In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) Proc. Principles of Data Mining and Knowledge Discovery (PKDD), 6th European Conference, Helsinki, Finland, August 2002, p. 275-286

Spatial Subgroup Mining Integrated in an Object-Relational Spatial Database

Willi Klösgen and Michael May

Fraunhofer Institute for Autonomous Intelligent Systems
Knowledge Discovery Team
D-53754 Sankt Augustin, Germany
{willi.kloesgen, michael.may}@ais.fraunhofer.de

Abstract. SubgroupMiner is an advanced subgroup mining system supporting multirelational hypotheses, efficient data base integration, discovery of causal subgroup structures, and visualization based interaction options. When searching for dependencies between subgroups and a target group, spatial subgroups with multirelational descriptions are explored. Search strategies of data mining algorithms are efficiently integrated with queries in an object-relational query language and executed in a database to enable scalability for spatial data.

1 Introduction: Mining Spatial Subgroups

The goal of spatial data mining is to discover spatial patterns and to suggest hypotheses about potential generators of such patterns. In this paper we focus on spatial patterns from the perspective of the subgroup mining paradigm. Subgroup Mining [7],[8],[9] is used to analyse dependencies between a target variable and a large number of explanatory variables. Interesting subgroups are searched that show some type of deviation, e.g. subgroups with an over proportionally high target share for a value of a discrete target variable, or a high mean for a continuous target variable.

This paper introduces the *SubgroupMiner*, an advanced subgroup mining system supporting multirelational hypotheses, efficient data base integration, discovery of causal subgroup structures, and visualization based interaction options. The goal is to provide a spatial mining tool applicable in a wide range of circumstances.

In this paper we focus on a representational issue that is at the heart of the whole approach: Representing spatial subgroups using an object-relational query language by embedding part of the search algorithm in a spatial database system (SDBS). Thus the data mining and the visualization in a Geographic Information System (GIS) share the same data. While this approach embraces the full complexity and richness of the spatial domain, most approaches to Spatial Data Mining export and pre-process the data from a SDBS. Our approach results in significant improvements in all stages of the knowledge discovery cycle:

1. **Data Access:** Subgroup Mining is partially embedded in a spatial database, where analysis is performed. No data transformation is necessary and the same data is

used for analysis and mapping in a GIS. This is important for the applicability of the system since pre-processing of spatial data is error-prone and complex.

2. **Pre-processing and analysis**: SubgroupMiner handles both numeric and nominal target attributes. For numeric explanatory variables on-the-fly discretisation is performed. Spatial and non-spatial joins are executed dynamically.
3. **Post-processing and Interpretation:** Similar subgroups are clustered according to degree of overlap of instances to identify multicollinearities. A Bayesian network between subgroups can be inferred to support causal analysis.
4. **Visualisation.** SubgroupMiner is dynamically linked to a GIS, so that spatial subgroups are visualized on a map. This allows the user to bring in background knowledge into the exploratory process, to perform several forms of interactive sensitivity analysis and to explore the relation to further variables and spatial features.

The paper is organized as follows. In Section 2, the representation of spatial data and spatial subgroups is discussed. Section 3 focuses on database integration using a sufficient statistics approach. Due to space restrictions, we will not elaborate on post-processing and visualization in this paper, but Section 4 puts the discussion into context by presenting an application example. Finally related work is summarized.

2 Representation of Spatial Data and of Spatial Subgroups

Representation of spatial data. Most modern Geographic Information Systems use an underlying Database Management System for data storage and retrieval. While both relational and object-oriented approaches exist, a hybrid approach based on object-relational databases is becoming increasingly popular. Its main features are:

1. A *spatial data base S* is a set of relations $R_1,...,R_n$ such that each relation R_i in S has a geometry attribute G_i or an attribute A_i such that R_i can be linked (joined) to a relation R_k in S having a geometry attribute G_k.
2. A *geometry attribute* G_i consists of ordered sets of *x-y*-coordinates defining points, lines, or polygons.
3. Different types of spatial objects (e.g. streets, buildings) are organized in different relations R_i, called *geographical layers***.** Each layer can have its own set of attributes A_1,..., A_n, called *thematic data*, and at most one geometry attribute G.

This representation extends a purely relational scheme since the geometry attribute is non-atomic. One of its strengths is that a query can combine spatial information with attribute data describing objects located in space.

