

# Ordinal Association Rules for Error Identification in Data Sets<sup>1</sup>

Andrian Marcus  
Jonathan I. Maletic<sup>2</sup>  
K. I. Lin

Division of Computer Science  
The Department of Mathematical Sciences  
The University of Memphis  
Campus Box 526429  
Memphis, TN 38152  
Phone: (901) 678-3140

amarcus@memphis.edu, jmaletic@memphis.edu, linki@msci.memphis.edu

## Paper ID: 257

**Abstract:** Association rules are a fundamental class of patterns that exist in data. These patterns have been widely utilized (e.g., market basket analysis) and extensive studies exist to find efficient association rule mining algorithms. Special attention is given in literature to the extension of binary association rules (e.g., ratio, quantitative, generalized, multiple-level, constrained-based, distance-based, composite association rules). A new extension of the Boolean association rules, *ordinal association rules*, that incorporates ordinal relationships among data items, is introduced. These rules are used to identify possible errors in data. An algorithm that finds these rules and identifies potential errors in data is proposed. The results of applying this method to a real-world data set are given.

**Keywords:** Data Mining, Data Cleansing, Association Rules, Ordinal Rules

---

<sup>1</sup> This research is supported in part by a grant from the Office of Naval Research.

<sup>2</sup> Contact Author

## **1. Introduction**

The quality of a large real world data set depends on a number of issues [English'99, Wang'95, Wang'96], but the source of the data is the crucial factor. Data entry and acquisition is inherently prone to errors both simple and complex. Much effort can be given to this front-end process, with respect to reduction in entry error, but the fact often remains that errors in a large data set are common. Unless an organization undertakes extreme measures in an effort to avoid data errors the field errors rates are typically around 5% or more [Orr'98, Redman'98]. To address this problem some organizations spend millions of dollars per year to detect data errors [Redman'96]. Coupled with the fact that the manual process of data cleansing is also laborious, time consuming, and prone to errors, methods to automatically detect or assist in detecting errors is of great interest.

The work presented here extends data mining techniques (i.e., Boolean association rules) and applies these extensions to the problem of detecting errors in databases. The types of errors we are trying to detect lie outside the standard integrity constraints. The method presented here aims to uncover relationships (e.g., numerical ordering or equality) between attributes that commonly occur over the data set and then use this information to identify attributes that do not conform to these uncovered (partial) orderings. While this constrains the types of errors that can be detected by our method, it does work to address part of the overall data cleansing process. Few methods exist that directly tackle this problem and our method represent a useful and practical approach. The proposed method is intended to be use in conjunction with existing ones to solve part of the data cleansing problem (i.e., identification of potential errors in data).

The following section summarizes the directly related work on association rules. This work forms the foundation for ordinal rules. Then a formal definition of ordinal rules is given and the details of the algorithms used for implementation. Results of identifying and using ordinal rules to find potential errors in a real world data set are then discussed.

## 2. Related Work

In this section, we survey the related work. Our work is directly related to association rule mining and therefore, we first outline the work in this area. Next, we discuss previous work on using database/data mining techniques for data cleansing (cleaning).

### 2.1 Data mining and Association rules

The term, *association rule* was first introduced by Agrawal et al. [Agrawal'93] in the context of market basket analysis. In this kind of analysis, the data set is defined as the *basket data*  $B = \{b_1, b_2, \dots, b_n\}$ , where each *basket*  $b_i \subseteq I$  is a collection of *items*, and where  $I = \{i_1, i_2, \dots, i_k\}$  is a set of  $k$  elements. An *association rule* in the database  $B$  is defined as follows.

$i_1 \Rightarrow i_2$  is an *association rule* if:

$i_1$  and  $i_2$  occur together in at least  $s\%$  of the  $n$  baskets, where  $s$  is the *support* of the rule; and of all baskets containing  $i_1$ , at least  $c\%$  contain  $i_2$ , where  $c$  is the *confidence* of the rule.

