Data Min Knowl Disc (2008) 16:313–348
DOI 10.1007/s10618-008-0092-3

Using association rules to mine for strong approximate dependencies

Daniel Sánchez · José María Serrano · Ignacio Blanco · Maria Jose Martín-Bautista · María-Amparo Vila

Received: 2 August 2006 / Accepted: 24 January 2008 / Published online: 30 March 2008

Abstract In this paper we deal with the problem of mining for approximate dependencies (AD) in relational databases. We introduce a definition of AD based on the concept of association rule, by means of suitable definitions of the concepts of item and transaction. This definition allow us to measure both the accuracy and support of an AD. We provide an interpretation of the new measures based on the complexity of the theory (set of rules) that describes the dependence, and we employ this interpretation to compare the new measures with existing ones. A methodology to adapt existing association rule mining algorithms to the task of discovering ADs is introduced. The adapted algorithms obtain the set of ADs that hold in a relation with accuracy and support greater than user-defined thresholds. The experiments we have performed show that our approach performs reasonably well over large databases with real-world data.

Keywords Approximate dependencies · Association rules · Data mining · Relational databases

1 Introduction

Functional dependencies (FD) are integrity constrains based on real world restrictions, that are used in the design process of a relational database. Data in a relational database

Responsible editor: M. J. Zaki.

D. Sánchez (✉)
E.T.S.I.I.T., C/ Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain
e-mail: daniel@decsai.ugr.es

D. Sánchez · I. Blanco · M. J. Martín-Bautista · M.-A. Vila
Department of Computer Science and A.I., University of Granada, Granada, Spain

J. M. Serrano
Department of Informatics, University of Jaén, Jaén, Spain

are stored in a set of tables, each one consisting of a fixed scheme (set of attributes). An instance of a given scheme is a set of tuples (rows), where each tuple pertains to the cartesian product of the domain of the attributes in the scheme, verifying certain integrity constraints. Given a relational scheme $RE = \{At_1, \ldots, At_m\}$ and $V, W \subset RE$ verifying $V \cap W = \emptyset$, the FD "V determines W", $V \rightarrow W$, holds in RE if and only if for every instance r of RE

$$\forall t, s \in r \quad \text{if } t[V] = s[V] \text{ then } t[W] = s[W] \tag{1}$$

The meaning of such FD is that given a value for V in a tuple, we can predict the value for W in the same tuple. In this sense, a FD can be represented by a theory (set of rules) describing the associations between values of V and values of W.

There has been a lot of interest in mining for FD in relational databases (Bitton et al. 1989, Mannila and Räihä 1992, Savnik and Flach 1993, Mannila and Räihä 1994, Bell 1995, 1997, Gunopulos et al. 1997, Flach and Savnik 1999). This is a difficult task since one single exception breaks the dependence. However, if the number of exceptions is not very high, such "FD with exceptions" are showing us interesting regularities that hold in the data. Moreover, usual problems such as the presence of noisy data can hide a FD by introducing false exceptions. Hence, relaxing the rule (1) that defines a FD and finding such relaxed FDs has been recognized as an interesting goal (Bosc et al. 1997).

The concept of FD has been relaxed in several ways and for different purposes. An important class of relaxed FD are fuzzy functional dependencies (FFD). A FFD replaces some of the elements of the definition of FD (in the rule 1) by their fuzzy counterparts. For example, one kind of FFD replaces the equality of values of attributes by a degree of resemblance given by a fuzzy resemblance relation [see (Bosc et al. 1997) for a detailed study]. However, some definitions of FFD are more restrictive than FDs, in the sense that a relation satisfying a FFD also satisfies a FD between the same attributes, and most of the times they are oriented to database design. On the contrary, we are interested in FD as predictive and previously unknown models that hold (with few exceptions) in the real world and provide some information about it.

Approximate dependencies (AD) (also called "partial determinations" and "partial functional dependencies" in the literature) are smoothed FDs, in the sense that some exceptions to the rule (1) are allowed. What is relaxed here is the universal quantifier that appears in the rule (1). This kind of regularities are less restrictive than FDs (i.e. a relation satisfying a FD also satisfies an AD), and hence they are better suited for our purposes.

Approximate dependencies are very useful. First, as the result of a data mining process, they give us information about dependencies between attributes, a first global view of what's going on in our database. From this point of view they can be useful as a summary, since an approximate dependence summarizes the information of the set of rules that form its associated theory. These two possibilities have been employed for example in Calero et al. (2003), Berzal et al. (2003) and Calero et al. (2004b,c). ADs can be also employed for data fusion, since they tell us about possible relations between attributes, and data compression, since when an AD $V \rightarrow W$ holds, storing a table with columns V and W is equivalent to store V and the theory of the AD, plus the

exceptions. The information provided by ADs and its theory has been also employed as a data mining alternative to the classic statistical correspondence analysis in Sánchez et al. (2003) and Calero et al. (2004a), in particular in order to obtain measures of strength of certain relations, such as similarity and refinement, between partitions and combinations of partitions of the same set of objects.

The discussion about partial dependencies goes back to Lukasiewicz (1970). Since then, many papers about AD mining have been published (Pawlak 1991, Ziarko 1991, Piatetsky-Shapiro 1992, Kivinen and Mannila 1995, Pfahringer and Kramer 1995, Kramer and Pfahringer 1996, Huhtala et al. 1998, 1999, Lopes et al. 2002). The main problem addressed in these papers is the definition of measures for assessing ADs. Many different proposals are available. However, most of them consider accuracy measures only.

This is a problem since, starting from different perspectives, some authors have pointed out that accurate ADs can be misleading, trivial and/or uninteresting. As an example, if there are no pair of tuples in a table with the same value of a set of attributes V, the dependence $V \rightarrow W$ trivially holds for any set of attributes W with total accuracy. We shall call these dependencies *trivial dependencies* from now on.

A first reference to this problem appears in Pfahringer and Kramer (1995). In this paper an AD (called *partial determination*) is seen as a theory that can be used to describe one-to-one associations between values of the antecedent and the consequent. The objective is to find dependencies such that the corresponding theory is not too complex. The complexity is measured as the amount of bits needed to encode both the theory and the exceptions. Hence, complexity is related to the number of exceptions, the number of rules in the theory, and the size of the antecedent and consequent of each rule. In Pfahringer and Kramer (1995) and Kramer and Pfahringer (1996) it is shown that the complexity of the theory for a given AD can be equal to the complexity of the original set of data, even for totally accurate dependencies. This is the case of trivial dependencies, for instance. In this case, the discovered dependencies are not useful for data compression, for instance.

The same problem is discussed in Piatetsky-Shapiro (1992), where a close relation between the cardinality K of the domain of a certain V and the quality of a dependence of the form $V \rightarrow W$ is pointed out. Following (Piatetsky-Shapiro 1992), random permutations of the values of W in the tuples of a table r should break any dependence $V \rightarrow W$, because the associations among values of V and W are lost. But if K is close to $n = |r|$ the accuracy of the dependence is almost the same for any permutation, and hence the dependence could be unreal. In other words, there is no evidence supporting the dependence in the data, so it is not a reliable result of a data mining process. Again the worst case arise if $V \rightarrow W$ is a trivial dependence, because the accuracy is 1 for any permutation we could perform on the values of W.

In this paper we introduce a new definition of ADs as association rules (Agrawal et al. 1993) with the semantics of Eq. 1. This kind of association rules are different from the usual ones in relational tables; instead of considering items as pairs (attribute,value) and transactions as tuples, we consider items as attributes and transactions as pairs of tuples, in accordance with Eq. 1. This definition has several advantages. First, as we shall show, the support is a valid measure of the degree of triviality of a dependence. This alternative to the proposals in Piatetsky-Shapiro (1992), Pfahringer and Kramer

(1995) and Kramer and Pfahringer (1996) has the advantage that it can be used in order to prune the search during the mining process, discarding in advance trivial dependencies. In particular, and this is another advantage of the proposal, any association rule mining algorithm can be adapted for mining non-trivial ADs. In addition, the accuracy of an AD can be measured by using any accuracy measure for the corresponding ARs. Finally, using fuzzy extensions of association rules (Delgado et al. 2003, Dubois et al. 2006) it is possible to look for ADs when we have imprecision associated to the data in what we call *fuzzy approximate dependencies* (Berzal et al. 2005).

The paper is organized as follows. In the next section we introduce our new definition of AD, measures and properties, and we show its suitability to solve the problem of mining non-trivial accurate dependencies. In Sect. 3 we compare our new approach to some existing approaches. Section 4 shows how to adapt any AR mining algorithm for mining ADs. An empirical evaluation of the approach is provided in Sect. 5. Finally, Sect. 6 contains our conclusions and future work.

2 A new definition of approximate dependence

In the rest of the paper, we shall use the following notation. We shall note an AD in an instance r of a relational scheme RE as $V \rightarrow W$, where $V, W \subset RE$ with $V \cap W = \emptyset$. Also let $dom(V) = \{v_1, \ldots, v_K\}$ and $dom(W) = \{w_1, \ldots, w_M\}$ be the values of V and W appearing in r, respectively. Finally, let $n = |r|$, and $n_{v_i} = |\{t \in r | t[V] = v_i\}|$, and $n_{w_j} = |\{t \in r | t[W] = w_j\}|$, and $n_{v_i w_j} = |\{t \in r | t[V] = v_i \text{ and } t[W] = w_j\}|$.

2.1 Approximate dependencies as association rules

From (1) a FD "$V \rightarrow W$" is a rule that relates the presence of pairs of tuples with the same value of V to the presence of pairs of tuples with the same value of W, with total accuracy. This idea has been used for example in Lopes et al. (2002). On the other hand, ARs relate the presence of items in a transaction. Hence, we can consider a FD as an AR by introducing the following interpretations of the abstract concepts of item, itemset and transaction: let $RE = \{At_1, \ldots, At_m\}$ be a relational scheme and let r be an instance of RE such that $|r| = n$.

Definition 2.1 An item is an object associated with an attribute of RE. For every attribute $At_k \in RE$ we note it_{At_k} the associated item.

Let us remark that the item it_{At_k} associated to the attribute At_k is an abstract object that is independent of the instances of RE and the domain of At_k. Its meaning and utility will be explained after Definition 2.3.

Definition 2.2 Let $V \subseteq RE$. Then we introduce the itemset I_V to be

$$I_V = \{it_{At_k} | At_k \in V\}$$

A particular case is the set of all the items associated with RE,

$$I_{RE} = \{it_{At_1}, \ldots, it_{At_m}\}$$

Definition 2.3 We introduce the set of transactions T_r to be the following: for each pair of tuples $<t, s> \in r \times r$ there is a transaction $ts \in T_r$ verifying

$$it_{At_k} \in ts \Leftrightarrow t[At_k] = s[At_k]$$

Every transaction in T_r corresponds to a pair of tuples in r. The presence of an item it_{At_k} in a transaction ts means that the tuples t and s have the same value of At_k.

On this basis we characterize an AD as an AR as follows:

Definition 2.4 Let $V, W \subset RE$ such that $V \cap W = \emptyset$. An approximate dependence $V \rightarrow W$ in the relation r is an association rule $I_V \Rightarrow I_W$ in T_r.

This way, an association rule $it_{At_k} \Rightarrow it_{At_j}$ in T_r has exactly the same meaning as the functional dependence $At_k \rightarrow At_j$ in r. Using this transformation, we can look for approximate dependencies in a table r by looking for the corresponding association rules in T_r. Despite the fact that $|T_r| = |r \times r| = n^2$, it is possible to obtain ADs following our approach with complexity $O(n)$ with respect to the number of tuples, as we shall show in Sect. 4.

