

Detecting Temporal Change in Event Sequences: An Application to Demographic Data

H. Blockeel¹, J. Fürnkranz², A. Prskawetz³, and F. C. Billari³

¹ Katholieke Universiteit Leuven
Department of Computer Science

Celestijnenlaan 200A, B-3001 Leuven, Belgium

`Hendrik.Blockeel@cs.kuleuven.ac.be`

² Austrian Research Institute for Artificial Intelligence

Schottengasse 3, A-1010 Wien, Austria

`juffi@oefai.at`

³ Max Planck Institute for Demographic Research

Doberaner Straße 114, D-18507, Rostock, Germany

`[fuernkranz,billari]@demogr.mpg.de`

Abstract. In this paper, we discuss an approach for discovering temporal changes in event sequences, and present first results from a study on demographic data. The data encode characteristic events in a person's life course, such as their birth date, the begin and end dates of their partnerships and marriages, and the birth dates of their children. The goal is to detect significant changes in the chronology of these events over people from different birth cohorts. To solve this problem, we encoded the temporal information in a first-order logic representation, and employed *Warmr*, an ILP system that discovers association rules in a multi-relational data set, to detect frequent patterns that show significant variance over different birth cohorts. As a case study in multi-relational association rule mining, this work illustrates the flexibility resulting from the use of first-order background knowledge, but also uncovers a number of important issues that hitherto received little attention.

1 Introduction

In this paper, we study the problem of discovering patterns that exhibit a significant change in their relative frequency of occurrence over time. As was already argued by [8], in many domains the step beyond discovering frequent item sets to the discovery of second-order phenomena like the temporal change in these frequencies is of crucial importance.

The analysis of life courses is such a problem. In the social sciences, and especially in demography and sociology, there has been a diffusion of the so-called *life course approach* [15]. One of the principal interests in that approach is the study of how the lives of humans change as far as the age and the sequencing and the number of crucial events are concerned. To study the evolution of a whole society, it is common to analyze successive *cohorts* of people, i.e., groups

of people that were born in the same period of time (e.g. the same year or the same decade).

Previous approaches to detecting change mostly propose special-purpose algorithms that had to treat time as special type of variable [12,8]. Instead, we suggest to address this problem by exploiting the power of a general, multi-dimensional data mining system. The system that we use—Warmr [9]—is based on the level-wise search of conventional association rule learning systems of the Apriori-family [1,11]. It extends these systems by looking for frequent patterns that may be expressed as conjunction of first-order literals. This expressive power allows to encode temporal relationships fairly easily. In fact, the system does not need to discriminate between temporal relations and other domain-dependent relations, which is typical for conventional solutions to sequence discovery [3,13], and lets them co-exist in a natural and straight-forward way.

2 The Dataset

The data for our analysis originate from the Austrian Fertility and Family Survey (FFS), which was conducted between December 1995 and May 1996. In the survey, retrospective histories of partnerships, births, employment and education were collected for 4,581 women and 1,539 men between ages 20 and 54. Hence, the Austrian FFS covers birth cohorts from 1941 to 1976. The retrospective histories of partnerships and fertility for each respondent allow us to determine the timing of all births in the current and any previous union. Moreover, information about the civil status of each partnership in any month of observation is available, which allows us to determine whether a union started as a marriage or whether it was transformed into a marriage later on.

We are interested in studying the main features that discriminate between the life courses of older and younger cohorts as far as the number of children, number and type of unions, fertility before and after unions, etc. are concerned. The present study should be considered a first step into that direction.

3 Multi-relational Data Mining – Using Warmr for Discovering Temporal Changes

The dataset under consideration is essentially a multi-relational dataset: a person’s life course is not described with a single tuple but by a set of tuples. Most common data mining algorithms expect the data to reside in a single table, and when the data are actually stored in a database with multiple tables they have to be preprocessed: one needs to derive a single table from the original data that hopefully retains as much of the original information as possible. This is a non-trivial task. Two directions are possible: one is to devise automatic preprocessing methods, another is to use data mining methods that can handle a multi-relational database, e.g., inductive logic programming (ILP) methods [10]. Our approach falls into the second category as we employ the ILP system Warmr [9] to detect interesting patterns.

```

female(159).
birth_date(159,6001).
children(159,2).
    child_birth_date(159,7810).
    child_birth_date(159,8706).
unions(159,2).
    union(159,7805, 8106).
    union(159,8306, -2).
marriage_date(159,7807).
marriage_date(159,8706).

subject(I) :- female(I),
              birth_date(I,X),
              X >= 4100, X<=6000.
gen40(I) :- birth_date(I,X),
            X >= 4100, X<4600.
child_out_of_wedlock_at_age(I,A) :-
    birth_date(I,X),
    child_birth_date(I,M),
    \+ (marriage_date(I,N), N=<M),
    A is (M-X)/100, A <= 35.
    
```