This generalizes to rules of type  $A \Rightarrow B$ , where  $A$  and  $B$  are disjoint sets of items instead of single items. An example of such rule in real life is: “50% of people who buy diapers, also buy beer, and 20% of all buyers buy diapers.” In this case, *diapers* and *beer* are the items, the confidence of the rule is 50%, and the support of the rule is 20%. Association rules of this type are also referred to in the literature as *classical* or *Boolean* association rules. For the purposes of this paper Boolean association rules between single items are considered as basis for subsequent definitions and the rules between item sets are considered generalizations.

Since its introduction, the problem of mining association rules from large databases has been subject of numerous studies. Some of them focus on developing faster algorithms for the classical method and/or adapting the algorithms to various situations, like parallel mining and incremental mining. Hipp et al. [Hipp'00] provides an excellent survey on this topic. Another direction is to define rules that modify some conditions of the classical rules to adapt to new applications (like imposing constraints on item sets, or adapting rules to time-series data). The

work of Ng et al. [Ng'98] contains a comprehensive list of references related to the above-mentioned studies.

Association rules as defined above apply to Boolean or categorical data. In practice, the information in many, if not most, databases is not limited to categorical attributes, but also contains much quantitative data. Unfortunately, the definition of categorical association rules does not translate directly to the case of quantitative attributes. It is therefore necessary to provide a definition of association rules for the case of a database containing quantitative attributes. Srikant et al. [Srikant'96] extended the categorical definition to include quantitative data. The basis for their definition is to build categorical events from the quantitative data by considering intervals of the numeric values. Thus, each basic event is either a categorical item or a range of numerical values. Such rules are called *quantitative association rules* [Srikant'96]. A more formal definition is given there, and confidence and support of the rule are slightly redefined. An example of such rule is “People who spent on bread between \$3-\$5, and on milk at the same time between \$1-\$2, usually spend between \$1.5-\$2 on butter in the same transaction”.

A stronger set of rules is defined in [Korn'98] as *ratio-rules*. A rule under this framework is expressed in the form: “Customers typically spend 1: 2: 5 dollars on bread: milk: butter”. This time the strength of the rule allows multiple applications, including data cleansing and outlier detection. However, the paper does not exploit this idea. It is only mentioned that the power of ratio-rules to reconstruct data could support the data cleansing process. Eigen system analysis is used to find these rules and induces the strength of the rules as well as a computational overhead.

A series of generalizations of quantitative association rules are defined in [Aumann'99]. From the perspective of this paper, it is useful to utilize the formulation of a general association rule as

$$\textit{population-subset} \Rightarrow \textit{interesting-behavior}$$

A further generalization is made in [Padmanabhan'98] where a general form of rules are considered: *body*  $\Rightarrow$  *head*, where *body* is a conjunction of atomic conditions of the form *attribute*

*op value*, and *head* is a single atomic condition of the form *attribute op value*, where  $op \in \{\leq, =, \geq\}$ .

Various other data mining problems are also related to data cleansing. Of special interest is the problem of outlier detection – where the goal is to find out exception in large data sets. Many different approaches have been proposed. Many of them are based on the notion of distance-based outliers [Knorr'98, Ramaswamy'00]. Other techniques such as *FindOut* [Yu'01] combine clustering and outlier detection.

In this paper, we focus on rules that describe ordinal relationships among attributes. The work by [Guillaume'00], which independently uses the term ordinal rules, is not related, and focuses on the development of ordinal objective measures. Likewise, in [Buchter'99], the authors produce association rules on ordinal data and their goal is more akin to the quantitative rules mentioned above.

## 2.2 Data cleansing techniques

Data are deemed “unclean” due to many different reasons. Thus, various techniques have been developed to tackle the problem of data cleansing. Largely, data cleansing is an interactive approach, as different sets of data have different rules determining the validity of data. Thus, many systems allow users to specify rules and transformations needed to clean the data. For example, [Raman'01] proposed the use of an interactive spreadsheet to allow users to perform transformation, while [Galhardas'99] allow users to specify rules and conditions on an SQL-like interface.

