On Objective Measures of Rule Surprisingness.

Alex A. Freitas

CEFET-PR (Federal Center of Technological Education), DAINF (Dept. of Informatics)
Av. Sete de Setembro, 3165, Curitiba - PR, 80230-901, Brazil
alex@dainf.cefetpr.br
http://www.dainf.cefetpr.br/~alex

Abstract. Most of the literature argues that surprisingness is an inherently subjective aspect of the discovered knowledge, which cannot be measured in objective terms. This paper departs from this view, and it has a twofold goal: (1) showing that it is indeed possible to define *objective* (rather than subjective) measures of discovered rule surprisingness; (2) proposing new ideas and methods for defining objective rule surprisingness measures.

1 Introduction

A crucial aspect of data mining is that the discovered knowledge (usually expressed in the form of “if-then” rules) should be somehow interesting, where the term interestingness is arguably related to the properties of surprisingness (unexpectedness), usefulness and novelty of the rule [Fayyad et al. 96]. In this paper we are interested in quantitative, objective measures of one of the above three properties, namely rule *surprisingness*.

In general, the evaluation of the interestingness of discovered rules has both an objective (data-driven) and a subjective (user-driven) aspect. In the particular case of surprisingness, however, most of the literature argues that this property of the discovered rules is inherently subjective - see e.g. [Liu et al. 97].

Hence, objective measures of rule surprisingness seem to be something missing, or at least underexplored, in the literature, and this is part of the motivation for the discussion presented in this paper. Another motivation, of course, is that objective measures of rule surprisingness have one important advantage. Objectiveness is strongly related to domain independence, while subjectiveness is strongly related to domain dependence. Hence, objective measures of rule surprisingness are, in principle, more generic than subjective measures. (However, subjective measures are still necessary - see below.)

We should emphasize that the aim of the paper is *not* to propose a new rule surprisingness measure. Note that any rule interestingness measure is part of the bias of the corresponding data mining algorithm, and it is well-known that any data mining bias has a domain-dependent effectiveness.

Hence, instead of investigating the effectiveness of any particular rule surprisingness measure, the aim of this paper is rather to suggest and discuss ideas that can be used to objectively measure the degree of rule surprisingness, in a generic way. We hope that this discussion will be a useful reference for other researchers who have a specific data mining problem to be solved, since these researches could use the ideas discussed in this paper to design their rule surprisingness measure.

It should be noted that the goal of our work is to complement, rather than to replace, the existing subjective measures of rule surprisingness. Actually, in practice both objective and subjective approaches should be used to select surprising rules. In our case, the new ideas for objective rule suprisingness measures proposed in this paper can be used as a kind of filter to select potentially surprising rules, among the many rules discovered by a data mining algorithm. We can then return to the user a much smaller number of potentially surprising rules, and let him/her judge the ultimate surprisingness of those rules only. Obviously, the user will be able to save a lot of his/her rule analyzis time with this approach, particularly when the data mining algorithm discovers a large number of rules.

This paper is organized as follows. Section 2 discusses how to measure the surprisingness of small disjuncts. Section 3 discusses how to measure the degree of surprisingness associated with individual attributes in a rule antecedent. Section 4 discusses how we can discover surprising knowledge by detecting occurrences of Simpson's paradox. Finally, section 5 concludes the paper.

2 On the Surprisingness of Small Disjuncts

A rule set can be regarded as a disjunction of rules, so that a given rule can be regarded as a disjunct. The size of a disjunct (rule) is the number of tuples satisfied by the rule's antecedent – i.e. the "if part" of the rule. Thus, small disjuncts are rules whose number of covered tuples is small, according to some specified criterion (e.g. a fixed threshold, or a more flexible criterion).

At first glance, it seems that small disjuncts are undesirable, since they have little generality and tend to be error-prone (see below). Based on this view, most data mining algorithms have a bias favoring the discovery of large disjuncts.

However, small disjuncts have the potential to capture truly unexpected relationships, in the data. For instance, [Provost & Aronis 96] report an application where a small disjunct discovered by a data mining algorithm was considered truly interesting and led to substantial new research in the application domain. Hence, it would be nice if the data mining algorithm could automatically evaluate the surprisingness of small disjuncts, reporting to the user only the most promising ones.