Fig. 1. On the left, a Prolog encoding of a typical entry in the database is shown. This snapshot represents a female person with id 159 who was born in January 1960. Up to the interview date (\approx December 1995), she had formed two unions. The first lasted from May 1978 to June 1981, and the second started in June 1983 (and has not ended at the time of the interview). Both unions were converted to marriages (July 1978 and June 1987) and in each union one child was born (October 1978 and June 1987). The right half shows a few predicate definitions that operated on the basic data encoding. The `subject/1` predicate served as a key, and could be used to filter the data (in this case to admit only persons from cohorts 1941–60). `gen40/1` encodes one of the class values (the cohort 1941–1945), and `child_out_of_wedlock_at_age/2` shows an example for abstract background knowledge that can be defined upon the base predicates, as well as an example for censoring events above a certain age.

As multi-relational data mining is a relatively unexplored field, few guidelines exist as to the methodology that should be followed. Consequently, this section describes the complete data mining process that we employed for addressing our problem. On the way, we will also describe the workings of *Warmr*, the first-order data mining system that we used, as well as the way the data was represented for the use by this system. Although the following sections will only describe the final data representation, it should be noted that this was not the result of a top-down design, but of repeated exploration of several options.

3.1 Data Preprocessing

Data preprocessing consisted of several steps. First, the data were converted into a Prolog format. A typical entry is shown in the left half of Fig. 1. From then on, preliminary experiments could be run, which were helpful to further improve the quality of the domain representation.

To this base representation, we added several pieces of background knowledge in the form of Prolog predicates that operate upon these operational predicates. These were mostly conversions from dates to ages (which is needed when one wants to find, e.g., changes in the average age of marrying etc.), but also included complex high-level predicates such as `child_out_of_wedlock_at_age/2`, which encodes whether a person had a child out of wedlock. Most importantly, we included the `</2` predicate, which allowed the system to compare ages at which

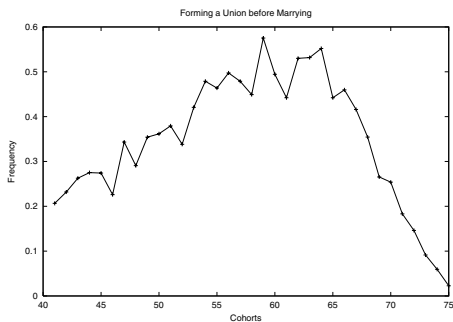


Fig. 2. Relative frequencies of people that had a union before they married. The steady increase up to 1960 is a true regularity, while the decline for the cohorts 1960 — 1975 is due to the fact that for increasingly many individuals of these cohorts, union formation and marriage has not yet taken place in their lives at the time of the interview (\approx December 1995), but will presumably take place later.

events take place in people’s life courses, and thus to describe their chronology. Note that this approach does not only allow to find sequences in the strict sense, but also partially ordered events, such as $A < B$, $A < C$, where the order of the events B and C is left open (both $A < B < C$ and $A < C < B$ are possible). It is precisely this facility which makes the use of a relational data mining system necessary, because this functionality cannot be achieved with a conventional, propositional association rule finder without using a pre-processing phase that encodes all possible event sequences as separate features. Note that such a procedure would basically reduce the task of the frequent pattern discovery to a single pass over all possible, pre-compiled sequences, which defeats the purpose of the efficient, level-wise search.