For querying multirelational spatial data a spatial database adds an operation called the *spatial join*. A spatial join links two relations each having a geometry attribute based on distance or topological relations (*disjoint*, *meet*, *equal*, *inside*, *contains*, *covers*, *coveredBy*, *overlap*) [2]. For supporting spatial joins efficiently, special purpose indexes like KD-trees or Quadtrees are used.

Pre-processing vs. dynamic approaches. A GIS representation is a *multi-relational* description using non-atomic data types (the geometry) and applying operations from computational geometry to compute the relation between spatial objects. Since most machine learning approaches rely on single-relational data with atomic data types only, they are not directly applicable to this type of representation. To apply them, a possibility is to pre-process the data and to join relevant variables from secondary tables to a single target table with atomic values only. The join process may include spatial joins, and may use aggregation. The resulting table can be analysed using standard methods like decision trees or regression. While this approach may often be practical, it simply sidesteps the challenges posed by multi-object-relational datasets.

In contrast, Malerba et al. [17] pre-process data stored in a object-relational database to represent it in a *deductive database*. Thus, spatial intersection between objects is represented in a derived relation `intersects(X,Y)`. The resulting representation is still multi-relational, but only atomic values are permitted, and relationships in Euclidean space are reduced to qualitative relationships.

Extracting data from a SDBS S to another format has its disadvantages:

1. The set of possible joins L between relations in S constrain the hypothesis space H. Since all spatial joins between geographical layers according to the topological relations described by Egenhofer [2] are meaningful, the set L is prohibitively large. If L includes the distance relation with a real-valued distance parameter, there are infinitely many possible joins. Thus, for practical and theoretical reasons, after pre-processing only part of the original space H will be represented in the transformed space H', so that the best hypothesis in H may not be part of H'.
2. Conversely, much of the pre-processing, which is often expensive in terms of computation and storage, may be unnecessary since that part of the hypothesis space may never be explored, e.g. because of early pruning.
3. Pre-processing leads to redundant data storage, and in applications where data can change due to adding, deleting or updating, we suffer the usual problems of non-normalized data storage well-known from the database literature.
4. Storing the respective data in different formats makes a tight integration between a GIS and the data mining method much more difficult to achieve.

An advantage of pre-processing is that once the data is pre-processed the calculation has not to be repeated, e.g. by constructing join indices [3]. However, a dynamic approach can get similar benefits from caching search results, and still have the original hypothesis space available.

For these reasons, our approach to spatial data mining relies on using a SDBS without transformation and pre-processing. Tables are dynamically joined. Variables are selected during the central search of a data mining algorithm, and inclusion depends on intermediate results of the process. Expensive spatial joins are performed only for the part of the hypothesis space that is really explored during search.

Spatial subgroups. Subgroups are subsets of analysis objects described by selection expressions of a query language, e.g. simple conjunctional attributive selections, or multirelational selections joining several tables. *Spatial subgroups* are described by a spatial query language that includes operations on the spatial references of objects. A spatial subgroup, for instance, consists of the enumeration districts of a city intersected by a river. A spatial predicate (*intersects*) operates on the coordinates of the spatially referenced objects *enumeration districts* and *rivers*.

Hypothesis Language. The *domain* is an object-relational database schema $S = \{R_1, ..., R_n\}$ where each R_i can have at most one geometry attribute G_i. Multirelational subgroups are represented by a concept set $C = \{C_i\}$, where each C_i consists of a set of conjunctive attribute-value-pairs $\{C_i.A_1=v_1,..., C_i.A_n=v_n\}$ from a relation in S, a set of links $L=\{L_i\}$ between two concepts C_j, C_k in C via their attributes A_m, A_k, where the link has the form $C_i.A_m$ θ $C_k.A_m$, and θ can be '=', a distance or topological predicate (*disjoint, meet, equal, inside, contains, covered by, covers, overlap, interacts*).

For example, the subgroup "districts with high rate of migration and unemployment crossed by the M60" is represented as

C={{district.migration=high,district.unemplyoment=high}, {road.name='M60'}}

L= {{ interacts(district.geometry, road.geometry)}}

Existential quantifiers of the links are problematic when many objects are linked, e.g. many persons living in a city or many measurements of a person. Then the condition that one of these objects has a special value combination will often not result in a useful subgroup. In this case, conditions based on *aggregates* such as counts, shares or averages will be more useful [12], [14].