The remaining of this section is divided into two parts. Subsection 2.1 reviews previous research on small disjuncts, while subsection 2.2 proposes our idea to objectively measure the surprisingness of small disjuncts.

2.1 A Review of Small Disjuncts

As mentioned above, small disjuncts are error-prone. Since they cover a small number of tuples, it is possible that they cover mainly noise. At first glance, it seems that this problem can be solved by simply discarding small disjuncts.

Unfortunately, however, prediction accuracy can be significantly reduced if all small disjuncts are discarded by the data mining algorithm, as shown by [Holte et al. 89]. This is a particularly serious problem in domains where the small disjuncts collectively match a large percentage of the number of tuples belonging to a given class [Danyluk & Provost 93]. The main problem is that a small disjunct can represent either a true exception occurring in the data or simply noise. In the former case the disjunct should be maintained, but in the latter case the disjunct is error prone and should be discarded. Unfortunately, however, it is very difficult to tell which is the case, given only the data.

[Holte et al. 89] suggested that one remedy for the problem of small disjuncts was to evaluate these disjuncts by using a bias different from the one used to evaluate large disjuncts. Hence, they proposed that small disjuncts be evaluated by a maximum-specificity bias, in contrast with the maximum-generality bias (favoring the discovery of more general rules) used by most data mining algorithms. [Ting 94] further investigated this approach, by using an instance-based learner (as far as we can go with the maximum-specificity bias) to evaluate small disjuncts.

It should be noted that the above literature has studied the effect of small disjuncts mainly in the classification accuracy of the discovered knowledge. In this paper we are rather interested in the effect of small disjuncts in the surprisingness of the discovered knowledge, as will be discussed in the next subsection.

2.2 Measuring the Surprisingness of Small Disjuncts

We propose that a small disjunct be considered as surprising knowledge to the extent that it predicts a class different from the class predicted by the minimum generalizations of the disjunct. The minimum generalizations of a disjunct are defined as follows. (Henceforth, we use simply the term disjunct to refer to small disjuncts.)

Let the disjunct be composed by the conjunction of m conditions, of the form $cond_1$ AND $cond_2$ AND ... $cond_m$, where each $cond_k$, k=1,...,m, is a triple of the form <Att op Val>, where Att is an attribute, op is a relational operator in the set $\{<,>,\geq,\leq,=\}$ and Val is a value belonging to the domain of Att. Also, following a common practice in data mining algorithms, if the attribute Att is continuous (real-valued), then op is in the set $\{<,>,\geq,\leq,\}$; whereas if the attribute Att is categorical then op is the “=“ operator.

A disjunct has m minimum generalizations, one for each of its m conditions. The k-th minimum generalization of the disjunct, associated with $cond_k$, k=1,...,m, is defined as follows. If Att in $cond_k$ is categorical, then the k-th minimum generalization of the disjunct is achieved by removing $cond_k$ from the disjunct. If Att in $cond_k$ is continuous, then the k-th minimum generalization of the disjunct can be defined in two ways, namely: (a) by removing $cond_k$ from the disjunct; (b) by adding a small value α to the value of the "cut-point" Val in $cond_k$, when op in $\{<,\leq\}$, or subtracting α from Val in $cond_k$, when op in $\{>,\geq\}$; where α is a user-defined, problem-dependent constant. It is up to the user to choose one of these two definitions of minimum generalization for continuous attributes, depending on the application.

To clarify the above definition (b) of minimum generalization, let us consider, as an example, the following rule condition: "Age < 23". Supposing that α is specified as 5, the minimum generalization of this condition would be "Age < 28". In practice there might be several continuous attributes, and different absolute values for α would be necessary, due to differences of scale among the attributes - for instance, for the attribute "Yearly Income", $\alpha = 5$ is too small. Obviously, it would be tedious to specify a different absolute value of α for each continuous attribute. A solution is to specify a relative value of α, such as 5%. Then, for each continuous attribute, the actual absolute value for α would be automatically calculated as 5% of the highest value observed for that attribute. An alternative definition of minimum generalization for continuous attributes could use some kind of percentile-based approach.