As it is well known, the support of an itemset I_V, $S(I_V)$, is the percentage of transactions containing the itemset. The support of a rule $I_V \Rightarrow I_W$, $S(I_V \Rightarrow I_W)$, is the support of the itemset formed by the union of the itemsets in the antecedent and consequent, and defines the percentage of data where the rule holds. Confidence is the classical accuracy measure for ARs, defined as

$$C(I_V \Rightarrow I_W) = \frac{S(I_V \Rightarrow I_W)}{S(I_V)} = \frac{S(I_V W)}{S(I_V)} \tag{2}$$

In order to assess the ADs, we can use the measures of accuracy and support of the corresponding ARs, in the following way:

Definition 2.5 The support and confidence of an AD $V \rightarrow W$ are the support and confidence of the corresponding AR $I_V \Rightarrow I_W$, i.e.,

$$S(V \rightarrow W) = S(I_V \Rightarrow I_W) \tag{3}$$

$$C(V \rightarrow W) = C(I_V \Rightarrow I_W) \tag{4}$$

In the same way, $S(V) = S(I_V)$ is the support of attribute V. An important property of this characterization is the following:

Proposition 2.1 *The dependence $V \rightarrow W$ is functional if and only if $C(I_V \Rightarrow I_W)$=1.*

Proof $C(I_V \Rightarrow I_W) = 1$ if and only if every pair of tuples that agree in V also agree in W, and that holds if and only if $V \rightarrow W$ is a FD. □

Let us remark that confidence is not the only accuracy measure for ARs existing in the literature. There are several alternative proposals that try to solve some of the inconveniences of confidence. We shall come back to this point in Sect. 2.3.

Table 1 Some data about three students

ID	Year	Course	Lastname
1	1991	3	Smith
2	1991	4	Smith
3	1991	4	Smith

Table 2 The set T_{r_3} of transactions for r_3

Pair	it_{ID}	it_{Year}	it_{Course}	$it_{Lastname}$
$\langle 1, 1\rangle$	1	1	1	1
$\langle 1, 2\rangle$	0	1	0	1
$\langle 1, 3\rangle$	0	1	0	1
$\langle 2, 1\rangle$	0	1	0	1
$\langle 2, 2\rangle$	1	1	1	1
$\langle 2, 3\rangle$	0	1	1	1
$\langle 3, 1\rangle$	0	1	0	1
$\langle 3, 2\rangle$	0	1	1	1
$\langle 3, 3\rangle$	1	1	1	1

Table 3 Some association rules in T_{r_3} corresponding to approximate dependencies in r_3

AR	Confidence	Support	AD
$\{it_{ID}\} \Rightarrow \{it_{Year}\}$	1	1/3	$ID \rightarrow Year$
$\{it_{Year}\} \Rightarrow \{it_{Course}\}$	5/9	5/9	$Year \rightarrow Course$
$\{it_{Year}, it_{Course}\} \Rightarrow \{it_{ID}\}$	3/5	1/3	$Year, Course \rightarrow ID$

To illustrate our new definition of AD, let r_3 be the relation in Table 1, and let $RE = \{ID,\ Year,\ Course,\ Lastname\}$. Then

$$I_{RE} = \{it_{ID}, it_{Year}, it_{Course}, it_{Lastname}\}$$

Table 2 shows the set of transactions T_{r_3}. Each row is the description of a transaction (associated with a pair of tuples), and each column is an item. The value 1 (resp. 0) in a cell means that the item is (resp. is not) in the transaction.

Table 3 contains some ARs that hold in T_{r_3}. They define ADs that hold in r_3. The confidence and support of the ARs in Table 3 measure the accuracy and support of the corresponding ADs.

One of the main advantages of our approach is that the support is a valid measure of triviality of an AD. Hence, looking for ADs with support above a high enough threshold, using any of the algorithms for mining AR existing in the literature, allow us to obtain non-trivial dependencies with the advantage that the support allows us

to prune the search, as in the case of ARs. We explain this point in detail in the next subsection.

2.2 Support of ADs and non-triviality

As detailed in the introduction, our goal is to avoid accurate but uninteresting ADs, in any of the (equivalent) senses of triviality, complexity of the underlying theory, or invariance to permutations of values in the consequent. These are different views of the same problem.

- An AD is trivial in a table when there are neither exceptions nor data where it holds. In this case, the dependence holds because there are no exceptions, but it is not supported by the data, meaning that we cannot guarantee that the AD is showing us a valid dependence that hold in the real world. The extreme case is that of a dependence $V \rightarrow W$ such that no pair of tuples has the same value in V. In general, a certain amount of data supporting the dependence is necessary in order to be confident about it.
- This problem is described in Pfahringer and Kramer (1995) from the point of view of the complexity of the theory underlying the dependence. The complexity is measured as the amount of bits needed to encode both the theory and the exceptions. The dependence is uninteresting if the number of bits needed to encode the projection on VW of the table is less or similar to the number of bits needed to encode the theory of the dependence $V \rightarrow W$, the values of V, and the exceptions. In other words, the dependence is interesting if it allows us to encode the same information in much less space. In Pfahringer and Kramer (1995) and Kramer and Pfahringer (1996) it is shown that the complexity of the theory for a given AD can be equal to the complexity of the original set of data, even for totally accurate dependencies. This is the case with trivial dependencies, for instance.
  The complexity of an AD is related to the number of exceptions, the number of rules in the theory, and the size of the projection on V of the table. For a very accurate dependence $V \rightarrow W$, the number of exceptions is assumed to be very low, and the size of the projection on V of the table is fixed, so the complexity depends basically on the number of rules, i.e., the number of different values of VW in the table. The goodness of the dependence depends on the ratio with respect to the number of bits needed to encode the projection on VW of the table. In other words, for a fixed set of rules, as the size of the table increases (by adding tuples corresponding to any rule in the theory), the AD is better. Hence, the quality of the AD is better as the number of different values of VW decreases and/or the number of tuples increases.
- Another point of view is introduced in Piatetsky-Shapiro (1992), where a close relation between the ratio K/n in a table r ($K = |dom(V)|, n = |r|$) and the quality of a dependence of the form $V \rightarrow W$ is pointed out. Following Piatetsky-Shapiro (1992), random permutations of the values of W in the tuples of a table r should break any dependence $V \rightarrow W$, because the associations among values of V and W are lost. But if K/n is close to 1, the accuracy of the dependence is almost the same for any permutation, and hence the dependence could be unreal.

In other words, there is no evidence in data supporting the dependence, so it is not a reliable result of a data mining process. Again the worst case arise if $V \rightarrow W$ is a trivial dependence, because the accuracy is 1 for any permutation we could perform on the values of W.

The conclusion of these three points of view is the same: as the number of different values of V appearing in a table r approaches $n = |r|$ (consequently, the number of different values of VW in the table approaches n as well), the quality of the dependence decreases, despite its accuracy, yielding uninteresting dependencies.

This problem can be solved by considering the support of V and $V \rightarrow W$. The support of V in a relation r can be calculated as

$$S(V) = \frac{1}{n^2} \sum_{i=1}^{K} n_{v_i}^2 \tag{5}$$

It is easy to see that $S(V)$ increases as K decreases and/or n increases since $\sum_{i=1}^{K} n_{v_i} = n$. In the same way, the support of an AD is related to both the number of different values of VW appearing in r and the number of tuples that agree with each value as follows:

$$S(V \rightarrow W) = \frac{1}{n^2} \sum_{i=1}^{K} \sum_{j=1}^{M} n_{v_i w_j}^2 \tag{6}$$

Clearly, the support will decrease if we replace some value of VW in r with a new value not being in $dom(V) \times dom(W)$. Finally, let us remark that the support of an attribute or dependence ranges from 0 (when $n = 0$) to 1, and its minimum value for a non-empty relation is $1/n$. This is also the value for a completely trivial dependence.

We can conclude that the support is a valid measure of triviality for ADs, since it behaves in the same way. Hence, the usual procedure for mining ARs with support above minimum threshold allows us to discard the corresponding trivial (to some extent) ADs as those rules with support below the threshold. This is one of the main advantages of our approach; at the same time it is based on the well-known theory of ARs, is simpler than the approach in Pfahringer and Kramer (1995), it allows us to employ existing algorithms and, contrary to Piatetsky-Shapiro (1992), it does not combine triviality and accuracy in a single measure; confidence or another accuracy measure for ARs is employed as well. Finally, by using AR mining algorithms, we are able to prune the search avoiding to explore a large amount of uninteresting dependencies by using any of the heuristics existing in the literature.

2.3 An alternative accuracy measure

Despite it is the most employed, confidence is not a suitable accuracy measure for association rules, as pointed out by different authors (Brin et al. 1997, Silverstein et al. 1998, Berzal et al. 2002). For instance, Piatetsky-Shapiro (1991) suggested that a suitable accuracy measure ACC for ADs should verify the following properties:

1. $ACC = 0$ if $S(V \rightarrow W) = S(V)S(W)n^2$
2. ACC monotonically increases with $S(V \rightarrow W)$ when other parameters remain the same.
3. ACC monotonically decreases with $S(V)$ (or $S(W)$) when other parameters remain the same.

However, confidence does not verify all these properties (Berzal et al. 2002). Among other problems, confidence is not able to detect neither statistical independence nor negative dependence between sets of items (Silverstein et al. 1998, Berzal et al. 2002).

This is a major drawback specially when there are items with very high support in our database, since these items are a potential source of trivial dependencies. For example, let us suppose a database with 16 attributes, one of them being an attribute A with support 0.999. Then, the confidence of any AD of the form $V \rightarrow A$ is in the interval $[1 - (10^{-3}/S(V)), 1]$. As an example, if $S(V) = 5 \times 10^{-3}$ then $C(V \rightarrow A) \in [0.8, 1]$. This way we can obtain up to 2^{15} trivial, misleading, dependencies. From the point of view in Piatetsky-Shapiro (1992), if the support of the attribute in the right part is very high, it is more probable that random permutations of its values introduce very few modifications in the dependence.

Different alternative measures have been proposed in order to solve the problems of confidence. Most of them introduce $S(W)$ in the expression, comparing the a priori probability that two tuples agree in W to the conditional probability in those cases where the tuples also agree in V (they are the same when both facts are statistically independent), or equivalently comparing the number of occurrences of a joint event with what would have been expected under a null hypothesis of independence. This is the case of measures like the *weighted relative accuracy*, also called *novelty"* (Lavrac et al. 1999), defined as $WRAcc(V \rightarrow W) = S(V)(C(V \rightarrow W) - S(W))$, the *lift*, also called *interest* (Silverstein et al. 1998), defined as $Lift(V \rightarrow W) = S(VW)/S(V)S(W)$, or the *conviction*, that can be written as $Conv(V \rightarrow W) = (S(V) - S(V)S(W))/(S(V) - S(VW))$ (Brin et al. 1997), among others. Let us remark that, to the extent that they are useful to measure the accuracy of ARs, any of these measures could be in principle employed in order to measure the accuracy of ADs.

In Berzal et al. (2002) we show that certainty factor (Shortliffe and Buchanan 1975), a rule uncertainty measure coming from the expert systems field and developed for the MYCIN system, is an alternative accuracy measure for ARs with good properties.