These aggregation conditions are included by aggregation operations (*avg*, *count*, *share*, *min*, *max*, *sum*) for an attribute of a selector. An average operation on a numerical attribute additionally needs labeled intervals to be specified.

C = (district.migration = high; building.count(id) = high)

L = (spatially_interact(district.geometry, building.geometry))

Extension: Districts with many buildings.

For *buildings.count(id),* labels *low, normal, high* and intervals are specified.

Multirelational subgroups have first been described in Wrobel [23] in an ILP setting. Our hypothesis language is more powerful due to numeric target variables, aggregations, and spatial links. Moreover, all combinations of numeric and nominal variables in the independent and dependent variables are permitted in the problem description. Numeric independent variables are discretised on the fly. This increases applicability of subgroup mining.

Representation of spatial subgroups in query languages. Our approach is based on an *object-relational* representation. The formulation of queries depends on non-atomic data-types for the geometry, spatial operators based on computational geometry, grouping and aggregation. None of these features is present in basic relational algebra or Datalog. An interesting theoretical framework for the study of spatial databases are constraint databases [15], which can be formulated as (non-trivial) extensions of relational algebra or Datalog. However, using SQL is more direct and much more practical for our purposes. The price to pay is that SQL extended by object-relational features is less amendable for theoretical analysis (but see [16]). For calculating spatial relationships spatial extensions of DBMS like *Oracle Spatial* can be used.

For database integration, it is necessary to express a multirelational subgroup as defined above as a query of a database system. The result of the query is a table representing the extension of the subgroup description. One part of this query defines the subset of the product space according to the *l* concepts and *l*-1 link conditions. The *from* part includes the *l* (not necessarily different) tables and the *where* part the *l*-1 link conditions (as they are given as strings or default options in the link specification; spatial extensions of SQL apply a special syntax for the spatial operations). Additionally the *where* part includes the conditions associated to the definition of selectors of concepts. Then the aggregation conditions are applied and finally the product space is projected to the target table (using the DISTINCT feature of SQL).

The complexity of the SQL statement is low for a single relational subgroup. Only the attributive selectors must be included in the *where* part of the query. For multirelational subgroups without aggregates and no distinction of multiple instances, the *from* part must manage possible duplicate uses of tables, and the *where* part includes the link conditions (transformed from the link specification) and the attributive selectors. For aggregation queries, a nested two-level select statement is necessary, first constructing the multirelational attributive part and then generating the aggregations. Multiple instances of objects of one table are treated by including the table in the *from* part several times and the distinction predicate in the *where* part.

The space of subgroups to be explored within a search depends on the specification of a relation graph which includes tables (object classes) and links. For spatial links the system can automatically identify geometry attributes by which spatial objects are linked, since there is at most one such attribute. A relation graph constrains the multirelational hypothesis space in a similar way as attribute selection constrains it for single relations.

3. Database Integration of Subgroup Mining

Subgroup mining search. This paper focuses on database integration of spatial subgroup mining. The basic subgroup mining algorithm is well-documented and only summarized here. Different subgroup patterns (e.g. for continuous or discrete target variables), search strategies and quality functions are described in [8], [9].

The search is arranged as an iterated *general to specific, generate and test procedure*. In each iteration, a number of parent subgroups is expanded in all possible ways, the resulting specialized subgroups are evaluated, and the subgroups are selected that are used as parent subgroups for the next iteration step, until a prespecified iteration depth is achieved or no further significant subgroup can be found. There is a natural partial ordering of subgroup descriptions. According to the partial ordering, a specialization of a subgroup either includes a further selector to any of the concepts of the description or introduces an additional link to a further table.

The statistical significance of a subgroup is evaluated by a quality function. As a standard quality function, SubgroupMiner uses the classical binomial test to verify if the target share is significantly different in a subgroup:

$$\frac{p-p_0}{\sqrt{p_0(1-p_0)}}\sqrt{n}\sqrt{\frac{N}{N-n}} \quad (1)$$

This *z*-score quality function based on comparing the target group share in the subgroup (p) with the share in its complementary subset balances four criteria: size of subgroup (n), relative size of subgroup with respect to total population size (N), difference of the target shares (p-p_0), and the level of the target share in the total population (p_0). The quality function is symmetric with respect to the complementary subgroup. It is equivalent to the χ^2-test of dependence between subgroup S and target group T, and the correlation coefficient for the (binary) subgroup and target group variables. For continuous target variables and the deviating mean pattern, the quality function is similar, using mean and variance instead of share p and binary case variance $p_0(1$-$p_0)$.