In any case, note that a minimum generalization produces a new disjunct which covers a superset of the tuples covered by the original disjunct. As a result, the class distribution of the tuples covered by the new disjunct (produced by minimum generalization of the original disjunct) can be significantly different from the class distribution of the tuples covered by the original disjunct.

Therefore, after a new disjunct is produced by minimum generalization, its predicted class (i.e. the consequent of the rule) is re-computed. More precisely, the class predicted by the new disjunct is determined by using the same procedure used by the data mining algorithm that discovered the original disjunct. (Typically, picking up the class with largest relative frequency among the tuples covered by the disjunct.)

We are now ready to define a way to objectively measure the surprisingness of small disjuncts discovered by a data mining algorithm. Let C be the class predicted by the original disjunct and C_k be the class predicted by the disjunct produced by the k-th minimum generalization of the original disjunct, k=1,...,m, as explained above. We then compare C against each C_k, k=1,...,m, and count the number of times C differs from C_k. This result, an integer number in the interval 0...m, is defined as the raw surprisingness of the original disjunct, denoted $DisjSurp_{raw}$. The higher $DisjSurp_{raw}$, the more surprising the disjunct is.

Note that the value of $DisjSurp_{raw}$ is significantly influenced by the number of conditions m in the disjunct, which is in turn a measure of syntactic complexity. Several data mining algorithms already use a measure of syntactic complexity as part of their inductive bias. In this case, to avoid confusion between measures of syntactic complexity and measures of disjunct surprisingness, we can render the latter somewhat more independent from the former by defining a normalized disjunct surprisingness measure, as follows: $DisjSurp_{norm} = DisjSurp_{raw} / m$; where m is the number of conditions of the disjunct. Clearly, $DisjSurp_{norm}$ takes on values in the interval [0..1]. The higher $DisjSurp_{norm}$, the more surprising the disjunct is. However, it should be noted that this normalized measure has a bias towards rules with fewer conditions, since it will probably be difficult for rules with many conditions to hold a high normalized value. The suitability of this bias depends on the application domain.

Finally, note that the above measure of small disjunct surprisingness is being proposed as a post-processing approach, applied once the rules have been discovered. This approach seems consistent with top-down, specialization-driven rule induction algorithms, that iteratively add conditions to a candidate rule, in order to specialize it. Note that this is the most common kind of rule induction algorithm used in practice.

3 On The Surprisingness of a Rule's Individual Attributes

Most rule surprisingness measures (or, more generally, rule interestingness measures) consider the rule antecedent as a whole, without paying attention to the individual attributes occurring in the rule antecedent - see e.g. the well-known rule interestingness measure proposed by [Piatetsky-Shapiro 91].

In some sense, these rule surprisingness measures are coarse-grained. However, two rules with the same value of a coarse-grained rule surprisingness measure can have very different degrees of interestingness for the user, depending on which attributes occur in the rule antecedent.

In order to evaluate the surprisingness of predicting attributes in a rule antecedent, denoted AttSurp, we propose an information-theory-based measure, as follows - see [Cover & Thomas 91] for a comprehensive review of information theory. First, we calculate InfoGain(A_i), the information gain of each predicting attribute A_i in the rule antecedent, using formulas (1), (2) and (3). In these formulas, Info(G) is the information of the goal (class) attribute G, Info(G|A_i) is the information of the goal attribute G given predicting attribute A_i, A_{ij} denotes the j-th value of attribute A_i, G_j denotes the j-th value of the goal attribute G, Pr(X) denotes the probability of X, Pr(X|Y) denotes the conditional probability of X given Y, and all the logs are in base 2. The index j in formulas (2) and (3) varies in the interval 1..n, where n is the number of goal attribute values. The index k in formula (3) varies in the interval 1..m, where m is the number of values of the predicting attribute A_i.

$$\text{InfoGain}(A_i) = \text{Info}(G) - \text{Info}(G|A_i) \quad (1)$$

where
$$\text{Info}(G) = -\sum_{j=1}^{n} \Pr(G_j) \log \Pr(G_j) \quad (2)$$

$$\text{Info}(G|A_i) = \sum_{k=1}^{m} \Pr(A_{ik}) \left(-\sum_{j=1}^{n} \Pr(G_j|A_{ik}) \log \Pr(G_j|A_{ik}) \right) \quad (3)$$

Attributes with high information gain are good predictors of class, when these attributes are considered individually, i.e. one at a time. However, from a rule interestingness point of view, it is likely that the user already knows what are the best predictors (individual attributes) for its application domain, and rules containing these attributes would tend to have a low degree of surprisingness for the user.

