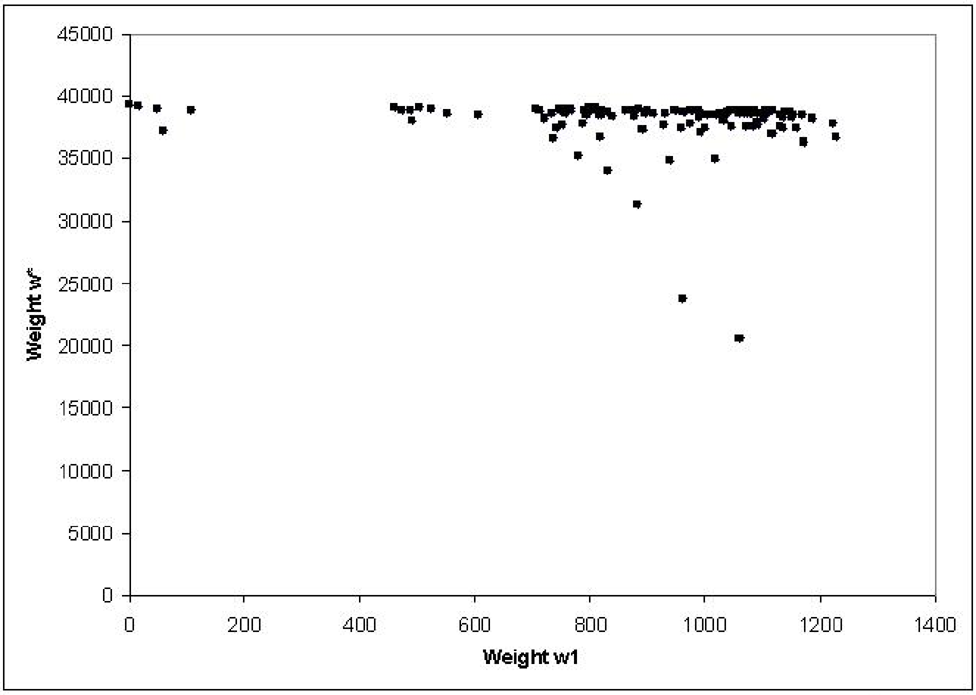


Figure 3.7: Hiiumaa, conformity plot, weight functions w_1, w_*.

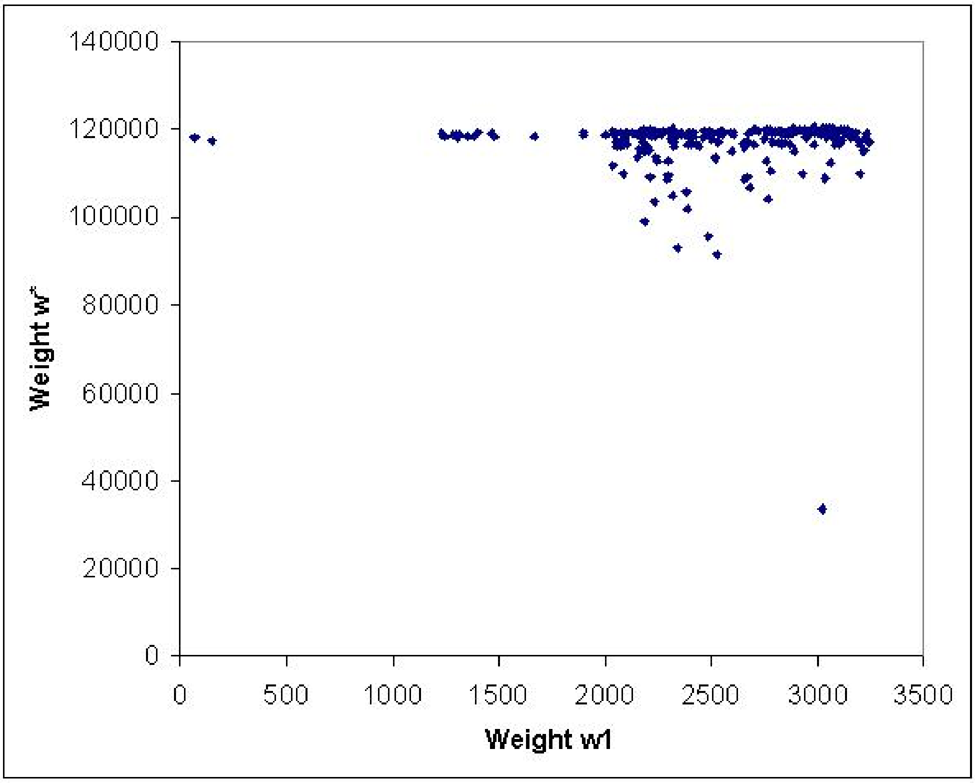


Figure 3.8: Saaremaa, conformity plot, weight functions w_1, w_*.

There is striking similarity between conformity plots for Hiiumaa and Saaremaa. It seems that conformity plot brings out similar structure in the data about island settlements. Such a plot makes outlier groups very obvious, structure of the main group, in the upper right is harder to see. Again, there is a problem of attaching semantic information to these clusters.

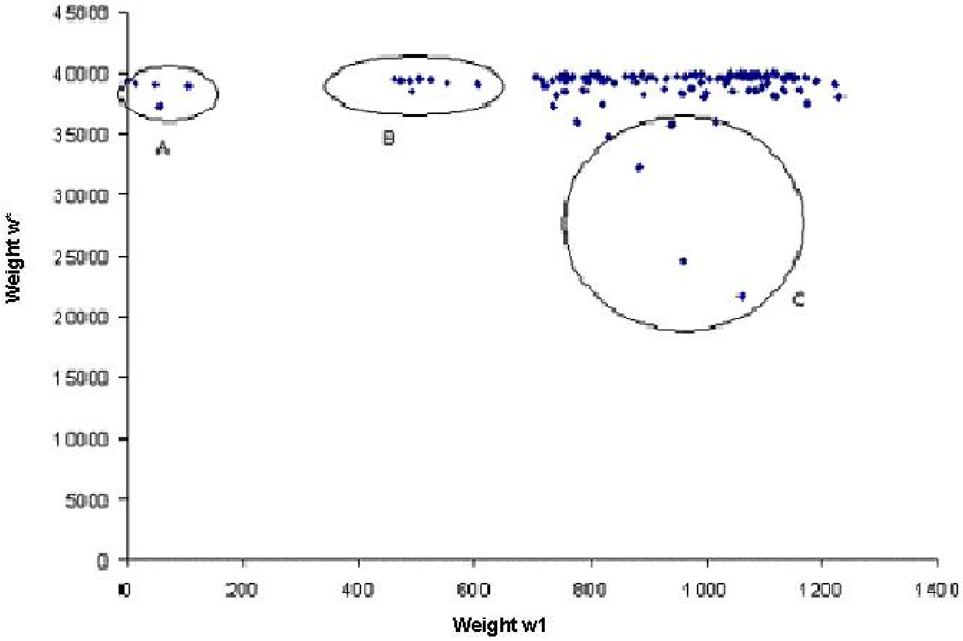


Figure 3.9: Hiiumaa, conformity plot with marked outlier clusters, weight functions w_1, w_*.

Semantic information for easy-to-detect outlier groups as shown in Figures 3.9 and 3.10 is same for both of the islands:
A: The most non-typical villages, people do not live there and villages have no social characteristics. But they have some economic activities, like harbor, custom, border guard, summer-cafe, etc, which are supervised from other (central) places.
B: The second clearly differentiated settlements group, has weaker social characteristics (no children in villages), than usual. They have also small harbors, coastal fishing, summer-cafes, sights etc. There are no private enterprises.
C: Large settlements and administrative centers, mentioned in previous section.

Main group (upper right) contains mostly settlements with the total population 10 to 50 people and having present both children, workers and elderly.

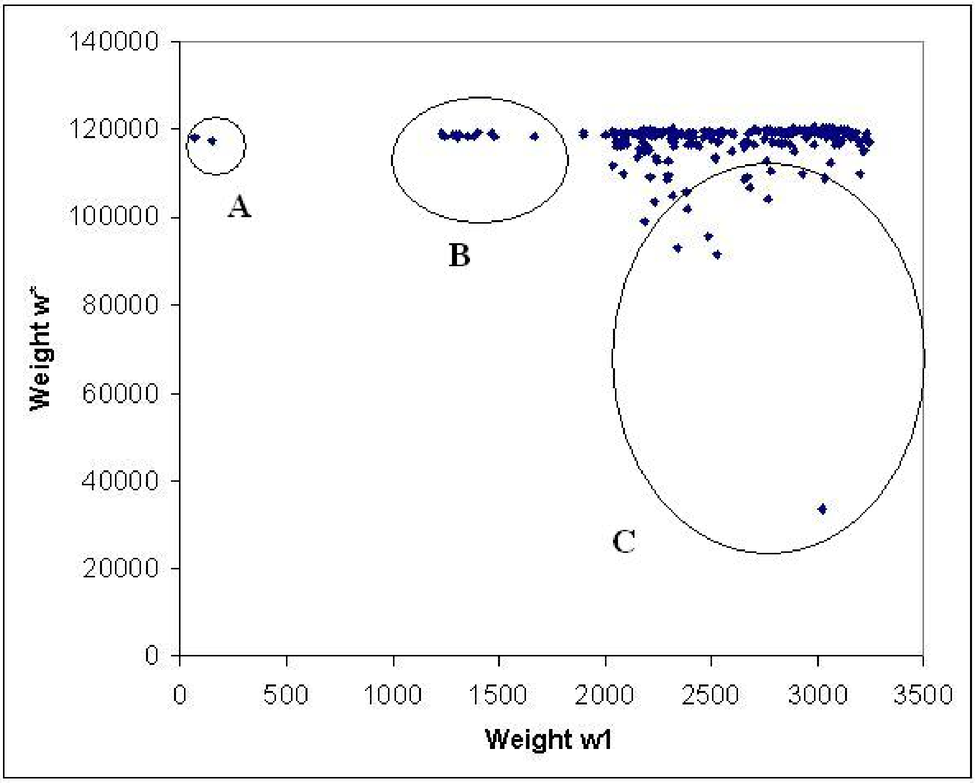


Figure 3.10: Saaremaa, conformity plot with marked outlier clusters, weight functions w_1, w_*.

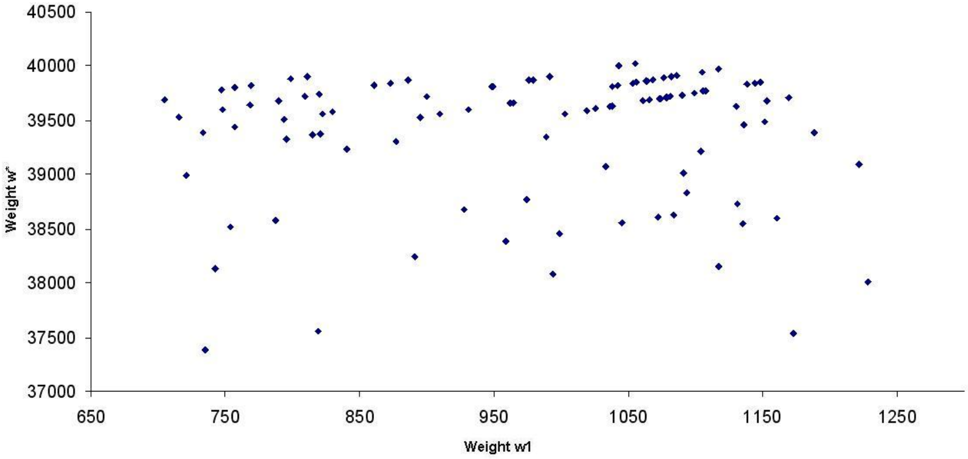


Figure 3.11: Hiiumaa, conformity plot for main group, weight functions w_1, w_*.

If we zoom into main group (Figure 3.11), we can see that elements of monotone system are arranged in perfect lines. There seems to be some kind of relation between the weight functions w_1 and w_* for binary data tables. As shown by the author in the article [15] there is a linear relation between values of functions w_* and w_1 for objects g_a and g_b that have same number of attributes. That is $|\{g_a\}'| = |\{g_b\}'|$. For the sake of simplicity let us presume that objects g_a and g_b differ by exactly two attributes $a, b \in M$, so that $g_aIa, \neg g_aIb, g_bIb, \neg g_bIa$. It is clear that we can transform one object into any other with same number of attributes with enough swaps between two attributes so such case suffices for showing the linear relationship.

Theorem 3 (Relation between w_1 and w_* for formal contexts)**.** *Let w_1 and w_* be frequency based monotone weight functions for objects as defined in section 2.2.2 and (G, M, I) be a formal context. Lets have two objects $g_a, g_b \in G$ and two attributes $a, b \in M$ so that $|\{g_a\}'| = |\{g_b\}'|$, $\{g_a\}' - \{a, b\} = \{g_b\}' - \{a, b\}$ and $g_aIa, \neg g_aIb, g_bIb, \neg g_bIa$. Then following linear relation holds:*

$$w_*(g_b) - w_*(g_a) = 2 \cdot (w_1(g_b) - w_1(g_a)). \tag{3.1}$$

Proof. There must exist a constant C describing the part of weight w_1 for g_a, g_b without a of b

$$w_1(g_a) = C + |\{a\}'| \tag{3.2}$$

$$w_1(g_b) = C + |\{b\}'| \tag{3.3}$$

$$w_1(g_b) - w_1(g_a) = |\{b\}'| - |\{a\}'|. \tag{3.4}$$

There must also exist a constant K describing the part of weight w_1 for g_a, g_b without a of b

$$w_*(g_a) = K + |\{a\}'| + |G| - |\{b\}'| \tag{3.5}$$

$$w_*(g_b) = K + |\{b\}'| + |G| - |\{a\}'| \tag{3.6}$$

$$w_*(g_b) - w_*(g_a) = 2 \cdot (|\{b\}'| - |\{a\}'|). \tag{3.7}$$

$$w_*(g_b) - w_*(g_a) = 2 \cdot (w_1(g_b) - w_1(g_a)). \tag{3.8}$$

□

Here we don't use such a relation much, but in an independent work by Innar Liiv such a relation is used to speed up calculation of w_* for sparse data tables by deducing w_* from faster to calculate w_1 [20], [21].

3.3 Case study: Minus technique

Minus technique allows us to find kernels and it usually gives somewhat better ordering than scale of conformity, though calculating it is computationally more demanding. Figures 3.12, 3.13, 3.14 and 3.15 show the settlements arranged by minus technique sequence – that is, by their order of removal from the system – and their weights while removed. Global maxima in these plots define the kernels.

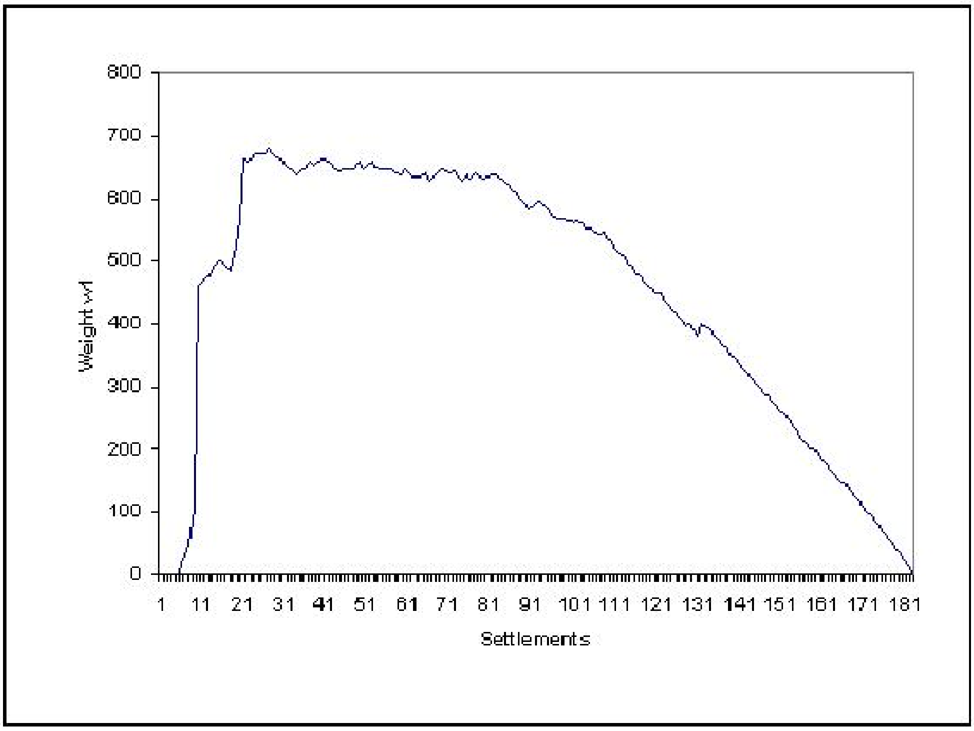


Figure 3.12: Hiiumaa, minus technique, w_1.

We can see from Figures 3.12, 3.13, 3.14 and 3.15 that plots for both islands are very similar. For both islands and both weight functions, kernels (settlements right from the global maximum) cover most of the monotone system so that it is easier to describe elements outside the kernels. For the weight function w_* these are big settlements and regional centers. For the weight function w_1 these are very small settlements with no population or population less than 10. The kernels are thus composed of mainly medium sized settlements. Local kernels, defined by the local maxima, don't help us here much in reducing the size of the kernel as the final part of minus technique sequence has smoothly decreasing weights. Again, there are difficulties in extracting the semantic information as it is not obvious from those plots

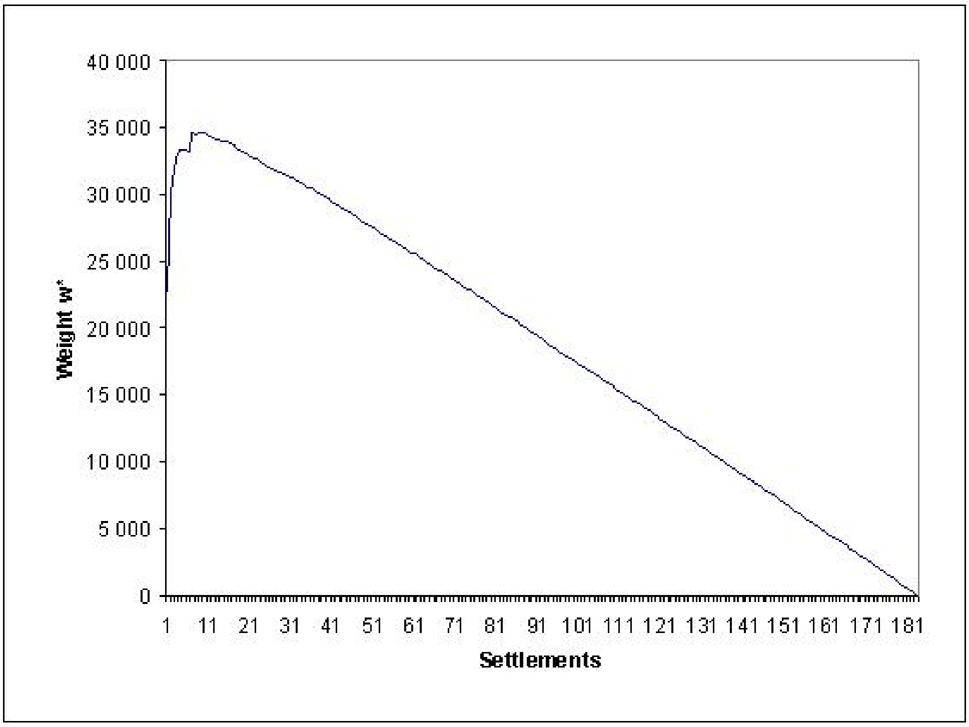


Figure 3.13: Hiiumaa, minus technique, w_*.

and has to be extracted from source data.

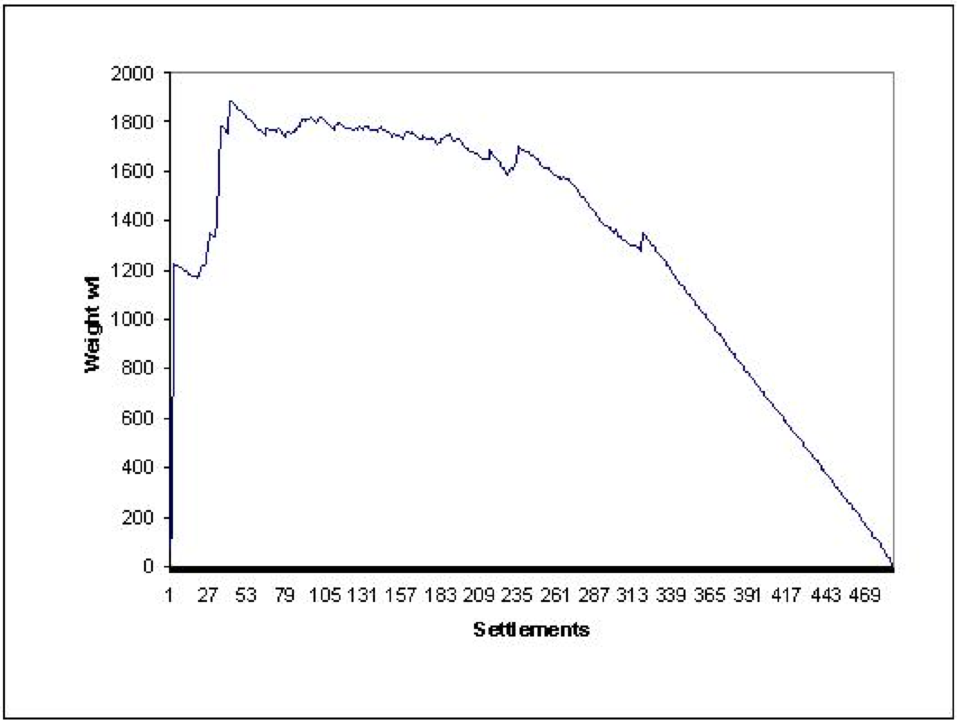


Figure 3.14: Saaremaa, minus technique, w_1.

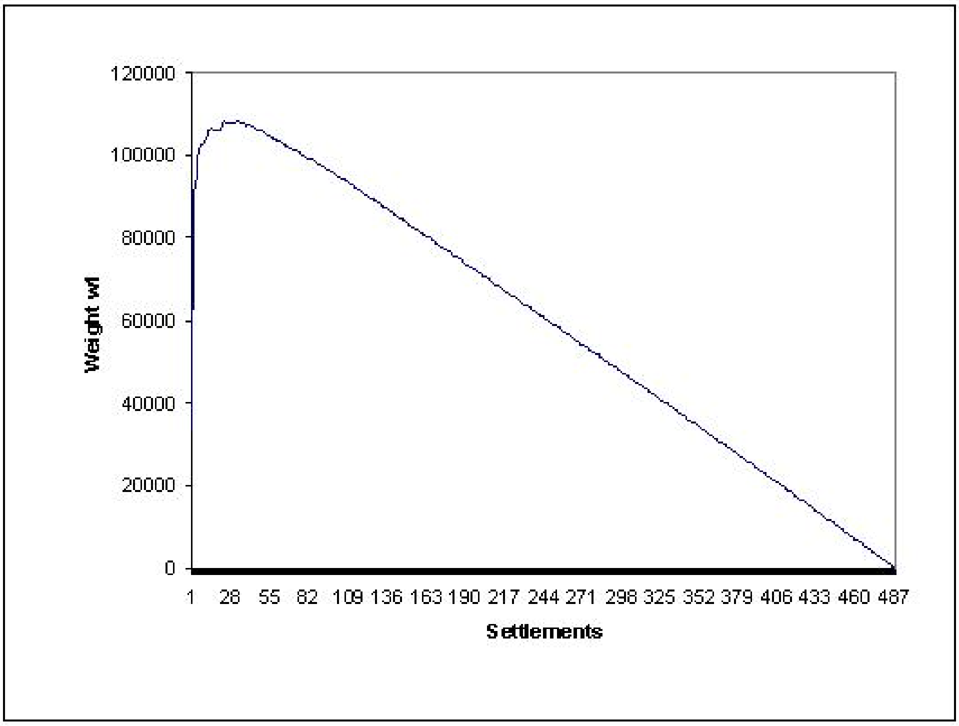


Figure 3.15: Saaremaa, minus technique, w_*.

How similar is minus technique sequence ordering to scale of conformism ordering for our case study? Scatter plots in Figures 3.16, 3.17, 3.18 and 3.19 relate settlements position in the minus technique sequence to its weight in the full system (scale of conformism). As we can see, ordering is pretty much the same for both cases, with some small differences.

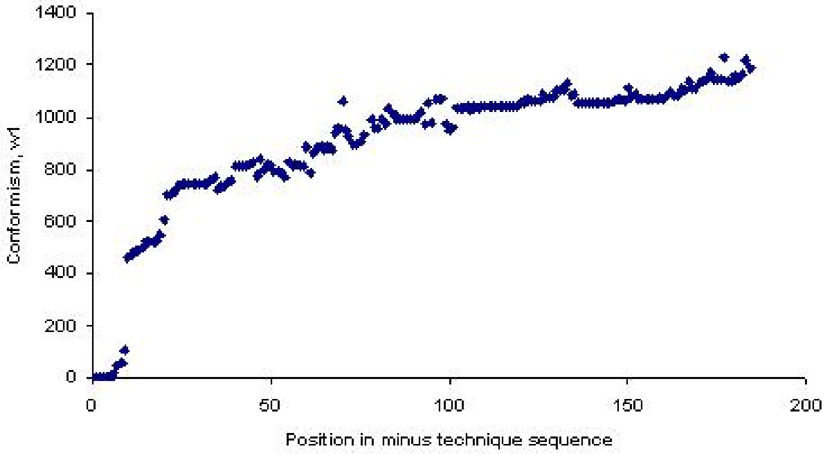


Figure 3.16: Hiiumaa, correlation between minus technique and scale of conformism, w_1.