Evaluation of contingency tables. To evaluate a subgroup description, a contingency table is statistically analyzed (tab 1). It is computed for the extension of the subgroup description in the target object class. To get these numbers, a multirelational query is forwarded to the database. Contingency tables must be calculated in an efficient way for the very many subgroups evaluated during a search task.

Tab 1. Contingency table for target migration=high vs. unemployment=high.

		Target		
		migration = high	¬migration = high	
Subgroup	unemployment=high,	16	19	35
	¬unemployment=high	47	496	543
		63	515	578

Sufficient statistics approach. We use a two-layer implementation [22], where evaluation of contingency tables is done in SQL, while the search manager is implemented in Java. A sufficient statistics approach is applied by which a single SQL query provides the aggregates that are sufficient to evaluate all successor subgroups. In the data server layer, within *one pass* over the database all contingency tables are calculated that are needed for the next search level. Thus not each single hypothesis queries the database, but a (next) population of hypotheses is treated concurrently to optimize data access and aggregation needed by these hypotheses. The search manager receives only aggregated data from the database so that network traffic is reduced. Besides offering scaling potential, such an approach includes the advantage of development ease, portability, and parallelization possibilities.

Construction of query. The central component of the query is the selection of the multirelational parent subgroup. This is why representation of multirelational spatial subgroup in SQL is required. To generate the aggregations (cross tables) for a parent subgroup, a nested select-expression is applied for multirelational parents. From the product table, first the expansion attribute(s), key-attribute for the primary table and target attribute are projected and aggregates calculated for the projection. Then the cross tables (target versus expansion attribute) are calculated. Efficient calculation of several cross tables, however, is difficult in SQL-implementations. An obvious solution could be based on building the union of several group-by operations (of target and expansion attributes). Although, in principle, several parallel aggregations could be calculated in one scan over the database, this is not optimised in SQL implementations. Indeed each union operation unnecessarily performs an own scan over the database. Therefore, to achieve a scalable implementation (at least for single relational and some subtypes of multirelational or spatial applications), the group-by operation has been replaced by explicit sum operations including case statements combining the different value combinations. Thus for each parent, only one scan over the database (or one joined product table) is executed. Further optimisations are achieved by combining those parents that are in the same joined product space (to eliminate unnecessary duplicate joins).

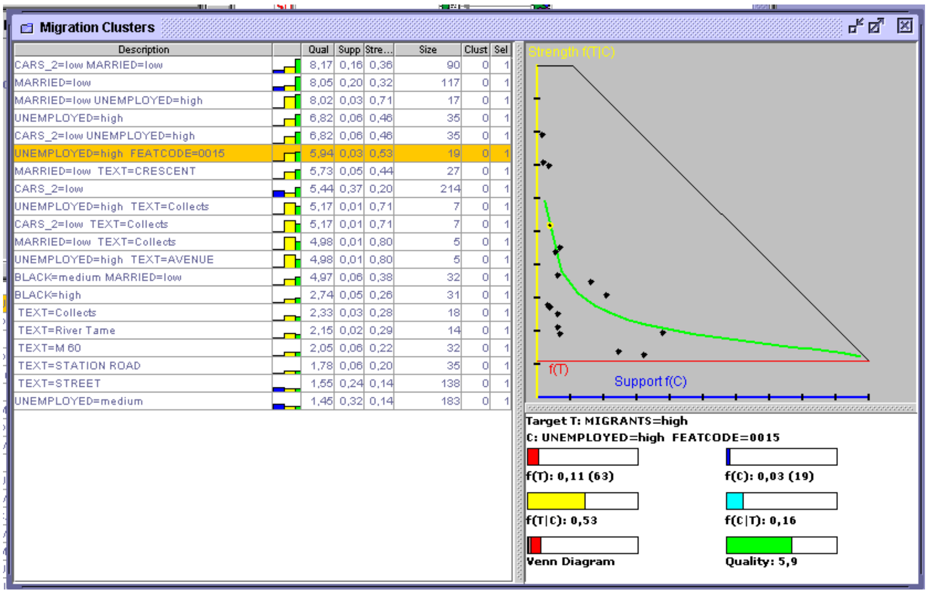
4 Application and Experiments

Application to UK Census Data. In this section we put the previous discussion in context. We describe a practical example that shows the interaction between spatial subgroup mining and a GIS mapping tool. The application has been developed within the IST-SPIN!-project, that integrates a variety of spatial analysis tools into a spatial data mining platform based on Enterprise Java Beans [18,19]. Besides Subgroup Mining these are Spatial Association rules [17], Bayesian Markov Chain Monte Carlo and the Geographical Analysis Machine GAM [20].