Definition 2.6 The certainty factor of $I_V \Rightarrow I_W$ is

$$CF(I_V \Rightarrow I_W) = \begin{cases} \dfrac{C(I_V \Rightarrow I_W) - S(I_W)}{1 - S(I_W)} & C(I_V \Rightarrow I_W) > S(I_W) \\ \dfrac{C(I_V \Rightarrow I_W) - S(I_W)}{S(I_W)} & C(I_V \Rightarrow I_W) \leq S(I_W) \end{cases} \tag{7}$$

assuming by agreement that if $S(I_W) = 1$ then $CF(I_V \Rightarrow I_W) = 1$ and if $S(I_W) = 0$ then $CF(I_V \Rightarrow I_W) = -1$.

As for the rest of measures we employ the alternative notation $CF(V \rightarrow W) = CF(I_V \Rightarrow I_W)$.

The certainty factor is interpreted as a measure of *variation* of the probability that the consequent is in a transaction when we consider only those transactions where the antecedent is. More specifically, it measures the decrease of the probability that the consequent is not in a transaction, given that the antecedent is. From a probabilistic point of view, and in the context of ADs, the absolute value of the positive (resp. negative) certainty factor measures the percentage of reduction of the probability that two tuples do not agree (resp. agree) in W when we know that they agree in V.

When the certainty factor is 0 then the antecedent and consequent of the rule are statistically independent. This property, that was proved for ARs in Berzal et al. (2002) holds for ADs as well, as the following proposition shows:

Proposition 2.2 *If V and W are statistically independent, then $CF(V \rightarrow W) = 0$.*

Proof V and W are statistically independent iff

$$\frac{n_{v_i w_j}}{n} = \frac{n_{v_i}}{n}\frac{n_{w_j}}{n} \quad \forall <v_i, w_j> \in dom(V) \times dom(W)$$

Hence

$$\begin{aligned} C(V \rightarrow W) = \frac{S(I_V \Rightarrow I_W)}{S(I_V)} &= \frac{\sum_{p=1}^{K}\sum_{q=1}^{M}\left(\frac{n_{v_i w_j}}{n}\right)^2}{\frac{1}{n^2}\sum_{i=1}^{K} n_{v_i}^2} \\ &= \frac{\sum_{p=1}^{K}\sum_{q=1}^{M}\frac{n_{v_p}^2}{n^2}\frac{n_{w_q}^2}{n^2}}{\frac{1}{n^2}\sum_{i=1}^{K} n_{v_i}^2} = \frac{\frac{1}{n^2}\sum_{p=1}^{K} n_{v_p}^2 \sum_{q=1}^{M}\frac{n_{w_q}^2}{n^2}}{\frac{1}{n^2}\sum_{i=1}^{K} n_{v_i}^2} \\ &= \sum_{q=1}^{M}\frac{n_{w_q}^2}{n^2} = S(W) \end{aligned}$$

and hence, $CF(V \rightarrow W) = 0$. □

A negative certainty factor means that the statistical dependence is negative (i.e. the presence of the antecedent is related to the absence of the consequent in the same transaction). Also, certainty factors verify the three properties stated in Piatetsky-Shapiro (1991). An additional property is $CF(I_V \Rightarrow I_W) \leq C(I_V \Rightarrow I_W)$.

In addition, we can point out some relationships to other measures in the literature that are shown in Berzal et al. (2002):

- Let $CF(V \rightarrow W)>0$ and $S(V)>0$ and $S(W)<1$. Then

$$CF(V \rightarrow W) = 1 - \frac{1}{Conv(V \rightarrow W)} \tag{8}$$

- Let $CF(V \rightarrow W)<0$ and $S(W)>0$. Then

$$CF(V \rightarrow W) = Lift(V \rightarrow W) - 1 \tag{9}$$

The proofs of these properties, together with a more detailed description of the properties of certainty factors can be found in Berzal et al. (2002). An advantage of certainty factors with respect to conviction and lift is that the possible values are bounded in $[-1, 1]$ and represent (in absolute value) percentages, so it is easier to interpret them. With respect to the weighted relative accuracy, the factor $(C(V \rightarrow W) - S(W))$ measures the absolute variation of probability, as the enumerator of the certainty factor. The factor $S(V)$ is introduced in order to trade off this variation with the generality of the rule as measured by $S(V)$. However, in the context of AR mining, these two factors are considered separately, by using a minimum support threshold in order to discard rules with low generality. Hence, it is not necessary to weight the accuracy measure with the support. The advantages of considering separately accuracy and support when mining for ADs were discussed in Sect. 2.2.

The following is another important property of certainty factors with respect to ADs:

Proposition 2.3 *The approximate dependence $V \rightarrow W$ is functional if and only if $CF(I_V \Rightarrow I_W) = 1$.*

Proof For any AR $I_1 \Rightarrow I_2$ in a set of transactions T, $C(I_1 \Rightarrow I_2) = 1$ if and only if $CF(I_1 \Rightarrow I_2) = 1$ (Berzal et al. 2002). Hence, $V \rightarrow W$ is a FD if and only if $C(I_V \Rightarrow I_W) = 1$ (Proposition 2.1), and $C(I_V \Rightarrow I_W) = 1$ if and only if $CF(I_V \Rightarrow I_W) = 1$. □

We call *strong approximate dependencies* those dependencies with certainty factor and support greater than two user-defined thresholds *minCF* and *minSupp* respectively. We assume that we are interested in finding rules with positive certainty factor, because ADs are positive associations.

2.4 An interpretation of support and accuracy based on the theory of an AD

The measures associated with ADs must have a clear interpretation, because before the mining process we must choose the thresholds *minCF* and *minSupp*, and when the ADs have been obtained we need to understand how good they are. If we look at the AD $V \rightarrow W$ as an AR $I_V \Rightarrow I_W$ in T_r, the interpretations of support and certainty factor are clear. However, we are looking for ADs in r and we would like to interpret the measures in r.

In this subsection we provide an interpretation of the support and accuracy of an AD based on the support and accuracy of the ARs that form the theory of the AD. The ARs that form the theory of an AD are defined in r and relate the presence of values of attributes in tuples, i.e. items are pairs (attribute, value) (that we note [attribute=value]) and transactions are tuples. First, we formalize the concept of theory of an AD.

Definition 2.7 The theory that describes an AD "$V \rightarrow W$" is the following set of ARs

$$Th_{[V \rightarrow W]} = \left\{ Ru_{ij} | \exists t \in r \text{ such that } t[V] = v_i \text{ and } t[W] = w_j \right\}$$

where Ru_{ij} is the association rule $[V = v_i] \Rightarrow [W = w_j]$. We shall note s_{ij}, c_{ij} and cf_{ij} the support, confidence and certainty factor of Ru_{ij}, respectively.

Let us remark that the ARs that form the theory of an AD are defined on r in the classical way, i.e., they relate the presence of items of the form (attribute,value) in

transactions corresponding to tuples. They are therefore different from ARs on the set of transactions T_r that define our ADs. However, as we shall show in this section, the measures of the former are related to those of the latter, and this relationship is very useful in order to understand the support and certainty factor of an AD.

2.4.1 Support

The following propositions relate the support of an AD to the support of the rules of its theory:

Proposition 2.4 *The support of an AD "$V \to W$" can be obtained from the support of the ARs of its theory as follows:*

$$S(V \to W) = \frac{1}{n^2} \sum_{Ru_{ij} \in Th_{[V \to W]}} n_{v_i w_j}^2 = \sum_{Ru_{ij} \in Th_{[V \to W]}} s_{ij}^2 \tag{10}$$

Proof If $Ru_{ij} \notin Th_{[V \to W]}$ then $n_{v_i w_j} = 0$. Hence,

$$S(V \to W) = \frac{1}{n^2} \sum_{i=1}^{K} \sum_{j=1}^{M} n_{v_i w_j}^2 = \frac{1}{n^2} \sum_{Ru_{ij} \in Th_{[V \to W]}} n_{v_i w_j}^2$$

By definition, $s_{ij} = n_{v_i w_j}/n$, so

$$S(V \to W) = \frac{1}{n^2} \sum_{Ru_{ij} \in Th_{[V \to W]}} n_{v_i w_j}^2 = \sum_{Ru_{ij} \in Th_{[V \to W]}} s_{ij}^2 \qquad \square$$

Proposition 2.5 *If every AR in $Th_{[V \to W]}$ has the same support s_0, then the following holds:*

1. $|Th_{[V \to W]}| = 1/s_0$
2. $S(V \to W) = s_0$

Proof

1. Trivial since $\sum_{Ru_{ij} \in Th_{[V \to W]}} s_{ij} = 1$
2. As a consequence of 1.

$$S(V \to W) = \sum_{Ru_{ij} \in Th_{[V \to W]}} s_{ij}^2 = \frac{1}{s_0} s_0^2 = s_0 \qquad \square$$

Following Proposition 2.5, an AD with support s_0 is as trivial as an AD described by a theory consisting of $1/s_0$ rules, each one with support s_0. Since $1/s_0$ is not always an integer, we can establish bounds for the complexity of s_0. Specifically, if $1/s_0$ is not an integer, let a be an integer such that $a < 1/s_0 < a + 1$. Then, the AD is less (resp. more) complex than an AD described by $a + 1$ (resp. a) rules with support $1/(a + 1)$ (resp. $1/a$). The number of rules and its support can give us an idea of the complexity of the original AD.

2.4.2 Confidence

We consider confidence in this subsection because the certainty factor is based on it, though only support and certainty factor are to be used in finding ADs.

Lemma 2.1 *Let* $f : \mathbb{R} \to \mathbb{R}$. *Then*

$$\sum_{i=1}^{K} f(n_{v_i}) = \sum_{Ru_{ij} \in Th_{[V \to W]}} c_{ij} f(n_{v_i}) \tag{11}$$

Proof For each $i \in \{1, \ldots, K\}$,

$$\sum_{j \in \{1,\ldots,M\} | Ru_{ij} \in Th_{[V \to W]}} c_{ij} = \sum_{j \in \{1,\ldots,M\} | Ru_{ij} \in Th_{[V \to W]}} \frac{n_{v_i w_j}}{n_{v_i}} = 1$$

In addition, $\forall i \in \{1, \ldots, K\} \exists j \in \{1, \ldots, M\}$ such that $Ru_{ij} \in Th_{[V \to W]}$. Hence

$$\begin{aligned} \sum_{i=1}^{K} f(n_{v_i}) &= \sum_{i=1}^{K} \left(\sum_{j \in \{1,\ldots,M\} | Ru_{ij} \in Th_{[V \to W]}} c_{ij} f(n_{v_i}) \right) \\ &= \sum_{Ru_{ij} \in Th_{[V \to W]}} c_{ij} f(n_{v_i}) \end{aligned}$$

□

Proposition 2.6 *The confidence of an AD "*$V \to W$*" can be obtained from the support and confidence of the ARs of its theory as follows:*

$$\frac{1}{C(V \to W)} = \sum_{Ru_{ij} \in Th_{[V \to W]}} \left(\left(\frac{s_{ij}^2}{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2} \right) \frac{1}{c_{ij}} \right) \tag{12}$$

Proof

$$\begin{aligned} C(V \to W) = \frac{S(I_V \Rightarrow I_W)}{S(I_V)} &= \frac{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2}{\frac{1}{n^2} \sum_{i=1}^{K} n_{v_i}^2} \\ &= \frac{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2}{\frac{1}{n^2} \sum_{Ru_{ij} \in Th_{[V \to W]}} c_{ij} n_{v_i}^2} \end{aligned}$$