In addition, the flexibility of the first-order background knowledge also facilitated *censoring* of the data. In social sciences, this term describes the situation that, when using data collected from interviews where people are asked about their past experience of events, only their life courses up to the age at interview are available to the analyst. Such a situation can, e.g., be seen in the decline of frequencies of people that formed a union before they were married (Fig. 2). As people born after 1965 were, at the time of the interview (\approx December 1995) 30 years or younger, it is quite natural that the probability for these people to have experienced certain events or event sequences in their life course is not the same as for people in their forties or older. Consequently, in preliminary experiments we discovered many rules because of the comparably low frequencies for marriage, union formation, and child birth for people born in the seventies.

After some experimentation, we decided to censor items in the following way:

- only people born in the forties or fifties were retained
- events in people’s lives that occurred after the age of 35 were ignored

Censoring was easily done by adding inequations in the relevant definitions of background knowledge (like in the last rule of Fig. 2). Naturally, looking only at events happening before the age of 35 severely limits our study, but increasing this age limit could only be done at the expense of reducing the data set (in order to increase the age limit to 40 and yet avoid artificial patterns, the set of subjects would need to be reduced to those people born before 1956). Our

choice is a compromise between keeping as much data as possible and looking at a reasonable set of events in people’s lives.

The original dataset contained 6120 entries, 1539 representing male subjects, and 4581 representing female subjects. As the distribution of male and female subjects is obviously skewed, we decided to only work with the larger, female group.¹ Again, this could easily be achieved by adding an appropriate condition in `subject/2` (see Fig. 1), which defines the key (as defined below).

3.2 Discovery of Frequent Patterns

Warmr [9] is an ILP system that discovers frequent patterns in a data set, where a “pattern” is defined as a conjunction of first order literals. For instance, in our data set patterns may look like this:

```
subject(S), married_at_age(S,A), child_at_age(S,B), B>A.
subject(S), child_at_age(S,A), child_at_age(S,B), B>A.
```

The first pattern describes a person `S` who married and subsequently had a child; the second one describes a person who had at least two children (with different ages).

Each pattern is associated with a so-called *key*: the frequency of a pattern is determined by counting the number of key items (i.e., the number of instantiations of the key variable) that satisfy the constraints stated in the pattern. In the above example, it is natural to have the `subject S` as the key.

Warmr can be considered a first-order upgrade of Apriori [1]; it performs a top-down level-wise search, starting with the key and refining patterns by adding literals to them. Infrequent patterns (i.e. patterns of which the frequency is below some predefined threshold) are pruned as are their refinements. We refer to [9] for more details.

3.3 Discovering Temporal Change

As described above, Warmr finds all patterns that have a frequency above a user-specified threshold. In a second phase, Warmr can combine these patterns into first order association rules; basically, these rules are of the form “*if LHS occurs, then RHS occurs with probability c*”. Contrary to conventional, propositional association rule finders, patterns may be formulated in first-order logic, which allows conditions to be linked together by sharing the same variables.

In order to limit the number of possible rules, the current version of Warmr expects that the user provides a list of possible patterns that can occur on the left-hand side (LHS) or right-hand side (RHS) of a rule. This is done with Warmr’s `classes` setting. More specifically, if there is a frequent pattern P and subsets A and B , such that $A \subset P$ is one of the `classes` and $B = P \setminus A$, then

¹ The over-sampling of female individuals was intentional in the family and fertility survey from which the data originate because fertility is more closely linked to female life courses.

Warmr returns the rules $A \Rightarrow B$ and $B \Rightarrow A$ (assuming they fulfill possible additional criteria, such as achieving a minimal confidence level).

In our application, we used the classes to separate people into different cohorts. After some experimentation, we decided to use four different cohorts, each encompassing a 5-year span. In Warmr, this could be simply encoded using generation literals as classes:

```
classes([gen40(_), gen45(_), gen50(_), gen55(_)]).
```

This tells Warmr that we expect to see rules of the following type²

```
gen40(S) => subject(S), child_at_age(S,A), child_at_age(S,B), B>A.
```