Apart from general approaches, in many cases there are specific data cleansing problems that need to be solved. For example, a customer database may contain many entries for the same person, spelled differently (e.g., J. Doe vs. John Doe). The merge/purge problem [Hernandez'98] aims at removing such duplicates. Typical database techniques like sort-merge or nested-loop can be applied.

All the approaches above typically require the user to specify the rules beforehand. While it is reasonable in many cases, it is also important that the data cleansing system be able to automatically discover rules and detect errors. This is the approach of this paper. One must note that the error identification part of the data cleansing problem is difficult and no single method can solve it entirely or completely automatically.

### 3. Ordinal Association Rules

The presented extensions and generalizations of the association rule concept can be used for the identification of possible erroneous data items with certain modifications. These considerations lead to a new extension of the association rule – *ordinal association rules* or simply *ordinal rules*.

The objective here is to find ordinal relationships between record attributes that tend to hold over a large percentage of records. If attribute A is less than B most of the time then a record that contains a B that is less than A may be in error. One flag on B may not mean much, but if a number of ordinal rules that deal with B are broken, the likelihood of error goes up. The following more formally defines this concept.

*Definition.* Let  $R = \{r_1, r_2, \dots, r_n\}$  a set of records, where each record is a set of  $k$  attributes  $(a_1, \dots, a_k)$ . Each attribute  $a_i$  in a particular record  $r_j$  has a value  $\phi(r_j, a_i)$  from a domain  $D$ . The value of the attribute may also be empty and is therefore included in  $D$ . The following relations (partial orderings) are defined over  $D$ , namely less or equal ( $\leq$ ), equal ( $=$ ) and, greater or equal ( $\geq$ ) all having the standard meaning.

Then  $(a_1, a_2, a_3, \dots, a_m) \Rightarrow (a_1 \mu_1 a_2 \mu_2 a_3 \dots \mu_{m-1} a_m)$ , where each  $\mu_i \in \{\leq, =, \geq\}$ , is a an *ordinal association rule* if:

- 1)  $a_1 \dots a_m$  occur together (are non-empty) in at least  $s\%$  of the  $n$  records, where  $s$  is the *support* of the rule;

- 2) and, in a subset of the records  $R' \subseteq R$  where  $a_1 \dots a_m$  occur together and  $\phi(r_j, a_1) \mu_1 \dots \mu_{m-1} \phi(r_j, a_m)$  is true for each  $r_j \in R'$ . Thus  $|R'|$  is the number of records that the rule holds for and the *confidence*,  $c$ , of the rule is the percentage of records that hold for the rule  $c = |R'|/|R|$ .

As can be seen from the definition, ordinal rules can be defined over two or more attributes. The work here currently focuses on ordinal rules that use only two attributes.

Ordinal association rules bear some similarity with the above-mentioned extensions of Boolean association rules. However, they are better suited to the problem of identifying possible errors in the type of data sets being analyzed for the following reasons:

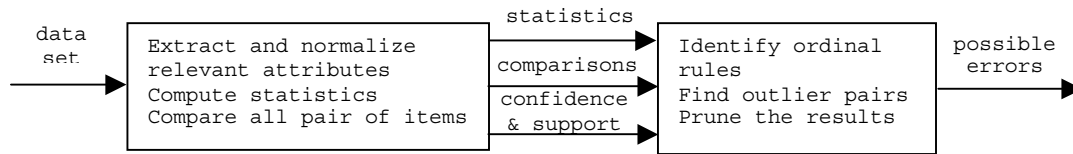
- They are easier and faster to compute than quantitative association rules or ratio-rules.
- Although they are weaker than quantitative association rules or ratio-rules, they give very good results in the case of finding (partial) ordering trends.
- Distance-based association rules (over interval data) [Miller'97] could be also used in this for this problem, but it is inherently hard to find the right intervals in the absence of specific domain knowledge, and the methods tend to be rather expensive.