(by Lemma 2.1 being $f(x) = x^2$), and hence

$$\frac{1}{C(V \to W)} = \frac{\frac{1}{n^2} \sum_{Ru_{ij} \in Th_{[V \to W]}} c_{ij} n_{v_i}^2}{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2} = \frac{\sum_{Ru_{ij} \in Th_{[V \to W]}} c_{ij} \left(\frac{s_{ij}}{c_{ij}} \right)^2}{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2}$$

$$\left(\text{because } c_{ij} = \frac{s_{ij}}{(n_{v_i}/n)} \text{ and hence } \left(n_{v_i}/n\right)^2 = \left(\frac{s_{ij}}{c_{ij}}\right)^2\right)$$

$$= \sum_{Ru_{ij} \in Th_{[V \to W]}} \left(\left(\frac{s_{ij}^2}{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2}\right)\frac{1}{c_{ij}}\right) \qquad \square$$

Corollary 2.1 *The inverse of the confidence of an AD "$V \to W$" is a weighted sum of the inverse of the confidences of each AR of the theory $Th_{[V \to W]}$, where the weight is the relative contribution of the rule to the support of the AD, i.e.*

$$\frac{1}{C(V \to W)} = \sum_{Ru_{ij} \in Th_{[V \to W]}} \left(\left(\frac{s_{ij}^2}{S(V \to W)}\right)\frac{1}{c_{ij}}\right)$$

Proof By Propositions 2.6 and 2.4. $\square$

On the basis of these results, the interpretation of confidence is the following:

Proposition 2.7 *If every AR in the theory of an AD "$V \to W$" has confidence c_0, then the confidence of the AD is c_0.*

Proof

$$\frac{1}{C(V \to W)} = \sum_{Ru_{ij} \in Th_{[V \to W]}} \left(\left(\frac{s_{ij}^2}{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2}\right)\frac{1}{c_{ij}}\right)$$

$$= \frac{1}{c_0} \sum_{Ru_{ij} \in Th_{[V \to W]}} \left(\frac{s_{ij}^2}{\sum_{Ru_{pq} \in Th_{[V \to W]}} s_{pq}^2}\right) = \frac{1}{c_0}$$

Hence, $C(V \to W) = c_0$. $\square$

2.4.3 Certainty factor

Finally, the following proposition provides a similar interpretation for the certainty factor of an AD.

Proposition 2.8 *If every AR in the theory of an AD "$V \to W$" has confidence c_0, and certainty factor cf_0, then the following holds:*

1. *Either $c_0 = 1$ or every item $[W = w_j]$ has support $s_1 = 1/M$ in r.*
2. *The certainty factor of the AD is cf_0.*

Proof Let $S([W = w_j])$ be the support of the item $[W = w_j]$. We shall show both 1 and 2.

1. Let $c_0 < 1$. Then either

$$cf_{ij} = cf_0 = \frac{c_0 - S([W = w_j])}{1 - S([W = w_j])} \quad \forall Ru_{ij} \in Th_{[V \to W]}$$

if $cf_0 > 0$, or

$$cf_{ij} = cf_0 = \frac{c_0 - S([W = w_j])}{S([W = w_j])} \quad \forall Ru_{ij} \in Th_{[V \to W]}$$

if $cf_0 \leq 0$. As both expressions for cf_{ij} are strictly monotonic, $S([W = w_j]) = s_1 \in [0, 1]\ \forall j \in \{1, \ldots, M\}$. On the other hand, since $\sum_{j=1}^{M} S([W = w_j]) = 1$, then $s_1 = 1/M$.

2. We shall consider two cases:
   - Let $c_0 = 1$. Then $cf_0 = 1$ (Berzal et al. 2002). On the other hand, $C(V \to W) = c_0 = 1$ (Proposition 2.7), and hence

$$CF(V \to W) = \frac{1 - S(W)}{1 - S(W)} = 1 = cf_0$$

   - Let $0 < c_0 < 1$. Then

$$cf_0 = \frac{c_0 - s_1}{1 - s_1}$$

     On the other hand

$$S(W) = \sum_{j=1}^{M} n_{w_j}^2 = M \frac{1}{M^2} = 1/M = s_1$$

     Hence

$$CF(V \to W) = \frac{C(V \to W) - S(W)}{1 - S(W)} = \frac{c_0 - s_1}{1 - s_1} = cf_0$$

     The proof for the case $-1 < c_0 \leq 0$ is similar, but using the expression for negative certainty factors. □

The following result integrates the interpretation of support, confidence and certainty factor:

Proposition 2.9 *If every AR in the theory of an AD "$V \to W$" has support s_0, confidence c_0, and certainty factor cf_0, then the following holds:*

1. *Every item $[V = v_i]$ has support $s_2 = 1/K$ in r.*
2. *$cf_0 \geq 0$.*

Proof

1. For every AR in $Th_{[V \to W]}$, the confidence is $c_{ij} = s_{ij}/S([V = v_i])$. Since $c_{ij} = c_0$ and $s_{ij} = s_0$ for every rule, $S([V = v_i])$ takes the same value for every $v_i \in dom(V)$. We name that value s_2. As $\sum_{i=1}^{K} S([V = v_i]) = 1$ it is obvious that $s_2 = 1/K$.

2. If every AR has the same support s_0, the number of ARs is $1/s_0$ (Proposition 2.5). Obviously, $1/s_0 \leq K \cdot M$, so $s_0 \geq \frac{1}{K \cdot M}$ and therefore

$$c_0 = \frac{s_0}{s_2} = \frac{s_0}{1/K} \geq \frac{1}{M} = s_1$$

Hence, $cf_0 \geq 0$. □

2.4.4 Conclusions

On the basis of the results in this subsection, the theory of an AD with support s_0, confidence c_0 and certainty factor $cf_0 \geq 0$ is as complex as a theory formed by $1/s_0$ ARs, each one with support s_0, confidence c_0 and certainty factor cf_0. Such a theory would hold in a relation whose size were a multiple of $1/s_0$, K and M. Though it is usual that $1/s_0$ is not an integer, we can use the nearest lower and higher integer values as an approximation, as we pointed out in our interpretation of support.

For example, if an AD "$V \rightarrow W$" has support $s_0 = 0.17$ then $1/s_0 \approx 5.18$, and we shall use the integer interval [5,6]. In such situation we say that the complexity of the AD is between the complexity of an AD described by 5 ARs, each one with support 1/5, and the complexity of an AD described by 6 ARs, each one with support 1/6. The confidence and certainty factor of the ARs will be c_0 and cf_0, respectively, in both models.

If $cf_0 < 0$ we can still use Proposition 2.8, but by Proposition 2.9 the rules in the theory cannot have the same support. However, as we pointed out at the end of the previous section, we are only interested in ADs with positive certainty factor.

The interpretation can help us to understand the significance of the values of support and certainty factor of an AD in terms of the quality of the theory of the AD. In the same way, it can make easier the choice of the thresholds *minSupp* and *minCF*.

3 Related work and comparison with our approach

In this section we compare our measures with other proposals available in the literature. In some cases, the comparison is based on an interpretation of the measures in terms of the theory of the AD.

3.1 Exceptions

Approximate dependencies are fuzzy dependencies with exceptions. The existing definitions of AD differ in the concept of *exception*, and the measure(s) of accomplishment of the AD (usually, based on the number of exceptions). The main approaches for the concept of exception are the following:

- **Exceptions as pairs of tuples.** This approach claims that an exception to a FD is an exception to the rule (1), i.e., a pair of tuples $t, s \in r$ verifying $t[V] = s[V]$ and $t[W] \neq s[W]$.

- **Exceptions as individual tuples.** A set $r_e \subseteq r$ is a set of exceptions of $V \rightarrow W$ if the dependence holds in $r \backslash r_e$. In general, the set r_e of exceptions is not unique. Two possible definitions of r_e are the following:
  - (Bra and Paredaens 1983) The set of exceptions is

$$r_e = \left\{ t \in r | \exists t' \in r, t[V] = t'[V] \text{ and } t[W] \neq t'[W] \right\} \tag{13}$$

    As we shall see later, this definition is related to the (previously existing) concept of boundary of rough sets (Pawlak 1982).
  - The minimal approach, formalized in Cubero et al. (1998), yet a measure according to it was defined previously in Kivinen and Mannila (1995). The set of exceptions r_e is any set verifying the following properties:
    1. $V \rightarrow W$ is a FD in $r \backslash r_e$
    2. $\forall t \in r_e$, $V \rightarrow W$ is not a FD in $(r \backslash r_e) \cup \{t\}$
    3. $\nexists r'_e \subset r$ verifying 1 and 2 such that $|r'_e| < |r_e|$

In our new approach (Sect. 2) we have considered exceptions as pairs of tuples that verify the antecedent but not the consequent in (1). There are two arguments that justify this approach.

1. The concept of exception as pairs of tuples leads to a unique set of exceptions for a given relation. For instance, let us consider the toy relation r of Table 4. If we consider exceptions as individual tuples we have five different sets of exceptions r_e^i with $i \in \{1, 2, 3, 4, 5\}$. Following the approach based on the boundary of rough sets (equivalently, De Bra and Paredaens), the set of exceptions is $r_e^1 = r$. Following the minimal approach the possible sets of exceptions are r_e^2, r_e^3, r_e^4, and r_e^5, shown in Table 5 (A-D).
   Obviously, in real-world tables, the number of possible sets of exceptions is usually much larger. On the contrary, the set of exceptions as pairs of tuples, that we call

Table 4 Toy relation r

Tuple	V	W
1	v_1	w_1
2	v_1	w_2
3	v_2	w_2
4	v_2	w_3

Table 5 Possible sets of exceptions following the minimal approach

(A) r_e^2		(B) r_e^3		(C) r_e^4		(D) r_e^5	
V	W	V	W	V	W	V	W
v_1	w_1	v_1	w_1	v_1	w_2	v_1	w_2
v_2	w_2	v_2	w_3	v_2	w_2	v_2	w_3

E, is unique for a given relation. Exceptions based on pairs of tuples are those pairs of tuples such that $t[V] = s[V]$ and $t[W] \neq s[W]$. For instance, in the case of the relation r, $E = \{<1, 2>, <3, 4>\}$.

2. The measures of accuracy based on individual tuples are the percentage of tuples that are not exceptions. However this is the percentage of tuples where the dependence holds, i.e., the support of the dependence. This fact is pointed out for example in Pfahringer and Kramer (1995). However, if we consider exceptions as pairs of tuples, accuracy and support are different in general. This is more intuitive and, as we have discussed in Sect. 2.2, very useful as a way to avoid trivial dependencies.

3.2 Proposals based on exceptions as individual tuples

Some accuracy measures based on exceptions as individual tuples are

- (Ziarko 1991). This measure is based on the theory of rough sets introduced in Pawlak (1982). Let $IND'(U)$ with $U \subseteq RE$ be the set of equivalence classes induced in r by the equivalence relation $IND(U)$, meaning "to agree in attribute U", provided that if U is a set of attributes, the tuples must agree in every atomic attribute that pertains to U. Hence

$$(t, s) \in IND(U) \Leftrightarrow t[U] = s[U]$$

$$IND'(U) = \{[t]_{IND(U)} | t \in r\}$$

  The accuracy of $V \rightarrow W$ is

$$k(V, W) = \frac{|POS(V, W)|}{|r|} \tag{14}$$

  where

$$POS(V, W) = \bigcup \{\underline{IND}(V, \Delta) | \Delta \in IND'(W)\}$$

  with

$$\underline{IND}(V, \Delta) = \bigcup \{\Gamma \in IND'(V) | \Gamma \subseteq \Delta\}$$

  If $k(V, W) = 1$ then $V \rightarrow W$ is a FD.
- Measure g_2 (Kivinen and Mannila 1995).

$$g_2(V \rightarrow W, r) = \frac{|\{t: t \in r, \exists s \in r: t[V] = s[V], t[W] \neq s[W]\}|}{|r|} \tag{15}$$

  This is an error measure. It takes value 0 if the dependence is functional, and a value close to 1 if the dependence clearly does not hold. This measure considers

exceptions in the sense of Bra and Paredaens (1983), see (Cubero et al. 1998). This is also related to the concept of inconsistency for rough sets (Pawlak 1982).