The rules are ordered according to their *interestingness*. Interestingness is defined in Warmr as the number of standard deviations the observed frequency of a pattern in an age group differs from the average frequency in the whole data set under consideration. More precisely, for a rule $A \Rightarrow B$, interestingness $d(A \Rightarrow B)$ is defined as

$$d(A \Rightarrow B) = \frac{p(B|A) - p(B)}{\sqrt{\frac{p(B)(1-p(B))}{n(A)}}} \approx \frac{\frac{n(A \wedge B)}{n(A)} - \frac{n(B)}{N}}{\sqrt{\frac{\frac{n(B)}{N}(1 - \frac{n(B)}{N})}{n(A)}}}$$

where $p(B) \approx \frac{n(B)}{N}$ is the probability/relative frequency of the pattern B in the entire data set, whereas $p(B|A) \approx \frac{n(A \wedge B)}{n(A)}$ is the probability/relative frequency of the same pattern occurring in the subgroup defined by the class A . Hence, the numerator computes the deviation of the expected frequency of pattern B (if it were independent of A) from its actual frequency of occurrence in the subgroup defined by pattern A . This difference is normalized with the standard deviation that could be expected if $p(B)$ were the true frequency of occurrence of B within A . Thus, $d(A \Rightarrow B)$ computes the number of standard deviations that the actual frequency is away from the expected frequency. In our case, the interestingness measure compares the expected number of individuals that satisfy a certain pattern in a cohort to the expected number of individuals that should satisfy this pattern if the occurrence of the pattern were independent from the birth cohort.

3.4 Filtering of Semantic Redundancy

Many of the discovered rules are syntactically different but semantically equivalent because redundant conditions are added to a rule (if, e.g., a constraint that specifies that a person has more than 3 children is followed by redundant tests for having more than 2 or 1 children) or the same situation is expressed in different

² As mentioned above, Warmr will also produce rules of the form `subject(S), child_at_age(S,A) ... => gen40(S)`. because the evaluation of rules $A \Rightarrow B$ and $B \Rightarrow A$ may differ. However, as characteristic rules of the form `gen40(S) => pattern` are more natural to interpret, we applied a filter to Warmr's output that only retained this kind of rules.

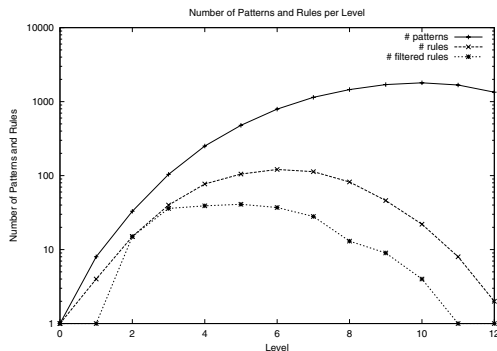


Fig. 3. The number of found patterns, the number of found rules and the number of semantically unique rules that remain after filtering. The scale on the y -axis is logarithmic.

ways (e.g., the above-mentioned constraint on the number of children can also be formulated using an equivalent number of `child_at_age/2` predicates together with inequalities that assure that they refer to successive events). Some (but not all) of these cases can be addressed by Warmr’s configurable language bias (e.g., by specifying that only one constraint on the number of children is admissible in a rule) and its constraint specification language (e.g., by specifying that literal $p(X)$ must not be added if literal $q(X)$ already occurs). We will return to this issue in Sect. 5.

To reduce the number of rules, we employ a simple filtering strategy: for rules that share the same frequencies for all its components ($n(A)$, $n(B)$, $n(A \wedge B)$) and hence have the same measure of interestingness $d(A \Rightarrow B)$, we simply assume that they are semantically equivalent. In such cases we automatically removed all rules except those that were found at the minimum level (i.e., all but the shortest rules).³

Figure 3 shows the number of frequent patterns found in one of our experiments, the number of rules generated from these patterns, and the number of rules that survived the filtering process. What also becomes apparent is that the number of irrelevant and redundant patterns increases with the level. At level 10, where the largest number of frequent patterns is discovered (1795), only four rules survive the filtering, and at subsequent levels none remain (0 frequencies are shown as 1 in the log-scale).

3.5 Visualizing the Results

Rules reported by Warmr to be interesting (in the sense that they describe patterns whose frequency in a certain age group deviates significantly from its av-

³ A better approximation of semantical equivalence would be to consider the actual sets of covered instances, instead of just their size. However, after inspecting a number of rules with the same interesting measure, we found no examples where rules with the same interestingness measure were not semantically equivalent, so we did not consider it worthwhile to implement this more accurate approximation. Concerning the choice of the simplest rule, when several rules had the same interestingness measure and the same complexity, we arbitrarily decided for one of them.