The process to identify potential errors in data sets using ordinal association rules is composed of the following steps:

- Prepare the data
- Find ordinal rules with a minimum confidence  $c$ .
- Identify data attributes that broke the rules and can be considered potential errors.

Here, the manner in which support of a rule is important differs from the typical data-mining problem. In error detection, we are looking for rules that hold for large number of records. But, a rule may hold for a small percentage of the overall records and not represent an error. Here, we assume all the discovered rules that hold for more than two records represent valid possible partial orderings. Future work will investigate user-specified minimum support. However, this will only change the number of initially identified patterns. Since only pairs of items are

considered, there can be at most  $C(M,2)$  patterns. Where  $M$  is the number of attributes (fields) of the record and  $N$  is the number of records in the database.



**Figure 1: The two main components of the prototype tools. The boxes correspond to each component and show the performed tasks. The arrows indicate the data flow in the system.**

Figure 1 describes two distinct components of the implementation we used for proof of concept. The first component 1) normalizes the data if necessary; 2). Computes for all records the minimum, maximum, mean, standard deviation, number of empty items of each attribute; 3) Compares, for each record, every pair of attributes. Only a single pass over the data set is necessary. The following information is then available for every pair of attributes  $(x, y)$  in all the records:

- Number of comparisons were performed,
- Number of times  $\phi(r_j, x) > \phi(r_j, y)$ ,
- Number of times  $\phi(r_j, x) = \phi(r_j, y)$ ,
- Number of times  $\phi(r_j, x) < \phi(r_j, y)$ ,
- Number of times  $\phi(r_j, x)$  is empty,
- Number of times  $\phi(r_j, y)$  is empty.

An array with the results of the comparisons is maintained in the memory. The array will only have  $C(M,2)$  elements.. Figure 2 contains the algorithm for this step. The complexity of this step is only  $O(N*M^2)$  where  $N$  is the number of records in the data set, and  $M$  is the number of fields. Usually  $M$  is much smaller than  $N$  (see next section for an example). The results of this algorithm are written to a temporary file for use in the next step of processing.

In the second step, the patterns are identified based on the chosen minimum confidence. There are several researched methods to determine the strength including interestingness and statistical

```

Algorithm compare items.
for each record in the data base (1..N)
    extract the record
    normalize or convert data
    update statistics
    for each attribute x in (1 .. M-1)
        for each attribute y in (x+1 ... M-1)
            compare the values in x and y
            update the comparisons array
        end for.
    end for.
    output the record with the normalized data
end for.
output the comparisons array
end algorithm.

```

**Figure 2: The algorithm for the first step. Complexity is  $O(N)$ .**

significance of a rule (minimum support and minimum confidence, chi-square test, etc.). Using confidence intervals [Johnson'98] to determine the minimum confidence is currently under investigation. However, previous work on the data set [Maletic'00] used in our experiment showed that the distribution of the data was not normal. Therefore, the minimum confidence was chosen empirically, several values were considered and the algorithm was executed. The results indicated that a minimum confidence between 98.8 and 99.7 provide best results (less number of false negative and false positives).

The second component extracts from the temporary file and stores in memory the data associated with the patterns. There are three types of patterns that are identified:

- $x = y$  with confidence  $c$ ;
- $x \geq y$  with confidence  $c$ ; and
- $x \leq y$  with confidence  $c$ .