- Measure g_3 (Kivinen and Mannila 1995).

$$g_3(V \rightarrow W, r) = \frac{\min\{|r_e|: r_e \subseteq r \text{ and } V \rightarrow W \text{ holds in } r \backslash r_e\}}{|r|} \tag{16}$$

Another error measure, taking values between 0 (FD) and 1. It is based on the minimal approach to exceptions.

Some other measures (Schlimmer 1993, Shen 1991) are referenced and briefly commented in Pfahringer and Kramer (1995).

3.2.1 Comparison

The following proposition provides an interpretation of Ziarko's measure on the basis of the rules in $Th_{[V \rightarrow W]}$:

Proposition 3.1 *Ziarko's measure $k(V, W)$ (14) can be expressed in the following way*

$$k(V, W) = \sum_{Ru_{ij} \in Th_{[V \rightarrow W]} | cf_{ij}=1} s_{ij} \tag{17}$$

Proof Each equivalence class in $IND'(V)$ (resp. $IND'(W)$) is associated with a value $v_i \in dom(V)$ (resp. $w_j \in dom(W)$). Let us note $CL(X, x)$ the equivalence class associated with the value x of attribute X. By definition, $k(V, W) = |POS(V, W)|/n$, $POS(V, W)$ being the union of the equivalence classes in $IND'(V)$ that are contained in an equivalence class pertaining to $IND'(W)$. For all $v_i \in dom(V)$, if the tuples in $CL(V, v_i)$ are in $POS(V, W)$ then there is a value $w_j \in dom(W)$ such that the tuples are in $CL(W, w_j)$. Hence the rule Ru_{ij} has confidence 1, and $|CL(V, v_i)| = n_{v_i} = n_{v_i w_j}$. Therefore, each rule with confidence 1 contributes with $n_{v_i w_j}$ to $POS(V, W)$, and with $n_{v_i w_j}/n = s_{ij}$ to $k(V, W)$. As $c_{ij} = 1$ iff $cf_{ij} = 1$, the contributions come from rules whose certainty factor is 1. □

The following proposition was shown in Cubero et al. (1998). The proof is based on the concept of exception by De Bra and Paredaens, which is on the basis of g_2.

Proposition 3.2 *The measure g_2 (15) can be expressed in the following way:*

$$g_2(V \rightarrow W, r) = 1 - k(V, W)$$

The measure g_3 can be interpreted on the basis of $Th_{[V \rightarrow W]}$ as well, as the following proposition shows:

Proposition 3.3 *The measure g_3 (16) can be expressed in the following way*

$$g_3(V \rightarrow W, r) = 1 - \sum_{i=1}^{K} max_{\{j \mid Ru_{ij} \in Th_{[V \rightarrow W]}\}}\{s_{ij}\} \tag{18}$$

Proof

$$g_3(V \to W, r) = \frac{\min\{|r_e|: r_e \subseteq r \text{ and } V \to W \text{ holds in } r \backslash r_e\}}{|r|}$$
$$= \frac{|r| - max\,\{|r_{oe}| : r_{oe} \subseteq r \text{ and } V \to W \text{ holds in } r_{oe}\}}{|r|}$$

If "$V \to W$" is a FD, there is no pair of ARs in the theory associated with $V \to W$ with the same antecedent (in that case, at least two tuples agree in V and don't agree in W, and the dependence is not functional). Moreover, every tuple in r supports only one AR of the theory (assuming, as it is usually the case, that r verifies the first normal form), so we can obtain a partition of the tuples based on the AR that each tuple supports. Hence, if we want to choose the greatest set of tuples where a given dependence "$V \to W$" is a FD, it is sufficient to choose for every $v_i \in dom(V)$ the tuples supporting the rule Ru_{ij} with higher support. Let Ru_{iJ} be such rule. The size of the set of tuples that supports Ru_{iJ} is $|r|s_{iJ}$. Therefore

$$max\,\{|r_{oe}| : r_{oe} \subseteq r \text{ and } V \to W \text{ holds in } r_{oe}\}$$
$$= |r| \sum_{i=1}^{K} max_{\{j \mid Ru_{ij} \in Th_{[V \to W]}\}}\{s_{ij}\}$$

Hence

$$g_3(V \to W, r) = \frac{|r| - |r| \sum_{i=1}^{K} max_{\{j | Ru_{ij} \in Th_{[V \to W]}\}}\{s_{ij}\}}{|r|}$$
$$= 1 - \sum_{i=1}^{K} max_{\{j | Ru_{ij} \in Th_{[V \to W]}\}}\{s_{ij}\} \qquad \square$$

With these results we can point out the following conclusions:

- A clear difference between the measures g_i $i \in \{2, 3\}$ and the rest is that g_i are error measures, while the rest are accuracy measures. The values $1 - g_i$ can be seen as accuracy measures.
- By Proposition 3.2, g_2 measures the distance from the AD to a FD, and $k(V, W)$ can be seen as the corresponding accuracy. In this sense, k is based on the concept of exception by De Bra and Paredaens.
- As $S(W)$ is not considered, $k = 1 - g_2$ does not verify the three properties stated in Piatetsky-Shapiro (1991). Neither do $1 - g_3$. Indeed, these measures don't see ADs as rules.
- The formula (17) tell us that both g_2 and k are rather strict, because all the tuples supporting an AR are considered as exceptions if only one single tuple breaks the AR. Hence, one single noisy tuple can reduce significantly the accuracy of the dependence, even from 1 to 0 in the case of $k(V, W)$. For instance, $k(V, W) = 1$ for a FD taking the same value for V in all tuples of r, but $k(V, W)$ becomes 0 if we introduce a new value for V in a tuple.

- The measure g_3 is less strict than g_2 (indeed $1 - g_3 \geq 1 - g_2$), because it considers other tuples than those supporting rules with $c_{ij} = 1$, see formula (18). It is less sensible to the incorporation of noisy tuples, in the sense that changes in the accuracy are smoother. Also measures based on pairs of tuples, including ours, are less sensible to erroneous tuples than other measures.

3.3 Proposals based on exceptions as pairs of tuples

Some measures are the following:

- Measure g_1 (Kivinen and Mannila 1995)

$$g_1(V \to W, r) = \frac{|\{(t, s)\colon t, s \in r, t[V] = s[V], t[W] \neq s[W]\}|}{|r|^2} \tag{19}$$

  In fact, this is not an accuracy measure, but an error measure. A value 0 indicates that the dependence is functional, and a value close to 1 indicates that the dependence clearly does not hold.
- Measures *pdep*, τ and μ (Piatetsky-Shapiro 1992). Let

$$pdep(W) = \sum_{j=1}^{M} \frac{|\{t \in r\colon t[W] = w_j\}|^2}{|r|^2} \tag{20}$$

  Then

$$pdep(V, W) = \frac{1}{|r|} \sum_{i=1}^{K} \sum_{j=1}^{M} \frac{|\{t \in r\colon t[V] = v_i, t[W] = w_j\}|^2}{|\{t \in r\colon t[V] = v_i\}|} \tag{21}$$

$$\tau(V, W) = \frac{pdep(V, W) - pdep(W)}{1 - pdep(W)} \tag{22}$$

$$\mu(V, W) = 1 - \frac{1 - pdep(V, W)}{1 - pdep(W)} \frac{|r| - 1}{|r| - K} \tag{23}$$

  Measures *pdep* and τ take value 1 if the dependence is functional, and close to 0 if the dependence clearly does not hold. As we shall discuss later, measure μ could be interpreted as a combination of accuracy and support of an AD.

Another measure based on this approach can be found in Russell (1989). It is similar to *pdep* (Piatetsky-Shapiro 1991). It is also referenced and briefly commented in Pfahringer and Kramer (1995).

3.3.1 Comparison

Some propositions will help us to compare our proposal with the aforementioned approaches.

Proposition 3.4 *The measure g_1 (19) can be expressed in the following way*

$$g_1(V \to W, r) = S(I_V) - S(I_V \Rightarrow I_W) \tag{24}$$

Proof

$$\begin{aligned} g_1(V \to W, r) &= \frac{\left|\{(t,s) \in r^2 : t[V] = s[V],\ t[W] \neq s[W]\}\right|}{|r|^2} \\ &= \frac{|\{(t,s) \in r^2 : t[V] = s[V]\}| - |\{(t,s) \in r^2 : t[V] = s[V], t[W] = s[W]\}|}{|r|^2} \\ &= S(I_V) - S(I_V \Rightarrow I_W) \end{aligned}$$

□

Corollary 3.1

$$g_1(V \to W, r) = S(V)\,(1 - C(V \to W)) \tag{25}$$

Proof From Proposition 3.4 and the definition of confidence (2). □

We can point out that

- g_1 is based on pairs of tuples. As we showed in previous sections, the accuracy of an AD increases as $S(V \to W)$ gets closer to $S(V)$, and reaches its maximum when $S(V \to W) = S(V)$. In this sense, g_1 measures the distance from the AD to a FD (Proposition 3.4). Also g_1 is shown to be related to confidence by formula (25).
- As $S(W)$ is not considered, $1 - g_1$ does not verify the three properties stated in Piatetsky-Shapiro (1991). Again, this measure does not see ADs as rules.

In the following, we focus on the relation among our measures and those proposed in Piatetsky-Shapiro (1992).

Proposition 3.5 $pdep(V) = S(V)$

Proof Trivial regarding Eqs. 5 and 20. □

The following proposition shows a nice expression for $pdep(V, W)$ based on $Th_{[V \to W]}$:

Proposition 3.6 *The measure $pdep(V, W)$ (21) can be expressed in the following way*

$$pdep(V, W) = \sum_{Ru_{ij} \in Th_{[V \to W]}} s_{ij} c_{ij} \tag{26}$$

Proof

$$pdep(V, W) = \frac{1}{|r|} \sum_{i=1}^{K} \sum_{j=1}^{M} \frac{|\{t \in r: t[V] = v_i, t[W] = w_j\}|^2}{|\{t \in r: t[V] = v_i\}|}$$

$$= \frac{1}{n} \sum_{i=1}^{K} \sum_{j=1}^{M} \frac{n_{v_i w_j}^2}{n_{v_i}} = \sum_{i=1}^{K} \sum_{j=1}^{M} \frac{n_{v_i w_j}}{n} \frac{n_{v_i w_j}}{n_{v_i}} = \sum_{i=1}^{K} \sum_{j=1}^{M} s_{ij} c_{ij}$$

$$= \sum_{Ru_{ij} \in Th_{[V \to W]}} s_{ij} c_{ij} \qquad \square$$

An interesting relation between the measures we are discussing is shown in the following proposition:

Proposition 3.7 *If every AR in the theory of an AD "$V \to W$" has confidence c_0, and certainty factor $cf_0 \geq 0$, then the following holds:*

1. $pdep(V, W) = C(V \to W) = c_0$.
2. $\tau(V, W) = CF(V \to W) = cf_0$.