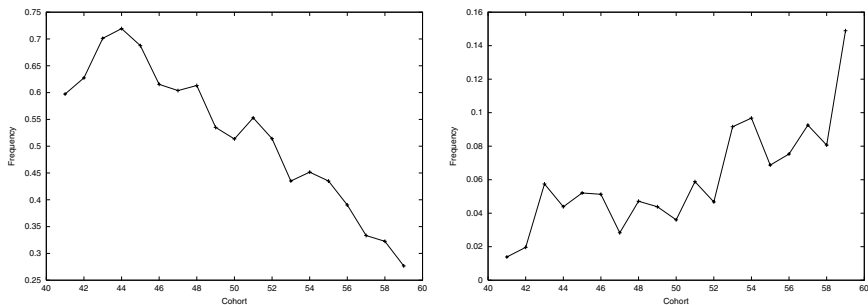


Fig. 4. The trends behind a very simple pattern and a very complex pattern. The left graph shows the negative trend in people that started their first union when they married. The pattern behind the right graph is explained in the text.

erage frequency in the entire population) were inspected manually, looking at the most interesting rules first. The temporal change of interesting patterns was visualized by plotting the frequency of occurrence of the pattern over 1-year cohorts from 1940 to 1960. This was very useful for assessing whether the found anomaly for one age group represented a trend throughout the years or only a temporary change in behavior. This phase could be enriched using a query language for trend shapes, such as the one proposed by [12].

4 Selected Results

In this section, we report on some of the results obtained in our case study. For these experiments we used the *Warmr* algorithm as implemented in the data mining tool *ACE-iiProlog* [6], version 1.1.6. We used the default settings for *Warmr*, except for a minimal support of 0.01 (the default is 0.1). No minimal confidence was specified for the generated rules.

The used background knowledge allowed the system to consider the events of child birth, start of a marriage and start of a union, and to order these events using inequalities, as well as test whether events occurred at the same time. To prevent infinite chains, the latter tests were restricted to ages originating from different event types (e.g., the system was not allowed to test the equality of two marriage dates), and only one equality test per pair of event types was allowed. In addition, all rules were initialized with the date of the first union. With this language, a *Warmr* run that went 16 levels deep took approximately 6 hours.

Figure 4 shows two discovered trends, one originating from a very simple rule, and the second from a fairly complex rule. The first pattern is the relation that people formed their first union when they married. This pattern was found with a negative deviation of 8.07 for the cohort 1956–1960, and with a positive deviation of 7.05 for the cohort 1941–1945. Together with its counter-part (Fig. 2), this pattern showed the strongest deviation of all discovered rules. Near these two

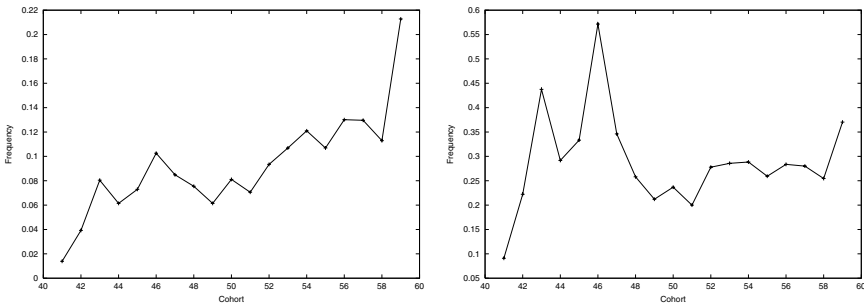


Fig. 5. People that have a child between their first union and a marriage over all people (left) and the same group over those people that had a marriage after their first union.

rules were several similar rules, that add minor restrictions to these rules (e.g., marrying at the time of the first union and having a child at a later time).

The second rule is shown below; it was found at level 7, with negative deviation of 2.17. Only a few rules at levels ≥ 8 survived the filtering process and none of them had a deviation of more than 2 standard deviations.

```
gen55(A) :- subject(A), first_union_at_age(A,B),
             married_at_age(A,C),C>B, child_at_age(A,D),D>C,
             child_at_age(A,E),E>B,
             married_at_age(A,F),F>E, child_at_age(A,G),G>E]
```

The first part of this rule states that persons satisfying this pattern married after their first union and had a child thereafter. The second part specifies that they had a child after their first union and after this child, they had both a marriage and a second child. Note that not all of the events in this rule are strictly ordered, and it is sometimes left open whether they refer to identical or different events. For example, both marriages referred to in the rule could be bound to the same marriage, in which case the pattern describes women that had at least two children, at least one of them before the marriage. However, all 6 event predicates could also refer to different events. The partial ordering of these events makes these patterns more general and may result in interesting combinations of subgroups.