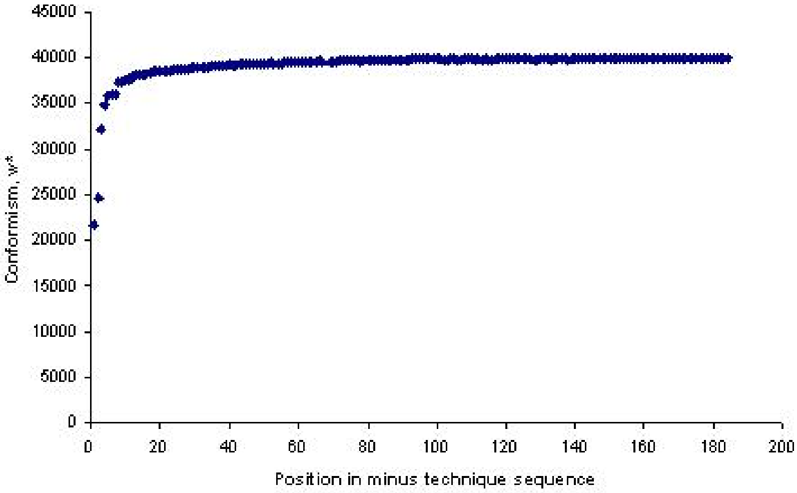


Figure 3.17: Hiiumaa, correlation between minus technique and scale of conformism, w_*.

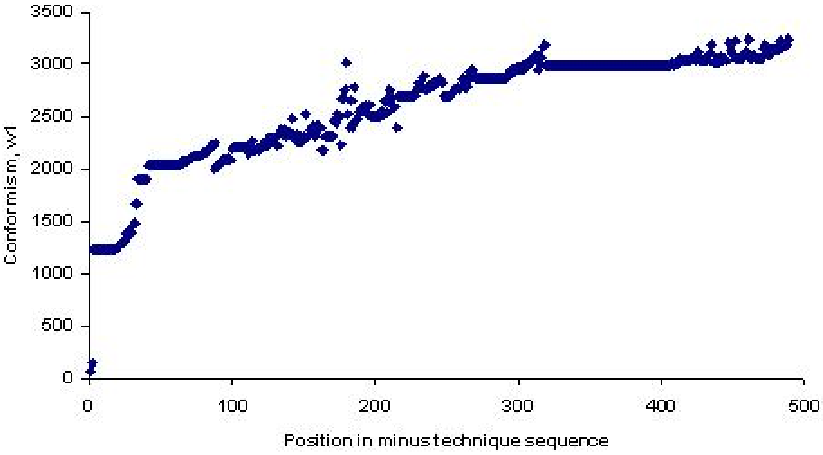


Figure 3.18: Saaremaa, correlation between minus technique and scale of conformism, w_1.

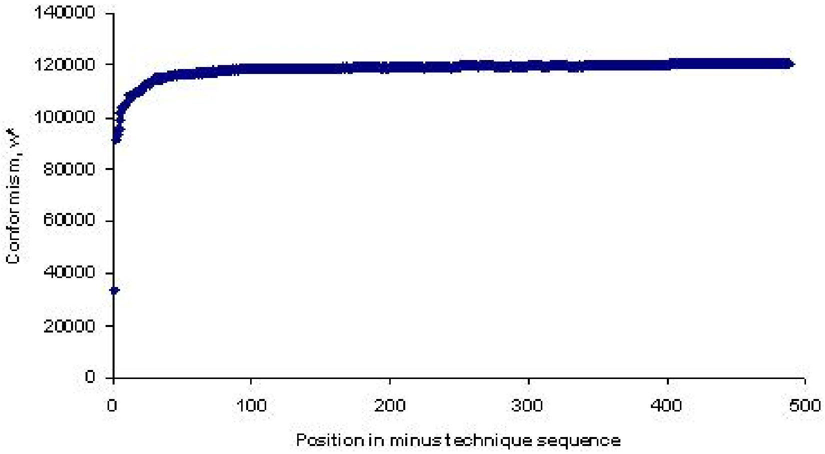


Figure 3.19: Saaremaa, correlation between minus technique and scale of conformism, w_*.

3.4 Best decision i.e. best concept chain

The problem of finding best decision and algorithms for it are based on the original work by Rein Kuusik [16]. Author of this thesis contributed with speed enhancements to the algorithm that provided the basis for authors bachelor thesis and year 2005 article [34]. In this thesis, for the first time, the problem of finding the best decision is defined in the way that is compatible with the language of FCA. Such a reframing motivates renaming the problem of finding the best decision, suggesting connection to decision trees, into the problem of finding the best concept chain.

Definition 9. *Best concept chain for the formal context (G, M, I) is the chain (A_1, B_1), (A_2, B_2), ... , (A_i, B_i), ... , (A_n, B_n) of the corresponding concept lattice that maximizes the area S covered by it, defined formally as:*

$$S = |\{(g, m) \in I | (\exists (A_i, B_i)) g \in A_i, m \in B_i\}|. \tag{3.9}$$

A chain of a concept lattice is a subset of concepts that has a linear order: there is a subconcept / superconcept relationship between any two concepts in a chain. More detailed information about lattice theory is found in the voluminous literature about the subject [13], [9] .

As we can see, such a definition is symmetrical: there is no need for dual definitions for objects and attributes. That is a novel result of reframing the problem in the terms of FCA. Previous work has described best decision as a sequence of attribute, value pairs, clearly unsymmetrical definition. This symmetry sadly does not hold for multivalued contexts as transformation from multivalued context into the single-valued formal context itself breaks the symmetry. It could be said that best concept chain is even more interesting in FCA terms than in its original, multi-valued form.

More traditional equivalent definition for best decision is following:

Definition 10. *Best decision for the formal context (G, M, I) is the sequence of attributes $m_1, m_2, ..., m_n$ where $m_i \in M$ that maximizes the area S covered by it, defined formally as:*

$$S = \sum_{i=1..n} |\{m_1, m_2, ..., m_i\}'| \tag{3.10}$$

or dually, sequence of objects $g_1, g_2, ..., g_m$ where $g_i \in G$ that maximizes the area

$$S = \sum_{i=1..m} |\{g_1, g_2, ..., g_i\}'|. \tag{3.11}$$

Figure 3.20 illustrates the equivalence of both definitions.

Best concept chain can be interpreted as a chain of concepts that describes the formal context best, in its sequence of attributes form, attributes are ordered by their descriptive strength, in its sequence of objects form, objects are ordered by their descriptive strength.

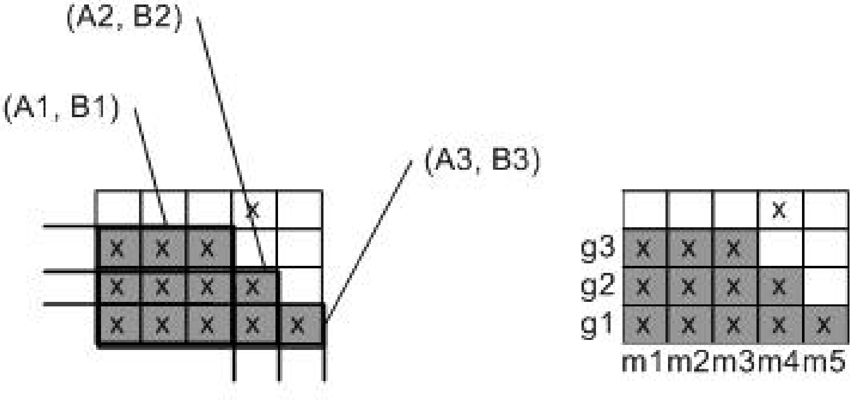


Figure 3.20: Area S (grayed) according to the definitions of best concept chain and best decision.

We can view the attributes $m \in M$ (there is, of course, dual definition for objects) as elements of the monotone system with the weight function $w(m, M) = |\{m\}'|$. Removal of an element would mean removing appropriate column from the context and removing all objects not in $\{m\}'$ from the data table. Minus technique sequence would then correspond to a greedy heuristic that does not guarantee the best concept chain, but usually a reasonably good concept chain.

To find the exact beset concept chain we can augment minus technique with backtracking. Lets have possibly partial set of attributes M, current best concept chain $Best$ and possibly partial concept chain under consideration $Partial$. Here we use sequence of attributes representation for the best concept chain. Then we can define backtracking brute-force algorithm recursively as follows:

```
BestCC($M$, $Best$, $Partial$):
if $|M| = 0$ then
    if $S(Best) < S(Partial)$ then
        Set $Best \leftarrow Partial$
    end if
else
    for all $m \in M$ ordered descending by $|\{m\}'|$ do
        Set $Best \leftarrow BestCC(M - m, Best, Partial + m)$
    end for
end if
Return $Best$
```

There are several enhancements that can be made to backtracking algorithms. We can prune the search by eliminating the hopeless branches of backtracking. As is claimed in the book about algorithm design by S. Skiena [30], good pruning techniques have stronger influence to the efficiency of a backtracking algorithm than any other factor.

There is a hard upper bound for the final area of any partial concept chain, as an addition of a new attribute can only reduce the number of objects in the partial context under consideration. Thus it is possible to place a hard upper limit for the final area of the partial concept chain. This limit was introduced in the work by R. Kuusik [16] and is called a **potential**.

Definition 11. *We define the potential of attribute m for the partial concept chain $Partial$ as $S(Partial) + |\{m\}'| \cdot |M|$ where M is the set of attributes in the partial context defined by the $Partial$.*

It is obvious that extending $Partial$ with m cannot give a concept chain with bigger area S than its potential.

That gives an improved version of backtracking algorithm:

```
BestCC($M$, $Best$, $Partial$):
if $|M| = 0$ then
  if $S(Best) < S(Partial)$ then
    Set $Best \leftarrow Partial$
  end if
else
  for all $m \in M$ ordered descending by $|\{m\}'|$ do
    if $S(Best) < (S(Partial) + |\{m\}'| \cdot |M|)$ then
      Set $Best \leftarrow BestCC(M - m, Best, Partial + m)$
    end if
  end for
end if
Return $Best$
```

Two further properties that allow for pruning were described in the article by the author of this thesis [34].

First property shows that there is no need to consider an element m_a that was already considered at one level higher in the backtracking tree.

Theorem 4 (Two-element area in a concept chain)**.** *For a set of attributes M from some formal (sub)context, if $|\{m_a\}'| \geq |\{m_b\}'|$ then*

$$S(\langle m_a, m_b \rangle) \geq S(\langle m_b, m_a \rangle) \tag{3.12}$$

and partial concept chains $\langle m_a, m_b \rangle$, $\langle m_b, m_a \rangle$ correspond to the same sub-context.

Second property shows that there is no need to consider an element m_a that was considered one level lower if it had exact same extent at lower level.

Theorem 5 (Invariant extent)**.** *For a set of attributes* M *from some formal (sub)context, if* $|\{m_a\}'| = |\{m_b, m_a\}'|$ *then there exists concept chain* $C_b = \langle m_b, ...\rangle$ *so that for any possible concept chain* $C_a = \langle m_a, ...\rangle$*:*

$$S(C_b) \geq S(C_a). \tag{3.13}$$

No proof is given as these properties are trivial.

These properties give final version of the algorithm for finding the best concept chain. Sets of elements that can be ignored are denoted by K (from upper level) and K' (from current level).

```
BestCC($M$, $Best$, $Partial$, $K$):
if $|M| = 0$ then
   if $S(Best) < S(Partial)$ then
      Set $Best \leftarrow Partial$
   end if
else
   Set $K' \leftarrow \{\}$
   for all $m \in M$ ordered descending by $|\{m\}'|$ do
      if $S(Best) < (S(Partial) + |\{m\}'| \cdot |M|)$ and not $m \in K$ and not $m \in K'$
      then
         Add $m$ into $K'$
         Set $Best \leftarrow BestCC(M - m, Best, Partial + m, K')$
         Add all $m_i$ where $|\{m_i\}'| = |\{m, m_i\}'|$ into $K'$
      end if
   end for
end if
Return $Best$
```

Figures 3.21, 3.22, 3.23 and 3.24 demonstrate the performance of all three different versions of the algorithm. Test rig had following hardware:

- processor Intel (R) Pentium (R) 4, 2.80 GHz
- 512 MB memory
- operating system Windows XP

Tests were run for different numbers of rows and columns, for random and structured data. Enhanced versions greatly outperform brute-force algorithm. Last version of an algorithm is also clearly faster than algorithm using only the potential for pruning.

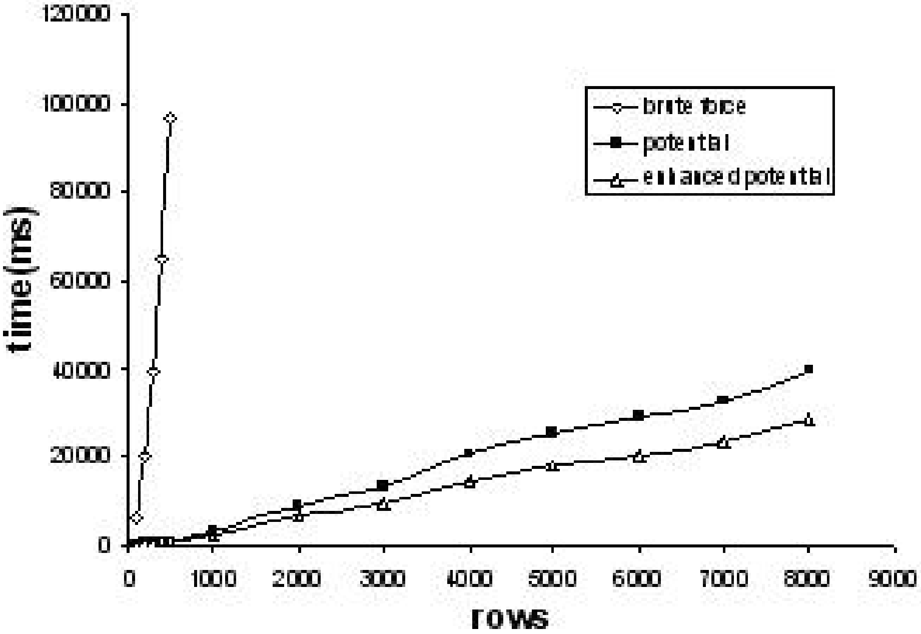


Figure 3.21: Performance of algorithms for finding the best concept chain. Random data, 5 columns.

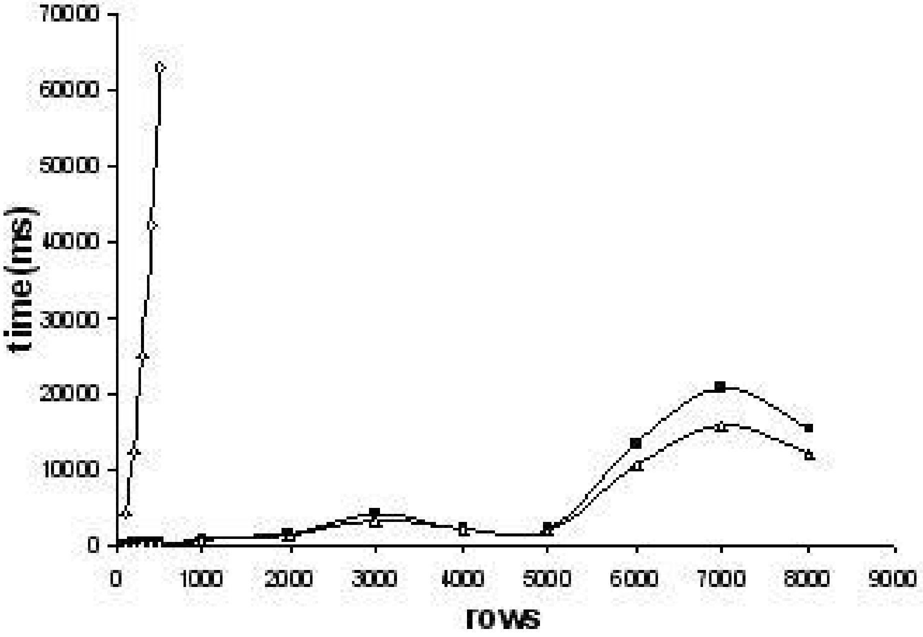


Figure 3.22: Performance of algorithms for finding the best concept chain. Structured data, 5 columns.

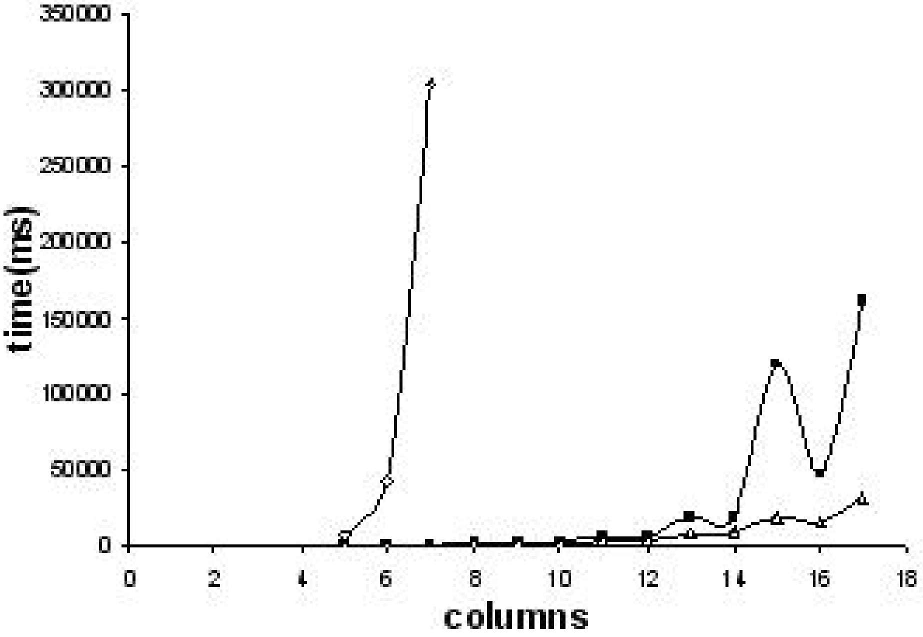


Figure 3.23: Performance of algorithms for finding the best concept chain. Random data, 100 rows.

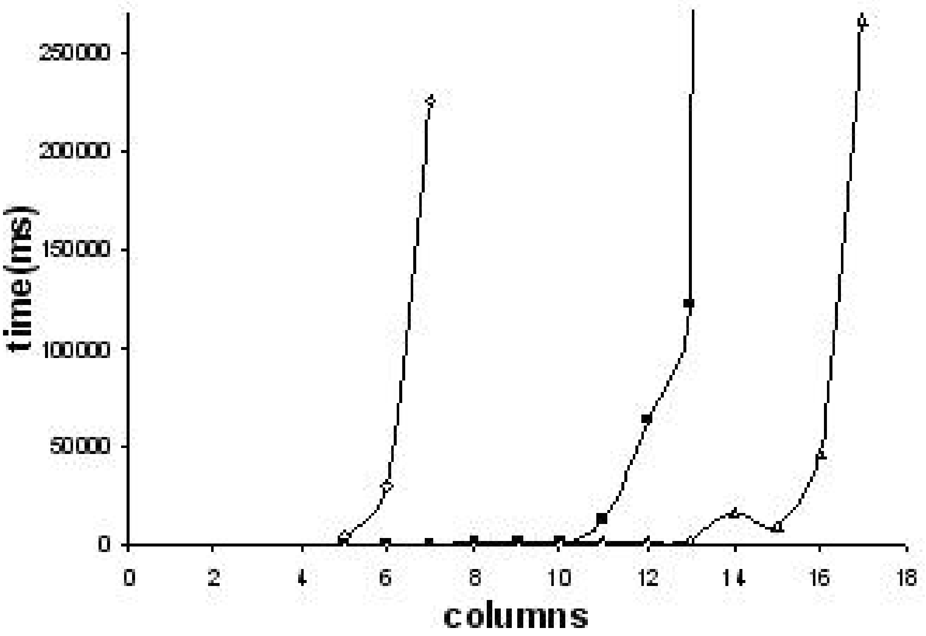


Figure 3.24: Performance of algorithms for finding the best concept chain. Structured data, 100 rows.

Chapter 4

MONOCLE Method for Knowledge Discovery

4.1 Motivation

As described in previous chapter, standard monotone system seriation methods make it hard to gain semantic information and meaning from the groups of objects they uncover. Formal concepts, on the other hand, have a nice semantic description - concepts intent. The problem with FCA is, that the number of concepts in the concept lattice can be too large for unaided analysis for quite a small context. MONOCLE method defines the formal concepts as elements of the monotone system, thus giving results that are semantically easy to interpret and reducing the size of a concept lattice. It is hard to argue why certain data analysis method is good or interesting. It is easy to define, for example, arithmetic mean, but why is it a good measure? Good data mining methods seem to have the properties of data compression and intuitive definition. Arithmetic mean compresses an arbitrarily large set of values into a single value that has intuitive definition as an "average value". While it may seem that seriation methods have no property of data compression, they reorder the data table into the form that allows the brain to compress the visual information in the data table into a small set of concepts. Their result has also very intuitive definition - it is the data table being analyzed. Sadly, for the large and complex context, brains ability to compress and interpret the data breaks down. MONOCLE method tries to compress the context by selecting concepts with a large area with little overlap. Its result is ordered set of formal concepts that has an intuitive and clear definition founded in logic, mathematics, philosophy and linguistics. This method was first presented in the 2008 article by the author of this thesis [35].

4.2 Concept Area

MONOCLE method is based on a modified version of concept area. Area of a concept is simply the product of the sizes of its extent and intent.

Definition 12. *For the concept* (A, B) *its area* $S(A, B)$ *is defined as* $S(A, B) = |A| \cdot |B|$.

Common methods for concept lattice reduction, like iceberg view, described by Stumme et al. [32] and common in association rule mining [3], are based on the size of an extent. Why should we consider the concept area as an alternative?

We can interpret the size of an extent as a measure of concepts applicability. For example, the concept *Animal* has a bigger extent than the concept *Wasp* and is applicable to more things. On the other hand, the size of an intent measures concepts information content. If we know that something is a *Wasp* and not just any *Animal* we have a much richer information about its potential behavior and properties.[1]

If we consider which concepts are deemed important enough to have their own words in natural language then it seems that the criteria combines both size of a concepts extent and intent (and perhaps some other factors). For example we have words for the *Animal*, *Insect*, *Mammal*, *Fish* ,*Wasp*, *Bee*, *Tiger* and *Striped Butterfly Fish (Chaetodon fasciatus)*. We have no word for yellow-black striped animals though the extent of this concept contains that of a *Wasp*, *Bee*, *Tiger* and *Striped Butterfly Fish* and the intent of that concept contains that of an *Animal*. It seems that the size of such concepts extent does not justify its relatively information poor intent. That is, natural language is structured more according to the concepts area than the size of concepts extent. Finding formal concepts from the data is a way of generating objective terminology for that domain. It enables selection and filtering of concepts according to the criteria that seems to have some correspondence to the structure of natural language.

Sometimes the number of attributes corresponds to some direct measure of interest: number of items in the shopping basket is correlated to income, number of economic functions performed in a settlement is correlated to the overall level of economic activity. Concept area corresponds here very nicely with a chunk of income and a chunk of economic activity.

The nature of a system studied may also necessitate an area based approach instead of an extent based one. If we have a system where a single unique object has a significant influence for the entire system then the extent based approach is clearly inappropriate. A single shopping basket is never of a significant importance for the shop, a single settlement may very well have such an importance for an economic system. These systems are characterized by B. Mandelbrot in his book “The

[1] Here we are making a simplifying assumption that all objects and attributes are of an equal importance.

(mis)Behavior of Markets" [24] as examples of "wild" randomness (Cauchy distribution) in contrast to the systems with "mild" randomness (Gaussian distribution). To quote Mandelbrot: "... But the difference between the extremes of Gauss and of Cauchy could not be greater. They amount to two different ways of seeing the world: one in which big changes are the result of many small ones, or another in which major events loom disproportionately large". Mandelbrot uses the software industry and Microsoft as an example of "wild" variation, a single object having great influence for the whole system, and argues that behavior of financial markets is also governed by this kind of distribution - the "fat tails".

4.3 MONOCLE method

We now introduce our **MONOCLE** (MONOtone Concept Lattice Elimination) method for knowledge discovery in binary data tables [35]. We treat concepts as elements of the monotone system and we define an appropriate **MONOCLE weight function**. Generally, the MONOCLE data analysis process is as follows:

1. Concept generation.
2. Generation of minus technique sequence of concepts using MONOCLE weight function.
3. Data analysis using subsets of suitable size from the top of the minus technique sequence and possibly using global and local kernels to suggest good cut-off points.

The MONOCLE weight function is monotone and is correlated with the concept area. By concept area we mean the product of extent size and intent size $|A| \cdot |B|$ of a concept (A, B). We modify the weight of each attribute and object in our area calculation by its "rareness". Such a definition preserves the symmetry between objects and attributes, peculiar to FCA. MONOCLE method should give same results even after the data table is transposed. Finally, we show that a certain invariance property holds for the MONOCLE weight function.

Definition 13. *Let W be the set of all concepts for some context and $H \subseteq W$. We denote the number of all concepts in H not containing the object g as $N_G(g, H)$ and define it formally as*

$$N_G(g, H) = \left| \left\{ (A, B) \mid (A, B) \in H, g \notin A \right\} \right| . \tag{4.1}$$

We denote the number of all concepts in H not containing the attribute m as $N_M(m, H)$ and define it formally as

$$N_M(m, H) = \left| \left\{ (A, B) \mid (A, B) \in H, m \notin B \right\} \right| . \tag{4.2}$$

Definition 14. *Let W be the set of all concepts for some context and $H \subseteq W$. Let the concept $x \in H$ have extent A and intent B. We define the MONOCLE weight function w(x, H) as*

$$w(x, H) = \left(|A| + \sum_{g \in A} N_G(g, H)\right) \cdot \left(|B| + \sum_{m \in B} N_M(m, H)\right) . \tag{4.3}$$

We illustrate MONOCLE weight function by examples from the Figure 4.1.