Proof

1. By Proposition 3.6, $pdep(V, W)$ is a weighted sum of the confidences c_{ij}, where the weight for every c_{ij} is s_{ij}. Since the addition of the support of all rules is 1, if every rule in the theory has confidence c_0 then $pdep(V, W) = c_0$. Also, by Proposition 2.7 $C(V \to W) = c_0$.
2. If $cf_0 = 1$ then $V \to W$ is a FD and $\tau(V, W) = 1$. Let us suppose $0 \leq cf_0 < 1$. By Proposition 2.8, $CF(V \to W) = cf_0$ and $S(W) = s_1$. By Proposition 3.5, $pdep(W) = S(W)$. We have just shown that under the assumptions above, $pdep(V, W) = c_0$. Since $cf_0 \geq 0$

$$\tau(V, W) = \frac{pdep(V, W) - pdep(W)}{1 - pdep(W)} = \frac{c_0 - s_1}{1 - s_1} = cf_0 \qquad \square$$

Our conclusions about the relation between these measures are:

- The measures $pdep(V, W)$ and $C(V \to W)$ are not always the same, though conceptually they are both the conditional probability that two tuples agree in W, given that they agree in V. In particular, $pdep(V, W) \geq S(W)$ (Piatetsky-Shapiro 1992), while $C(V \to W)$ can be lesser than $S(W)$. The reason is that the probability is estimated in different ways. For instance, let r be that of Table 6. Then $pdep(V, W) = 7/10 = 0.7$ and $C(V \to W) = 11/17 \approx 0.647$. However, we have shown (Proposition 3.7) that the equality $pdep(V, W) = C(V \to W)$ holds at least when all the ARs in the theory of $V \to W$ have the same support, confidence and certainty factor.
- As a consequence, the measures $\tau(V, W)$ and $CF(V \to W)$ are not always the same, though $pdep(W) = S(W)$. Obviously, if $pdep(V, W) = C(V \to W)$ then $\tau(V, W) = C(V \to W)$.

Table 6 A table r where $pdep(V, W) \neq C(V \rightarrow W)$

V	W
v_1	w_1
v_1	w_2
v_1	w_1
v_1	w_1
v_2	w_1

- In our opinion, μ merges in one single measure the accuracy of the AD and its reliability, the latter being related to the value K, and hence to the support of both the antecedent and the AD (if we modify a tuple by introducing a new value of V, and K becomes $K + 1$, then the support decreases). If $K = 1$ then $\mu(V, W) = \tau(V, W)$, otherwise $\mu(V, W) \leq \tau(V, W)$ (Piatetsky-Shapiro 1992). The only case in which μ cannot discard an AD with very low support is when the AD is in fact a FD. In that case, $pdep(V, W) = \tau(V, W) = \mu(V, W) = 1$.
  Contrary to μ, our approach is to separate accuracy and support. We think that the use of two measures instead of only one is better for several reasons. From the theoretical point of view, it gives us more information about the AD. In practice, support allow us for a more efficient mining of data, since it can be used (as in the case of ARs) for bounding the search.

As a final comment, an important difference between k, g_2 and g_3 (based on individual tuples) and the rest of measures (based on pairs of tuples) is the amount of information they deal with. The expressions (17) and (18) show that only tuples supporting ARs that verify certain conditions are considered by k, g_2 and g_3. Hence, if we change a discarded tuple in a way that does not affect the set of ARs that verify the conditions, the accuracy of the dependence does not change. For instance, introducing a new value not in $dom(V)$ neither in $dom(W)$ in a discarded tuple does not change the accuracy.

On the contrary, measures based on pairs of tuples take into account every tuple in r, and the changes described above would change the accuracy in many cases (though smoothly, as we discussed before). This claim is easy to show by the definition of CF (Definition 2.6), and the expressions for g_1 (24) and $pdep$ (26) (also τ and μ, as based on $pdep$, consider every tuple).

4 Our methodology to discover approximate dependencies

By Definition 2.4 we can obtain AD mining algorithms by adapting efficient algorithms for association rule mining. However, the direct application of these algorithms to look for rules in T_r has some drawbacks related to efficiency, because the number of transactions is the square of the number of tuples. Our goal here is to solve those problems and to provide a way to obtain efficient algorithms for AD mining by adapting AR mining algorithms. Let us remark that, as we shall show, the proposed methodology can be applied to adapt any AR mining algorithm, provided that to deal with several transactions at a time is not an inconvenience.

4.1 Direct application of AR mining algorithms

To mine for strong dependencies $V \rightarrow W$ in r is equivalent to search for strong association rules $I_V \Rightarrow I_W$ in T_r by using one of the many existing algorithms. Most of them explore the lattice of itemsets with respect to set inclusion in a first step, searching for itemsets with enough support, i.e. itemsets whose support is above a threshold *minSupp* given as input. In a second step, and starting from these so-called *frequent* itemsets, the strong rules are obtained.

The first part of the algorithm is the most computationally expensive, and its complexity is comprised (in the worst case) at least of a linear factor in the number of transactions (because we need to explore the set of transactions to calculate the support of an itemset), another linear factor in the number of levels of the lattice (usually equal to the number of items), and an exponential factor in the number of items, that corresponds to the number of possible itemsets. Many algorithms available in the literature try to improve efficiency in different ways.

The complexity of a levelwise algorithm like APRIORI when employed for mining ADs on T_r has two main factors:

- A factor $2^m m$, m being the number of items (attributes) and
- a factor n^2 since $|T_r| = n^2$.

The first factor cannot be improved in the worst case, but the second one can be reduced to n, so that the final complexity is the same that the complexity of an AR mining algorithm (though obviously, both perform different tasks!). We explain this in the next section.

4.2 Efficient calculation of support

It is possible to calculate the support of an itemset I_V in T_r by exploring only once the set of n tuples in r instead of exploring the corresponding set of n^2 transactions. This solution is based on the following result:

Proposition 4.1 *The support of an itemset I_V is*

$$S(I_V) = \frac{1}{n^2} \sum_{i=1}^{K} \sum_{p=1}^{n_{v_i}} (2p - 1) \tag{27}$$

Proof From (5) and given that for each number $x \in \mathbb{N}$, with $x > 0$

$$x^2 = \sum_{p=1}^{x} (2p - 1)$$ □

Algorithm 1 employs this result to obtain $S(I_V)$ in T_r in time $O(n)$. The reduction in complexity from n^2 to n is achieved since each time a tuple t_i is scanned we are implicitly scanning $2i - 1$ transactions in T_r, corresponding to every pair of tuples of the

Algorithm 1 Algorithm to obtain the support of an itemset I_V

1. $S(I_V) \leftarrow 0$
2. For each $i \in \{1, \ldots, K\}$
   (a) $N(V, v_i) \leftarrow 0$
3. For each $t \in r$
   (a) $N(V, t[V]) \leftarrow N(V, t[V]) + 1$
   (b) $S(I_V) \leftarrow S(I_V) + 2N(V, t[V]) - 1$
4. Exit: $S(I_V)/n^2$ is the support of the itemset I_V.

form $t_j t_i$ or $t_i t_j$ with $j \leq i$. Among these we know that only those with $t_j[V] = t_i[V]$ contribute to the support of I_V.

Since the number of tuples t_j with $j \leq i$ such that $t_j[V] = t_i[V]$ is stored in $N(V, t_i[V])$ and updated in step 3a, we know that the number of transactions $t_j t_i$ or $t_i t_j$ supporting I_V is $2N(V, t_i[V]) - 1$, hence $S(I_V)$ is updated accordingly in step 3b.

4.3 Adapting existing algorithms

In principle, algorithms to obtain frequent itemsets with items of the form $<At, a>$ in r can be adapted to obtain frequent itemsets I_V in T_r as follows:

1. Add a variable S_V for each itemset I_V.
2. Calculate S_V at the same time that the support of $<V, v> \forall v \in dom(V)$ as in Algorithm 1 (we note here $<V, v>$ the itemset $\{<At_k, a_{kl}> | At_k \in V, a_{kl} \in dom(At_k)\}$).
3. Eliminate the variables $N(V, v)$ once $S(I_V)$ is obtained.

The rest of the process depends on the algorithm of our choice. Notice that

- The support of items of the form $<At, a>$ in r is calculated only as a way to obtain the support of I_{At} in T_r. The corresponding variables are eliminated once $S(I_{At})$ has been obtained.
- While calculating the support of an itemset I_V, the support of every value of V (i.e., every combination of values of attributes in V appearing in r) is calculated, despite all their subsets were frequent or not.
- For those algorithms using candidate generation, the property "every subset of a frequent itemset must be a frequent itemset" holds also for itemsets of the form I_V. Obviously, candidate generation considers itemsets of the form I_V with $V \subseteq ER$ only.

Once the frequent itemsets of the form I_V are found, strong dependencies can be obtained in the usual way. It is easy to obtain the certainty factor from the confidence of the rule and the support of the consequent.

The adaptation is possible in principle when to consider several transactions of T_r at a time (as we do) is not an inconvenience for the heuristics and techniques employed by the algorithm.

In order to test the viability of this approach we have adapted algorithm Apriori. In the next section we discuss about the complexity of the modified algorithms and, in

particular, we show that our adapted version of Apriori performs reasonably well on real databases. Adaptation of more efficient algorithms is left to future research.

4.4 Complexity

From now on we shall call items of the form $<At, a>$ AR-items. In the same way items of the form it_{At} will be called AD-items. Following this idea we shall talk of AR-itemset, AD-itemset, AR-algorithm and AD-algorithm. In the previous section we showed how to adapt an AR-algorithm to obtain an AD-algorithm. The question is, how do the modifications affect to the complexity and performance of the original algorithm? Of course, the answer to this question may vary depending on the specific features of each AR-algorithm, but what is sure is that the number of tuples and itemsets are the main factors affecting complexity. Taking into account only that information, our conclusions are the following:

- Complexity in the worst case (i.e. every AR-itemset is frequent and hence every AD-itemset is frequent) is the same for ARs and ADs, since in both cases the algorithms will calculate the support of every AR-itemset.
- Both AR and AD-algorithms have a factor that is linear in the number of tuples.
- The rest of the complexity depends on the number of itemsets considered, but there isn't a fixed relation between the number of AR-itemsets to be calculated and stored for both algorithms. In particular, it depends on the minimum support considered. This is illustrated in Example 1.

Example 1 Suppose we apply an AR-algorithm and the corresponding adaptation for AD on Table 7 with $minsupp = 0.3$. In a first step, the AR-algorithm will calculate the support of every AR-item and will find that only the items $<At_1, a>$, $<At_2, f>$ and $<At_3, k>$ are frequent (support is $1/3>0.3$). It is easy to see that any combination of these items is also frequent in Table 7, so in the whole process the AR-algorithm will calculate and store the support of 19 itemsets (15 in the first step and 4 more corresponding to the 2-items and 3-items that can be obtained from the three frequent items). The AD-algorithm will calculate the support of 15 items in the first step as well, in order to obtain the support of it_{At_1}, it_{At_2} and it_{At_3}, but they are not frequent (support is $2/9<0.3$), so the algorithm will stop.