Our application are UK 1991 census data for Stockport, one of the ten districts in Greater Manchester, UK. Census data provide aggregated information on demographic attributes such as persons per household, cars per household, unemployment, migration, long-term-illness. Their lowest level of aggregation are so called *enumeration districts*. Also available are detailed geographical layers, among them streets, rivers, buildings, railway lines, shopping areas. Data are provided to the project by the partners Manchester University and Manchester Metropolitan University.

Assume we are interested in enumeration districts with a high migration rate. We want to find out how those enumeration districts are characterized, and especially what distinguishes them from other enumeration districts not having a high migration rate. Spatial subgroup discovery helps to answer this question by searching the hypothesis space for interesting deviation patterns with respect to the target attribute.

The target attribute T is then *high migration rate*. A concept C found in the search is *Enumeration districts with high unemployment crossed by a railway line*. Note that this subgroup combines spatial and non-spatial features. The deviation pattern is that the proportion of districts satisfying the target T is higher in districts that satisfy pattern C than in the overall population ($p(T|C)>p(T)$).



Description		Qual	Supp	Stre...	Size	Clust	Sel
CARS_2=low MARRIED=low		8,17	0,16	0,36	90	0	1
MARRIED=low		8,05	0,20	0,32	117	0	1
MARRIED=low UNEMPLOYED=high		8,02	0,03	0,71	17	0	1
UNEMPLOYED=high		6,82	0,06	0,46	35	0	1
CARS_2=low UNEMPLOYED=high		6,82	0,06	0,46	35	0	1
UNEMPLOYED=high FEATCODE=0015		5,94	0,03	0,53	19	0	1
MARRIED=low TEXT=CRESCENT		5,73	0,05	0,44	27	0	1
CARS_2=low		5,44	0,37	0,20	214	0	1
UNEMPLOYED=high TEXT=Collects		5,17	0,01	0,71	7	0	1
CARS_2=low TEXT=Collects		5,17	0,01	0,71	7	0	1
MARRIED=low TEXT=Collects		4,98	0,01	0,80	5	0	1
UNEMPLOYED=high TEXT=AVENUE		4,98	0,01	0,80	5	0	1
BLACK=medium MARRIED=low		4,97	0,06	0,38	32	0	1
BLACK=high		2,74	0,05	0,26	31	0	1
TEXT=Collects		2,33	0,03	0,28	18	0	1
TEXT=River Tame		2,15	0,02	0,29	14	0	1
TEXT=M 60		2,05	0,06	0,22	32	0	1
TEXT=STATION ROAD		1,78	0,06	0,20	35	0	1
TEXT=STREET		1,55	0,24	0,14	138	0	1
UNEMPLOYED=medium		1,45	0,32	0,14	183	0	1

Target T: MIGRANTS=high
C: UNEMPLOYED=high FEATCODE=0015

f(T): 0,11 (63)
f(C): 0,03 (19)
f(T|C): 0,53
f(C|T): 0,16
Venn Diagram
Quality: 5,9

