Decision trees with optimal joint partitioning

Djamel A. Zighed[1], Gilbert Ritschard[2], Walid Erray[1], and Vasile-Marian Scuturici[1]

[1] ERIC Laboratory, University of Lyon 2, C.P.11 F-69676 Bron Cedex, France
zighed@univ-lyon2.fr

[2] Dept of Econometrics, University of Geneva, CH-1211 Geneva 4, Switzerland
ritschard@themes.unige.ch

Abstract. Decision tree methods generally suppose that the number of categories of the attribute to be predicted is fixed. Breiman et al., with their Twoing criterion in CART, considered gathering the categories of the predicted attribute into two supermodalities. In this paper, we propose an extension of this method. We try to merge the categories in an optimal unspecified number of supermodalities. Our method, called *Arbogodaï*, allows during tree growing for grouping categories of the target variable as well as categories of the predictive attributes. It handles both categorical and quantitative attributes. At the end, the user can chose to generate either a set of single rules or a set of multi-conclusion rules that provide interval like predictions.

1 Introduction

Induction trees are among the most popular supervised methods proposed in the literature. They are appreciated for the simplicity and the high efficacy of the algorithms, for their ease of use and for the easily interpretable results they provide. Hastie et al. [15], p. 313, designate them as the learning tool that comes closest to the requirements of an "off-the-shelf" method.

Many induction tree methods have been proposed so far in the literature. Some like ID3 [20], C4.5 [21] and CHAID [17, 18] build n-ary trees, others like CART [6] produce binary trees or, like SIPINA [27, 28], latticed graphs that generalize trees by allowing the merging of nodes.

All these methods were originally intended for categorical attributes and require therefore that quantitative variables be discretized. This discretization can be done at once before growing the tree. Most of the tree growing methods, nevertheless, handle quantitative variables in an automatic manner by dynamically choosing the optimal discretization thresholds at each node [26, **?**, **?**][]. Some methods also attempt to reduce the number of categories of nominal attributes by partitioning them into a smaller number of classes. CART, for example, merges the categories into two new super-modalities at each new split. This has the advantage of avoiding to uselessly increase the number of nodes. Indeed, the higher the number of nodes, the greater are the chances that some of them will have too few cases to get reliable estimates of the response classes probabilities.

There are two main ways for partitioning the values of the nominal predictive attributes.

1. The first is for instance a characteristic feature of CHAID [17]. At each node, the local discriminating power of each categorical attribute is tested using all possible partitions of its values. Partitions in two or more groups are explored. Thus, for each split, a predictor is selected simultaneously with its locally best partition.
2. The second strategy is used for instance by Breiman et al. [6] in their CART method. At each node, CART looks only for the best bi-partition of each predictor. It generates thus only binary trees.

With their Twoing criterion, the authors of CART propose however also a strategy that extends their principle to the response variable. When the response is multi-valued, using Twoing is equivalent to seek, for every predictor, simultaneously the best bi-partition of its values and the best bi-partition of the response values. The Twoing is the value of the Gini impurity for the best couple of bi-partitions and is used for selecting the split variable at each node.

In this paper, we extend the principle of a simultaneous search of a double bi-partition. We combine the CHAID and CART approaches. Like CHAID we look at each step for the best not necessarily binary partition of the attributes. Like CART with Twoing we explore also the partitioning of the values of the target variable. Unlike CART, we do not, however, restrict ourself to bi-partitions. At each step we look for the simultaneous grouping of the predictor values and of the target variable values that optimizes the chosen criterion. This gives rise to a new induction tree method that we call *Arbogodaï*. This kind of tree is characterized by a number of value classes of the target variable that varies from one node to the other. It is dynamically determined at each new split. When the majority class in a leaf contains several response values, the corresponding prediction rule becomes a multiple conclusion rule. For instance, we can get a rule like "a female customer aged between 30 and 40 with a monthly income ranging from 4000 to 5000 euros will chose a red or blue car". Indeed, we can easily compute which of the two colors is more frequent in the leaf. Hence, we can also derive classical simple rules. With *Arbogodaï*, the user has the possibility to chose the kind of rule that best suits her/his needs.

The paper is organized as follows. Section 2 sets the framework and recalls the goal and principle of induced decision trees. In Section 3, we motivate the simultaneous n-ary partitioning of the target and predictor values. Section 4 describes the simultaneous row-column merging heuristic. The *Arbogodaï* tree growing process that seeks at each step the optimal joint merging of target and predictor values is described in Section 5. Section 6 discusses the multiple conclusion nature of the generated rules and how to measure their prediction error rates. It reports also some experiments with a set of benchmark datasets. In Section 7, we propose an in depth study of the simultaneous merging heuristic. Further developments are briefly discussed in the concluding section.

2 Principle of induction trees and notations

Let Ω be the population concerned by the learning problem. The profile of any member ω of Ω is described by p variables, $X_1, \ldots, X_p$, called either exogenous variables, predictive attributes or predictors. These variables can be qualitative or quantitative. The set of values taken by X_j is denoted by $\mathcal{X}_j$. Each variable X_j, $j = 1, \ldots, p$ can be seen as a mapping $X_j : \Omega \rightarrow \mathcal{X}_j$, where $\mathcal{X}_j$, the domain of the values of X_j, is any not necessarily finite set. We consider also a target attribute C, sometimes called response, endogenous or dependent variable, and designate by $\mathcal{C}$ the set of response values. Like the X_j's, C can be qualitative or quantitative. Since the attributes X_j and the target variable C take only a finite number of different values in a given dataset, the sets $\mathcal{X}_j$ and $\mathcal{C}$ are finite. We denote by m_j the number of different values taken by the attribute X_j and by ℓ the number of different response values c_i. Thus, $\mathcal{C} = \{c_1, \ldots, c_\ell\}$.