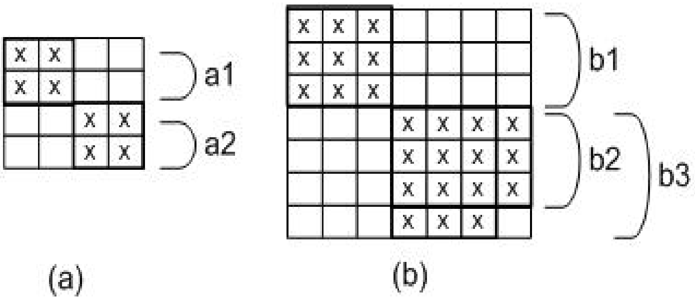


Figure 4.1: Two sample contexts with the concepts marked inside the table by borders and labeled outside the table by their extents.

Each object and attribute of the concepts in the set $H = \{a_1, a_2\}$ for the context (a) is not contained in exactly one concept, so $N_G(g, H) = 1$ and $N_M(m, H) = 1$ for any object g or attribute m in the context (a). Weights for the concepts in the context (a) are

$$\begin{gathered} w(a_1, \{a_1, a_2\}) = w(a_2, \{a_1, a_2\}) \\ = \big((1+1) + (1+1)\big) \cdot \big((1+1) + (1+1)\big) \\ = 16 \; . \end{gathered} \tag{4.4}$$

Both a_1, a_2 and a_2, a_1 are correct minus technique sequences,

$$w(a_1, \{a_1\}) = w(a_2, \{a_2\}) = \big((0+1) + (0+1)\big) \cdot \big((0+1) + (0+1)\big) = 4 \; . \tag{4.5}$$

so the corresponding sequence of F_{min} is $16,\ 4$; $\{a_1, a_2\}$ is a kernel.

For the context (b):

$$w(b_1, \{b_1, b_2, b_3\}) = (3+3+3) \cdot (3+3+3) = 81 \; ; \tag{4.6}$$

$$w(b_2, \{b_1, b_2, b_3\}) = w(b_3, \{b_1, b_2, b_3\}) = (2+2+2) \cdot (2+2+2+3) = 54 \; . \tag{4.7}$$

Two minus technique sequences are b_2, b_1, b_3 and b_3, b_1, b_2 and the corresponding sequence of F_{min} is $54,\ 36,\ 12$; thus kernel is $\{b_1, b_2, b_3\}$. Here, minus technique sequence is clearly different from the simple area calculation $|A| \cdot |B|$ where b_1 would be the first element removed from the system.

4.3.1 Invariance Property

We now demonstrate that we can change certain contexts in certain ways that preserve the weights of corresponding concepts in the old and new contexts.

Let us consider the three pairs of contexts, where concepts do not overlap, shown in Figure 4.2.

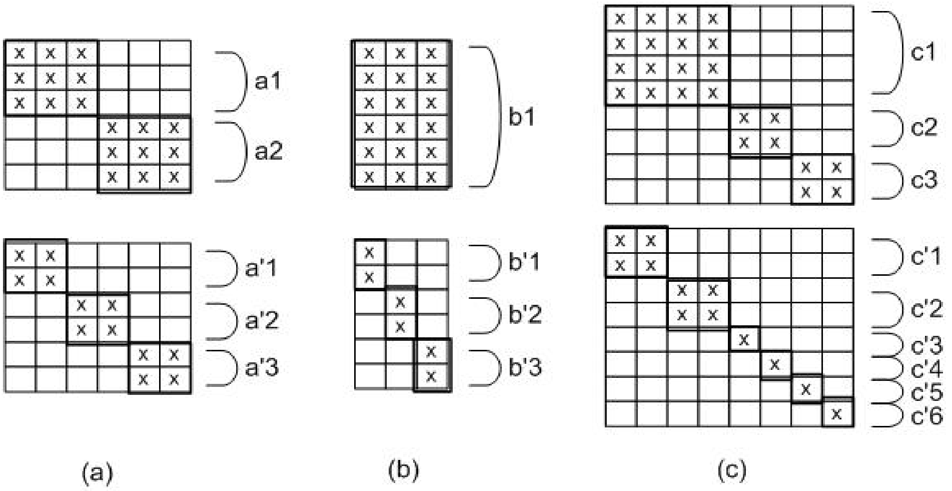


Figure 4.2: Three pairs of contexts with non-overlapping concepts. The concepts are marked inside the table by borders and labeled outside the table by their extents.

The set of objects G and the set of attributes M are unchanged for the pairs. For each concept in the upper contexts, we create r concepts in the lower contexts, leaving extent and intent size ratios between the concepts unchanged. For the pair (a) $r = 3/2$, for the pair (b) $r = 3$ and for the pair (c) $r = 2$. We can see that the weights of corresponding concepts are equal, for example:

$$w(a_1, \{a_1, a_2\}) = w(a'_1, \{a'_1, a'_2, a'_3\}) = 36 \tag{4.8}$$

$$w(b_1, \{b_1\}) = w(b'_1, \{b'_1, b'_2, b'_3\}) = 18 \tag{4.9}$$

$$w(c_1, \{c_1, c_2, c_3\}) = w(c'_1, \{c'_1, c'_2, c'_3, c'_4.c'_5, c'_6\}) = 144 \tag{4.10}$$

$$w(c_2, \{c_1, c_2, c_3\}) = w(c'_3, \{c'_1, c'_2, c'_3, c'_4.c'_5, c'_6\}) = 36 \tag{4.11}$$

$$F_{min}(\{a_1, a_2\}) = F_{min}(\{a'_1, a'_2, a'_3\}) = 36 \tag{4.12}$$

$$F_{min}(\{b_1\}) = F_{min}(\{b'_1, b'_2, b'_3\}) = 18 \tag{4.13}$$

$$F_{min}(\{c_1, c_2, c_3\}) = F_{min}(\{c'_1, c'_2, c'_3, c'_4.c'_5, c'_6\}) = 36\ . \tag{4.14}$$

We now demonstrate that property formally.

Theorem 6 (Invariance property). *Let W be the system of non-overlapping concepts with set of objects G and set of attributes M. Let W' be another system of non-overlapping concepts with set of objects G' and set of attributes M' so that $|G| = |G'|$ and $|M| = |M'|$. Let r be a rational number so that for the sets of concepts defined by any pair of natural numbers n, m*

$$H = \big\{(A, B) \bigm| |A| = n, |B| = m, (A, B) \in W\big\} \tag{4.15}$$

$$H' = \big\{(A', B') \bigm| |A'| = \frac{n}{r}, |B'| = \frac{m}{r}, (A', B') \in W'\big\} \tag{4.16}$$

it holds that

$$|H'| = r \cdot |H|. \tag{4.17}$$

Then for any $c = (A, B) \in H$ and $c' = (A', B') \in H'$

$$w(c, W) = w(c', W'). \tag{4.18}$$

Proof. We can see that for non-overlapping concepts

$$|A| + \sum_{g \in A} N_G(g, W) = |W| \cdot |A| \tag{4.19}$$

$$|B| + \sum_{m \in B} N_M(m, W) = |W| \cdot |B| \ . \tag{4.20}$$

We also know that

$$\big|W'\big| \cdot \big|A'\big| = r \cdot |W| \cdot \frac{|A|}{r} = |W| \cdot |A| \tag{4.21}$$

$$\big|W'\big| \cdot \big|B'\big| = r \cdot |W| \cdot \frac{|B|}{r} = |W| \cdot |B| \ . \tag{4.22}$$

Thus

$$w(c', W') = (\big|W'\big| \cdot \big|A'\big|) \cdot (\big|W'\big| \cdot \big|B'\big|) = (|W| \cdot |A|) \cdot (|W| \cdot |B|) = w(c, W) \ . \tag{4.23}$$

□

4.4 Case Study: socio-economic data

We now return to our case study of Saaremaa and Hiiumaa from Section 3.1 and apply MONOCLE method to these data. Our set of objects consists of settlements and the set of attributes consists of various social and economic characteristics like the presence of a school, a kindergarten, shops or certain types of industry. Here we exclude demographic attributes (number of children, workers and elderly) that were included in our research presented in Section 3.1 and [14] as these would tend to dominate the results and information provided by these attributes is somewhat less interesting than that from the more qualitative attributes. We have also applied the MONOCLE method to data with all the attributes present and results are generally consistent with those from Section 3.1 and in a more explicit and easier to interpret form. For Hiiumaa $|G| = 184$ and $|M| = 206$; for Saaremaa $|G| = 488$ and $|M| = 234$. The attribute sets are mostly similar, however some attributes are present for only one island, hence some differences.

The set of concepts for Hiiumaa contained 380 concepts, the set of concepts for Saaremaa contained 1823 concepts. The weights of minus technique sequences are presented in Figure 4.3.

As we can see from the Figure 4.3, the global kernels H_G and S_G are quite large. Smallest local kernel for Hiiumaa is H_L that is still pretty large. Smallest local kernel for Saaremaa S_1 contains 17 concepts, pretty good size for the general overview of the system. For Hiiumaa we select “almost” a local kernel H_2 that contains 10 concepts instead of the too large H_L. We also select subset H_2 that is equal in size to S_2 and S_1 that is equal in size to H_1 for comparison.

We present the concept lattices for Hiiumaa corresponding to H_1 and H_2 as the Figure 4.4. Note that concepts corresponding to intersections of extents and intents are also added. Lattices were generated from data tables that contained only concepts in H_1 or H_2 , using Galicia [1]. Markings for concepts in H_1 or H_2 were added later.

The following list is the tail of the minus technique sequence, numbered backwards: concepts in H_1 (all 17) and H_2 (first ten). Numbering corresponds to Figure 4.4. If extent or intent is large, we provide only its size. We use the format: Weight $w(x_n, H_n)$; {extent}, {intent}.

1. Weight 116; {Kärdla, Käina}, (58 attributes)

2. Weight 250; (68 settlements), {summer cabins}

3. Weight 382; {Käina}, (83 attributes)

4. Weight 574; {Kärdla}, (101 attributes)

5. Weight 625; {Emmaste}, (41 attributes)

6. Weight 747; (17 settlements), {summer cabins, beach}

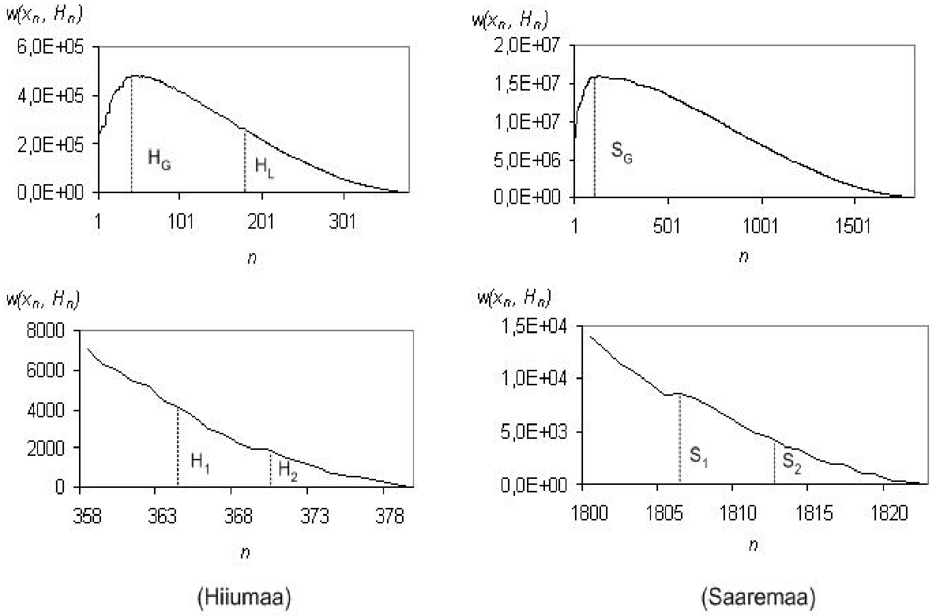


Figure 4.3: Minus technique sequences for Hiiumaa and Saaremaa. Tails of sequences, containing most interesting concepts, are presented separately below main sequence. Several kernels and cut-off points used for following analysis $H_G, H_L, H_1, H_2, S_G, S_1, S_2$ are marked with dashed lines.

7. Weight 1050; {Kärdla, Käina, Emmaste}, (25 attributes)
8. Weight 1352; (22 settlements), {agriculture}
9. Weight 1536; (32 settlements), {housing}
10. Weight 1974; {Käina, Emmaste}, (27 attributes)
11. Weight 1976; {Kärdla, Emmaste}, (28 attributes)
12. Weight 2277; {Kärdla, Kõrgessaare, Käina}, (16 attributes)
13. Weight 2717; {Nõmme}, (22 attributes)
14. Weight 3000; {Kõrgessaare, Käina}, (19 attributes)
15. Weight 3620; (15 settlements), {summer cabins, housing}
16. Weight 4130; (22 settlements), {beach}

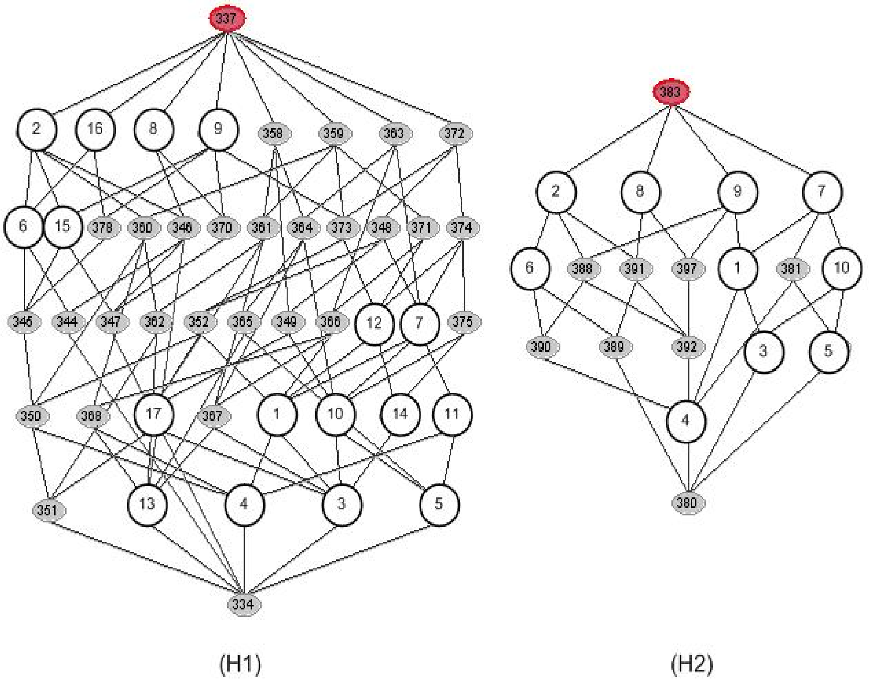


Figure 4.4: Lattices H_1 and H_2 for Hiiumaa. Concepts in H_1 and H_2 are marked with big numbered circles.

17. Weight 4444; {Kassari, Käina}, (17 attributes)

We present the concept lattices for Saaremaa corresponding to S_1 and S_2 as the Figure 4.5.

Following is the list of concepts for Saaremaa.

1. Weight 179; {Kuressaare}, (179 attributes)
2. Weight 272; (68 settlements), {landing places for fishing boats}
3. Weight 456; (87 settlements), {summer cabins}
4. Weight 936; {Kuressaare, Orissaare}, (52 attributes)
5. Weight 1204; {Kuressaare, Nasva}, (47 attributes)
6. Weight 1878; (55 settlements), {agriculture}
7. Weight 2007; {Kuressaare, Kärla}, (44 attributes)
8. Weight 2409; {Kuressaare, Valjala}, (39 attributes)

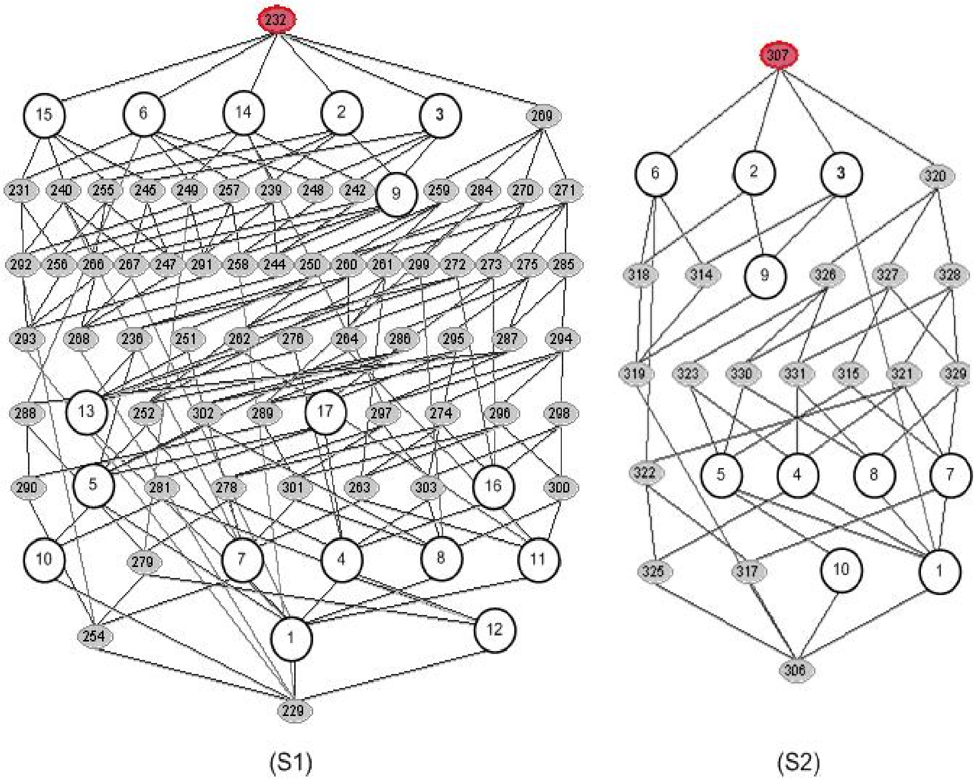


Figure 4.5: Lattices S_1 and S_2 for Saaremaa. Concepts in S_1 and S_2 are marked with big numbered circles.

9. Weight 3330; (32 settlements), {landing places for fishing boats, summer cabins}
10. Weight 3582; {Nasva}, (54 attributes)
11. Weight 4448; {Kuressaare, Liiva}, (35 attributes)
12. Weight 4910; {Orissaare}, (58 attributes)
13. Weight 5681; {Kuressaare, Kudjape}, (30 attributes)
14. Weight 6534; (56 settlements), {housing}
15. Weight 7449; (41 settlements), {sights}
16. Weight 8085; {Kuressaare, Orissaare, Liiva}, (24 attributes)

17. Weight 8550; {Kuressaare, Valjala, Tornimäe, Kärla}, (17 attributes)

There is a clear division between concepts describing small monofunctional settlements (agriculture, summer cabins) and larger regional centers (Kärdla, Käina, Kuressaare). That division is fundamental to the data and not the artifact of MONOCLE method - there are very few settlements that are neither monofunctional nor regional centers. The division seems to be clearer in the case of Saaremaa where larger centers, represented by the "artificial" concept in the upper right corner of lattices S_1 and S_2 do not have attributes common with concepts describing monofunctional settlements. Upper right "artificial" concept for S_1 has value {several enterprises with turnover over million crowns}, {Kudjape, Nasva, Kuressaare, Kärla, Liiva, Orissaare, Tornimäe, Valjala}. Role of Saaremaa's capital Kuressaare seems to be more important as that of Kärdla for Hiiumaa as {Kuressaare} is the extent of the last concept in the minus technique sequence.

We finally compare graphs presented in Figure 4.3 to that of random data, having same frequency of ones and size as the data table for Hiiumaa. Figure 4.6 shows graph for random data and we can see that graph for minus technique sequence is heavily influenced by the internal structure of data.

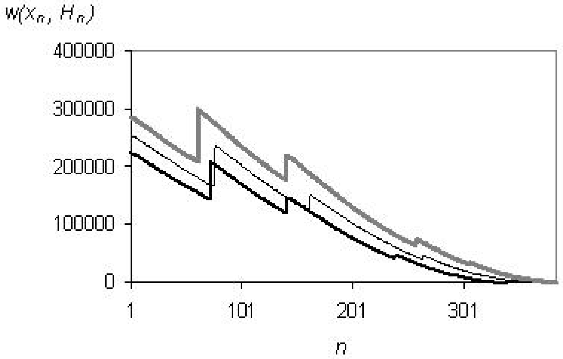


Figure 4.6: Three minus technique sequences for random data tables where size and frequency of ones were same as that of Hiiumaa. Sawteeth correspond to different simple concept areas.

Running speeds for building the lattice and finding the minus technique sequence ranged from couple of seconds for Hiiumaa without demographic data to couple of minutes for Saaremaa with demographic data. Hardware was ordinary desktop computer and the program was written in Python. Detailed discussion of speed and complexity issues is outside the scope of this thesis.

4.5 Case Study: epidemiology

This case studies the epidemiological data about dengue fever outbreaks in Brazil. These data were kindly supplied by professor Åke Sivertun from Linköping University. Input data table describes the number of infections per month for the districts of the Brazil. It contains data about 201 districts, in the shape shown in Table 4.1.

District	jan	feb	...
Alto da Boa Vista	24.2	12.1	...
Anchieta	14.9	52.0	...
...	...	...	...

Table 4.1: Fragment of the input. The table shows the number of infections per 100 000 people.

Our first task is to transform the data that is in real numbers into the binary as required by the MONOCLE method. The transformation should also relate concept areas to some interesting epidemiological property.

We transform the input by replacing each month column with three quartile interval columns. We set our first quartile strictly to zero due to the large amount of zero values. Other three parts are equal. Quartile values are calculated over the population of entire data table and not for the each row or column separately. Data table contains 1 wherever the rate of infection is greater than the corresponding quartile. For our previous table we found the quartiles 0, 12.9, 50.2. Such an arrangement relates concept area to the intensity of an infection. The result is demonstrated in Table 4.2.

District	jan>0	jan>12.9	jan>50.2	feb>0	feb>12.9	feb>50.2	...
Alto da Boa Vista	1	1	0	1	0	0	...
Anchieta	1	1	0	1	1	1	...
...	...	...	...	...	...	...	...

Table 4.2: Fragment of the transformed table.

Following are the top concepts according to minus technique for the year 2001. As we can see, they describe the wave of illness enveloping certain districts.

1. Weight 1704; {142 districts}, {'feb1', 'march1', 'march2', 'april1', 'april2', 'may1', 'may2', 'may3', 'june1', 'june2', 'july1', 'dec1'}

2. Weight 1860; {144 districts}, {'jan1', 'feb1', 'march1', 'april1', 'april2', 'may1', 'may2', 'june1', 'june2', 'july1', 'dec1'}

3. Weight 2422; {144 districts}, {'march1', 'march2', 'april1', 'april2', 'april3', 'may1', 'may2', 'may3', 'june1', 'june2'}

Total number of ones in the transformed input table [2] was 4071, the top concept (concept 1) covers 1704 of them, that is describes almost half of the context. Figure 4.7 illustrates these waves more clearly. We can see, that there are great similarities for these three different descriptions of wave of illness, however there are also some differences: concept 2 describes longer lasting, but less intense wave, concept 3 describes shorter but more intense wave. This suggests, difference in the development of epidemic between some districts.

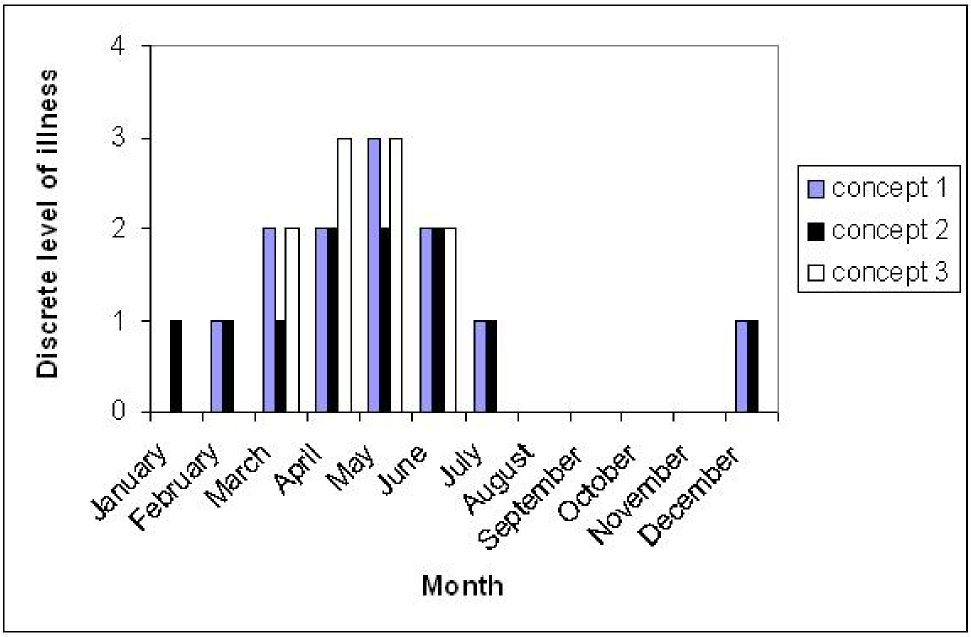


Figure 4.7: Waves of illness, as described by 3 top concepts for year 2001.

Following are the top concepts according to minus technique for the year 2002.

1. Weight 1705; {155 districts}, {'jan1', 'jan2', 'feb1', 'feb2', 'feb3', 'march1', 'march2', 'march3', 'april1', 'april2', 'may1'}
2. Weight 1848; {146 districts}, {'jan1', 'jan2', 'feb1', 'feb2', 'march1', 'march2', 'march3', 'april1', 'april2', 'may1', 'nov1'}
3. Weight 2506; {141 districts}, {'jan1', 'jan2', 'jan3', 'feb1', 'feb2', 'feb3', 'march1', 'march2', 'march3', 'april1', 'april2'}

Total number of ones in the transformed input table was 3436, the top concept covers 1705 of them, Figure 4.8 provides graphical description.