This is done with a single pass over the comparisons file (complexity  $O(C(M,2))$ ). Then for each record in the data set, each pair of attributes that correspond to a pattern it is check to see if the values in those fields within the relationship indicated by the pattern. If they are not, each field is marked as possible error. Of course, in most cases only one of the two values will actually be an error. Once every pair of fields that correspond to a rule is analyzed, the average number of possible error marks for each marked field is computed. Only those fields that are marked as possible errors more times than the average are finally marked as containing high probability

errors. Again, the average value was empirically chosen as threshold to prune the possible errors set. Other methods to find such a threshold, without using domain knowledge or multiple experiments, are under investigation. The time complexity of this step is  $O(N * C(M, 2))$ , and the analyzes of each record is done entirely in the main memory. Figure 3 shows the algorithm used in the implementation of the second component. The results are stored so that for each record and field where high probability errors were identified, the number of marks is shown.

```

Algorithm analyze records.
for each record in the data base (1...N)
    for each pattern in the pattern array (1...C(M,2) maximum)
        determine rule type and pairs
        compare item pairs
        if pattern NOT holds
            then mark each item as possible error
    end for.
    compute average number of marks
    select and output only the high probability marked errors
end for.
end algorithm.

```

**Figure 3: Algorithm for the second step. Time complexity  $O(N)$ .**

## 4. Experiments & Results

Two sets of experiments were executed to date. The first experiment is for proof of concept and we manually developed the data set. The second data set is from a real world application from the U.S. Navy.

### 4.1. Proof of Concept Results

We used synthetically generated data to validate the algorithms. A set of data with 100 attributes and 10,000 records was randomly generated. Each attribute had a known distribution and range. Then a number of errors were introduced. A number of these errors broke the existing ordering in data and additionally, a number were statistical outliers. Using statistical measures (e.g., means, standard deviation, etc.) some of these errors were not identifiable. Using the ordinal rules in the manner described above, all of the errors that broke the orderings were identified. By combining the two methods (i.e., identification of statistical outliers and ordinal rules) all the induce errors were detected. The number of false positives and false negatives was in direct correlation with the chosen confidence for ordinal rules. The best value for the confidence is

very data dependent and we are currently investigating methods to identify this value automatically.

#### **4.2. Naval Personnel Data Set Results**

The second set of experiments was performed on real world data supplied by the Naval Personnel Research, Studies, and Technology (NPRST). The data set is part of the officer personnel information system including midshipmen and officer candidates. Similar data sets are in use at personnel records departments in companies all over the world. The size of this particular data set is around 230,000 records with over 700 attributes. A large percentage of the attributes in the data set are empty. Many of the attributes represent dates of particular events (e.g., first enlistment, promotion dates, etc.). The rest of the attributes typically represent domain specific codes. For the experiment, a subset of this data set was chosen representing an important class of Navy personnel and contained 32,721 records with 226 attributes.

The attributes of type date are the only ones examined. Given that these attributes are all of the same type, the comparison operators make perfect sense and the generated ordinal rules map directly into the problem domain. For instance, all dates in an individual record should be greater than their date of birth. While this may seem an obvious relationship, the use of ordinal rules should automatically uncover such relationships (obvious or not).

This is not to say that ordinal rules cannot be applied to data of different types. Interesting patterns among fields of different types can emerge and be useful in the detection of errors. Alternatively, the data can be normalized in some manner and we are conducting a third set of experiments that uses a different data set and attempts to normalize and use attributes of different types.

Figure 4 summarizes some of the results of this experiment. At this time the results are under investigation by the domain experts at NPRST. Following their validation, fine-tuning of the method will be possible based on the number of actual errors that were identified among the possible ones. The third set of experiments, mentioned above, has data of a less sensitive nature

and actual erroneous data elements will be shown in a future report that will focus on more heterogeneous data.