More complicated rules have to be interpreted with caution, as can be seen from Fig. 5. Its left graph shows the trend for people that have a child between their first union and a marriage as discovered by the system, but the second graph, which normalizes this number over the number of people that had their first union before they married, shows that trend was mostly due to the trend in the normalizing pattern.

5 Lessons Learnt

An interesting, somewhat unexpected result of our explorations was that adding domain knowledge not only slows down the system but does not necessarily

yield more interesting results. The reason for this is that frequently the same relationships were also found in the form of the conjunctions of the operational predicates that were used to define the high-level predicates in the first place. So while we started our exploration by adding high-level predicates that appeared to be interesting to us (such as `child_out_of_wedlock_at_age/2`), we eventually ended up in removing most of these patterns because they unnecessarily slowed down the system’s performance.⁴ Warmr currently allows to formulate simple, syntax-oriented constraints (such as “do not add literal $p(X)$ if the literal $q(X)$ already occurs”), but this is insufficient for our application, where, e.g., the transitivity of $<$ causes many dependencies that cannot be avoided using the current constraint specification language. A more powerful language for expressing semantic constraints is clearly a promising topic for further research.

In particular for higher values of the frequency threshold, we often encountered the problem that negative deviations are often missed by the system. The reason for this is that patterns that occur significantly less frequently than expected are often below the frequency threshold. This is not a big problem in our application because a negative deviation for, e.g., the early cohorts 1941–1945 is typically paired with a positive deviation for the later cohorts 1956–1960, which is easily detected by the system. As we visually inspect the found patterns over all cohorts, it does not matter which deviation is found by the system.

In general, however, these issues may hint at an important shortcoming: in domains where frequency thresholds cannot be applied for effectively reducing the number of candidates (or can only reduce them at the expense of losing interesting patterns) the level-wise search pioneered by Apriori might in fact not be a suitable choice. In particular if the search is performed in memory, alternative search strategies [5,16] might be considered because of their more flexible pruning strategies. It is an open research problem which strategy is more appropriate for multi-relational data mining systems like Warmr.

6 Related Work

The problem of detecting temporal change in frequent patterns by grouping objects over time was also studied by [4]. In their application, they discovered trends in student admissions to UCI in the years 1993–1998. Their approach was limited to detecting changes in propositional patterns (which they called contrast sets), while we consider change in temporal patterns.

The problem of discovering frequent episodes in event sequences could also be solved by other techniques [3,13]. These could then be post-processed with similar techniques for detecting deviations over time. In fact, [12] discuss such a two-phase solution to the problem of discovering trends in text databases. The first phase consists of discovering frequent phrases in documents, i.e., in sequences of words using the advanced sequence discovery algorithm described in [14]. In the

⁴ The problem occurs less frequently in propositional tasks, but there too it is recognized that if such dependencies do exist, one needs special techniques to handle them [7].

second phase, the frequencies of the phrases over given time groups are determined, and their shape can be queried using the query language described in [2], which could, in principle, be replaced by our technique of detecting significant deviations from the mean over different periods of time.

The above-mentioned approaches basically treat the problem as a basket analysis problem with time as a special, designated variable that allows to integrate multiple baskets into single rules, as long as certain temporal constraints (e.g., a maximum window size) are followed. Our approach, however, is more general in the sense that it does not give a special role to time. Temporal relations are represented in the same way as other domain-dependent knowledge (even though this was not the main focus of this particular application). As a consequence, we are not only searching for sequential patterns (i.e., strict temporal orders of the form $A < B < C$), but for more general, graph-like structures (such as $A < B$ and $A < C$).

Several people have recognized the importance of taking dependencies into account when searching for association rules, and proposed solutions, e.g., in the form of *itemset closures* [7]; some of that work is currently being extended to first-order association rules [De Raedt; personal communication], but we have no knowledge of existing publications in this area.