[2] more formally, number of relations in the formal context

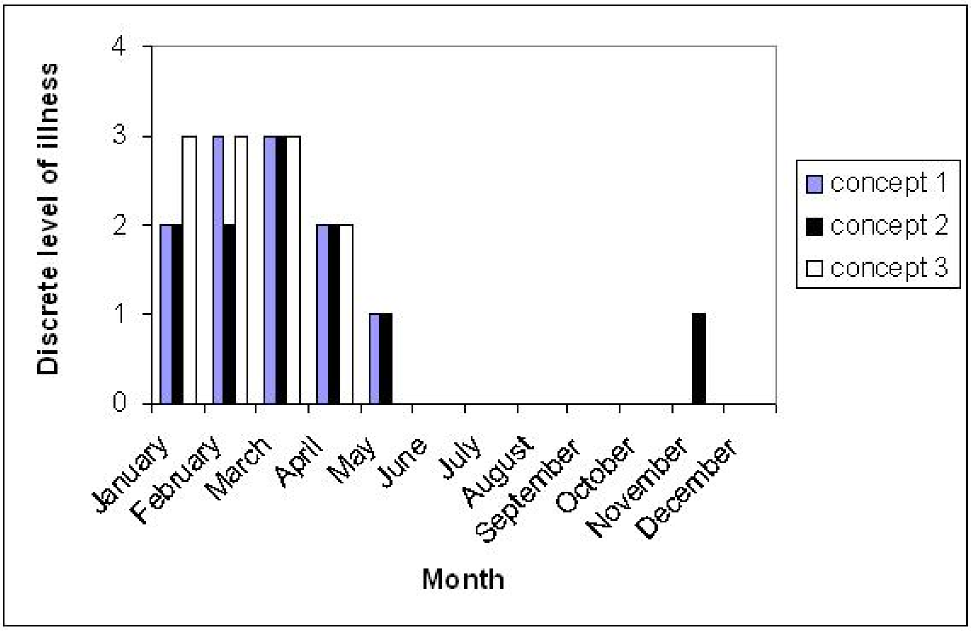


Figure 4.8: Waves of illness, as described by 3 top concepts for year 2002.

These results demonstrate that MONOCLE method is clearly applicable to diverse problem domains. It is remarkable that top concepts correspond to continuous waves as there is nothing in the algorithm that would force the intents to be continuous time intervals, in fact there is no concept of time encoded into the algorithm and no formal way to detect which attributes are corresponding to adjacent months and which are not. Such a result can be explained by a fact that if some set districts had similar epidemiological pattern in time intervals t_1 and t_3, then it is likely that they had similar epidemiological pattern in time interval t_2, between time intervals t_1 and t_3.

4.6 Discussion

MONOCLE method seems to give results, that are easy to understand, sensible and as compact as needed. It performed well in two different case studies. Issues of speed and complexity were not addressed. MONOCLE method requires the generation of entire concept lattice. It prunes this lattice for the human analyst but not in the algorithmic sense. This algorithm had enough speed for our case studies but it will not scale up very well. Some interesting fast heuristics are given by Ene, Horne, Milosavljevic, Rao, Schreiber and Tarjan in their year 2008 work about role minimization [10]. All their methods apply to formal concepts and the heuristic they describe as giving best results is basically a minus technique from the monotone systems theory. Elements of their monotone system are from the joint set of objects and attributes $G + M$, elements weight is either $|\{g\}'|$ or $|\{m\}'|$, and removal of an element removes the formal concept generated by it - $(\{g\}'', \{g\}')$ or $(\{m\}', \{m\}'')$ - from the context. It is interesting to note that they seem to have arrived to their results independently from FCA and monotone systems theory as neither is referenced in their work.

Chapter 5

How to measure formal concepts importance for the different contexts

5.1 Motivation

Sometimes we have different contexts that share either the set of attributes or the set of objects with each other. We may have gathered same type of information about different groups of objects like, for example, settlements in different islands and we may want to know how these contexts are different and how they are similar. Or we may want to compare the contexts that correspond to the same set of objects but with the different sets of attributes to check how dependent our analysis is to the selection of attributes or those sets of attributes focus on different interesting aspects and we want to compare and integrate those separate views. This is illustrated by Figure 5.1.

5.2 Simple measures

How should we evaluate and present the information about similarities and differences of different contexts? One approach would be to find a single numerical measure corresponding to similarity of two contexts. Another approach would be to measure if a concept is relatively more important in one context than in another. Here we use the latter concept based approach as it can show both differences and similarities between two contexts, while context based approach would roll it all into one number and our MONOCLE method provides us with means to select a limited subset of concepts so we don't have to be afraid of information overload. As such we are interested in measures *uniqueness(A, context, othercontext)* and *uniqueness(B, context, othercontext)* that measure whether concept (A, B) is unique for $context$ as

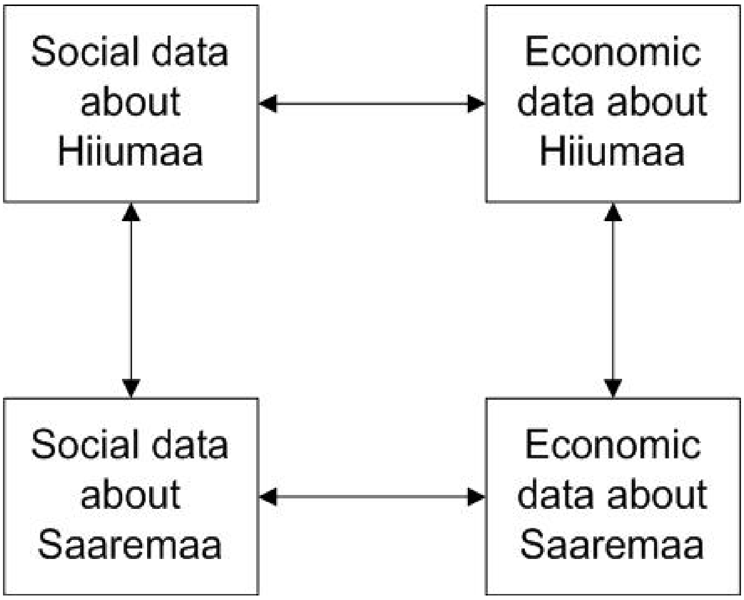

Social data about Hiiumaa
Economic data about Hiiumaa
Social data about Saaremaa
Economic data about Saaremaa

Figure 5.1: Some possible comparisons based on the case study of Saaremaa and Hiiumaa.

compared to $othercontext$ based on concepts relative importance.

However, finding such a measure is not a trivial task. One simple way is to define the importance of a concept (A, B) for the context (G, M, I) as a ratio of its strict area a_S to the full area of a context $\frac{|A| \cdot |B|}{|G| \cdot |M|}$.

We define strict concept area a_S as follows:

Definition 15. *For the context $K = (G, M, I)$ and the intent $B \subseteq M$:*

$$a_S(B, K) = |B| \cdot |B'| \tag{5.1}$$

and dually, for the extent $A \subseteq G$:

$$a_S(A, K) = |A| \cdot |A'|. \tag{5.2}$$

where A' is the set of attributes common to all the objects in A and B' is the set of objects possessing the attributes in B as in definition 2.

We will speak of the strict concept area as a kind of **cover**: raw weight of a concept for a certain context. We will introduce different kinds of covers later and we will use covers as a basis for calculating **measures** that evaluate concepts uniqueness for a certain context compared to another context.

If we calculate concepts uniqueness for the context K compared to the other context K_O then either concepts intent $B = B_O$ and contexts set of attributes $M =$

M_O or concepts extent $A = A_O$ and contexts set of objects $G = G_O$. That is, if we define concepts importance as a ratio between such measures then either attributes or objects cancel out. Then we divide concepts importance for the context K by the sum of its importance for K and K_O so our result will be in the 0..1 scale where 0.5 means that concept is equally important for both contexts, 1 means that the concept c is maximally important for context K as compared to the context K_O and 0 means the opposite.

Definition 16. *We define strict* ***ratio measure*** *for uniqueness r_S for the concept (A, B) from the context $K = (G, M, I)$ and compared to the context $K_O = (G_O, M_O, I_O)$ as follows:*

If $M = M_O$ then we take $A = B'$ for the context K and $A_O = B'$ for the context K_O

$$r_S(B, K, K_O) = \frac{\frac{a_S(B,K)}{|G|}}{\frac{a_S(B,K)}{|G|} + \frac{a_S(B,K_O)}{|G_O|}} = \frac{\frac{|A|}{|G|}}{\frac{|A|}{|G|} + \frac{|A_O|}{|G_O|}} \tag{5.3}$$

and dually, if $G = G_O$ then we take $B = A'$ for the context K and $B_O = A'$ for the context K_O

$$r_S(A, K, K_O) = \frac{\frac{a_S(A,K)}{|M|}}{\frac{a_S(A,K)}{|M|} + \frac{a_S(A,K_O)}{|M_O|}} = \frac{\frac{|B|}{|M|}}{\frac{|B|}{|M|} + \frac{|B_O|}{|M_O|}}. \tag{5.4}$$

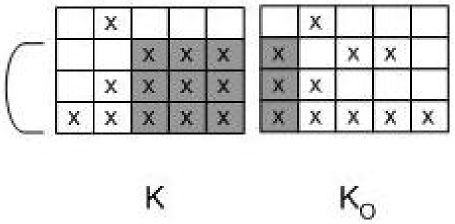


Figure 5.2: Contexts K, K_O and a concept $c = (A, B)$. Area covered by concept c that is relevant for calculating measure r_S is shaded.

For example, we can calculate the measure r_S for the concept $c = (A, B)$ shown in Figure 5.2 as

$$r_S(A, K, K_O) = \frac{\frac{3}{5}}{\frac{3}{5} + \frac{1}{5}} = \frac{3}{4}. \tag{5.5}$$

We can also give a general definition for the ratio measure

Definition 17. *We define* ***ratio measure*** *for uniqueness r for the concept (A, B) from the context $K = (G, M, I)$ and compared to the context $K_O = (G_O, M_O, I_O)$ as follows:*

$$r(B, K, K_O) = \frac{\frac{cover(B,K)}{total(K)}}{\frac{cover(B,K)}{total(K)} + \frac{cover(B,K_O)}{total(K_O)}} \tag{5.6}$$

and dually,

$$r(A, K, K_O) = \frac{\frac{cover(A,K)}{total(K)}}{\frac{cover(A,K)}{total(K)} + \frac{cover(A,K_O)}{total(K_O)}}. \tag{5.7}$$

where cover is measure for concept cover and total is measure for contexts size.

If we calculate r_S for an intent B containing many attributes (or dually, for an extent A containing many objects) then it is likely that $r_S = 1$ as G_O might not contain any objects that match to B. Furthermore, such a measure ignores all objects in G_O that almost match to B. One possible measure that tries to fix those problems is inspired by the monotone frequency based weight function (see Definition 8) and we denote it here as a measure r_L. We define loose concept area a_L as follows:

Definition 18. *For the context $K = (G, M, I)$ and the intent $B \subseteq M$:*

$$a_L(B, K) = \sum_{g \in G} |\{g\}' \cap B| \tag{5.8}$$

and dually, for the extent $A \subseteq G$:

$$a_L(A, K) = \sum_{m \in M} |\{m\}' \cap A| \tag{5.9}$$

Loose concept area a_L is similar to frequency based monotone weight function as defined in Definition 8.

Definition 19. *We define the measure of uniqueness r_L for the concept (A, B) from the context $K = (G, M, I)$ and compared to the context $K_O = (G_O, M_O, I_O)$ as follows:*

If $M = M_O$:

$$r_L(B, K, K_O) = \frac{\frac{a_L(B,K)}{|G|}}{\frac{a_L(B,K)}{|G|} + \frac{a_L(B,K_O)}{|G_O|}}. \tag{5.10}$$

and dually, if $G = G_O$:

$$r_L(A, K, K_O) = \frac{\frac{a_L(A,K)}{|M|}}{\frac{a_L(A,K)}{|M|} + \frac{a_L(A,K_O)}{|M_O|}}. \tag{5.11}$$

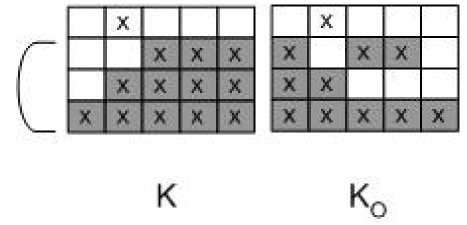


Figure 5.3: Contexts K, K_O and a concept $c = (A, B)$. Area a_L covered by concept c that is relevant for calculating measure r_L is shaded.

For example, we can calculate the measure r_L for the concept $c = (A, B)$ shown in Figure 5.3 as

$$r_L(A, K, K_O) = \frac{\frac{12}{5}}{\frac{12}{5} + \frac{10}{5}} = \frac{6}{11}. \tag{5.12}$$

5.3 Measure for evaluating concept as a classifier

We can treat concepts uniqueness as a hypotheses. And as in classical statistics we don't want to accept concept as unique when its uniqueness may be explained as a probable result of random noise. Not only should concept be relatively more important in one context then in another, it should also cover sufficient amount of the context to make it unlikely that its uniqueness is a result of random noise. A number of the measures I will propose are based on the probability measure for rule evaluation as described by Witten and Frank in their book about data mining [46]. That is we evaluate concepts accuracy as a classifier, if it is a good classifier between two contexts then we consider it unique for that context. Probability measure for rule evaluation is a probability of a completely random rule giving an equally good, or better, improvement in accuracy as the rule under consideration.

How to calculate the probability of a random rule performing as well or better as some classifier? As described by Witten and Frank [46], if we have a dataset that contains T examples (for our purposes that is usually number of examples in $K \cup K_O$), total number of instances of certain class K in that dataset P, total number of instances that the rule R selects t and number of instances of that class that the rule selects p then the probability that of t cases selected in random, exactly i are in class K is

$$Pr(t, i, K) = \frac{\binom{P}{i} \cdot \binom{T-P}{t-i}}{\binom{T}{t}}. \tag{5.13}$$

The probability that a random rule will do as well or better than rule R is

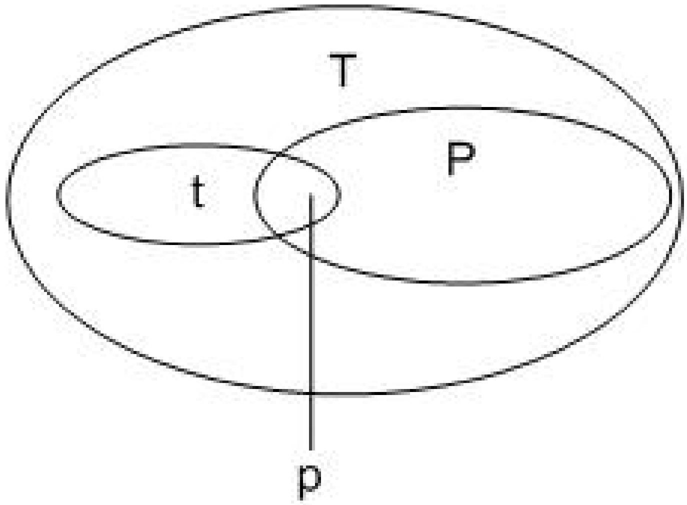


Figure 5.4: Sets T, P, t and p.

$$m(R) = \sum_{i=p}^{min(t,P)} Pr(t, i, K). \tag{5.14}$$

If the total number of instances T is large, a good approximation for the $Pr(t, i, K)$ is

$$Pr(t, i, K) = \binom{t}{i} \cdot \left(\frac{P}{T}\right)^{i} \cdot \left(1 - \frac{P}{T}\right)^{t-i}. \tag{5.15}$$

This probability is often approximated by an incomplete beta function $I_x(a, b)$ and that is what we use in our implementation [1]. Relation between incomplete beta function $I_x(a, b)$ and $m(R)$ approximation is

$$\sum_{i=p}^{t} \binom{t}{i} \cdot \left(\frac{P}{T}\right)^{i} \cdot \left(1 - \frac{P}{T}\right)^{t-i} = I_{\frac{P}{T}}(p, t - p + 1). \tag{5.16}$$

Now we define measure $m(R)$ in terms of contexts and concepts. We can relate total number of instances T in a dataset to total sizes of contexts K and K_O.

Here we usually use number of ones in the context as a measure of its total size if not mentioned otherwise. We could also use number of objects or attributes or their product, but the number of ones suits our area based approach better. In such a

[1] While Witten and Frank[46] recommend incomplete beta function for values of t greater than a dozen or so, my testing of Scientific Python library implementation revealed no approximation error even for values of t as low as 1.

case uniqueness of the concept is a measure of the concept being uniquely important to the system described by the context. That is, concept can be uniquely important to one context as compared to another even when both contexts object and attribute sets are equal and both concepts have equal extents and intents when one context contains less ones than another and therefore the concept makes up a bigger part of the system described by the context. P is related to the total size of context K. Author has to confess that the argument above feels somewhat vague. Evaluation of different methods for calculating contexts total size is a research area that could certainly benefit from future study.

Number of instances of a class p is related to the value of concepts cover over context K and difference $t - p$ is related to the value of concepts cover over context K_O.

Definition 20. *We define the* ***classifier measure*** *for concept similarity k as*

$$k(B, K, K_O) = I_{\frac{total(K)}{total(K)+total(K_O)}}(cover(B, K), cover(B, K_O) + 1) \tag{5.17}$$

and dually,

$$k(A, K, K_O) = I_{\frac{total(K)}{total(K)+total(K_O)}}(cover(A, K), cover(A, K_O) + 1). \tag{5.18}$$

In case of the ratio measure r high values of measure are interpreted as a sign of concepts uniqueness for context K as opposed to context K_O. In case of the classifier measure k low values of measure are interpreted as a sign of concepts uniqueness for context K as opposed to context K_O as they indicate a low probability for generating a better random classifier.

A simple way to define classifier measure is to use the set of objects in contexts as total and concepts extent as a cover.

Definition 21. *We define* ***simple classifier measure*** *for uniqueness $\hat{k}$ for the concept (A, B) from the context $K = (G, M, I)$ and compared to the context $K_O = (G_O, M_O, I_O)$ as follows:*

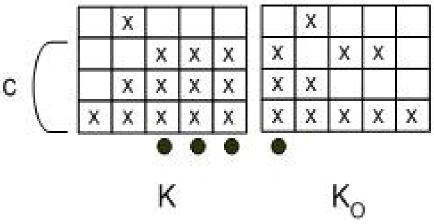
If $M = M_O$ then we take $A = B'$ for the context K and $A_O = B'$ for the context K_O

$$\hat{k}(B, K, K_O) = I_{\frac{|G|}{|G|+|G_O|}}(|A|, |A_O| + 1) \tag{5.19}$$

and dually, if $G = G_O$ then we take $B = A'$ for the context K and $B_O = A'$ for the context K_O

$$\hat{k}(A, K, K_O) = I_{\frac{|M|}{|M|+|M_O|}}(|B|, |B_O| + 1). \tag{5.20}$$

For example, we can calculate the measure $\hat{k}$ for the concept $c = (A, B)$ shown in Figure 5.5 as follows:



	X					X			
		X	X	X	X		X	X	
	X	X	X	X	X	X			
X	X	X	X	X	X	X	X	X	X
		●	●	●	●				

Figure 5.5: Contexts K, K_O and a concept $c = (A, B)$. Columns included in $|B|, |B_O|$ are marked with black dots.

$$\hat{k}(A, K, K_O) = I_{\frac{5}{10}}(3, 1 + 1) = 0.3125 \tag{5.21}$$

As we will see in the later section dealing with our case study, simple classifier measure is not symmetrical as $\hat{k}(B, K, K_O)$ will tend to have much lower values for single attribute intents than single object extents. So by design $\hat{k}(B, K, K_O)$ will never evaluate the concept of a single regional center as unique. This is at odds with our area based MONOCLE approach and our other measures will be based on the concept area.

5.4 Weighted concept area

We now introduce the weighted concept area based cover calculations. General idea is, that each object g (or dually an attribute m) of a context has a weight w between 0..1 that denotes its similarity to the concept (A, B). Such a weight is dependent of the size of concepts intent B (dually and extent) N and the size of intersection between g' and intent B that we denote n (dually an m' and A). Concepts area a is then defined simply as

$$a = \sum_{g \in G} |\{g\}' \cap B| \cdot w(|\{g\}' \cap B|, |B|) = \sum_{g \in G} n \cdot w(n, N) \tag{5.22}$$

and dually,

$$a = \sum_{m \in M} |\{m\}' \cap A| \cdot w(|\{m\}' \cap A|, |A| = \sum_{m \in M} n \cdot w(n, N). \tag{5.23}$$

We can now define previously defined strict and loose concept covers as weighted concept area based covers with trivial weight functions.

Definition 22.

$$w_S(n,N) = \begin{cases} 0: & n < N \\ 1: & n = N \end{cases} \tag{5.24}$$

$$a_S(B,K) = \sum_{g \in G} |\{g\}' \cap B| \cdot w_S(|\{g\}' \cap B|, |B|). \tag{5.25}$$

And dually

$$a_S(A,K) = \sum_{m \in M} |\{m\}' \cap A| \cdot w_S(|\{m\}' \cap A|, |A|). \tag{5.26}$$

Definition 23.

$$w_L(n,N) = 1. \tag{5.27}$$

$$a_L(B,K) = \sum_{g \in G} |\{g\}' \cap B| \cdot w_L(|\{g\}' \cap B|, |B|) = \sum_{g \in G} |\{g\}' \cap B|. \tag{5.28}$$

And dually

$$a_L(A,K) = \sum_{m \in M} |\{m\}' \cap A| \cdot w_L(|\{m\}' \cap A|, |A|) = \sum_{m \in M} |\{m\}' \cap A|. \tag{5.29}$$

We now introduce the **fuzzy concept cover** $\bar{a}_p$ that has non-trivial weight function which is basically a probability of obtaining worse match to the concept by generating data in random. First we define probability p for having a relation gIm (1 in the data table).

Definition 24. *We denote by* p *the probability of having a relation* gIm *between* g *and* m *for* $m \in B$ *for contexts* K *and* K_O*, or equivalently, relative frequency of 1's in the corresponding slice* $B \times G$ *of the data tables.*

$$p = \frac{|\{gIm \mid m \in B\}| + |\{gI_Om \mid m \in B\}|}{(|G| + |G_O|) \cdot |B|}. \tag{5.30}$$

And dual case, if we use concepts extent A*:*

$$p = \frac{|\{gIm \mid g \in A\}| + |\{gI_Om \mid g \in A\}|}{(|M| + |M_O|) \cdot |A|}. \tag{5.31}$$

Definition 25. *According to the Wolfram MathWorld [42] the probability of obtaining more successes than the* n *observed in a binomial distribution is* $I_p(n+1, N-n)$*.*

$$\bar{w}_p(n,N) = \begin{cases} 1 - I_p(n, N-n+1) & : n < N \\ 1 & : n = N \end{cases} \tag{5.32}$$

$$\bar{a}_p(B,K) = \sum_{g \in G} |\{g\}' \cap B| \cdot \bar{w}_p(|\{g\}' \cap B|, |B|). \tag{5.33}$$

And dually

$$\bar{a}_p(A,K) = \sum_{m \in M} |\{m\}' \cap A| \cdot \bar{w}_p(|\{m\}' \cap A|, |A|). \tag{5.34}$$

We mostly combine fuzzy concept cover with a classifier based similarity measure and we denote such combination as $\bar{k}_p$.

Definition 26. *We define classifier based measure for concept similarity with fuzzy concept cover as $\bar{k}_p$.*

$$\bar{k}_p(B,K,K_O) = I_{\frac{total(K)}{total(K)+total(K_O)}}(\bar{a}_p(B,K), \bar{a}_p(B,K_O)+1) \tag{5.35}$$

and dually,

$$\bar{k}_p(A,K,K_O) = I_{\frac{total(K)}{total(K)+total(K_O)}}(\bar{a}_p(A,K), \bar{a}_p(A,K_O)+1). \tag{5.36}$$

Measures k_L and k_S for concept covers a_L and a_S are defined by the same pattern.

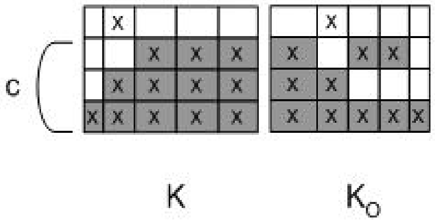


Figure 5.6: Contexts K, K_O and a concept $c = (A,B)$. Area $\bar{a}_p$ covered by concept c that is relevant for calculating measure $\bar{k}_p$ is shaded. Different column widths illustrate different weights $\bar{w}_p$.

For example, we can calculate the measure $\bar{a}_p$ for the concept $c = (A,B)$ shown in Figure 5.6 as follows:

First we calculate p:

$$p = \frac{12+10}{(5+5)\cdot 3} = \frac{11}{15}. \tag{5.37}$$

For this example we need weights $\bar{w}_p$ only for three cases:

$$\bar{w}_{\frac{11}{15}}(1,3) = 1 - I_{\frac{11}{15}}(1, 3-1+1) = 1 - 0.98 \approx 0.02$$
$$\bar{w}_{\frac{11}{15}}(2,3) = 1 - I_{\frac{11}{15}}(2, 3-2+1) = 1 - 0.82 \approx 0.18$$
$$\bar{w}_{\frac{11}{15}}(3,3) = 1$$

Now we can calculate area $\bar{a}_p(A, K)$

$$\begin{aligned} \bar{a}_{\frac{11}{15}}(A, K) = 3 \cdot 1 + \\ 3 \cdot 1 + \\ 3 \cdot 1 + \\ 2 \cdot 0.18 + \\ 1 \cdot 0.02 = 9.38 \end{aligned} \tag{5.38}$$

and area $\bar{a}_p(A, K_O)$

$$\begin{aligned} \bar{a}_{\frac{11}{15}}(A, K_O) = 3 \cdot 1 + \\ 2 \cdot 0.18 + \\ 2 \cdot 0.18 + \\ 2 \cdot 0.18 + \\ 1 \cdot 0.02 = 4.1 \end{aligned} \tag{5.39}$$

Now we can calculate the measure $\bar{k}_p(A, K, K_O)$. We use number of ones as a measure for the total size of a context giving us 13 for K and 24 for the combination of K and K_O.

$$\bar{k}_{\frac{11}{15}}(A, K, K_O) = I_{\frac{13}{24}}(9.38, 4.1 + 1) \approx 0.195 \tag{5.40}$$

Our classifier measure 0.195 indicates that concept c is somewhat unique for the context K as compared to K_O. Value 0.195 falls short from the 0.05 threshold commonly used in statistics so we cannot conclude uniqueness with certainty.