On the other hand, the user would tend to be more surprised if (s)he saw a rule containing attributes with low information gain. These attributes were probably considered as irrelevant by the users, and they are kind of irrelevant for classification when considered individually, one at a time. However, attribute interactions can render an individually-irrelevant attribute into a relevant one, and this phenomenon is intuitively associated with rule surprisingness. Therefore, all other things (such as the prediction accuracy, coverage and completeness of the rule) being equal, we argue that rules whose antecedent contain attributes with low information gain are more surprising than rules whose antecedent contain attributes with high information gain. This idea can be expressed mathematically by defining AttSurp as follows:

$$\text{AttSurp} = 1 \,/\, \left(\sum_{i=1}^{\#att} \text{InfoGain}(A_i) \,/\, \#att \right), \tag{4}$$

where $\text{InfoGain}(A_i)$ is the information gain of the i-th attribute occurring in the rule antecedent and #att is the number of attributes occurring in the rule antecedent. The larger the value of AttSurp, the more surprising the rule is.

4 On the Surprisingness of the Occurrence of Simpson's Paradox

4.1 A Review of Simpson's Paradox

Simpson's paradox [Simpson 51] can be defined as follows. Let a population be partitioned into two mutually exclusive and exhaustive populations, denoted Pop_1 and Pop_2, according to the value of a given binary attribute, denoted 1stPartAtt (First Partitioning Attribute). Let G be a binary goal attribute, which takes on a value indicating whether or not a given situation of interest has occurred in a population, and let G_1 and G_2 be the value of the attribute G in each of the respective populations Pop_1 and Pop_2. Let $Pr(G_1)$ and $Pr(G_2)$ denote the probability that the situation of interest has occurred in Pop_1 and Pop_2, respectively. Assume that $Pr(G_1) > Pr(G_2)$.

Let us now consider the case where both the populations Pop_1 and Pop_2 are further partitioned, in parallel, according to the value of a given categorical attribute, denoted 2ndPartAtt. Let this attribute have m distinct categorical values. We can now compute the probability Pr(G) in each population, for each of these m categories, which we denote by G_{ij}, where i=1,2 is the id of the population and j=1,...,m is the id of the value of 2ndPartAtt. Let $Pr(G_{1j})$ and $Pr(G_{2j})$ denote the probability that the situation of interest has occurred in Pop_1 and Pop_2, in the j-th category of 2ndPartAtt, j=1,...,m.

Finally, Simpson's paradox occurs when, although the overall value of Pr(G) is higher in Pop_1 than in Pop_2, i.e. $Pr(G_1) > Pr(G_2)$, in *each* of the categories produced by 2ndPartAtt the value of Pr(G) in Pop_1 is lower than or equal to its value in Pop_2, i.e. $Pr(G_{1j}) \leq Pr(G_{2j})$, j=1,...,m. The paradox also occurs in the dual situation, i.e. when $Pr(G_1) < Pr(G_2)$ but $Pr(G_{1j}) \geq Pr(G_{2j})$, j=1,...,m.

Some real-life examples of the occurrence of this paradox are mentioned in [Wagner 82]. For instance, the paradox occurred in a comparison of tuberculosis deaths in New York City and Richmond, Virginia, during the year 1910. Overall, the tuberculosis mortality rate of New York was lower than Richmond's one. However, the opposite was observed when the data was further partitioned according to two racial categories: white and non-white. In both the white and non-white categories, Richmond had a lower mortality rate. In terms of the above notation, the 1stPartAtt was *city*; the situation of interest measured by attribute G was the *occurrence of death* in a tuberculosis case; and the 2ndPartAtt was *racial category*.