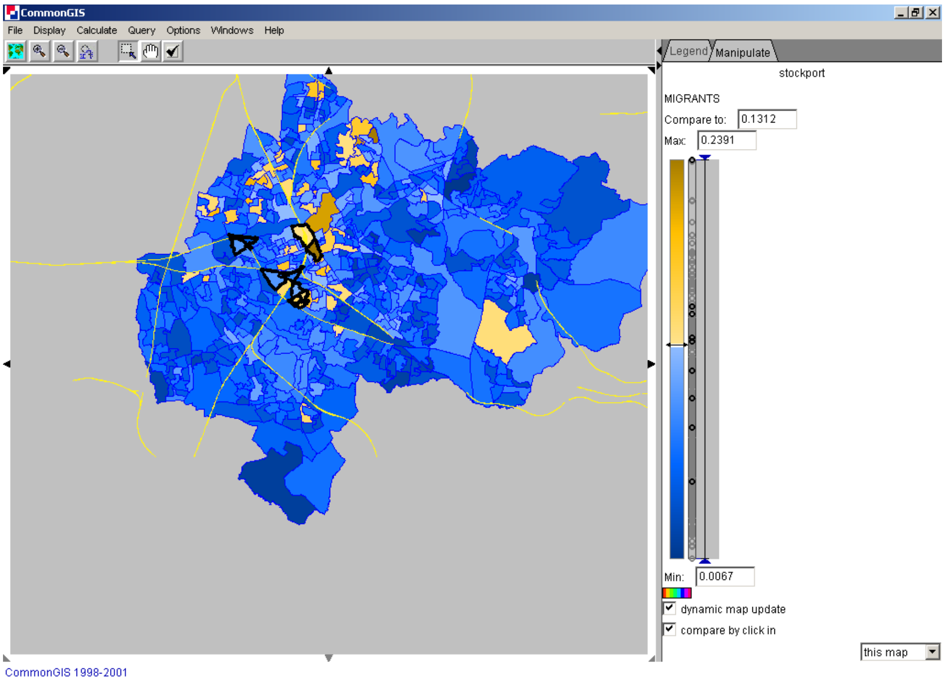
Fig 1. Overview on subgroups found showing the subgroup description (left). Bottom right side shows a detail view for the overlap of the concept *C* (e.g. located near a railway line) and the target attribute *T* (high unemployment rate). The window on the right top plots p(*T*|*C*) against p(*C*) for the subgroup selected on the left and shows *isolines* as theoretically discussed in [8].

Another – this time purely spatial – subgroup found is *Enumeration district crossed by motorway M60*. This spatial subgroup induces a homogenous cluster taking the form of a physical spatial object. Spatial objects can often act as causal proxies for causally relevant attributes not part of the search space.

A third – this time non-spatial – subgroup found is *Enumeration districts with low rate of households with 2 cars and low rate of married people*. By spotting the subgroup on the map we note that is a spatially inhomogeneous group, but with its center of gravity directed towards the center of Stockport.

The way data mining results are presented to the user is essential for their appropriate interpretation. We use a combination of cartographic and non-cartographic displays linked together through simultaneous dynamic highlighting of the corresponding parts. The user navigates in the list of subgroups (fig. 1), which are dynamically highlighted in the map window (fig. 2). As a mapping tool, the SPIN!-platform integrates the CommonGIS system [1], whose strengths lies in the dynamic manipulation of spatial statistical data. Figure 1 and 2 show an example for the migrant scenario, where the subgroup discovery method reports a relation between districts with high migration rate and high-unemployment.

Scalability Results. Spatial analysis is computationally demanding. In this section we summarize preliminary results on scalability. The simplest subgroup query provides all the information sufficient for the evaluation of all single relational successors of a set of single relational parent subgroup descriptions. These descriptions are constructed for one iteration step of specialization in the target object class including only attributes from this target class. Especially when the target object class contains many attributes that are used for descriptions of subgroups and the other (secondary) object classes contain much fewer attributes, these descriptions will constitute the main part of the search space.



Compare to: 0.1312

Max: 0.2391

Min: 0.0067

☑ dynamic map update

☑ compare by click in

this map

Fig. 2. Enumeration districts satisfying the subgroup description C (high unemployment rate and crossed by a railway line) are highlighted with a thicker black line. Enumeration districts also satisfying the target (high migration rate) are displayed in a lighter color.

The performance requirements will strongly increase when multirelational subgroups are evaluated, because joins of several tables are needed. Two types of multirelational queries can be distinguished following two specialization possibilities. A multirelational subgroup can be specialized by adding a further conjunctive selector to any of its concepts or by adding a further concept in the concept sequence via a new link. The multirelational case involving many new links still requires many dynamic joins of tables and is not generally scalable.

The one-scan solution is nearly linear in the number of tuples in the one relational case (independent of the number of attributes, if this number is small thus that the cross table calculation is dominated by the organization of the scan. For many attributes, computation time is also proportional in the total number of discrete attribute values), and calculating sufficient statistics needs for large databases several orders less time than the version based on union operators which needs many scans. A detailed analysis of computation times for the different query versions and types of multirelational applications is performed in a technical report [11].