5.5 General, entropy based measure

In the previous section we presented the idea of a weighted area concept cover and also three methods for calculating it: strict area, loose area and fuzzy concept cover. Strict area gives weight 1 for all the rows that match our concept fully and weight 0 for all the other rows. Loose area gives weight 1 for all the rows. Fuzzy concept cover is somewhere in between of these two extremes having weights increase with the quality of the match, up to 1 in the case of a full match. While it would nice to have one best measure, author of this thesis is not aware of any criteria according which we could make such a decision. There is also no reason to believe that fuzzy concept cover is only possible correct measure between those extremes. Still, it is inconvenient to use a lot of different, unrelated measures.

In this section we present a new measure whose behavior depends of a real valued parameter changing from 0.5 to 1.0. For the value of 0.5 it behaves like a loose cover, for the value 1.0 it behaves like strict cover, having its behavior change gradually between those two extremes as the parameter increases. That way we are able to use one method for our concept cover calculation though instead of a single valued result we will get a sequence of values. It may turn out that all values indicate approximately same level of concept uniqueness. If not, we can clearly see what kind of results we get from strict cover - like and loose cover - like behavior ranges.

Our cover calculation is based on the Shannon information entropy [29] for two possible values. In our case Shannon information entropy measures an uncertainty in having a particular attribute value in the row match the concept (dually object value in the column match the concept).

Definition 27. *Shannon information entropy [29] H(p1, p1) for two possible values with probabilities p1 and p2 is*

$$H(p1, p2) = -p1 \cdot log(p1) - p2 \cdot log(p2) \tag{5.41}$$

We now define entropy based weight function w_u for upper bound u.

Definition 28. *Entropy based weight function* $w_u(n, N)$ *for upper bound u where* $0.5 \geq u < 1$ *is*

$$w_u(n, N) = \begin{cases} H(u, 1-u)/H(0.5, 0.5) & : \frac{n}{N} < 0.5 \\ H(u, 1-u)/H(\frac{n}{N}, 1-\frac{n}{N}) & : 0.5 \geq \frac{n}{N} \leq u \\ 1 & : \frac{n}{N} > u \end{cases} \tag{5.42}$$

Behavior of the function w_u is illustrated by Figures 5.7, 5.5, 5.9 [2] . Those figures show the relation between w_u, w_L and w_S. We want to prove formally that w_u behaves like w_L when $u = 0.5$ and that w_u behaves like w_S when the value of u approaches 1.

Theorem 7 (Equivalence of $w_{0.5}$ and w_L). *For any* $n, N \geq 0$ *where* $n \leq N$

$$w_{0.5}(n, N) = w_L(n, N). \tag{5.43}$$

[2] These figures were generated with DISLIN scientific data plotting library [2].

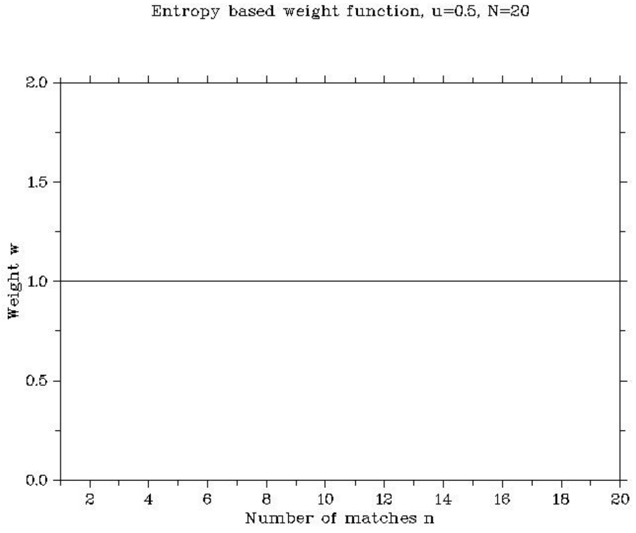


Figure 5.7: Weight function $w_{0.5}$ is equal to w_L having constant value 1.

Proof. Proof is trivial.

From Definition 28 we get when $u = 0.5$

$$w_{0.5}(n, N) = \begin{cases} H(0.5, 1 - 0.5)/H(0.5, 0.5) & : \frac{n}{N} < 0.5 \\ H(0.5, 1 - 0.5)/H(\frac{n}{N}, 1 - \frac{n}{N}) & : 0.5 \geq \frac{n}{N} \leq 0.5 \\ 1 & : \frac{n}{N} > 0.5 \end{cases} \tag{5.44}$$

That, we can simplify into

$$w_{0.5}(n, N) = \begin{cases} H(0.5, 0.5)/H(0.5, 0.5) & : \frac{n}{N} < 0.5 \\ H(0.5, 0.5)/H(0.5, 0.5) & : \frac{n}{N} = 0.5 \\ 1 & : \frac{n}{N} > 0.5 \end{cases} \tag{5.45}$$

And finally, using Definition 23

$$w_{0.5}(n, N) = w_L(n, N) = 1. \tag{5.46}$$

□

Theorem 8 (Equivalence of $\lim_{u \to 1} w_u$ and w_S)**.** *For any finite integers $n, N \geq 0$ where $n \leq N$*

$$\lim_{u \to 1} w_u(n, N) = w_S(n, N). \tag{5.47}$$

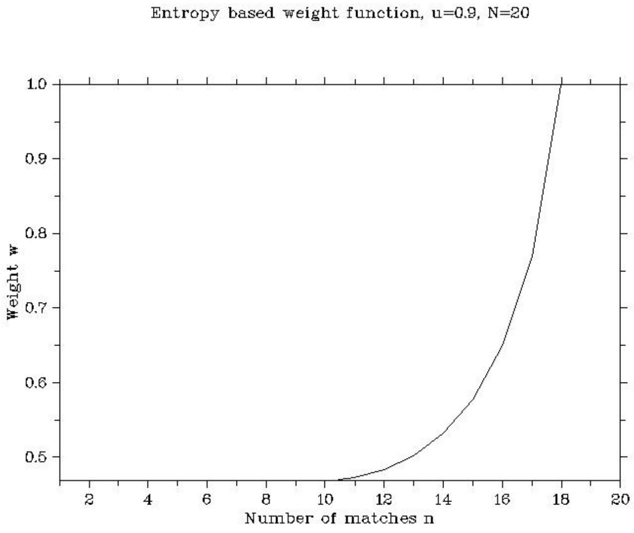


Figure 5.8: Weight function $w_{0.9}$ has value 1 for $n > 18$. For lower values of n, the value of function gradually decreases.

Proof. From Definition 28 we get when $\lim_{u\to 1}$

$$\lim_{u\to 1} w_u(n,N) = \begin{cases} \lim_{u\to 1} H(u,1-u)/H(0.5,0.5) & : \frac{n}{N} < 0.5 \\ \lim_{u\to 1} H(u,1-u)/H(\frac{n}{N},1-\frac{n}{N}) & : 0.5 \geq \frac{n}{N} \leq u \\ 1 & : \frac{n}{N} > u \end{cases} \tag{5.48}$$

From Definition 27, because $log(1) = 0$ and it is know that $\lim_{x\to 0} x \cdot log(x) = 0$

$$\lim_{u\to 1} H(u,1-u) = \lim_{u\to 1}(-u \cdot log(u) - (1-u) \cdot log(1-u)) = 0 \tag{5.49}$$

If n and N are finite then

$$\lim_{u\to 1} w_u(n,N) = \begin{cases} 0 & : \frac{n}{N} < 0.5 \\ 0 & : 0.5 \geq \frac{n}{N} \leq u \\ 1 & : \frac{n}{N} > u \end{cases} \tag{5.50}$$

And finally, using Definition 22

$$\lim_{u\to 1} w_u(n,N) = w_S(n,N) \tag{5.51}$$

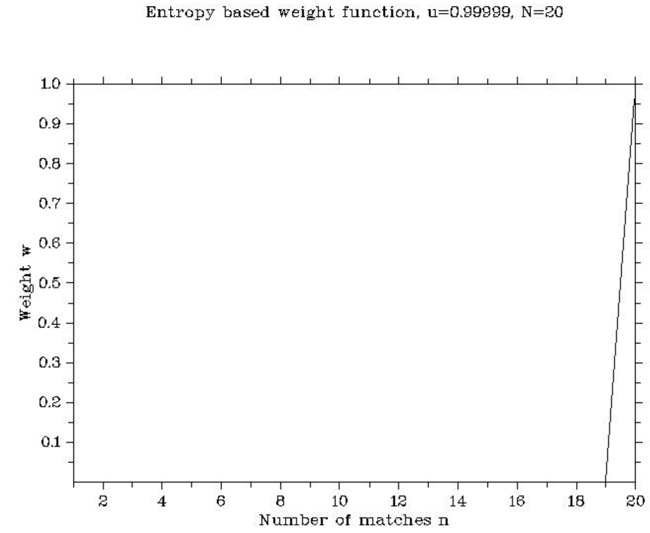


Figure 5.9: Weight function $w_{0.99999}$ approximates w_S having value 1 when $n = N$ and value close to 0 in lower range.

for any finite integers $n, N \geq 0$ where $n \leq N$.

□

We calculate cover a_u as a weighted concept area.

$$a_u(B,K) = \sum_{g \in G} |\{g\}' \cap B| \cdot w_u(|\{g\}' \cap B|, |B|) = \sum_{g \in G} |\{g\}' \cap B|. \tag{5.52}$$

And dually

$$a_u(A,K) = \sum_{m \in M} |\{m\}' \cap A| \cdot w_u(|\{m\}' \cap A|, |A|) = \sum_{m \in M} |\{m\}' \cap A|. \tag{5.53}$$

Definition 29. *We define classifier measure for concept similarity with entropy based concept cover as k_u.*

$$k_u(B, K, K_O) = I_{\frac{total(K)}{total(K)+total(K_O)}}(a_u(B,K), a_u(B,K_O) + 1) \tag{5.54}$$

and dually,

$$k_u(A, K, K_O) = I_{\frac{total(K)}{total(K)+total(K_O)}}(a_u(A,K), a_u(A,K_O) + 1). \tag{5.55}$$

As all those definitions follow the same pattern as corresponding definitions for strict and loose weight functions, it is trivial to show that $a_{0.5}$ and $k_{0.5}$ are equivalent to a_L and k_L and $\lim_{u\to 1} a_u$ and $\lim_{u\to 1} k_u$ are equivalent to a_S and k_S.

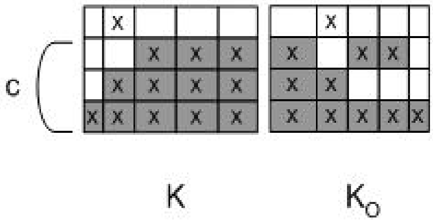


Figure 5.10: Contexts K, K_O and a concept $c = (A, B)$. Area $a_{0.9}$ covered by concept c that is relevant for calculating measure $k_{0.9}$ is shaded. Different column widths illustrate different weights $w_{0.9}$.

For example, we can calculate the measure $a_{0.9}$ for the concept $c = (A, B)$ shown in Figure 5.10 as follows:

For this example we need weights $w_{0.9}$ only for three cases:

$$\begin{aligned} w_{0.9}(1,3) &= H(0.9, 0.1)/H(0.5, 0.5) \approx 0.33/0.69 \approx 0.48 \\ w_{0.9}(2,3) &= H(0.9, 0.1)/H(\frac{2}{3}, \frac{1}{3}) \approx 0.33/0.64 \approx 0.51 \\ w_{0.9}(3,3) &= 1 \end{aligned}$$

Now we can calculate area $a_{0.9}(A, K)$

$$\begin{aligned} a_{0.9}(A, K) \approx 3 \cdot 1+ & \\ 3 \cdot 1+ & \\ 3 \cdot 1+ & \\ 2 \cdot 0.51+ & \\ 1 \cdot 0.48 = 10.5 & \end{aligned} \tag{5.56}$$

and area $a_{0.9}(A, K_O)$

$$\begin{aligned} a_{0.9}(A, K_O) \approx 3 \cdot 1+ & \\ 2 \cdot 0.51+ & \\ 2 \cdot 0.51+ & \\ 2 \cdot 0.51+ & \\ 1 \cdot 0.48 = 6.54 & \end{aligned} \tag{5.57}$$

Now we can calculate the measure $k_{0.9}(A, K, K_O)$. We use number of ones as a measure for the total size of a context giving us 13 for K and 24 for the combination of K and K_O.

$$k_{0.9}(A, K, K_O) \approx I_{\frac{13}{24}}(10.5, 6.54 + 1) \approx 0.357 \tag{5.58}$$

Our classifier measure 0.357 indicates that concept c seems slightly unique for the context K as compared to K_O. Value 0.357 falls clearly short from the 0.05 threshold commonly used in statistics so we cannot conclude uniqueness with any certainty.

5.6 Comparison between Saaremaa and Hiiumaa

We now return to our case study of Saaremaa and Hiiumaa. While previous results regarding the case study were achieved in cooperation with Karin Lindroos, these results were achieved solely by the author. We take 17 top concepts as sorted by our MONOCLE method and calculate a number of measures for one island as compared to another. We mostly use classifier measures k but we also add ratio measures r_S and r_L for comparison. For ratio measures r uniqueness is shown by high values, for classifier measures k uniqueness is shown by low values. We try to evaluate agreement between different measures and concept uniqueness for islands. Concepts that are not clearly measured as unique and values of measures that disagree with other measures are bolded in the following tables.

Legend for Hiiumaa:

1. KrKn: {Kärdla, Käina}, (58 attributes)
2. s: (68 settlements), {summer cabins}
3. Kn: {Käina}, (83 attributes)
4. Kr: {Kärdla}, (101 attributes)
5. E: {Emmaste}, (41 attributes)
6. sb: (17 settlements), {summer cabins, beach}
7. KrKnE: {Kärdla, Käina, Emmaste}, (25 attributes)
8. a: (22 settlements), {agriculture}
9. h: (32 settlements), {housing}
10. KnE: {Käina, Emmaste}, (27 attributes)
11. KrE: {Kärdla, Emmaste}, (28 attributes)

Concept	r_L	r_S	$\hat{k}$	$\bar{k}_p$	$k_{0.5} = k_L$	$k_{0.75}$	$k_{0.9}$	$k_1 = k_S$
KrKn	1.0	0.5994	0.075	0.0001	0.0002	0.0	0.0	0.0
s	0.6576	0.6576	0.0001	0.0019	0.0019	0.0019	0.0019	0.0019
Kn	1.0	0.6017	0.2738	0.0	0.0	0.0	0.0	0.0
Kr	1.0	0.588	0.2738	0.0	0.0002	0.0001	0.0	0.0
E	1.0	0.5763	0.2738	0.0273	0.0298	0.0138	0.0088	0.0
sb	0.6927	0.6396	0.012	0.0027	0.002	0.0024	0.0034	0.0038
KrKnE	1.0	0.5917	0.0205	0.0112	0.0099	0.0024	0.0	0.0
a	0.5148	0.5148	**0.4501**	0.6865	0.6865	0.6865	0.6865	0.6865
h	0.6025	0.6025	0.0411	0.1426	0.1426	0.1426	0.1426	0.1426
KnE	1.0	0.6007	0.075	0.0025	0.0026	0.0005	0.0	0.0
KrE	1.0	0.5928	0.075	0.0085	0.0075	0.0014	0.0	0.0
KrKgKn	1.0	0.5927	0.0205	0.0165	0.0228	0.0094	0.0004	0.0
N	1.0	0.5901	0.2738	0.0303	0.0174	0.0188	0.0084	0.0
KgKn	1.0	0.5991	0.075	0.0075	0.0111	0.0034	0.0003	0.0
sh	0.7006	0.6379	0.0144	0.0023	0.0013	0.0018	0.0032	0.0044
b	0.5933	0.5933	0.1008	0.237	0.237	0.237	0.237	0.237
KsKn	1.0	0.6156	0.075	0.0008	0.0012	0.0005	0.0002	0.0

Table 5.1: Evaluation of importance for top 17 concepts for Hiiumaa as compared to Saaremaa

12. KrKgKn: {Kärdla, Kõrgessaare, Käina}, (16 attributes)
13. N: {Nõmme}, (22 attributes)
14. KgKn: {Kõrgessaare, Käina}, (19 attributes)
15. sh: (15 settlements), {summer cabins, housing}
16. b: (22 settlements), {beach}
17. KsKn: {Kassari, Käina}, (17 attributes)

Legend for Saaremaa:

1. Ku: {Kuressaare}, (179 attributes)
2. f: (68 settlements), {landing places for fishing boats}
3. s: (87 settlements), {summer cabins}
4. KuO: {Kuressaare, Orissaare}, (52 attributes)
5. KuN: {Kuressaare, Nasva}, (47 attributes)

Concept	r_L	r_S	$\hat{k}$	$\bar{k}_p$	$k_{0.5} = k_L$	$k_{0.75}$	$k_{0.9}$	$k_1 = k_S$
Ku	1.0	**0.4701**	**0.7262**	0.1035	0.2556	0.1267	0.0077	0.0
f	0.8368	0.8368	0.0	0.0	0.0	0.0	0.0	0.0
s	0.3424	0.3424	1.0	0.9989	0.9989	0.9989	0.9989	0.9989
KuO	1.0	0.5238	**0.5274**	0.0003	0.0005	0.0007	0.0001	0.0
KuN	1.0	0.551	**0.5274**	0.0	0.0	0.0	0.0	0.0
a	0.4852	0.4852	**0.6475**	0.4068	0.4068	0.4068	0.4068	0.4068
KuKl	1.0	0.5011	**0.5274**	0.0068	0.021	0.0172	0.0033	0.0
KuV	1.0	0.5415	**0.5274**	0.0	0.0001	0.0002	0.0001	0.0
fs	0.751	**0.4622**	0.0167	0.2682	**0.5267**	0.368	0.101	0.0001
N	1.0	0.5358	**0.7262**	0.0001	0.0002	0.0001	0.0001	0.0
KuL	1.0	0.5331	**0.5274**	0.0008	0.0015	0.0039	0.0014	0.0
O	1.0	0.5218	**0.7262**	0.0001	0.0004	0.0004	0.0001	0.0
KuKd	1.0	0.6035	**0.5274**	0.0	0.0	0.0	0.0	0.0
h	0.3975	0.3975	0.9753	0.902	0.902	0.902	0.902	0.902
g	0.3557	0.3557	0.9939	0.9715	0.9715	0.9715	0.9715	0.9715
KuOL	1.0	0.5494	0.383	0.0009	0.0005	0.0032	0.0018	0.0
KuVTKl	1.0	0.6053	0.2781	0.0	0.0	0.0	0.0	0.0

Table 5.2: Evaluation of importance for top 17 concepts for Saaremaa as compared to Hiiumaa

6. a: (55 settlements), {agriculture}
7. KuKl: {Kuressaare, Kärla}, (44 attributes)
8. KuV: {Kuressaare, Valjala}, (39 attributes)
9. fs: (32 settlements), {landing places for fishing boats, summer cabins}
10. N: {Nasva}, (54 attributes)
11. KuL: {Kuressaare, Liiva}, (35 attributes)
12. O: {Orissaare}, (58 attributes)
13. KuKd: {Kuressaare, Kudjape}, (30 attributes)
14. h: (56 settlements), {housing}
15. g: (41 settlements), {sights}
16. KuOL: {Kuressaare, Orissaare, Liiva}, (24 attributes)
17. KuVTKl: {Kuressaare, Valjala, Tornimäe, Kärla}, (17 attributes)

As we can see, simple classifier measure $\hat{k}$ accounts for most of the disagreements with other measures. Most of those disagreements are in measures for concepts describing large settlements in Saaremaa. That is because $\hat{k}(B, K, K_O)$ counts only size of extent and size of intent is invisible for it and context for Saaremaa has more objects than Hiiumaa and it is quite likely that random rule picks couple of objects form Saaremaa and none from Hiiumaa. If we calculate classifier measure based on concept area then that probability changes. For our case study, measure $\hat{k}$ seems to be problematic because it discriminates systematically against concepts describing Saaremaa. Measure $\hat{k}$ also disagrees with other measures over the uniqueness of concept a (agricultural settlements) for some reason. Only other disagreement is over uniqueness of concept fs (landing places for fishing boats, summer houses) from the context of Saaremaa. According to loose measures r_L and k_L this concept is not unique for Saaremaa, according to stricter measures it is. That seems to be a legitimate clash between different definitions of similarity. Similar behavior is present in the concept Ku describing Kuressaare, capital of Saaremaa. Kuressaare is classified as clearly unique by strict measures but loose measures r_L classifies it not unique and measure k_L classifies it as slightly but not clearly unique as value 0.26 is clearly above 0.05 limit common in statistics.

Most of the top concepts generated by MONOCLE are unique, only concepts not unique are concept a (agriculture) for Hiiumaa and concepts s (summer houses), h (housing), g (sights) for Saaremaa.

Entropy based measure k_u is in agreement with other measures, k_L and k_S agree well with r_L and r_S and fuzzy classifier measure $\bar{k}_p$ is mostly within minimum and maximum values of k_u, except for concepts KrKnE, KrE, N for Hiiumaa, and even then it is not very far off. It seems sensible to confine our concept similarity analysis to measure k_u for greater ease and simplicity.

5.7 Visual analysis with entropy based measure

We can plot the values of k_u for the different values of u. Such a plot allows us quickly detect if there is any significant difference in the measures over the range 0.5...1 for u. For our case study, it turns out that measures are mostly in good agreement.

Plots for some pairs of important concepts are given below.

In Figures 5.11 and 5.12 we can see the plots for regional centers of Hiiumaa and Saaremaa, Kärdla and Kuressaare. We can see that both measures decrease monotonically as value of u increases. There are no exact matches for those objects in other context. Value of k_u stays always below 0.0002 for Kärdla, so it can be declared unique [3] with high confidence. For Kuressaare value of k_u decreases monotonically

[3] that is: declared unique for Hiiumaa as opposed to Saaremaa, but for shortness we will omit that from now on, if not necessary for clarity.

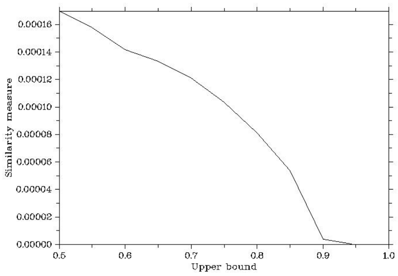


Figure 5.11: Hiiumaa: Kärdla (Kr), 101 attributes

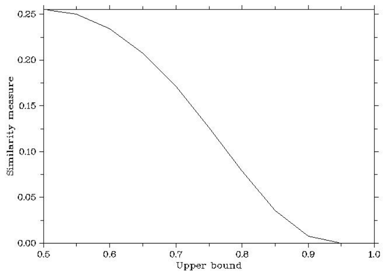


Figure 5.12: Saaremaa: Kuressaare (Ku), 179 attributes

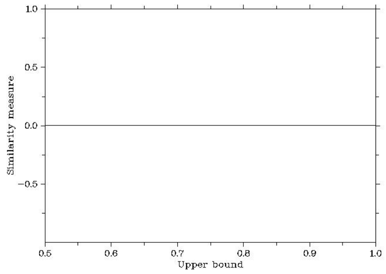


Figure 5.13: Hiiumaa: summer cabins (s), 63 objects

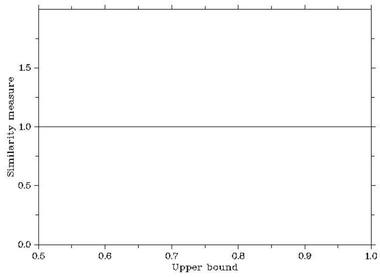


Figure 5.14: Saaremaa: summer cabins (s), 87 objects

from 0.25 to 0, reaching 0.05 at about $u = 0.85$. That confirms our deduction from previous section that Kuressaare is clearly unique by the strict measures ($u > 0.85$), but by looser measures ($u < 0.85$) it is slightly, but not clearly unique.

In Figures 5.13 and 5.14 we can see the plots for the concept *summer cabins* for Hiiumaa and Saaremaa. We can see that measures stay constant, because for one-attribute concept there can be only full match with maximum row weight and no match that multiplies row weight by zero. Measure k_u classifies that concept as clearly unique for Hiiumaa for the reason that there is not that much difference between objects covered by the concept when compared with the difference between total sizes of contexts.

In Figures 5.15 and 5.16 we can see the plots for the concepts corresponding to pairs of larger towns for Hiiumaa and Saaremaa. Measure k_u classifies both concepts as clearly unique for the entire range of u. Plot for concept Kuressaare, Orissaare is interesting as there is maximum near $u = 0.7$ instead of constant or monotonically

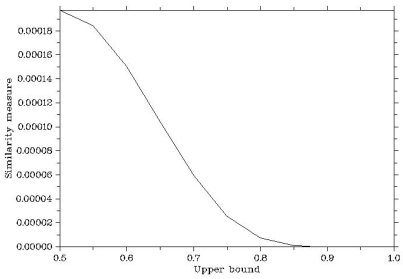


Figure 5.15: Hiiumaa: Kärdla, Käina (KrKn), 58 attributes

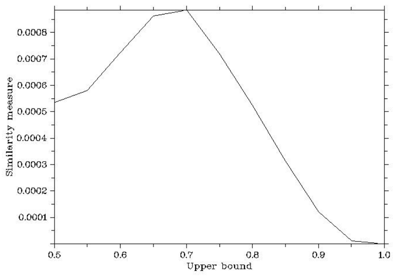


Figure 5.16: Saaremaa: Kuressaare, Orissaare (KuO), 52 attributes

decreasing curve.

For some concepts, it is also useful to plot the values of entropy based concept area a_u and different components that contribute to this area. Here we define the component as a subset of concepts intent and a particular object belongs to the component defined by the intersection of attribute values of the object and the concept. Sadly, this approach is unlikely to scale up well as the number of different subsets can grow exponentially. For our case study it works quite well and scaled up version of such a method is a possible area for future study.