An important issue at this point was the estimation of the probability that one particular data element identified as being potentially erroneous is in fact an error. Two measures at record level are of interest in this respect: number of rules broken by a record ( $BR_i$ ) and the number of attributes (fields) in a record that broke at least one rule and is marked as possible error ( $FR_i$ ) as described in the previous paragraph. For both measures the subscript  $i$  is a record identifier (the value of the key attribute(s)). In this respect those records that have a high number of broken rules that are broken by a relatively smaller number of attributes are most likely to contain actual records. Thus, for each record marked with possible errors we computed  $BR_i/FR_i$ . The larger

```

Data set characteristics
- 226 fields of date type
- 32721 records

Ordinal rules
- 17408 ordinal rules identified with 99.5 minimum confidence
- 1589 ordinal rules have confidence less than 100 (are broken)

Possible errors
- Records with possible errors (broke at least one rule): 5053
- Records that broke more than one rule: 2649
- Records with  $BR_i/FR_i > 2.0$ : 179
- Max number of rules broken by one record: 89
- Max number of fields in one record that broke some rule: 85
- Max number of rules broken by one field: 100

```

**Figure 4: Results of the experiment.**

the value of this measure, the higher the probability that the record contains actual errors. Within these records, those fields that are marked as breaking the most rules are the top candidates for being errors.

The results were compared with results of standard statistician outlier detection methods obtained in previous work [Maletic'00]. These possible errors not only matched many of the previously discovered ones, but also yielded (as expected) a number of errors unidentified by the other methods. The distribution of the data dramatically influenced the error identification process in the previous utilized methods. Ordinal rules are not influenced as much by the distribution of the data and is proving to be more robust.

Another interesting aspect of ordinal rules is that a Boolean association rule is implied (contained) within. Although for data cleansing purposes the ordering and quantitative nature of the rules are of interest, the embedded association rule provides valuable information about the distribution of the empty data elements. Missing data is an important problem in data cleansing and data analysis in general. This is also a matter under consideration as another possible use of the ordinal rules.

## 5. Conclusions

Association rule mining proves to be useful in identifying not only interesting patterns for fields such as market basket analysis or census data, but also, by extension to ordinal association rules, patterns that uncover errors in other kind of data sets. The classical notion of association rules has been extended to include ordinal relationships between pairs of numerical attributes, thus defining ordinal association rules. This extension allows the uncovering of stronger rules that yielded potential errors in the data set, while keeping the computation simple and efficient. These results are currently under detailed investigation by domain experts for the data set. This research address two important steps in defining and building a framework for automated data cleansing namely defining error types and automatically identify possible errors.

The results of the current experiments are promising and new ones are in progress to extend the use of the ordinal rules to cope with attributes of different types and also to find relationships between rules that involve more than two attributes.

## 6. References

- [Agrawal'93] Agrawal, R., Imielinski, T., and Swami, A., (1993), "Mining Association rules between Sets of Items in Large Databases", in Proceedings of ACM SIGMOD International Conference on Management of Data, Washington D.C., May, pp. 207-216.
- [Aumann'99] Aumann, Y. and Lindell, Y., (1999), "A Statistical Theory for Quantitative Association Rules", in Proceedings of KDD99, San Diego, CA, pp. 261 - 270.
- [Buchter'99] Buchter, O. and Wirth, R., (1999), "Exploration of Ordinal Data Using Association Rules", *Knowledge and Information Systems*, vol. 1, no. 4, November.