A further optimisation is achieved by substituting the parallel cross table calculation in SQL by a stored procedure, which is run (as the SQL query) in the database. It sequentially scans the (product) table and incrementally updates the cells of the cross tables. We are currently evaluating the performance of this solution compared with the SQL implementation. The SQL implementation, however, is easy portable to other data base systems. Only some specific expressions (case statement) must be adapted.

5 Related work

Subgroup mining methods have been first extended for multirelational data by Wrobel [22]. SubgroupMiner allows flexible link conditions, an extended definition of multirelational subgroups including numeric targets, aggregate operations in links, spatial predicates, and is database integrated.

Knobbe et al. [12] and Krogel et al. [14], although in a non-spatial domain, apply a static pre-processing step that transforms a multirelational representation into a single table. Then, standard data mining methods such as decision trees can be applied. Static pre-processing typically has the disadvantages summarized in sec. 2 and must be restricted to avoid generating impractically large propositional target datasets.

Malerba and Lisi [17] apply an ILP approach for discovering association rules between spatial objects using first order logic (FOL) both for data and subgroup description language. They operate on a deductive relational database (based on Datalog) that extracts data from a spatial database. This transformation includes the precalculation of all spatial predicates, which as before ([12]) can be unnecessarily complex. Also the logic-based approach cannot handle numeric attributes and needs discretizations of numerical attributes of (spatial) objects. On the other side, the expressive power of Datalog allows to specify prior knowledge, e.g. in the form of rules or hierarchies. Thus the hypothesis language is in this respect more powerful than the hypothesis language in SubgroupMiner. In [17] the same UK census data set is used, as both approaches are developed within the scope of the IST-10536-SPIN! project [4].

In [12] it is pointed out that aggregations (*count*, *min*, *avg*, *sum* etc.) are more powerful than ILP approaches to propositionalisation, which typically induce binary features, expressed in FOL restricting to existence aggregates.

Ester et al. [3] define neigborhood graphs and neighborhood indices as novel data structures useful for speeding up spatial queries, and show how several data mining methods can be built upon them. Koperski et al. [13] propose a two-level search for association rules, that first calculates coarse spatial approximations and performs more precise calculations to the result set of the first step.

Several approaches to extend SQL to support mining operations have been proposed, e.g. to derive sufficient statistics minimizing the number of scans [5]. Especially for association rules, a framework has been proposed to integrate the query for association rules in database queries [6]. For association rules, also architectures for coupling mining with relational database systems have been examined [21]. Siebes and Kersten [23] discuss approaches to optimize the interaction of subgroup mining (KESO) with DBMSs. While KESO still requires a large communication overhead between database system and mining tool, database integration for subgroup mining based on communicating sufficient aggregates has not been implemented before.

6 Conclusion and Future Work

Two-layer database integration of multirelational subgroup-mining search strategies has proven as an efficient and portable architecture. Scalability of subgroup mining for large datasets has been realized for single relational and multi-relational applications with a not complex relation graph. The complexity of a multirelational application mainly depends of the number of links, the number of secondary attributes to be selected, the depth of the relation graph, and the aggregation operations. Scalability is also a problem, when several tables are very large. Some spatial predicates are expensive to calculate. Then sometimes a grid for approximate (quick) spatial operations can be selected that is sufficiently accurate for data mining purposes.

We are currently investigating caching options to combine static and dynamic links, so that links can be declared as static in the relation graph. The join results are stored and need not be calculated again. The specification of textual link conditions and predicates in the relation graph that are then embedded into a complex SQL query has proven as a powerful tool to construct multirelational spatial applications.

Spatial analysis requires further advancements of subgroup mining systems. Basic subgroup mining methods discover correlations or dependencies between a target variable and explanatory variables. Spatial subgroups typically overlap with attributive subgroups. For the actionability of spatial subgroup mining results, it is important to analyze the causal relationships of these attributive and spatial variables. These relationships are analyzed by constraint-based Bayesian network techniques. Details of the causal analysis methods for subgroup mining are presented in [10].

Acknowledgments. Work was partly funded by the European Commission under IST-1999-10563 SPIN! – Spatial Mining for Data of Public Interest. We would like to thank Jim Petch, Keith Cole and Mohammed Islam from Manchester University and Chrissie Gibson from Manchester Metrop. Univ. for making available census data.