Let us first look at the concept *summer cabins, beach* (sb). Figure 5.17 shows measure k_u and Figures 5.18 and 5.19 show concept cover a_u for Hiiumaa and Saaremaa. We can see that beaches without summer cabins are much more prevalent in Saaremaa than in Hiiumaa. That concept is also clearly unique for Hiiumaa. We can see that the cover of non-exact-match components exceeds the exact match at about $u = 0.85$ for both contexts.

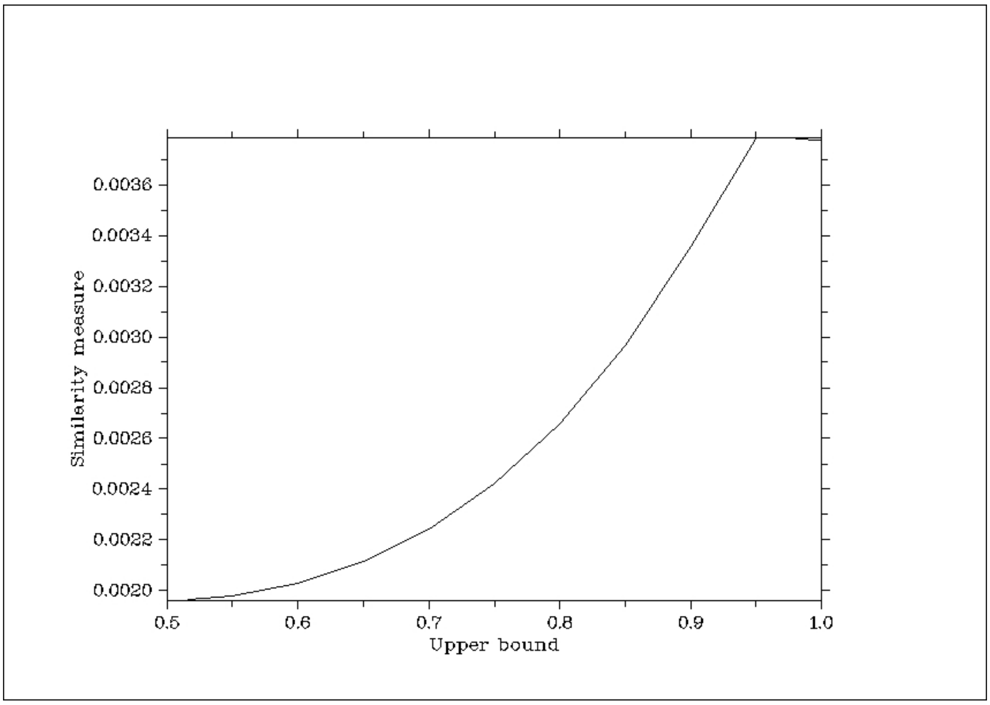


Figure 5.17: Concept summer cabins, beach (sb). Measure k_u for uniqueness for Hiiumaa.

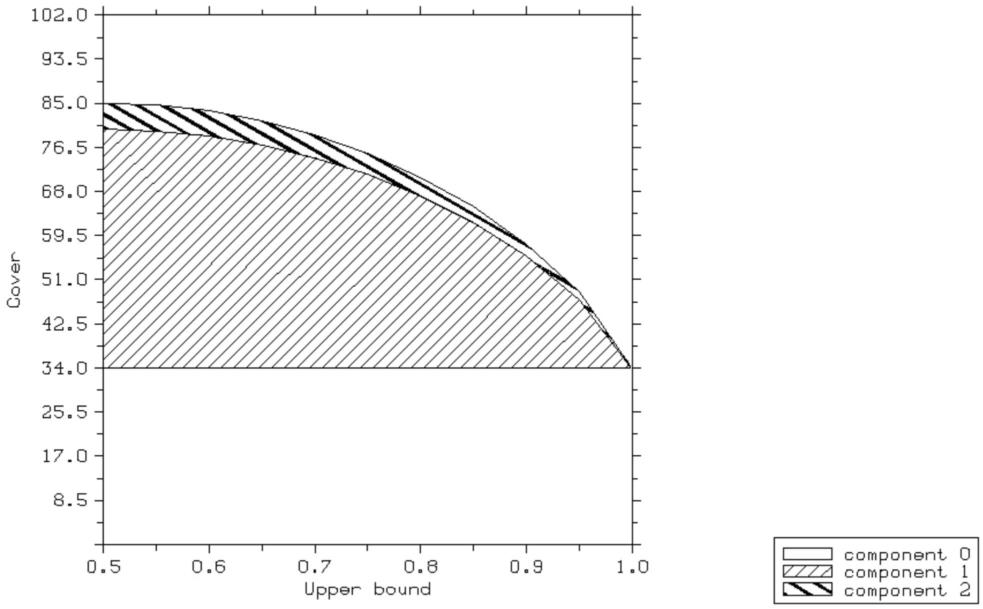


Figure 5.18: Concept summer cabins, beach. Measure a_u for concept cover for Hiiumaa. Component 0: summer cabins, beach; component 1: summer cabins; component 2: beach

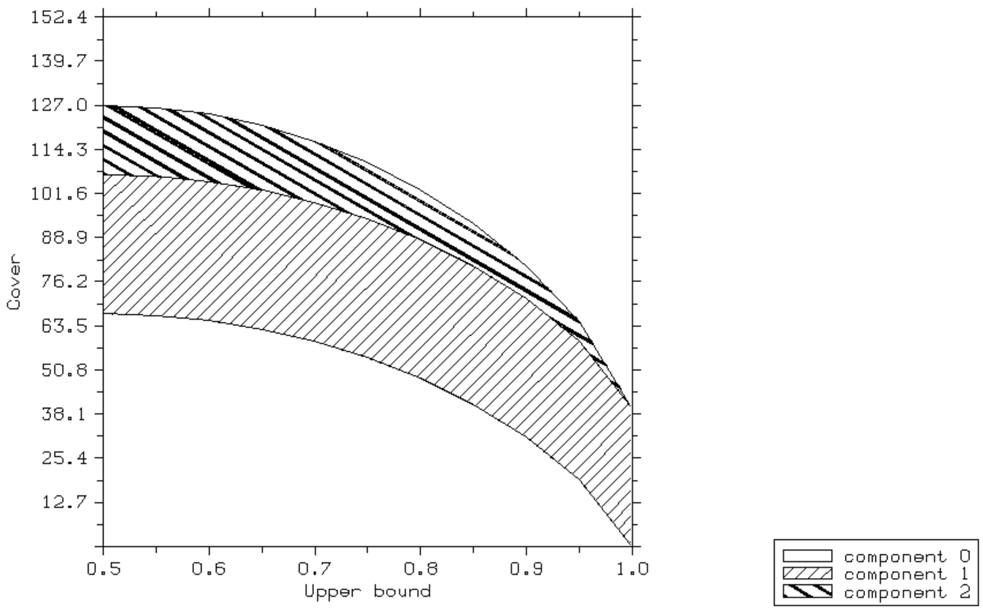


Figure 5.19: Concept summer cabins, beach. Measure a_u for concept cover for Saaremaa. Component 0: summer cabins; component 1: summer cabins, beach; component 2: beach

As mentioned in previous section and illustrated in Figure 5.20, the concept *landing places for fishing boats, summer cabins* (fs) is classified as unique for values of $u > 0.9$ but is not unique in lower range. Figures 5.21 and 5.22 show concept cover a_u for Saaremaa and Hiiumaa. We can see that the reason for difference between low and high values of k_u is lack of landing places for fishing boats that means low amount of cover for higher values of u combined with lots of summer houses that contribute to the cover when values of u are low.

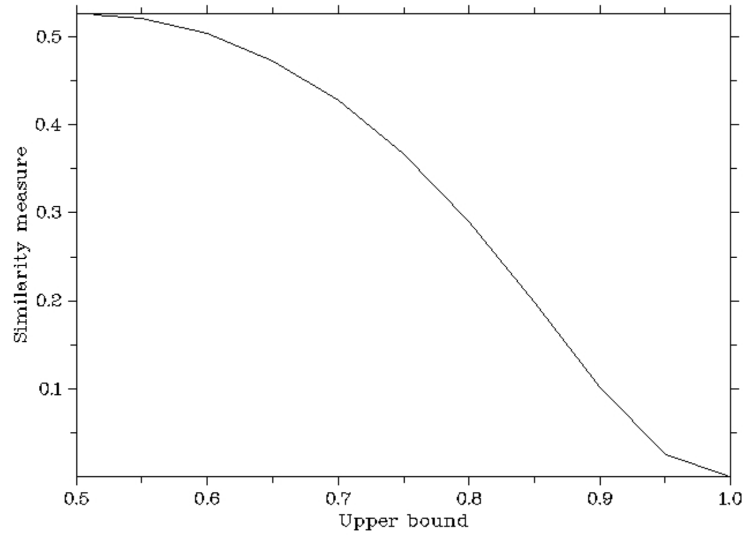


Figure 5.20: Concept for landing places for fishing boats, summer cabins (fs). Measure k_u for uniqueness for Saaremaa.

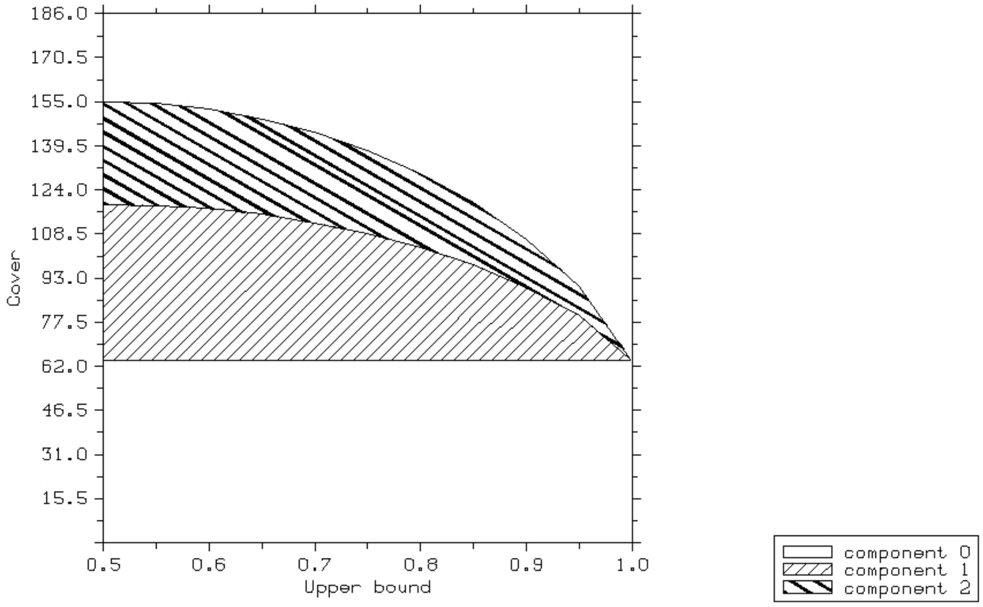


Figure 5.21: Concept: landing places for fishing boats, summer cabins. Measure a_u for concept cover for Saaremaa. Component 0: landing places for fishing boats, summer cabins; component 1: summer cabins; component 2: landing places for fishing boats

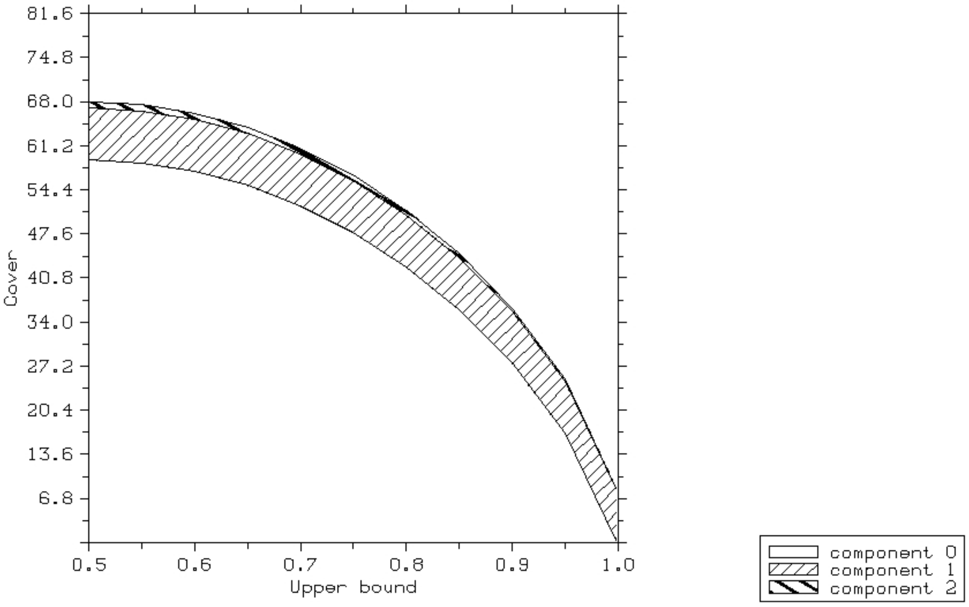


Figure 5.22: Concept: landing places for fishing boats, summer cabins. Measure a_u for concept cover for Hiiumaa. Component 0: summer cabins; component 1: landing places for fishing boats, summer cabins; component 2: landing places for fishing boats

We now turn to concepts that describe larger settlements. In a previous section we compared similarity plots for concepts describing Kärdla (Figure 5.11), Kuressaare (Figure 5.12), Kärdla combined with Käina (Figure 5.15), Kuressaare combined with Orissaare (Figure 5.16). Following are similarity plots for rest of the concepts describing larger settlements from the top 9 concepts as found by the MONOCLE method.

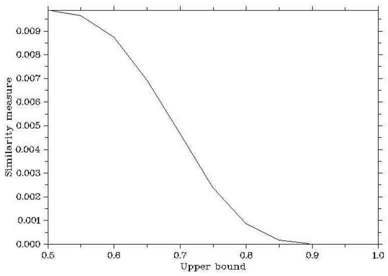


Figure 5.23: Hiiumaa: Kärdla, Käina, Emmaste (KrKnE), 25 attributes

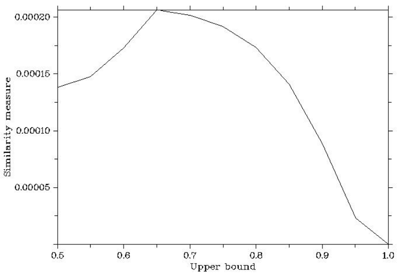


Figure 5.24: Saaremaa: Kuressaare, Valjala (KuV), 39 attributes

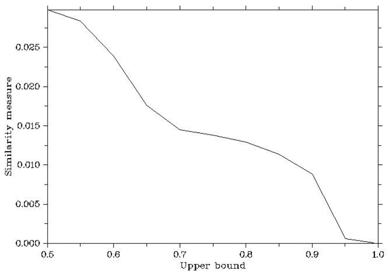


Figure 5.25: Hiiumaa: Emmaste (E), 41 attributes

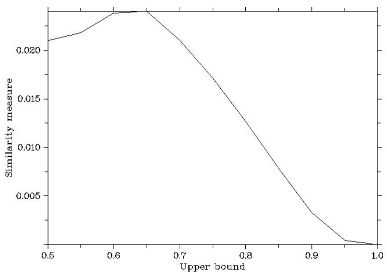


Figure 5.26: Saaremaa: Kuressaare, Kärla (KuKl), 44 attributes

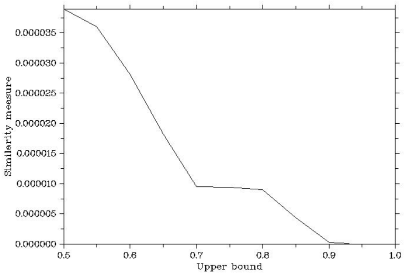


Figure 5.27: Hiiumaa: Käina (Kn), 83 attributes

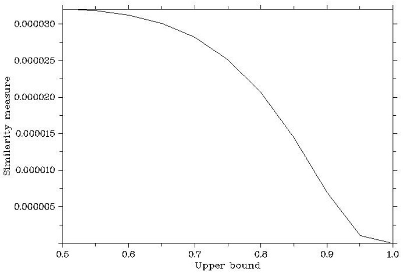


Figure 5.28: Saaremaa: Kuressaare, Nasva (KuN), 47 attributes

All those concepts, except the concept for Kuressaare (Ku), are clearly unique to their respective contexts. While highest value of k_u for Kuressaare is $k_{0.5} \approx 0.25$, the next highest value for k_u is $k_{0.5} \approx 0.025$ for the concept describing Emmaste (E), a tenfold difference. That is a curious difference between Kuressaare and rest of the settlements.

Lets examine the components of the concept cover for the concept Ku, describing Kuressaare.

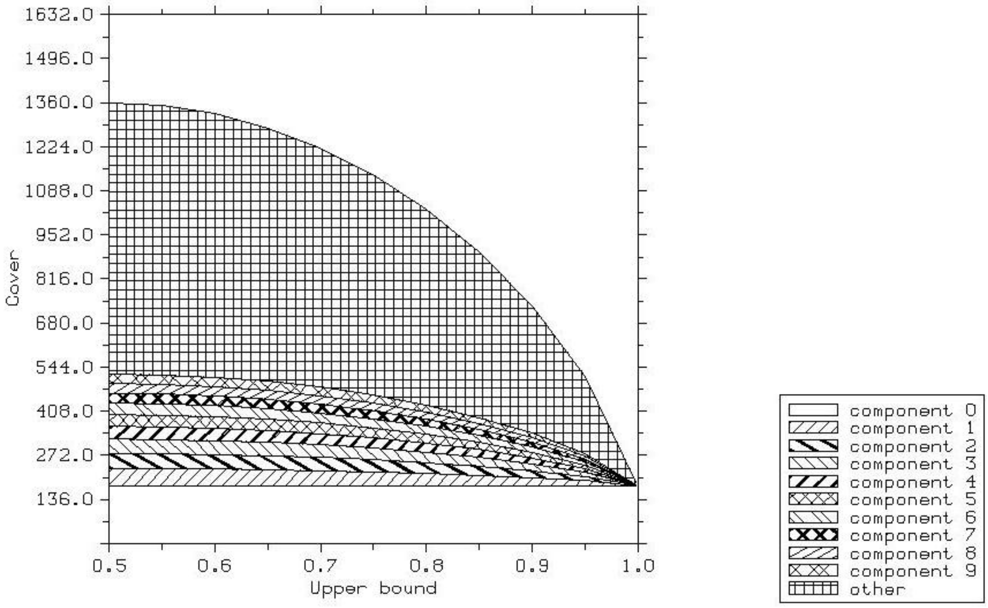


Figure 5.29: Kuressaare (Ku). Measure a_u for concept cover for Saaremaa. Component 0: Kuressaare, 179 attributes; component 1: Orissaare, 52 attributes match; component 2: Nasva, 47 attributes match; component 3: Kärla, 44 attributes match; component 4: Valjala, 39 attributes match; component 5: summer houses, 37 objects; component 6: Liiva, 35 attributes match; component 7: Tornimäe, 31 attributes match; component 8: Kudjape, 30 attributes match; component 9: Kihelkonna, 29 attributes match

We can see from Figures 5.29 and 5.30 that Kärdla and Käina from Hiiumaa match the intent of Kuressaare better with their 87 and 69 matching attributes than any settlement from Saaremaa, where best match is Orissaare with 52 attributes. That is quite remarkable as Hiiumaa is smaller socio-economic system. That is the reason why measure k_u does not show Kuressaare as clearly unique for Saaremaa. There is also a contribution from other large settlements of Hiiumaa. Much of the concept

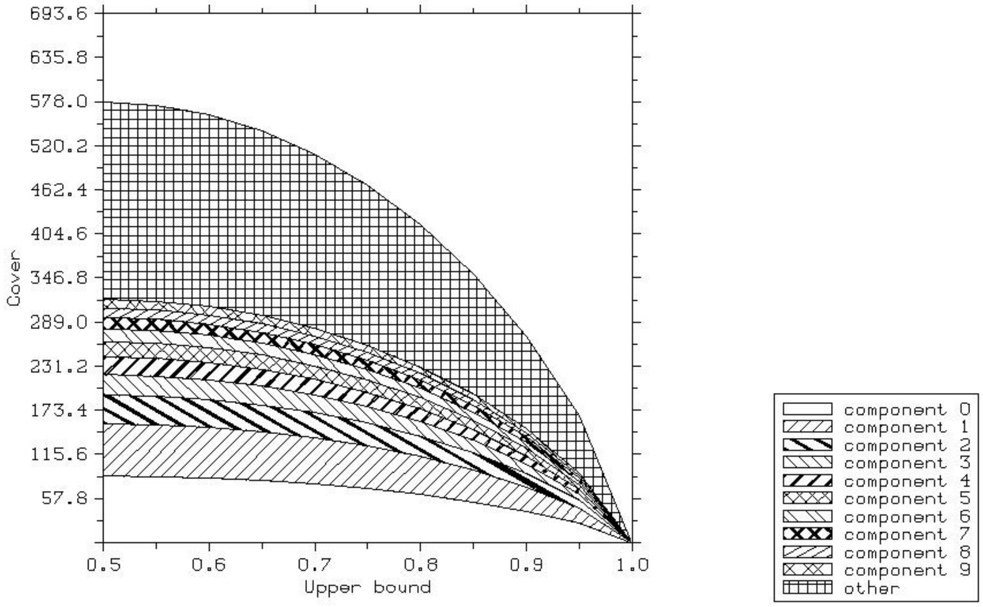


Figure 5.30: Kuressaare (Ku). Measure a_u for concept cover for Hiiumaa. Component 0: Kärdla, 87 attributes match; component 1: Käina, 69 attributes match; component 2: Emmaste, 37 attributes match; component 3: summer houses, 27 objects; component 4: Kõrgessaare, 23 attributes match; component 5: Kassari, 20 attributes match; component 6: Nõmme, 17 attributes match; component 7: Männamaa, 15 attributes match; component 8: Sõru, 12 attributes match; component 9: Palade, 12 attributes match

cover for Saaremaa is made up from the ”other” components- that is the mass of smaller settlements sharing some attributes with Kuressaare and from the Kuressaare itself.

Let us now examine the concept cover a_u of several other concepts describing large settlements: KuO, Kr, KrKnE.

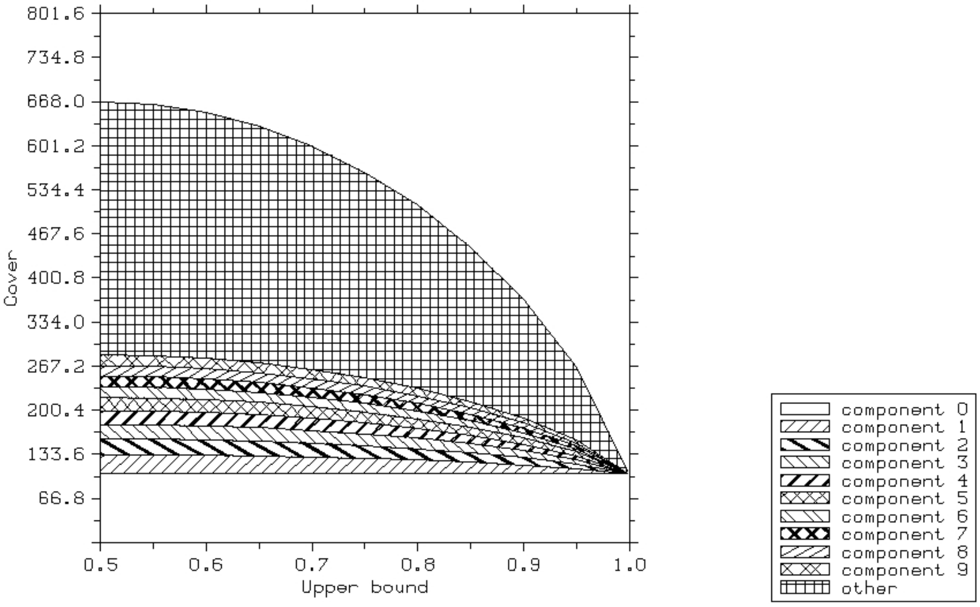


Figure 5.31: Kuressaare, Orissaare (KuO). Measure a_u for concept cover for Saaremaa. Component 0: Kuressaare, Orissaare, 52 attributes; component 1: housing, 29 objects; component 2: Liiva, 24 attributes match; component 3: Kärla, 22 attributes match; component 4: Valjala, 21 attributes match; component 5: Tornimäe, 20 attributes match; component 6: Nasva, 16 attributes match; component 7: Salme, 16 attributes match; component 8: Lümanda, 16 attributes match; component 9: Mustjala, 16 attributes match

Kärdla and Käina still match the concept better than anything from Saaremaa. However, difference in area covered by ”other” components is here much larger than for the concept Ku. Components 7-9 for Hiiumaa have already quite a small area, while components 7-9 for Saaremaa have stable area of 16. It seems that Orissaare has some attributes that are not widespread in Hiiumaa.