Some authors have drawn attention to Simpson's paradox in the context of data mining - see e.g. [Glymour et al. 97]. However, most of this literature regards this paradox as a kind of danger, or obstacle, for data mining algorithms. In particular, the existence of this paradox in a given data set can easily fool a data mining algorithm, causing the algorithm to misinterpret a given relationship between some attributes. For instance, decision-tree learners usually build a tree by selecting one attribute at a time. Hence, they can select an attribute that seems to have a certain relationship with a given class, when in reality the true relationship (taking into account attribute interactions) is the reverse of the apparent one.

Instead of considering Simpson's paradox as an obstacle, in this paper we are interested in the potential that the occurrence of Simpson's paradox offers for the discovery of truly surprising knowledge, as discussed in the next subsection.

4.2 Discovering Surprising Knowledge via the Detection of Simpson's Paradox

We suggest to make a data mining algorithm to explicitly search for occurrences of Simpson's paradox and to report the discovered occurrences for the user, as a kind of surprising knowledge.

This search can be performed by the Algorithm 1 below. The input for the algorithm is a list L_G of user-defined binary goal attributes, each of them indicating whether or not a given situation of interest has occurred. The algorithm below is specified in a high level of abstraction, so the two statements that identify the attributes to be put in lists L_1 and L_2 can be expanded in different procedures, using different criteria, as long as three conditions hold: (a) all attributes in L_1 are binary; (b) all attributes in L_2 are categorical; (c) any goal attribute contained in L_G does not appear in L_1 nor in L_2. Note that these conditions are not very strict, and in particular they allow the possibility that an attribute is contained in both L_1 and L_2 (since binary attributes are a particular case of categorical attributes). This possibility justifies the use of the condition $A_2 \neq A_1$ in the third FOR EACH statement of Algorithm 1. In practice, this and other more strict conditions may be directly implemented in the two statements that identify the attributes to be put in lists L_1 and L_2, when Algorithm 1 is refined to achieve a particular implementation.

Algorithm 1 only detects occurrences of Simpson's paradox. Extending the algorithm to explain why the paradox has occurred is beyond the scope of this paper.

```
INPUT: list of user-defined goal attributes, denoted L_G
BEGIN
   identify attributes that can be used as 1stPartAtt and put them in list L_1
   identify attributes that can be used as 2ndPartAtt and put them in list L_2
   FOR EACH goal attribute G in L_G
        FOR EACH attribute A_1 in L_1
             partition population into Pop_1 and Pop_2, according to the values of A_1
             Pr(G_1) = Pr(G="yes"|A_1=1)
             Pr(G_2) = Pr(G="yes"|A_1=2)
             FOR EACH attribute A_2 in L_2 such that A_2 ≠ A_1
                FOR  i=1,2
                    partition Pop_i into m new populations Pop_i1 ... Pop_im,
                      according to the values of A_2
                    FOR j=1,...,m
                      Pr(G_ij) = Pr(G="yes"|A_1=i,A_2=j)
                IF ( Pr(G_1) > Pr(G_2)  AND  Pr(G_1j) ≤ Pr(G_2j), j=1,...,m )
                  OR ( Pr(G_1) < Pr(G_2)  AND  Pr(G_1j) ≥ Pr(G_2j), j=1,...,m )
                    report the occurrence of the paradox to the user
END
```

Algorithm 1: Search for occurrences of Simpson's paradox.

5 Conclusion

We cannot overemphasize that a rule surprisingness measure (or, more generally, a rule interestingness measure) is a bias, and so there is no universally best rule surprisingness measure across all application domains. Each researcher or practitioner must adapt/invent a rule surprisingness measure to his/her particular target problem.

Hence, as mentioned in the introduction, in order to render the contribution of the paper generic, the main goal of this paper was not to introduce yet another rule surprisingness measure. Rather, this paper had the twofold goal of: (1) showing that it is possible to define *objective* (rather than subjective) measures of discovered rule *surprisingness*, unlike what we might infer from the literature; (2) proposing new ideas for defining objective rule surprisingness measures, which will hopefully be useful for other data mining researchers.