7 Conclusions

In this paper, we demonstrated a way for exploiting the generality of *Warmr*, a multi-relational data mining system, for the task of discovering temporal changes in sequential patterns. The generality of *Warmr* allows a straight-forward encoding of time-dependent information and a seamless integration with additional background knowledge, which facilitates ease of experimentation and flexibility in incorporating new knowledge. In particular, during our experimentation, we frequently changed the system's view of the data, e.g., by censoring recent cohorts and late events in people's life courses. Such changes could be handled by simple changes in the background knowledge, while the underlying data representation could remain the same.

On the other hand, our current experiments clearly show the need for handling dependencies in first order association rule mining in general. A specifically interesting consequence of this is that the use of background knowledge may hurt the performance of systems such as *Warmr*, rather than improving it. This is an important issue, as the possibility to use background knowledge is one of the advantages of ILP approaches. In the current version of *Warmr*, dependencies can to some extent be handled using syntactical constraints on clauses, but this is clearly insufficient to handle semantic dependencies between background knowledge definitions (e.g. the transitivity of $<$). Further research is needed to address such problems in first-order association rule mining.

Acknowledgements. Hendrik Blockeel is a post-doctoral fellow of the Fund for Scientific Research (FWO) of Flanders. The Austrian Research Institute for

Artificial Intelligence is supported by the Austrian Federal Ministry of Education, Science and Culture. The authors wish to thank the Advisory Group of the FFS programme of comparative research for its permission, granted under identification number 75, to use the FFS data on which this study is based. The views expressed in this paper are the authors' own views and do not necessarily represent those of the Max Planck Institute for Demographic Research.

References

1. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, (eds.) *Advances in Knowledge Discovery and Data Mining*, pp. 307–328. AAAI Press, 1995.
2. R. Agrawal, G. Psaila, E. L. Wimmers, and M. Zait. Querying shapes of histories. In *Proceedings of the 21st Conference on Very Large Databases (VLDB-95)*, Zürich, Switzerland, 1995.
3. R. Agrawal and R. Srikant. Mining sequential patterns. In *Proceedings of the International Conference on Data Engineering (ICDE)*, Taipei, Taiwan, 1995.
4. S. D. Bay and M. J. Pazzani. Detecting group differences: Mining contrast sets. *Data Mining and Knowledge Discovery*, 2001. To appear.
5. R. J. Bayardo Jr. and R. Agrawal. Mining the Most Interesting Rules. In *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 145–154, 1999.
6. H. Blockeel, L. Dehaspe, B. Demoen, G. Janssens, J. Ramon, and H. Vandecasteele. Executing query packs in ILP. In J. Cussens and A. Frisch (eds.) *Proceedings of the 10th International Conference on Inductive Logic Programming (ILP-2000)*, pp. 60–77, London, UK, 2000. Springer-Verlag.
7. J.-F. Boulicaut and A. Bykowski. Frequent closures as a concise representation for binary data mining. In *Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD-00)*, pp. 62–73, Kyoto, Japan. Springer-Verlag.
8. S. Chakrabarti, S. Sarawagi, and B. Dom. Mining surprising patterns using temporal description length. In *Proceedings of the 24th Conference on Very Large Databases (VLDB-98)*, New York, NY, 1998.
9. L. Dehaspe and H. Toivonen. Discovery of frequent Datalog patterns. *Data Mining and Knowledge Discovery*, 3(1):7–36, 1999.
10. S. Džeroski and N. Lavrač (eds.). *Relational Data Mining: Inductive Logic Programming for Knowledge Discovery in Databases*. Springer-Verlag, 2001. To appear.
11. J. Hipp, U. Güntzer, and G. Nakhaeizadeh. Algorithms for association rule mining – a general survey and comparison. *SIGKDD explorations*, 2(1):58–64, June 2000.
12. B. Lent, R. Agrawal, and R. Srikant. Discovering trends in text databases. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining (KDD-97)*, Newport Beach, CA, 1997.
13. H. Mannila, H. Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.
14. R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *Proceedings of the 5th International Conference on Extending Database Technology (EDBT)*, Avignon, France, 1996.

15. L. J. G. van Wissen and P. A. Dykstra (eds.). *Population Issues: An Interdisciplinary Focus*. Kluwer Academic/Plenum Publishers, New York, 1999.
16. G. I. Webb. Efficient search for association rules. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2000)*, pp. 99–107, Boston, MA, 2000.