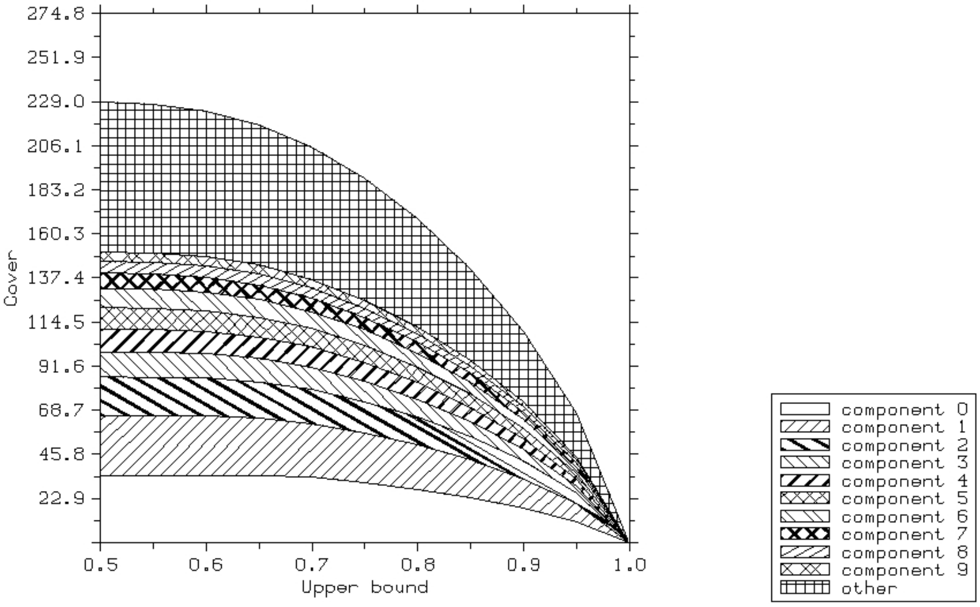


Figure 5.32: Kuressaare, Orissaare (KuO). Measure a_u for concept cover for Hiiumaa. Component 0: Kärdla, 35 attributes match; component 1: Käina, 31 attributes match; component 2: Emmaste, 20 attributes match; component 3: Kõrgessaare, 13 attributes match; component 4: housing, 12 objects; component 5: Kassari, 11 attributes match; component 6: beach, 10 objects; component 7: Suuremõisa, 8 attributes match; component 8: (Mangu, Sarve, Sääre), (housing, beach); component 9: Tärkma, 5 attributes match

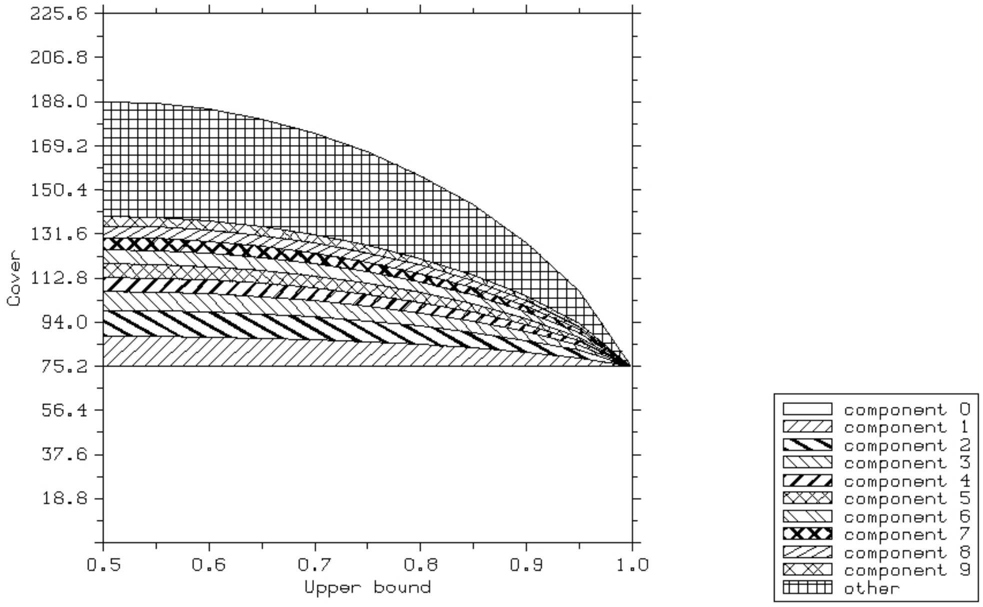


Figure 5.33: Kärdla, Käina, Emmaste (KrKnE). Measure a_u for concept cover for Hiiumaa. Component 0: Kärdla, Käina, Emmaste, 25 attributes; component 1: sights, 14 objects; component 2: Kõrgessaare, 11 attributes match; component 3: Suuremõisa, 8 attributes match; component 4: church, sights, 3 objects; component 5: Nurste, 6 attributes match; component 6: Kassari, 6 attributes match; component 7: Männamaa, 5 attributes match; component 8: Palade, 5 attributes match; component 9: music, arts, 4 objects

We find again that ”other” components contribute much more to concept cover for Saaremaa than for Hiiumaa. Component corresponding to the concept KrKnE itself is a big part of the concept cover at $a_{0.5}$, much bigger than Kuressaare was for its concept cover at $a_{0.5}$.

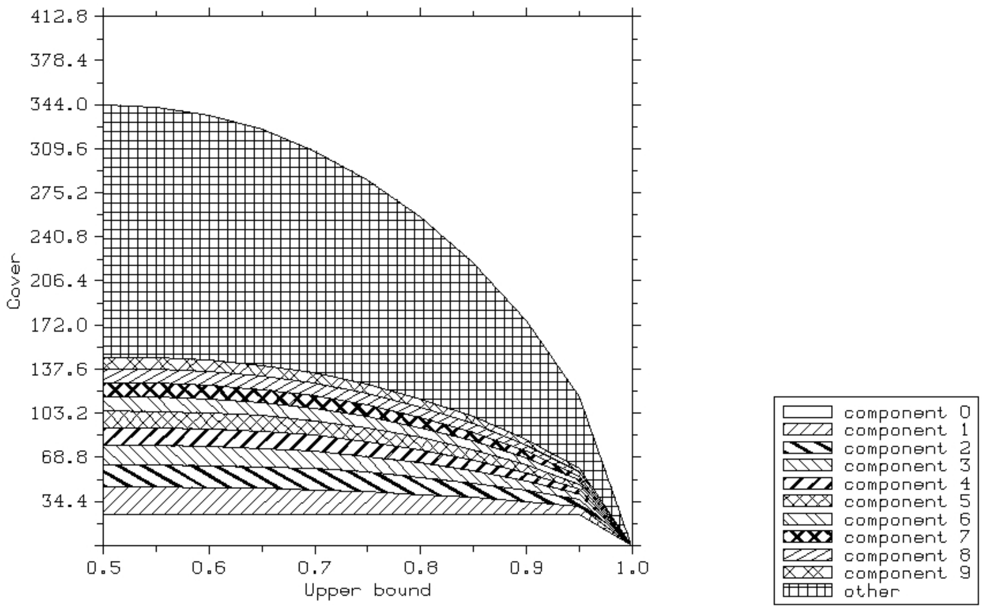


Figure 5.34: Kärdla, Käina, Emmaste (KrKnE). Measure a_u for concept cover for Saaremaa. Component 0: Kuressaare, 24 attributes match; component 1: sights, 22 objects; component 2: Kärla, 17 attributes match; component 3: Orissaare, 15 attributes match; component 4: Liiva, 14 attributes match; component 5: Kihelkonna, 13 attributes match; component 6: Valjala, 11 attributes match; component 7: Leisi, 11 attributes match; component 8: Tornimäe, 11 attributes match; component 9: Mustjala, 9 attributes match

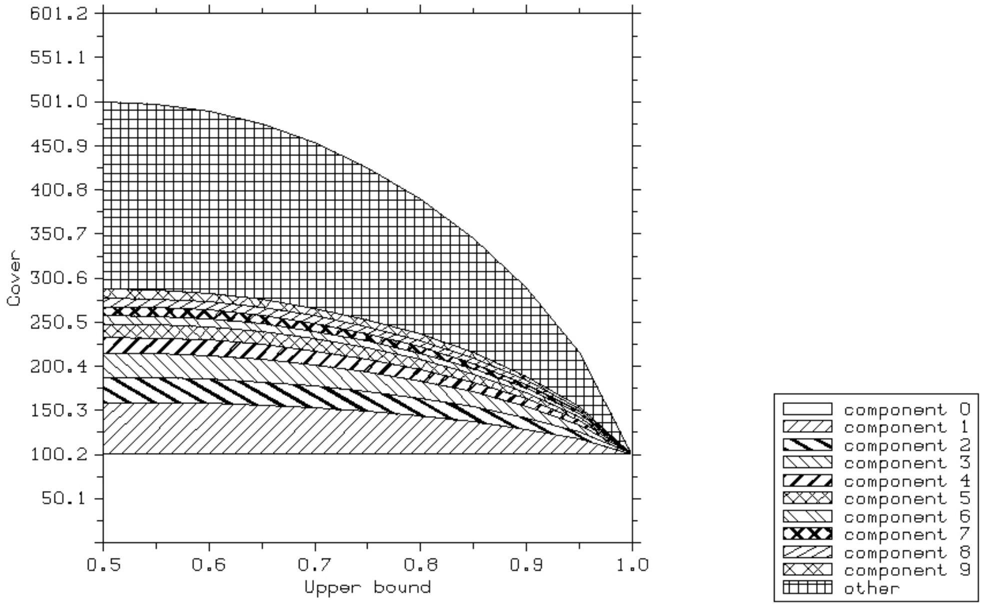


Figure 5.35: Kärdla (Kr). Measure a_u for concept cover for Hiiumaa. Component 0: Kärdla, 101 attributes; component 1: Käina, 58 attributes match; component 2: summer houses, 30 objects; component 3: Emmaste, 28 attributes match; component 4: Kõrgessaare, 18 attributes match; component 5: Kassari, 15 attributes match; component 6: beach, summer houses, 5 objects; component 7: Nõmme, 10 attributes match; component 8: Männamaa, 10 attributes match; component 9: Nurste, 10 attributes match

Concept cover $a_{0.5}$ for Hiiumaa is 501 and for Saaremaa 931 giving a ratio $501/931 \approx 0.54$. For the concept of Kuressaare (Ku) those figures were respectively 1360 and 578 giving a lower ratio of $578/1360 \approx 0.42$. If we eliminate component 0-s from each of the 4 concept covers to neutralize influence of object that the concept is based on and to find out if there is some tendency for settlements in both islands to resemble their regional centers then we find that new ratios are $400/854 \approx 0.47$ for Hiiumaa and $491/1181 \approx 0.42$ for Saaremaa. That seems to confirm the hypothesis that settlements in the island tend to resemble their regional centres as ratio for Hiiumaa is higher.

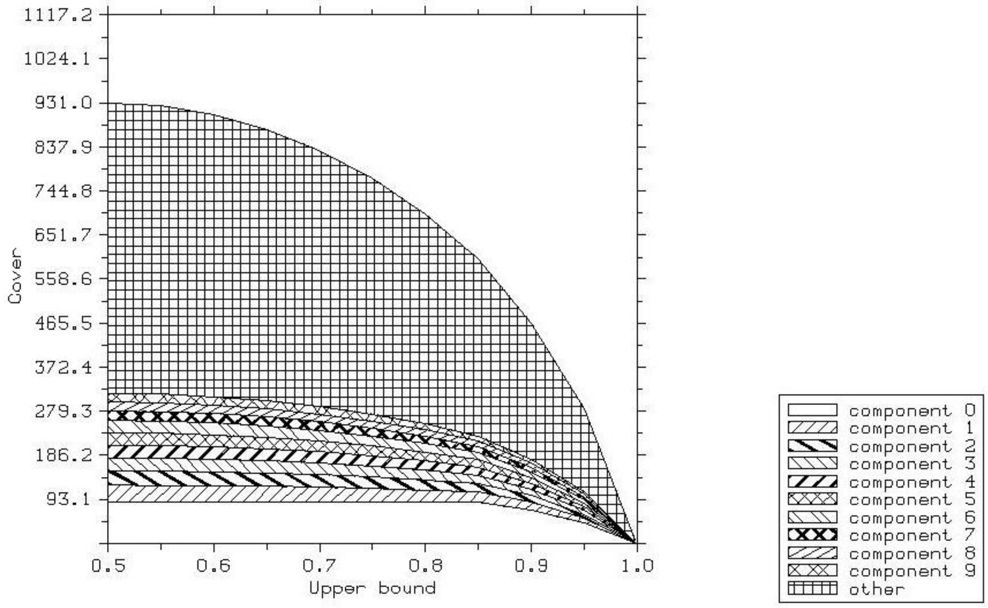


Figure 5.36: Kärdla (Kr). Measure a_u for concept cover for Saaremaa. Component 0: Kuressaare, 87 attributes match; component 1: Orissaare, 36 attributes match; component 2: summer houses, 30 objects; component 3: Liiva, 28 attributes match; component 4: Kärla, 27 attributes match; component 5: Valjala, 26 attributes match; component 6: landing places for fishing boats, summer houses, 13 objects; component 7: Kihelkonna, 20 attributes match; component 8: landing places for fishing boats, 19 objects; component 9: landing places for fishing boats, beach, summer houses, 6 objects

5.8 Comparison between contexts with different attributes

Attributes selected for analysis can influence the results of data mining. In our previous case study of Saaremaa and Hiiumaa we focused on the economy related, binary attributes like presence of housing, beach and so on. We excluded population related attributes that describe the total population in the settlement, number of children, work-aged persons, elderly. These attributes are few in number - 20 total - but there are many relations between the objects and those attributes, thus they can have a strong influence on the results. These attributes follow the general pattern: $x < 10$, $10 \leq x < 50$, $50 \leq x < 100$, $x \geq 100$, presence of x; where x is one of the following: total population, number of children, number of workers, number of elderly. We use similarity measure k_u to estimate the influence of those new attributes to results. We find eight [4] top concepts according to MONOCLE for each island with and without the population related attributes, then we plot values of $k_u(A, K, K_O)$ for each concept (A, B). Eight top MONOCLE concepts would be a good amount for quick analysis, so our comparison should demonstrate if that sort of analysis would have results that are not appropriate for different attribute set. As a comparison, we make similar plots for the different islands with same attributes for values of $k_u(B, K, K_O)$ for a total of 8 comparisons, two for each line in Figure 5.1. Influence of attribute selection to the result is non obvious and important question and thus a good way to test the usefulness of similarity measure k_u.

Population related attributes, as described here, have several flaws from data mining viewpoint. For our area based method it would be advisable to replace attributes $x < 10$, $10 \geq x < 50$, $50 \geq x < 100$ with attributes $x > 0$, $x \geq 10$, $x \geq 50$ where greater population corresponds to greater concept area. As trivially correlated attributes are not recommended in data mining literature [31], it would also be advisable to remove population related attributes that are sum of children, work-aged and elderly and attributes of the type *presence of x*. However, our aim here is to test the similarity measure k_u for comparing the difference between different attribute sets and not actual data analysis. Therefore we will use unchanged attributes.

Figures 5.37 to 5.44 demonstrate that different attribute sets influence the results greatly. If we count the number of totally unique concepts, that is, concepts with a similarity curve as visual straight line at $k_u = 0$ then we find 26 of such concepts for comparison between contexts with different attribute sets and about 12 for comparison between contexts corresponding to different islands. That is, contexts describing same island with different attributes differ more than contexts describing different islands with same attributes. That result is very important for further study of Saaremaa and Hiiumaa as, so far, selection of attributes has unfortunately not been given much thought. Ability to compare differences between contexts with different sets of objects to differences between contexts with different sets of attributes is a nice property

[4] 8 is the number of different line styles in DISLIN Scientific Plotting Library [2]

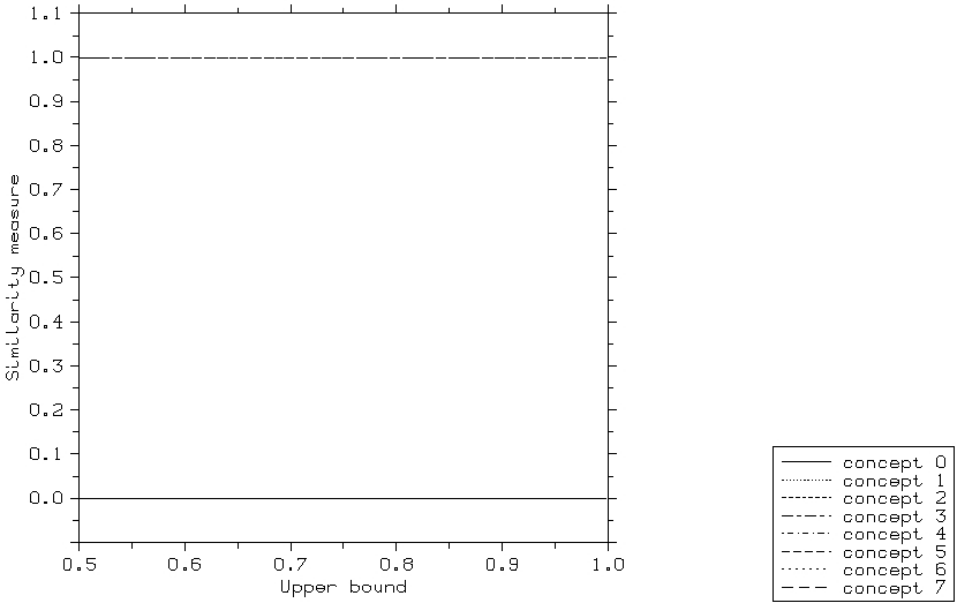


Figure 5.37: Measure $k_u(A, K, K_O)$ for top 8 concepts for Hiiumaa with the population related attributes (K) as compared to Hiiumaa without the population related attributes (K_O). Line where $k_u = 1$ consists of two concepts, line where $k_u = 0$ consists of six concepts.

of similarity measure as it makes such differences quantifiable and co-measurable.

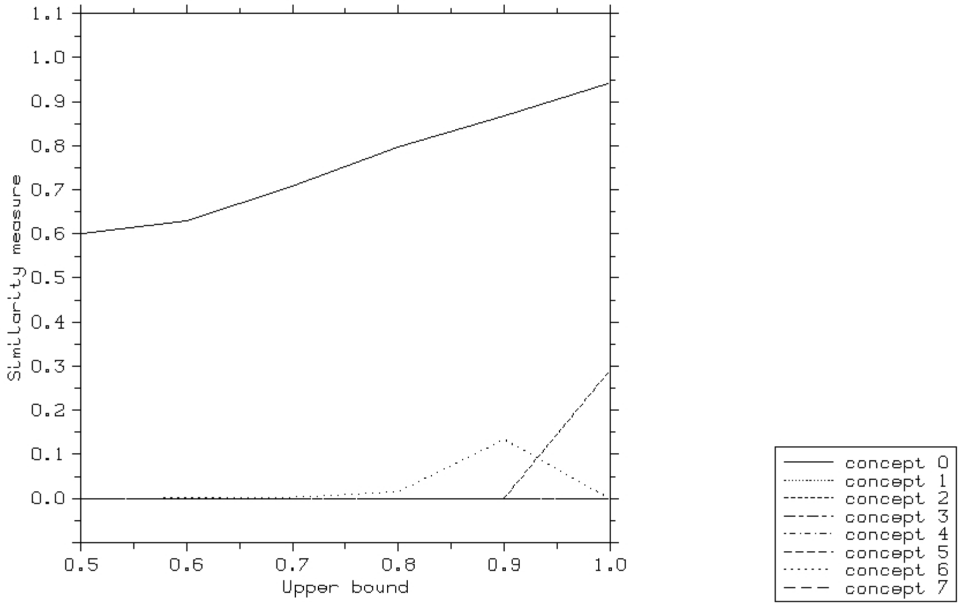


Figure 5.38: Measure $k_u(A, K, K_O)$ for top 8 concepts for Hiiumaa without the population related attributes (K) as compared to Hiiumaa with the population related attributes (K_O).

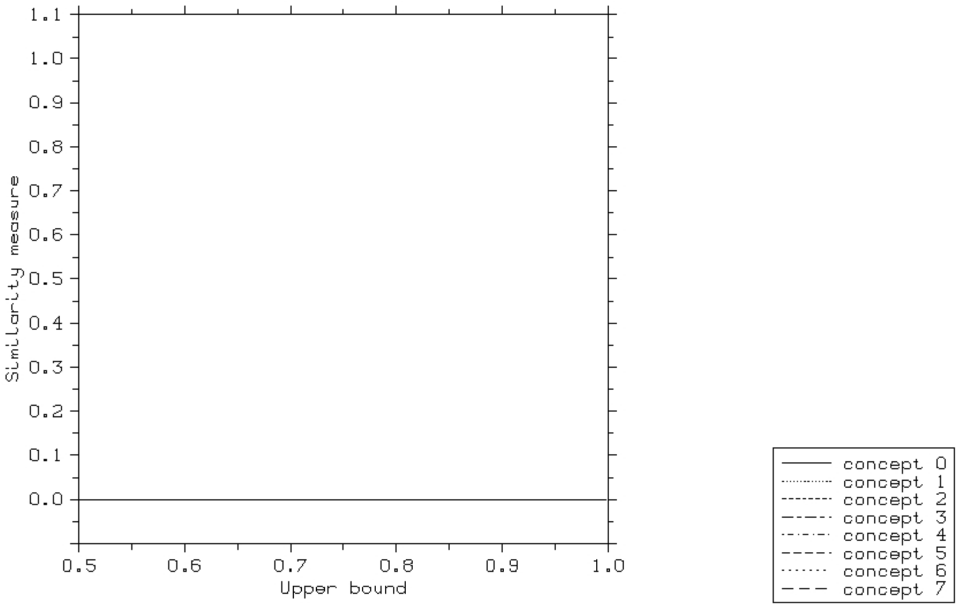


Figure 5.39: Measure $k_u(A, K, K_O)$ for top 8 concepts for Saaremaa with the population related attributes (K) as compared to Saaremaa without the population related attributes (K_O).

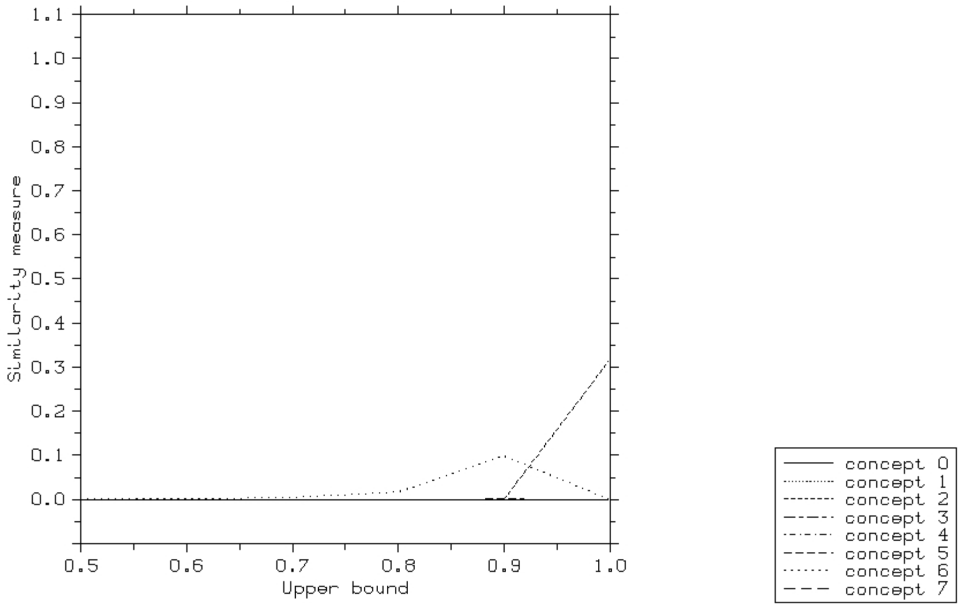


Figure 5.40: Measure $k_u(A, K, K_O)$ for top 8 concepts for Saaremaa without the population related attributes (K) as compared to Saaremaa with the population related attributes (K_O).

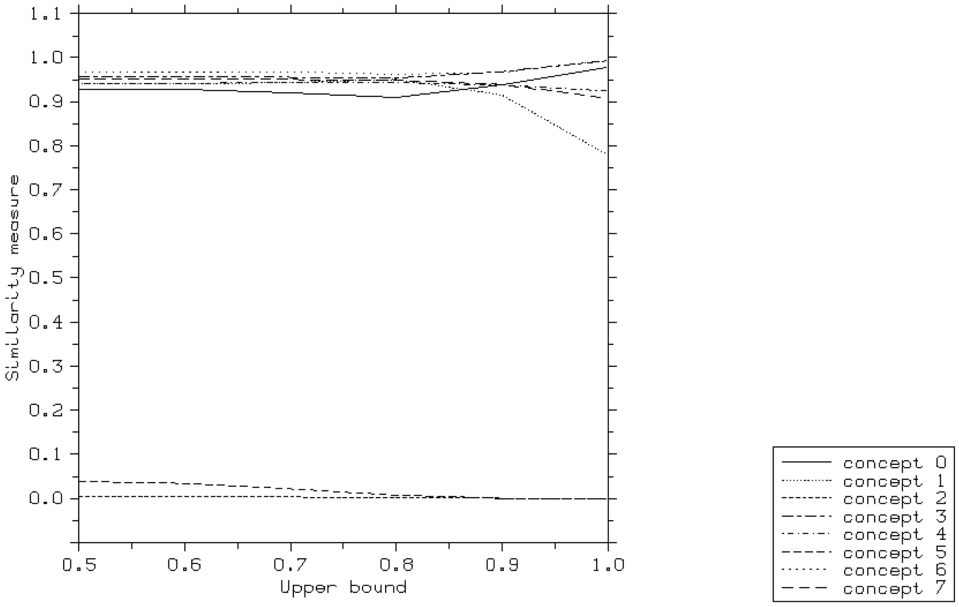


Figure 5.41: Measure $k_u(B, K, K_O)$ for top 8 concepts for Hiiumaa with the population related attributes (K) as compared to Saaremaa with the population related attributes (K_O).