- [English'99] English, J., (1999), "Plain English on Data Quality", DM Review, Webzine, Date Accessed: 02/10/99, <http://www.dmreview.com>.
- [Galhardas'99] Galhardas, H., Florescu, D., Shasha, D., and Simon, E., (1999), "An Extensible Framework for Data Cleaning".: Institute National de Recherche en Informatique et en Automatique.
- [Guillaume'00] Guillaume, S., Khenchaf, A., and Briand, H., (2000), "Generalizing Association Rules to Ordinal Rules", in Proceedings of The Conference on Information Quality (IQ2000), Massachusetts Institute of Technology, October 20-22, pp. 268-282.
- [Hernandez'98] Hernandez, M. and Stolfo, S., (1998), "Real-world Data is Dirty: Data Cleansing and The Merge/Purge Problem", *Data Mining and Knowledge Discovery*, vol. 2, no. 1, January, pp. 9-37.
- [Hipp'00] Hipp, J., Guntzer, U., and Nakhaeizadeh, G., (2000), "Algorithms for Association Rule Mining - A General Survey and Comparison", *SIGKDD Explorations*, vol. 2, no. 1, June 2000, pp. 58-64.
- [Johnson'98] Johnson, R. A. and Wichern, D. W., (1998), *Applied Multivariate Statistical Analysis*, 4th ed., Prentice Hall.
- [Knorr'98] Knorr, E. M. and Ng, R. T., (1998), "Algorithms for Mining Distance-Based Outliers in Large Datasets", in Proceedings of 24th Int. Conf. Very Large Data Bases, New York, pp. 392-403.
- [Korn'98] Korn, F., Labrinidis, A., Yannidis, K., and Faloutsos, C., (1998), "Ratio Rules: A New Paradigm for Fast, Quantifiable Data Mining", in Proceedings of 24th VLDB Conference, New York, pp. 582--593.
- [Maletic'00] Maletic, J. I. and Marcus, A., (2000), "Data Cleansing: Beyond Integrity Checking", in Proceedings of The Conference on Information Quality (IQ2000), Massachusetts Institute of Technology, October 20-22, pp. 200-209.
- [Miller'97] Miller, R. J. and Yang, Y., (1997), "Association Rules over Interval Data", *ACM SIGMOD*, vol. 26, no. 2, May, pp. 452-461.
- [Ng'98] Ng, R. T., Lakshmanan, S., L. V., Han, J., and Pang, A., (1998), "Exploratory Mining and Pruning Optimizations of Constrained Association Rules", in Proceedings of ACM SIGMOD, Seattle, Washington, June, pp. 13-24.

- [Orr'98] Orr, K., (1998), "Data Quality and Systems Theory", *CACM*, vol. 41, no. 2, February, pp. 66-71.
- [Padmanabhan'98] Padmanabhan, B. and Tuzhilin, A., (1998), "A Belief-Driven Method for Discovering Unexpected Rules", in Proceedings of KDD 98, pp. 94-100.
- [Raman'01] Raman, V. and Hellerstein, J. M., (2001), "Potter's wheel: an interactive data cleaning system", in Proceedings of 27th International Conference on Very Large Databases, Rome, pp. To appear.
- [Ramaswamy'00] Ramaswamy, S., Rastogi, R., and Shim, K., (2000), "Efficient Algorithms for Mining Outliers from Large Data Sets", in Proceedings of 2000 ACM SIGMOD Intl. Conference on Management of Data, Dallas, pp. 427-438.
- [Redman'96] Redman, T., (1996), *Data Quality for the Information Age*, Artech House.
- [Redman'98] Redman, T., (1998), "The Impact of Poor Data Quality on the Typical Enterprise", *CACM*, vol. 41, no. 2, February, pp. 79-82.
- [Srikant'96] Srikant, R., Vu, Q., and Agrawal, R., (1996), "Mining Association Rules with Item Constraints", in Proceedings of SIGMOD International Conference on Management of Data, Montreal, Canada, June, pp. 1-12.
- [Wang'95] Wang, R., Storey, V., and Firth, C., (1995), "A Framework for Analysis of Data Quality Research", *IEEE Transactions on Knowledge and Data Engineering*, vol. 7, no. 4, August, pp. 623-639.
- [Wang'96] Wang, R., Strong, D., and Guarascio, L., (1996), "Beyond Accuracy: What Data Quality Means to Data Consumers", *Journal of Management Information Systems*, vol. 12, no. 4, Spring, pp. 5-34.
- [Yu'01] Yu, D., Sheikholeslami, G., and Zhang, A., (2001), "FindOut: Finding Outliers in Very Large Datasets", *Knowledge and Information Systems*, pp. To appear.