References

1. G. Andrienko, Andrienko, N. Interactive Maps for Visual Data Exploration, *International Journal of Geographical Information Science* 13(5), 355-374, 1999
2. M. J. Egenhofer. Reasoning about Binary Topological Relations, *Proc. 2nd Int. Symp. on Large Spatial Databases*, Zürich, Switzerland, 143-160,1991
3. M. Ester, Frommelt, A., Kriegel, H.P, Sander, J. Spatial Data Mining: Database Primitives,

Algorithms and Efficient DBMS Support, *Data Mining and Knowledge Discovery,* 2, 1999

4. IST-10536-SPIN!-project web site, http://www.ccg.leeds.ac.uk/spin/
5. G. Graefe, Fayyad, U., Chaudhuri, S. On the efficient gathering of sufficient statistics for classification from large SQL databases. *Proc. of the 4th Intern. Conf. on Knowledge Discovery and Data Mining,* Menlo Park: AAAI Press, 204-208, 1998
6. T. Imielinski, Virmani, A. A Query Language for Database Mining. *Data Mining and Knowledge Discovery,* Vol. 3, Nr. 4, 373–408, 2000
7. W. Klösgen. Visualization and Adaptivity in the Statistics Interpreter EXPLORA. In *Proceedings of the 1991 Workshop on KDD,* ed. Piatetsky-Shapiro, G., 25-34, 1991
8. W. Klösgen. Explora: A Multipattern and Multistrategy Discovery Assistant. *Advances in Knowledge Discovery and Data Mining,* eds. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Cambridge, MA: MIT Press, 249–271, 1996
9. W. Klösgen. Subgroup Discovery. Chapter 16.3 in: *Handbook of Data Mining and Knowledge Discovery,* eds. Klösgen, W., Zytkow, J., Oxford University Press, New York, 2002
10. W. Klösgen. Causal Subgroup Mining. To appear.
11. W. Klösgen, May, M. Database Integration of Multirelational Subgroup Mining. Technical Report. Fraunhofer Institute AIS, Sankt Augustin, Germany, 2002
12. A. Knobbe, de Haas, M., Siebes, A. Propositionalisation and Aggregates. In *Proc.* PKDD 2001, eds. De Raedt, L., Siebes, A., Berlin:Springer, 277-288, 2001
13. K. Koperski, Adhikary, J. Han, J. Spatial Data Mining, Progress and Challenges, Vancouver, Canada, Technical Report, 1996
14. M. Krogel, Wrobel, S. Transformation-Based Learning Using Multirelational Aggregation, *Proc. ILP 2001*, eds. Rouveirol, C., Sebag, M., Springer, 142-155, 2001
15. G. Kuper, Libkin, L., Paredaens (eds.). *Constraint Databases*, Berlin:Springer, 2000
16. L. Libkin, Expressive Power of SQL, *Proc. of the 8th International Conference on Database Theory (ICDT01)*, eds. Bussche, J, Vianu, V., Berlin:Springer, 1-21, 2001
17. D. Malerba, Lisi, F.. Discovering Associations between Spatial Objects: An ILP Application. Proc. *ILP 2001,* eds. Rouveirol, C., Sebag, M., Berlin: Springer, 156-163, 2001
18. M. May. Spatial Knowledge Discovery: The SPIN! System. *Proc. of the 6th EC-GIS Workshop, Lyon*, ed. Fullerton, K., JRC, Ispra, 2000
19. M. May, Savinov, A. An Architecture for the SPIN! Spatial Data Mining Platform, *Proc. New Techniques and Technologies for Statistics, NTTS 2001*, 467-472, Eurostat, 2001
20. Openshaw, S., Turton, I., Macgill, J. and Davy, J. Putting the Geographical Analysis Machine on the Internet, in Gittings, B. (ed.) *Innovations in GIS 6,* 1999
21. S. Sarawagi, Thomas, S., Agrawal, R. Integrating Association Rule Mining with Relational Database Systems. *Data Mining and Knowledge Discovery*, 4, 89-125, 2000
22. A. Siebes, Kersten, M. KESO: Minimizing Database Interaction. *Proc. of the 3rd Intern. Conf. on Knowledge Discovery and Data Mining,* Menlo Park: AAAI Press, 247-250, 1998
23. S. Wrobel. An Algorithm for Multi-relational Discovery of Subgroups. In *Proc. of First PKDD*, eds. Komorowski, J., Zytkow, J., Berlin:Springer, 78-87, 1997