More precisely, the main new ideas proposed in this paper were: (a) a method for measuring the surprisingness of discovered small disjuncts, essentially based on how much the prediction of the *small disjunct* differs from the predictions of its *minimum generalizations*; (b) an information-theoretic, *fine-grain* method for measuring the surprisingness of a discovered rule by considering the surprisingness of *individual attributes* in the rule antecedent, rather than the rule antecedent as a whole (the conventional coarse-grain approach); (c) a method for discovering surprising knowledge via the *explicit detection* of occurrences of *Simpson's paradox*, in the form of a high-level algorithm specifically designed for this task.

One limitation of this paper is that our discussion has not taken into account the interaction between rules in the discovered rule set. In principle, however, the issue of rule interaction is somewhat orthogonal to the issue of individual rule surprisingness, in the sense that the measure of rule interaction (typically a measure of rule overlapping) is often independent of the measure of individual rule surprisingness (or, more generally, interestingness). Hence, it should be possible to use the rule surprisingness measures proposed in this paper together with rule interaction measures. The reader interested in rule selection procedures taking into account rule interaction is referred to [Gebhardt 91], [Major & Mangano 95].

A natural direction for further research is to implement the new ideas for defining objective rule surprisingness measures proposed by this paper in some data mining algorithm(s), in order to evaluate their effectiveness in some real-world data sets.

References

[Cover & Thomas 91] T.M. Cover and J.A. Thomas. *Elements of Information Theory.* John Wiley & Sons, 1991.

[Danyluk & Provost 93] A.P. Danyluk & F.J. Provost. Small disjuncts in action: learning to diagnose errors in the local loop of the telephone network. *Proc. 10th Int. Conf. Machine Learning*, 81-88, 1993.

[Fayyad et al. 96] U.M. Fayyad, G. Piatetsky-Shapiro and P. Smyth. From data mining to knowledge discovery: an overview. In: U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy. *Advances in Knowledge Discovery & Data Mining*, 1-34. AAAI/MIT, 1996.

[Gebhardt 91] F. Gebhardt. Choosing among competing generalizations. *Knowledge Acquisit.*, 3, 1991, 361-380.

[Glymour et al. 97]. C. Glymour, D. Madigan, D. Pregibon and P. Smyth. Statistical themes and lessons for data mining. *Data Mining and Knowledge Discovery* 1(1), 11-28. 1997.

[Holte et al. 89] R.C. Holte, L.E. Acker and B.W. Porter. Concept learning and the problem of small disjuncts. *Proc. Int. Joint Conf. AI (IJCAI-89),* 813-818.

[Kamber & Shinghal 96] M. Kamber & R. Shinghal. Evaluating the interestingness of characteristic rules. *Proc. 2nd Int. Conf. Knowledge Discovery & Data Mining,* 263-266. AAAI, 1996.

[Liu et al. 97] B. Liu, W. Hsu and S. Chen. Using general impressions to analyze discovered classification rules. *Proc. 3rd Int. Conf. Knowl. Disc. & Data Mining*, 31-36. AAAI, 1997.

[Major & Mangano 95]. J.A. Major and J.J. Mangano. Selecting among rules induced from a hurricane database. *J. Intelligent Information Systems* 4(1), Jan./95, 39-52.

[Piatetsky-Shapiro 91] G. Piatetsky-Shapiro. Discovery, analysis and presentation of strong rules. In: G. Piatetsky-Shapiro and W.J. Frawley. *Knowledge Discovery in Databases*, 229-248. AAAI, 1991.

[Provost & Aronis 96] F.J. Provost and J.M. Aronis. Scaling up inductive learning with massive parallelism. *Machine Learning* 23(1), Apr./96, 33-46.

[Simpson 51] E. H. Simpson. The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society, series B, 13*, 1951, 238-241.

[Ting 94] K.M. Ting. The problem of small disjuncts: its remedy in decision trees. *Proc. 10th Canadian Conf. Artificial Intelligence*, 91-97. 1994.

[Wagner 82] Simpson's paradox in real life. *The Amer. Statist.*, 36(1), Feb./82, 46-48.