Table 7 A toy relation with three attributes

At_1	At_2	At_3
a	f	k
a	f	k
b	g	l
c	h	m
d	i	n
e	j	o

Now, suppose we apply both algorithms again with $minsupp = 0.2$. The AR-algorithm will do the same work that in the previous case, but the AD-algorithm will find in the first step that every item is frequent. In fact, it is easy to see that every AD-itemset in Table 7 is frequent when $minsupp = 0.2$, so the support of all the AR-itemsets will be calculated in order to obtain the support of every AD-itemset. The final amount of AR-itemsets calculated will be 35 (though only the support of 15 of them will be maintained in memory in every step, since once the global support of an AD-itemset is obtained, the memory employed for the variables that store the support of the corresponding AR-itemsets is set free). □

Hence, the performance of the AD-algorithm with respect to the corresponding AR-algorithm depends on the database since it is related to the number of itemsets and the minimum support considered. This conclusion is confirmed by some of the experiments in the next section.

5 Empirical evaluation

In previous sections we have motivated the use of support in order to determine how trivial an AD is, and we have proposed certainty factor as accuracy measure for ADs. Furthermore, we have shown the theoretical properties of this approach.

In this section we evaluate the proposal from an empirical point of view, with three objectives in mind:

- To show that the approach performs well in real-life applications, i.e., it is possible to implement algorithms for mining non-trivial ADs that expend a reasonable amount of time and space (Sect. 5.1).
- To show that a significant number of trivial dependencies are pruned by using a minimum support threshold, and hence the contribution of the approach to the task of finding non-trivial dependencies is significant (Sect. 5.2). In the same section, another objective is to show that certainty factors are useful in order to discard a large amount of misleading dependencies that are not detected by confidence.
- To show that the non-trivial ADs obtained are useful in practice, i.e., in real-life applications (Sect. 5.3).

5.1 Implementation and performance

We have adapted the algorithm Apriori to discover ADs, following the ideas in Sect. 4.3. The language employed was JAVA, and the experiments were performed with a PC Pentium IV 2.0 Ghz under Windows XP. Let us remark that the objective of this implementation was not to obtain the fastest algorithm, but to show that it is possible to implement algorithms for mining non-trivial ADs that expend a reasonable amount of time and space.

For our experiments, we have employed different databases implemented in ORACLE 9i and accessed through the JDBC interface. We have not used intermediate files to store the database.

Table 8 Average performance of our adaptation of Apriori to find ADs in ten executions

Database	Attributes	Tuples	Values	Time (s)	Mem (Kb)
Ahalone	9	4177	675.22	3.3	1617
Hepatitis	20	155	18.3	40.19	13682
Mushroom	23	8124	5.17	2496	128577
Adult	15	32561	1476.4	2315.9	94801
Breastwis	10	699	9.2	20.9	11089
Breastwis × 64	10	44736	9.2	616.3	11089
Breastwis × 128	10	89472	9.2	1246	11089
Chess	7	28056	8.29	13.2	254
Letter	17	20000	16.59	72.1	3774

The databases were obtained from the UCI Machine Learning Repository.[1] The databases "Breastwis x64" and "Breastwis x 128" were obtained from "Breastwis" (Winsconsin-breast-cancer database) by concatenation of copies of the database as detailed in Huhtala et al. (1998). Table 8 shows the name, number of attributes and tuples, and average number of values per attribute for each database.

In order to evaluate the performance both in time and space of our algorithm we have performed ten executions on each database using $minsupp = 0.05$ and looking for itemsets with at most five items. Average time and space are shown in Table 8. The maximum time expended was around 42 min for the database Mushroom; the maximum memory employed was around 126 Mb for the same database.

As a way to assess to what extent this performance is reasonable, we have verified that the time employed is in average similar (sometimes better and sometimes worst) to the time employed by the Apriori algorithm for mining ordinary association rules (not those defining ADs in our approach) on the same databases, using an implementation with the same language and database management system, and running on the same PC. This is in accordance with our conclusions in Sect. 4.4 about complexity. In all the cases, mining for ADs needed more memory that mining for ARs, but the amount of memory employed was affordable for an ordinary PC (less than 128 MB in all our experiments).

These results support the claim that our adaptation to ADs performs reasonably well in the sense that the memory needed is affordable and the time expended is comparable to that employed in other mining tasks with similar theoretical complexity and under similar conditions (database size, implementation language, PC employed, etc. . .) such as mining for ARs in a table. Let us remark that this comparison makes no other sense since, obviously, mining ordinary ARs and mining ADs in the same table are completely different and non-comparable tasks.

[1] http://www.ics.uci.edu/mlearn/MLRepository.html.

5.2 Detecting misleading ADs

Trivial dependencies are dependencies with very high accuracy and very low support. In real databases it is easy to find many trivial dependencies generated by sets of attributes with very low joint support, as we explained in Sect. 2.2. At the same time, taking into account the support but using confidence lead us to obtain misleading ADs that correspond to independence or negative dependence, as we explained in Sect. 2.3.

The first problem can be solved by mining for ADs with support above a minimum threshold. The second one can be solved by using an accuracy measure incorporating the support of the consequent, as certainty factor. We deal with these problems in the next sections. In both of them we have employed the databases in Table 8.

5.2.1 Effectiveness of support

Any dependence containing a key of a relation in the left part is totally accurate. For a single key with a attributes in a table with $a+b+1$ attributes, we can obtain in the order of 2^b trivial dependencies. For example, in the database mushroom (23 attributes), for every key with 2 attributes we can obtain in the order of 2^{20} trivial (uninteresting for our purposes) dependencies. Even if we restrict the number of attributes to be at most 4 in the antecedent, the number of potential trivial ADs is huge.

In practice it is usual to find several attributes with very low support, close to the support of keys (the minimum possible), that generate many trivial, in the sense of accurate but uninteresting, rules. For example, as shown in Sánchez (1999), the AD *BirthDate* $\rightarrow$ *Sex* in a database about surgical operations in the University Hospital of Granada has a rather high accuracy, as can be seen in Table 9, but it is clearly counterintuitive. We can see that most of the existing measures (except perhaps μ, that takes into account the support, and *CF*) consider that this AD is rather accurate. However, the support of the AD is $\approx 1.8E^{-4}$. That support is equivalent to the support of an AD whose theory contained $\approx$ 5484 ARs with the same support. This theory is too complex and hence the rule is unimportant, as intuitively expected.

In order to show the effectiveness of support in detecting trivial ADs, we have measured in every database the percentage of non-trivial dependencies for different values of minimum support (i.e., for different degrees of triviality). Only dependencies with a maximum of four attributes in the left and one on the right were considered. The accuracy was assessed using different minimum thresholds for certainty factor.

The results are shown in Tables 10–12. The column for minsupp = 0 indicates the number of dependencies obtained without taking into account the support. The tables show the number and percentage of trivial dependencies eliminated for minsupp = 0.05, minsupp = 0.1 and minsupp = 0.25. For databases Abalone and Letter there

Table 9 Accuracy for *BirthDate* $\rightarrow$ *Sex* in surgical operations

Measure	k	g_1	g_2	g_3	*pdep*	τ	μ	C	*CF*
Value $\approx$	0.78	$7.2E^{-5}$	0.22	0.08	0.89	0.78	0.44	0.72	0.43

Table 10 Effectiveness of support in detecting trivial dependencies (percentage of detection) for different minimum support thresholds, minCF = 0.4

Database	0	0.05	%	0.1	%	0.25	%
Abalone	1264	0	100	0	100	0	100
Hepatitis	23168	3953	83	2903	87.5	264	98.8
Mushroom	89906	40772	55	11031	87	422	99
Adult	6937	335	95	156	97.8	0	100
Breastwis	3151	1275	60	773	75.5	51	98.4
Chess	0	0	–	0	–	0	–
Letter	10321	0	100	0	100	0	100

Table 11 Effectiveness of support in detecting trivial dependencies (percentage of detection) for different minimum support thresholds, minCF = 0.7

Database	0	0.05	%	0.1	%	0.25	%
Abalone	1164	0	100	0	100	0	100
Hepatitis	7844	688	92.3	417	94.7	47	99.4
Mushroom	45179	20878	54	5632	87.5	287	99.3
Adult	5404	191	96.5	89	98.4	0	100
Breastwis	1654	587	64.5	335	80	25	98.4
Chess	0	0	–	0	–	0	–
Letter	141	0	100	0	100	0	100

Table 12 Effectiveness of support in detecting trivial dependencies (percentage of detection) for different minimum support thresholds, minCF = 0.9

Database	0	0.05	%	0.1	%	0.25	%
Abalone	1021	0	100	0	100	0	100
Hepatitis	3126	17	95.5	2	99.9	1	99.9
Mushroom	21681	11132	49	3343	85	239	98.9
Adult	3193	130	96	46	98.5	0	100
Breastwis	534	164	70	91	83	7	98.6
Chess	0	0	–	0	–	0	–
Letter	0	0	–	0	–	0	–

is a maximum percentage of reduction since the support of dependencies is very low. Database Chess does not contain any dependence with certainty factor above 0.4 (the lower minCF value considered). In the rest of the cases, the reduction is very important, with a minimum of 49%, showing that support allows us to detect and eliminate a very large amount of misleading dependencies.

5.2.2 *Certainty factor versus confidence*

As we explained in Sect. 2.3, confidence is not able to detect neither statistical independence nor negative dependence between V and W in a dependence $V \rightarrow W$. This is very important since that means that the dependence does not hold.

An example of such ADs, shown in Sánchez (1999), is *Sex* $\rightarrow$ *Transport*, relating the sex of a patient to the mean of transport employed for arriving at the urgency service of the University Hospital of Granada. The confidence of that dependence is 0.97, when indeed *Sex* and *Transport* are intuitively independent, as the certainty factor (10^{-3}) confirms.

We have measured the percentage of reduction in the number of accurate ADs when using certainty factor instead of confidence. We made the experiments for all the databases using different values of support. Again, only dependencies with a maximum of four attributes in the left and one in the right were considered. The results in Tables 13 and 14 show that using certainty factors we obtain an important reduction of the number of dependencies obtained. The dependencies discarded correspond to negative dependencies and independencies. By the theoretical properties of certainty factor, all the strong dependencies using certainty factor as accuracy measure are also strong following confidence.

5.3 Some application experiences

Our approach to approximate dependencies, and a fuzzy extension of it (Berzal et al. 2005) have been applied in several practical situations. In Sánchez et al. (2003), a data mining alternative to the classical correspondence analysis in statistics was introduced on the basis of our new definition of AD. This tool was employed to analyze correspondences between user and scientific knowledge in an agricultural environment, in particular to compare the perception of soils by farmers and pedologist, with the objective of improving performance of farms. Data was obtained by polling farmers cultivating

Table 13 Effectiveness of certainty factor against confidence: number of strong dependencies obtained by each measure and percentage of dependencies discarded by certainty factor for different minimum accuracy thresholds, minsupp = 0.01

Database	0.25			0.5			0.75		
	Conf	CF	%	Conf	CF	%	Conf	CF	%
Abalone	2	0	100	0	0	–	0	0	–
Hepatitis	34174	16924	50.5	32110	7772	75.8	14909	1982	86.7
Mushroom	200666	119971	40	133829	72222	46.1	83320	38844	53.4
Adult	6806	1512	87.8	5517	856	84.5	3377	476	86
Breastwis	2285	2023	11.5	2026	1873	7.6	1410	740	47.6
Chess	22	0	100	0	0	–	0	0	–
Letter	148	54	63.5	12	5	58.3	0	0	–

Table 14 Effectiveness of certainty factor against confidence: number of strong dependencies obtained by each measure and percentage of dependencies discarded by certainty factor for different minimum accuracy thresholds, minsupp = 0.05

Database	0.25			0.5			0.75		
	Conf	CF	%	Conf	CF	%	Conf	CF	%
Abalone	0	0	–	0	0	–	0	0	–
Hepatitis	16859	7102	58	16015	2477	84.5	6535	373	94.3
Mushroom	84608	52509	58	65991	33786	49	43707	18416	57.9
Adult	2558	527	79.4	2219	276	87.6	1374	155	88.8
Breastwis	1457	1308	10.3	1311	1208	7.9	913	488	46.6
Chess	2	0	100	0	0	–	0	0	–
Letter	8	0	100	0	0	–	0	0	–

olives in the southeast of Spain, in the context of a European project supported by the FEDER programme, and the obtained dependencies were analyzed by pedologist. However, the proposed correspondence analysis can be employed in many other problems.