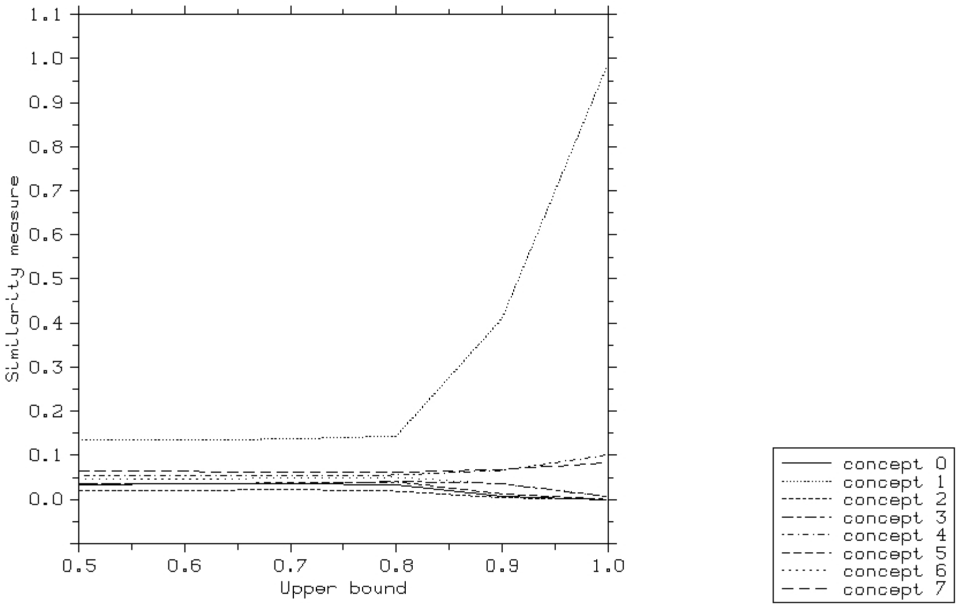


Figure 5.42: Measure $k_u(B, K, K_O)$ for top 8 concepts for Saaremaa with the population related attributes (K) as compared to Hiiumaa with the population related attributes (K_O).

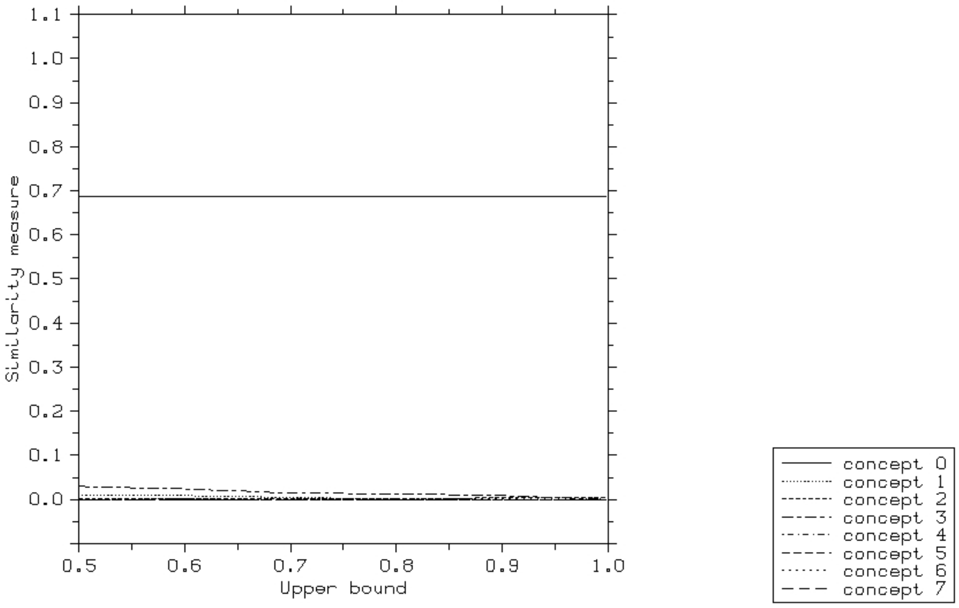


Figure 5.43: Measure $k_u(B, K, K_O)$ for top 8 concepts for Hiiumaa without the population related attributes (K) as compared to Saaremaa without the population related attributes (K_O).

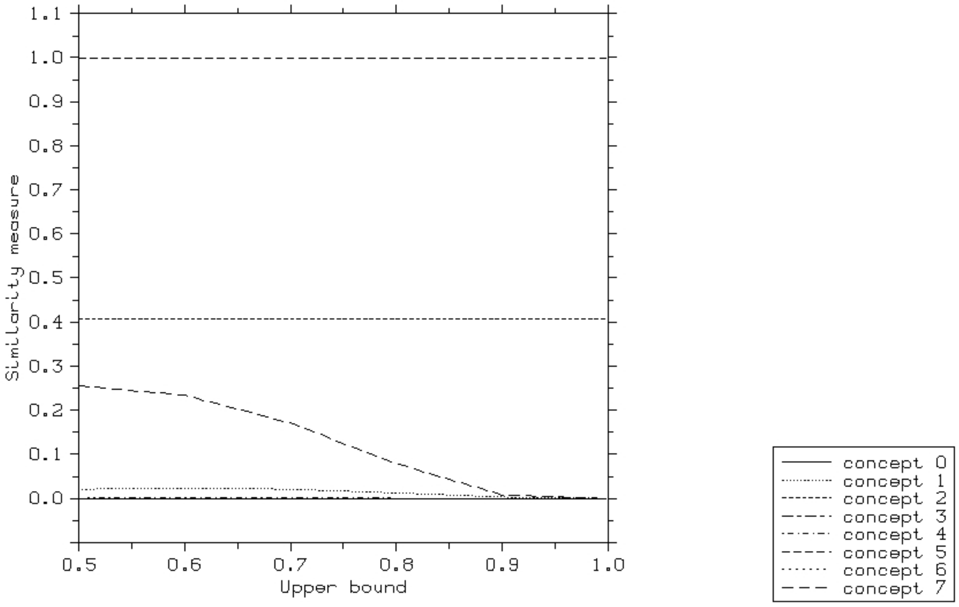


Figure 5.44: Measure $k_u(B, K, K_O)$ for top 8 concepts for Saaremaa without the population related attributes (K) as compared to Hiiumaa with the population related attributes (K_O).

Chapter 6

Conclusions

6.1 Discussion

Main topic of this thesis was unification of formal concept analysis with the theory of monotone systems. There is indeed much similarity: results of monotone system seriation, “blocks”, are essentially formal concepts; problems like finding the best decision, developed initially without any connection to FCA, have natural counterpart - best concept chain. Such connections suggest similarity between two fields and possibilities for future research about combining them.

The work on this thesis was greatly influenced by the case study of Hiiumaa and Saaremaa. The aim of data analysis has been getting the results that are compact, easy to understand, unambiguous and describe the data well. Combination of monotone systems methods with FCA was motivated by the difficulties of extracting semantic information from monotone systems results. MONOCLE method, proposed here, gives results that can be both compact and contain semantic information in an easily accessible form. Central idea of the MONOCLE method is concept area: we are not interested only in the size concepts intent or extent but in the product of their sizes. Our case studies seem to confirm the validity of MONOCLE approach as results are both interesting and sensible. Entropy based concept similarity measure aids further in interpretation of the results and in the evaluation of concepts uniqueness for certain context. Such a similarity measure can be used independently of the MONOCLE method, in the other fields of FCA.

6.2 Contributions of the thesis

To summarize, main contributions of this thesis are:

- Presentation of the theory of monotone systems in a way compatible with the language of FCA.

- Conformity plot visualization method, that gave good results for our case study.
- Redefinition of the problem of finding the best decision as the problem of finding the best concept chain and the discovery of its symmetry in regards to objects and attributes.
- Enhancements to the algorithm for finding the best concept chain.
- MONOCLE method for data mining and knowledge discovery.
- Entropy based similarity measure for evaluating the uniqueness of a formal concept.
- Application of known and new monotone systems methods to the case study about Hiiumaa and Saaremaa.

Of these results, unification of FCA with the theory of monotone systems, MONOCLE method and entropy based similarity measure are probably mor important.

6.3 Directions for further study

This is very much a work in progress. One area for future study would be algorithmic enhancements. MONOCLE method, for example, requires the generation of entire concept lattice. It prunes this lattice for the human analyst but not in the algorithmic sense. While the speed of it was adequate for our case studies, it will not scale up for very large databases. Work by Tarjan et. al. [10] describes a fast monotone heuristic for a related, but not same task, and comparison of its results to that of the MONOCLE method would be of a great interest.

Comparing the results of MONOCLE method with other related methods like clustering, association rules and so on would help greatly in defining the area of applicability for the method.

Another necessary area of further study would be finding the new applications for the described methods. One possible application area would be the presentation of the search results from databases as concepts instead of a flat list.

Possibly most important area for future study would be the further unification of different methods. For example, both monotone systems and formal concept analysis have obvious connections to the well established research fields of bipartite graphs and association rule mining. Such an unification would show these methods not as separate tools but as different branches of the same general and powerful approach.

Bibliography

[1] Galicia home page. http://www.iro.umontreal.ca/∼valtchev/galicia/, Dec. 2007.

[2] Dislin home page. http://www.mps.mpg.de/dislin/, Aug. 2009.

[3] R. Agrawal and T. Imielinski. Mining association rules between sets of items in large databases. In *WProceedings of ACM SIGMOD International Conference on Management of Data*, 1993.

[4] P. Becker, J. Hereth, and G. Stumme. Toscanaj: An open source tool for qualitative data analysis. *Advances in Formal Concept Analysis for Knowledge Discovery in Databases*, pages 1–2, 2002.

[5] J. Bertin. *Graphics and Graphic Information Processing*. Berlin: Walter de Gruyter, 1981.

[6] R. Bělohlávek and V. Vychodil. Formal concept analysis with constraints by closure operators. *Lecture Notes in Artificial Intelligence*, pages 176–191, 2005.

[7] C. Carpiento and G. Romano. Using concept lattices for text retrieval and mining. *Lecture Notes in Computer Science; Formal Concept Analysis*, 3626:161–179, 2005.

[8] J. Czekanowski. Zur differentialdiagnose der neandertalgruppe. *Korrespondentblat der Deutschen Gesellschaft für Anthropologie, Ethnologie und Urgeschichte*, 6 $\setminus$ 7:44–47, 1909.

[9] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order 2nd Revised ed.* Cambridge University Press, 2002.

[10] A. Ene, W. Horne, N. Milosavljevic, P. Rao, R. Schreiber, and E. Tarjan, Robert. Fast exact and heuristic methods for role minimization problems. In *Proceedings of the 13th ACM symposium on Access control models and technologies*, page 1. ACM.

[11] B. Ganter and R. Wille. *Formal Concept Analysis, Mathematical Foundation.* Springer, 1998.

[12] B. Ganter and R. Wille. Formal contexts and concept lattices. In *General Lattice Theory*, pages 591–605. Birkhäuser-Verlag, 2003.

[13] G. Grätzer. *General Lattice Theory*. Birkhäuser-Verlag, 2003.

[14] K. Juurikas, A. Torim, and L. Võhandu. Multivariate data visualization in social space. In *Procs. of IADIS International Conf. on Applied Computing*, pages 427–432, 2006.

[15] K. Juurikas-Lindroos, A. Torim, and L. Võhandu. Mitmemõõtmeliste andmete visualiseerimine sotsiaalses ruumis uurimus Hiiumaa näitel (multivariate data visualization in social space: case study about Hiiumaa). In *Interdistsiplinaarsus sotsiaalteadustes. I : Eesti sotsiaalteaduste VI aastakonverents*, pages 391–411, 2007. in Estonian.

[16] R. Kuusik. Application of theory of monotonic systems for decison trees generation. *Transactions of Tallinn University of Technology*, 705, 1989.

[17] E. N. Kuznetsov and M. I. B. Analysis of the distribution functions in an organization. *Automation and Remote control*, pages 1325–1332, 1982. http://www.datalaundering.com/download/organiza.pdf.

[18] S. O. Kuznetsov. On stability of a formal concept. *Annals of Mathematics and Artificial Intelligence*, 49:101–115, 2007.

[19] L. Lakhal and G. Stumme. Efficient mining of association rules based on formal concept analysis. *Lecture Notes in Computer Science; Formal Concept Analysis*, 3626:180–195, 2005.

[20] I. Liiv. Visualization and data mining method for inventory classification. *Proceedings of the 2007 IEEE International Conference on Service Operations and Logistics and Informatics*, pages 472–477, 2007.

[21] I. Liiv. *Pattern Discovery Using Seriation and Matrix Reordering: A Unified View, Extensions and an Application to Inventory Management*. PhD thesis, Tallinn University of Technology, 2008.

[22] I. Liiv. Implementation of Bertin's reorderable matrices with billion elements. Manuscript, 2009.

[23] K. Lindroos. *Mapping Social Structures by Formal Non-Linear Information Processing Methods: Case Studies of Estonian Islands Environments*. PhD thesis, Tallinn University of Technology, 2008.

[24] B. Mandelbrot, Benoit. *The (mis)Behaviour of Markets*. Profile Books, 2005.

[25] T. McCormick, W., J. Schweitzer, P., and W. White, T. Problem decomposition and data reorganization by a clustering technique. *Operations Research*, 20 (5):993–1009, 1972.

[26] I. Mullat. Extremal monotonic systems (in russian). *Automation and Remote Control*, (5):130–139, 1976. [WWW]: http://www.datalaundering.com/download/extrem01.pdf (31.12.2007) (in English).

[27] W. M. Petrie, Flinders. Sequences in prehistoric remains. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 29:295–301, 1899.

[28] U. Priss. Linguistic applications of formal concept analysis. *Lecture Notes in Computer Science; Formal Concept Analysis*, 3626:149–160, 2005.

[29] C. E. Shannon. Prediction and entropy of printed english. *The Bell System Technical Journal*, 30:50–64, 1951.

[30] S. Skiena, Steven. *The Algorithm Design Manual*. Springer-Verlag, 1998.

[31] T. Soukop and I. Davidson. *Visual Data Mining*. Wiley, 2002.

[32] G. Stumme, R. Taouil, and L. Bastide, Y.and Lakhal. Conceptual clustering with iceberg concept lattices. In *Proc. GI-Fachgruppentreffen Maschinelles Lernen*, page 763. Universität Dortmund, 2001.

[33] T. Tilley, R. Cole, P. Becker, and P. Eklund. A survey of formal concept analysis support for software engineering activities. *Lecture Notes in Computer Science; Formal Concept Analysis*, 3626:250–271, 2005.

[34] A. Torim and R. Kuusik. Problem and algorithms for finding the best decison. *WSEAS Transactions on Information Science and applications*, 9:1462–1469, 2005.

[35] A. Torim and K. Lindroos. Sorting concepts by priority using theory of monotone systems. *Lecture Notes in Computer Science; Conceptual Structures: Knowledge Visualization and Reasoning*, 5113:175–188, 2008.

[36] L. Võhandu, R. Kuusik, A. Torim, E. Aab, and G. Lind. Some monotone systems algorithms for data mining. In *WSEAS Transactions on Information Science & Applications*, pages 802–809, 2006.

[37] P. Valtchev, D. Grosser, C. Roume, and H. Hacene, R. Galicia: an open platform for lattices. In *Using Conceptual Structures: Contrib. to the 11th ICCS*, pages 241–254, 2003.

[38] L. Vyhandu. Rapid data analysis methods (in russian). *Transactions of Tallinn University of Technology*, 464:21–39, 1979.

[39] L. Vyhandu. Some methods to order objects and variables in data systems (in russian). *Transactions of Tallinn University of Technology*, 482:43–50, 1980.

[40] L. Vyhandu. Fast methods for data processing (in russian). *Computer Systems: Computer methods for revealing regularities*, pages 20–29, 1981.

[41] L. Vyhandu. Fast methods in exploratory data analysis. *Transactions of Tallinn University of Technology*, 705:3–13, 1989.

[42] W. Weinstein, Eric. Binomial distribution. http://mathworld.wolfram.com/BinomialDistribution.html, 2009.

[43] R. Wille. Restructuring lattice theory: an approach based on hierarchies of concepts. *Ordered Sets*, pages 445–470, 1982.

[44] R. Wille. Conceptual knowledge processing in the field of economics. *Lecture Notes in Computer Science; Formal Concept Analysis*, 3626:226–249, 2005.

[45] R. Wille, G. Stumme, and B. Ganter. *Formal Concept Analysis: Foundations and Applications*. Springer, 2005.

[46] H. I. Witten and F. Eibe. *Data Mining*. Academic Press, 2000.

Formaalsed mõisted monotoonsete süsteemide teoorias

Lühikokkuvõte

Formaalne mõistete analüüs ja monotoonsete süsteemide teooria on mõlemad hästituntud andmekavandamise ja teadmushõive meetodid, mille seoseid on senimaani vähe uuritud. Selles dissertatsioonis uuritaksegi neid seoseid ja pakutakse välja mõned uued teadmushõive meetodid, mis kombineerivad mõlema lähenemise omadusi.

Formaalset mõistet iseloomustavad tema ekstensioon, objektide hulk, ja intensioon, nende objektide ühiste atribuutide hulk. Nende vahel on defineeritud teatud matemaatiline seos, mille kohaselt need hulgad peavad olema teatud mõttes lokaalselt maksimaalsed.

Monotoonsete süsteemide teooria põhineb monotoonsetel kaalufunktsioonidel ja seda kasutatakse tihti järjestamiseks: andmetabeli korrastamiseks peidetud struktuuri avamise eesmärgil. See töö väidab, et selline struktuur on formaalsete mõistete hulk.

Töös on läbiva näite ja tulemusena kasutatud Saare- ja Hiiumaa asulate sotsiaalmajanduslike andmete analüüsi. Selle uuringu eesmärgiks oli majandusliku ja sotsiaalse arengu mustrite leidmine.

Töös pakutakse välja konformsusgraafiku visualisatsioon, mis andis Saare- ja Hiiumaa uuringus häid tulemusi ja defineeritakse parima otsuse leidmise probleem ümber parima mõisteahela leidmise probleemina, mis võimaldab näha probleemi sümmeetriat objektide ja atribuutide osas. Pakutakse välja mõned algoritmika-alased parandused parima mõisteahela leidmiseks.

Monotoonsete süsteemide meetoditel ja formaalsel mõistete analüüsil on teatud probleeme suurte andmetabelitega. Siin töös pakutakse välja MONOCLE meetod andmekaevandamiseks ja teadmushõiveks, mis üritab neid raskuseid leevendada. MONOCLE meetodi tulemiks on oma tähtsuse järgi järjestatud formaalsete mõistete jada.

Pakutakse välja ka meetod võrdlemaks neid tulemusi üle erinevate kontekstide, nagu Hiiumaa ja Saaremaa: entroopiapõhine sarnasusmõõt formaalse mõiste unikaalsuse hindamiseks.

Võtmesõnad: andmekaevandamine, teadmushõive, informatsiooni visualiseerimine, monotoonsed süsteemid, formaalne mõistete analüüs.

Publications by the author

- Torim A., Kuusik R., Describing data table with best decision, Proceedings of the 5th WSEAS Int. Conf. on SIMULATION, MODELING AND OPTIMIZATION, Corfu, Greece, August 17-19, 2005 (pp152-157)

- Torim A., Kuusik R., Problem and Algorithms for Finding the Best Decision, WSEAS Transactions on INFORMATION SCIENCE and APPLICATIONS, Issue 9 , Volume 2, September 2005, pp. 1462-1470

- Juurikas K., Torim A., Võhandu L., (2006). Multivariate Data Visualization in Social Space. In: Proceedings of the IADIS International Conference Applied Computing 2006: IADIS International Conference Applied Computing; San Sebastian, Spain; 2006, Feb 25-28. IADIS Press, 2006, 427 - 432.

- Võhandu, Leo; Kuusik, Rein; Torim, Ants; Aab, Eik; Lind, Grete (2006). Some Monotone Systems Algorithms for Data Mining. WSEAS Transactions on Information Science and Applications, 4(3), 802 - 809.

- Võhandu, Leo; Kuusik, Rein; Torim, Ants; Aab, Eik; Lind, Grete (2006). Some algorithms for data table (re)ordering using Monotone Systems. In: Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2006), Madrid, Spain, February 15-17, 2006: 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2006), Madrid, Spain, February 15-17, 2006. Madrid: 2006, 417 - 422.

- Torim, A.; Lindroos, K. (2008). Sorting Concepts by Priority Using Theory of Monotone Systems. In: Conceptual Structures: Knowledge Visualization and Reasoning: ICCS'08 - Conceptual Structures: Knowledge Visualization and Reasoning, 7-11 July, Toulouse, France. Heidelberg: Springer-Verlag, 2008, (Lecture Notes in Computer Science; 5113), 175 - 188.

Curriculum Vitae (in Estonian)

1. Isikuandmed

Ees- ja perekonnanimi: Ants Torim
Sünniaeg ja -koht: 11. veebruar 1977, Tallinn
Kodakondsus: Eesti

2. Kontaktandmed

Aadress: Raja 15, Tallinn 12168
Telefon: +37256462814
E-posti aadress: torim@staff.ttu.ee

3. Hariduskäik

Õppeasutus (nimetus lõpetamise ajal)	Lõpetamise aeg	Haridus (eriala/kraad)
Tallinna Tehnikaülikool	2000	informaatika / tehnikateaduste bakalaureus
Tallinna Tehnikaülikool	2003	informaatika / tehnikateaduste magister

4. Keelteoskus

Keel	Tase
Eesti keel	Kõrgtase
Inglise keel	Kõrgtase
Soome keel	Kesktase
Vene keel	Algtase

5. Teenistuskäik

Teenistuse aeg	Tööandja nimetus	Ametikoht
2000-2004	Tallinna Tehnikaülikool	Assistent
2004-2009	Tallinna Tehnikaülikool	Lektor

6. Teadustegevus (publikatsioonid)

- Torim A., Kuusik R., Describing data table with best decision, Proceedings of the 5th WSEAS Int. Conf. on SIMULATION, MODELING AND OPTIMIZATION, Corfu, Greece, August 17-19, 2005 (pp152-157)

- Torim A., Kuusik R., Problem and Algorithms for Finding the Best Decision, WSEAS Transactions on INFORMATION SCIENCE and APPLICATIONS, Issue 9 , Volume 2, September 2005, pp. 1462-1470

- Juurikas K., Torim A., Võhandu L., (2006). Multivariate Data Visualization in Social Space. In: Proceedings of the IADIS International Conference Applied Computing 2006: IADIS International Conference Applied Computing; San Sebastian, Spain; 2006, Feb 25-28. IADIS Press, 2006, 427 - 432.

- Võhandu, Leo; Kuusik, Rein; Torim, Ants; Aab, Eik; Lind, Grete (2006). Some Monotone Systems Algorithms for Data Mining. WSEAS Transactions on Information Science and Applications, 4(3), 802 - 809.

- Võhandu, Leo; Kuusik, Rein; Torim, Ants; Aab, Eik; Lind, Grete (2006). Some algorithms for data table (re)ordering using Monotone Systems. In:Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2006), Madrid, Spain, February 15-17, 2006: 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2006), Madrid, Spain, February 15-17, 2006. Madrid: 2006, 417 - 422.

- Torim, A.; Lindroos, K. (2008). Sorting Concepts by Priority Using Theory of Monotone Systems. In: Conceptual Structures: Knowledge Visualization and Reasoning: ICCS'08 - Conceptual Structures: Knowledge Visualization and Reasoning, 7-11 July, Toulouse, France. Heidelberg: Springer-Verlag, 2008, (Lecture Notes in Computer Science; 5113), 175 - 188.

7. Kaitstud lõputööd

Ants Torim, magistrikraad (teaduskraad), 2003, (juh) Rein Kuusik, OORMON: objektorienteeritud raamistik monotoonsete süsteemide loomiseks, Tallinna Tehnikaülikool

8. Teadustöö põhisuunad

Loodusteadused ja tehnika, Arvutiteadused (Andmeanalüüs, andmekaevandamine, formaalne mõistete analüüs, monotoonsed süsteemid)

Curriculum Vitae

1. Personal data

Name: Ants Torim
Date and place of birth: 11 February 1977, Tallinn
Citizenship: Estonia

2. Contact information

Address: 15 Raja Street, Tallinn 12168
Telefon: +37256462814
E-posti aadress: torim@staff.ttu.ee

3. Education

Educational institution	Year of graduation	Education
Tallinn University of Technology	2000	B.Sc. (informatics)
Tallinn University of Technology	2003	M.Sc. (informatics)

4. Language skills

Keel	Tase
Estonian	High level
English	High level
Finnish	Intermediate level
Russian	Basic level

5. Professional employment

Teenistuse aeg	Tööandja nimetus	Ametikoht
2000-2004	Tallinna Tehnikaülikool	Assistent
2004-2009	Tallinna Tehnikaülikool	Lektor

6. Scientific work (publications)

- Torim A., Kuusik R., Describing data table with best decision, Proceedings of the 5th WSEAS Int. Conf. on SIMULATION, MODELING AND OPTIMIZATION, Corfu, Greece, August 17-19, 2005 (pp152-157)
- Torim A., Kuusik R., Problem and Algorithms for Finding the Best Decision, WSEAS Transactions on INFORMATION SCIENCE and APPLICATIONS, Issue 9 , Volume 2, September 2005, pp. 1462-1470
- Juurikas K., Torim A., Võhandu L., (2006). Multivariate Data Visualization in Social Space. In: Proceedings of the IADIS International Conference Applied Computing 2006: IADIS International Conference Applied Computing; San Sebastian, Spain; 2006, Feb 25-28. IADIS Press, 2006, 427 - 432.
- Võhandu, Leo; Kuusik, Rein; Torim, Ants; Aab, Eik; Lind, Grete (2006). Some Monotone Systems Algorithms for Data Mining. WSEAS Transactions on Information Science and Applications, 4(3), 802 - 809.
- Võhandu, Leo; Kuusik, Rein; Torim, Ants; Aab, Eik; Lind, Grete (2006). Some algorithms for data table (re)ordering using Monotone Systems. In:Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2006), Madrid, Spain, February 15-17, 2006: 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2006), Madrid, Spain, February 15-17, 2006. Madrid: 2006, 417 - 422.
- Torim, A.; Lindroos, K. (2008). Sorting Concepts by Priority Using Theory of Monotone Systems. In: Conceptual Structures: Knowledge Visualization and Reasoning: ICCS'08 - Conceptual Structures: Knowledge Visualization and Reasoning, 7-11 July, Toulouse, France. Heidelberg: Springer-Verlag, 2008, (Lecture Notes in Computer Science; 5113), 175 - 188.

7. Defended theses

Ants Torim, M.Sc., 2003, supervisor Rein Kuusik, OORMON: objektorienteeritud raamistik monotoonsete süsteemide loomiseks (OORMON: object-oriented framework for the study of monotone systems), Tallinn University of Technology

8. Research interests

Natural sciences and technology, computer sciences (data analysis, data mining, formal concept analysis, monotone systems)