In Calero et al. (2003, 2004b,c) we have introduced a methodology that employ fuzzy approximate dependencies to perform a high-level analysis of data. After that, the corresponding set of association rules can be employed to analyze the dependencies obtained, or they can be employed to look for associations between values of attributes that are not related by a strong approximate dependence. The methodology was applied to the analysis of soil properties, specifically color as an aggregation of other soil properties. Soil experts found some interesting dependencies relating soil properties as *clay percentage, organic carbon percentage, useful water* and *dry chroma* among others.

A different application can be found in Berzal et al. (2003), where we employed our approach to discover approximate dependencies in the STULONG database in the context of the Discovery Challenge of the ECML/PKDD 2003 conference. STULONG provides information about a twenty years lasting longitudinal study of the risk factors of the atherosclerosis in a population of 1417 middle aged men, performed in the Faculty of Medicine of the Charles University, and the Charles University Hospital of Prague. A large analysis was performed and many interesting dependencies were obtained involving attributes about social factors, like education, and physical activities, smoking, alcohol consumption, blood pressure, cholesterol levels and body mass index.

6 Conclusions and future research

We have introduced a new definition of approximate dependence in a table r as an association rule in a set of transactions T_r. The set T_r is obtained from r so that association rules in T_r have the semantics of functional dependencies in r. Support and accuracy are employed to assess the dependencies, and those with values of these measures over user-defined thresholds are called strong approximate dependencies.

We have shown the advantages of the new approach. First, we have shown that support is a valid measure of the degree to which a dependence is trivial (i.e., it has high accuracy but is not supported by the data). The importance of avoiding trivial dependencies was discussed from different points of view in Piatetsky-Shapiro (1992), Pfahringer and Kramer (1995) and Kramer and Pfahringer (1996). Additionally, support can be employed to bound a levelwise search for dependencies as in the case of association rules. In this sense, we have proposed a way to adapt any association rule mining algorithm for mining non-trivial dependencies, and we have applied it to the Apriori algorithm.

We have proposed certainty factor as the accuracy measure for approximate dependencies, and we have shown its suitability by means of its properties, a theoretical comparison with other proposals, and our experiments. In particular, we have shown that certainty factor performs better than confidence. As a way to interpret in r the support and certainty factor obtained from the set of transactions T_r, we have formulated these measures in terms of the support and accuracy of the rules that form the theory of the dependence, i.e., the set of rules relating values of the attributes in the antecedent and consequent of the dependence.

Finally, we have shown that it is possible to develop algorithms for mining dependencies that expend a reasonable amount of space and time, and we have illustrated the utility of the new definition of approximate dependence by means of references to applications in real-world problems. In some of these references, the new definition of approximate dependence was employed to define a data mining counterpart of the classical statistical correspondence analysis, and to define fuzzy approximate dependencies as an extension to deal with different types of imprecision in data.

As future work we plan to adapt algorithms other than Apriori in order to obtain more efficient algorithms, and to keep applying the new approach to other real-world problems.

References

Agrawal R, Imielinski T, Swami A (1993) Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD conference, pp 207–216

Bell S (1995) Discovery and maintenance of functional dependencies by independencies. In: Proceedings of the first international conference on knowlege discovery and data mining (KDD'95), pp 27–32

Bell S (1997) Dependency mining in relational databases. In: Proceedings of the ECSQARU-FAPR'97, pp 16–29

Berzal F, Blanco I, Sánchez D, Vila M (2002) Measuring the accuracy and interest of association rules: A new framework. Intell Data Anal 6:221–235

Berzal F, Cubero J, Sánchez D, Serrano J, Vila MA (2003) Finding fuzzy approximate dependencies within STULONG data. In: Berka P (ed) Proceedings of the ECML/PKDD 2003 workshop on discovery challenge, pp 34–46

Berzal F, Blanco I, Sánchez D, Serrano J, Vila MA (2005) A definition for fuzzy approximate dependencies. Fuzzy Set Syst 149:105–129

Bitton D, Millman J, Torgersen S (1989) A feasibility and performance study of dependency inference. In: Proceedings of the 5th international conference on data engineering, pp 635–641

Bosc P, Lietard L, Pivert O (1997) Functional dependencies revisited under graduality and imprecision. In: Annual meeting of NAFIPS, pp 57–62

Bra PD, Paredaens J (1983) Horizontal decompositions for handling exceptions to functional dependencies. Adv Database Theor 2:123–144

Brin S, Motwani R, Ullman J, Tsur S (1997) Dynamic itemset counting and implication rules for market basket data. SIGMOD Rec 26(2):255–264

Calero J, Delgado G, Sánchez-Marañón M, Sánchez D, Serrano J, Vila MA (2003) Helping user to discover association rules. a case in soil color as aggregation of other soil properties. In: Proceedings of the 5th international conference on enterprise information systems, ICEIS'03, pp 533–540

Calero J, Delgado G, Sánchez D, Serrano J, Vila MA (2004a) A proposal of fuzzy correspondence analysis based on flexible data mining techniques. In: López-Díaz M, Gil M, Grzegorzewski P, Hyrniewicz O, Lawry J (eds) Soft methodology and random information systems. Advances in soft computing series. Springer, pp 447–454

Calero J, Delgado G, Sánchez-Marañón M, Sánchez D, Vila MA, Serrano J (2004b) An experience in management of imprecise soil databases by means of fuzzy association rules and fuzzy approximate dependencies. In: Proceedings of the 6th international conference on enterprise information systems, ICEIS'04, pp 138–146

Calero J, Delgado G, Serrano J, Sánchez D, Vila MA (2004c) Fuzzy approximate dependencies over imprecise domains. an example in soil data management. In: Proceedings of the IADIS international conference applied computing 2004, pp 396–403

Cubero J, Cuenca F, Blanco I, Vila M (1998) Incomplete functional dependencies versus knowledge discovery in databases. In: Proceedings of the EUFIT'98, Aachen, Germany, pp 731–74

Delgado M, Marín N, Sánchez D, Vila M (2003) Fuzzy association rules: general model and applications. IEEE Trans Fuzzy Syst 11(2):214–225

Dubois D, Hüllermeier E, Prade H (2006) A systematic approach to the assessment of fuzzy association rules. Data Min Knowl Disc 13(2):167–192

Flach P, Savnik I (1999) Database dependency discovery: a machine learning approach. AI Commun 12(3):139–160

Gunopulos D, Mannila H, Saluja S (1997) Discovering all most specific sentences by randomized algorithms. In: Afrati F, Kolaitis P (eds) Proceedings of the international conference on database theory, pp 215–229

Huhtala Y, Karkkainen J, Porkka P, Toivonen H (1998) Efficient discovery of functional and approximate dependencies using partitions. In: Proceedings of the 14th international conference on data engineering, pp 392–401

Huhtala Y, Karkkainen J, Porkka P, Toivonen H (1999) TANE: an efficient algorithm for discovering functional and approximate dependencies. Comput J 42(2):100–111

Kivinen J, Mannila H (1995) Approximate dependency inference from relations. Theor Comput Sci 149(1):129–149

Kramer S, Pfahringer B (1996) Efficient search for strong partial determinations. In: Proceedings of the 2nd international conference on knowledge discovery and data mining (KDD'96), pp 371–374

Lavrac N, Flach P, Zupan B (1999) Rule evaluation measures: a unifying view. In: LNAI 1364. Springer-Verlag, pp 74–185

Lopes S, Petit J, Lakhal L (2002) Functional and approximate dependency mining: Database and FCA points of view. J Expt Theor Artif Intell 14:93–114

Lukasiewicz J (1970) Die logishen grundlagen der wahrscheinilchkeitsrechnung. In: Borkowski L (ed) Jan Lukasiewicz - Selected Works. North Holland Publishing Company, Amsterdam, London, Polish Scientific Publishers, Warsaw, pp 16–63

Mannila H, Räihä K (1992) On the complexity of inferring functional dependencies. Discrete Appl Math 40:237–243

Mannila H, Räihä K (1994) Algorithms for inferring functional dependencies. Data Knowl Eng 12(1):83–99

Pawlak Z (1982) Rough sets. Int J Comput Inf Sci 11(5):341–356

Pawlak Z (1991) Rough sets: theoretical aspects of reasoning about data. Kluwer Academic Publishing, Dordrecht

Pfahringer B, Kramer S (1995) Compression-based evaluation of partial determinations. In: Proceedings of the first international conference on knowledge discovery and data mining (KDD'95), pp 234–239

Piatetsky-Shapiro G (1991). Discovery, analysis, and presentation of strong rules. In: Piatetsky-Shapiro G, Frawley W (eds) Knowledge discovery in databases. AAAI/MIT Press, pp 229–238

Piatetsky-Shapiro G (1992) Probabilistic data dependencies. In: Zytkow J (ed) Proceedings of machine discovery workshop, pp 11–17

Russell S (1989) The use of knowledge in analogy and induction. Pitman Publishing

Sánchez D (1999) Adquisición de relaciones entre atributos en bases de datos relacionales (Translates to: Acquisition of relationships between attributes in relational databases) (in Spanish). PhD thesis, Department of Computer Science and Artificial Intelligence, University of Granada

Sánchez D, Serrano J, Vila M, Aranda V, Calero J, Delgado G (2003) Using data mining techniques to analyze correspondences between user and scientific knowledge in an agricultural environment. In: Piattini M, Filipe J, Braz J (eds) Enterprise information systems IV. Kluwer Academic Publishers, Hingham, MA, USA pp 75–89

Savnik I, Flach P (1993) Bottom-up induction of functional dependencies from relations. In: Piatetsky-Shapiro G (ed) Knowledge discovery in databases, papers from the 1993 AAAI workshop. AAAI, pp 174–185

Schlimmer J (1993) Efficiently inducing determinations: a complete and systematic search algorithm that uses optimal pruning. In: Piatetsky-Shapiro G (ed) Proceedings of the 10th international conference on machine learning, pp 284–290

Shen W (1991) Discovering regularities from large knowledge bases. In: Proceedings of the 8th international workshop on machine learning, pp 539–543

Shortliffe E, Buchanan B (1975) A model of inexact reasoning in medicine. Math Biosci 23:351–379

Silverstein C, Brin S, Motwani R (1998) Beyond market baskets: generalizing association rules to dependence rules. Data Min Knowl Disc 2:39–68

Ziarko W (1991) The discovery, analysis and representation of data dependencies in databases. In: Piatetsky-Shapiro G, Frawley W (eds) Knowl discovery databases. AAAI/MIT Press, pp 195–209