The goal of induction trees is then to generate a model $\phi(X_1, \ldots, X_p)$ in the form of a decision tree for predicting the value of C from the knowledge of the values taken by the predictive attributes. The tree ϕ is induced from a training sample $\Omega_L \subset \Omega$. The validation of the predictive model is done on a test sample $\Omega_T \subset \Omega$ distinct from the former, $\Omega_L \cap \Omega_T = \emptyset$.

The growing process of the tree is quite simple. As illustrated in Figure 1, the set Ω_L is iteratively split by means of, at each step, one of the predictive attributes $X_1, \ldots, X_p$.

The leaves of the tree obtained at each step t of the growing process define a partition S_t of Ω_L that becomes finer and finer with t. The root of the tree corresponds to the trivial partition $S_0 = \{\Omega_L\}$.

The goal is to get a partition with each leaf (class of the partition) as pure as possible, a pure leaf being one in which all the individuals have the same value for the predicted attribute. The leaf must indeed contain enough individuals to be reliable.

The tree given in Figure 1 partitions Ω_L in three subsets corresponding to the nodes s_2, s_3 and s_4. In leaf s_3 for example, we have the set of cases of Ω_L that take values $X_1 =$ male and $X_2 < 5000$. At step t, the partition S_t is derived from the previous one S_{t-1} by seeking the best leaf-attribute couple (s_k, X_j), i.e.

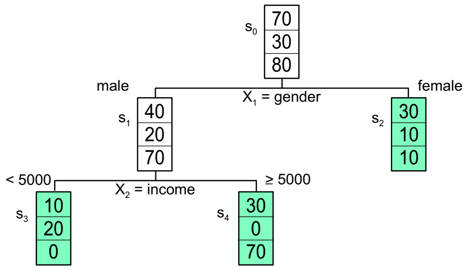

Fig. 1. An induced tree

Table 1. Contingency table defined by X_j at a node s

	x_{j1}	$\dots$	x_{jk}	$\dots$	x_{jm_j}	Total
c_1	n_{11}	$\dots$	n_{1k}	$\dots$	n_{1m_j}	$n_{1.}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$		$\vdots$	$\vdots$
c_i	n_{i1}	$\dots$	n_{ik}	$\dots$	n_{im_j}	$n_{i.}$
$\vdots$	$\vdots$		$\vdots$	$\ddots$	$\vdots$	$\vdots$
c_ℓ	$n_{\ell 1}$	$\dots$	$n_{\ell k}$	$\dots$	$n_{\ell m_j}$	$n_{\ell .}$
Total	$n_{.1}$	$\dots$	$n_{.k}$	$\dots$	$n_{.m_j}$	n

that for which the splitting of $s_k \in S_{t-1}$ according to the values of X_j maximizes the gain of information on the target variable between S_{t-1} and S_t. Formally, letting $G(S_{t-1}, s_k, X_j)$ be the gain of information when s_k is split with attribute X_j, we seek at step t the leaf-attribute couple (s_v, X_u) such that

$$G(S_{t-1}, s_v, X_u) = \max_{k;j} G(S_{t-1}, s_k, X_j)$$

The gain of information is usually measured as the reduction in uncertainty for the target variable or as the increase in the strength of association between the partition and the target variable. The growing process stops when the criterion can no longer be improved, i.e. when $G(S_{t-1}, s_v, X_u) \leq 0$, or when some other stopping criterion is reached.

Let n be the grand total of cases in node s, n_{ik} the number of cases with value c_i for the target variable in the class (leaf) s_k of the partition S of the cases in s, $n_{.k}$ the total number of cases in leaf s_k, $n_{i.}$ the total number of cases with value c_i in s. The corresponding observed frequencies are denoted respectively by f_{ik}, $f_{.k}$ and $f_{i.}$, and $f_{i|k} = n_{ik}/n_{.k}$ stands for the conditional frequency of value c_i in the leaf s_k. To be rigorous, the n's and f's should be indexed by the node label s. We omit it to avoid cumbersome notations.

At any node s of a tree, an attribute X_j defines a partition of the cases in s. This partition is described by the columns of the $\ell \times m_j$ contingency table (Table 1) that cross-tabulates the target variable (rows) with X_j (columns).

The criteria used to measure the gain of information brought by a split defined by X_j are computed from this table. For instance, some methods try to maximize the reduction in uncertainty as measured by entropies. In this case, the uncertainty after the split is defined as the weighted mean of the uncertainty of the columns of the contingency Table 1

$$I(S) = \sum_{k=1}^{m_j} \frac{n_{.k}}{n} h(f_{1|k}; \dots; f_{i|k}; \dots; f_{\ell|k}) \tag{1}$$

where $h()$ is, for example, the Shannon entropy, $-\sum_{i=1}^{\ell} f_{i|k} \log_2 f_{i|k}$, or the quadratic entropy, also known as the Gini diversity index, $\sum_{i=1}^{\ell} f_{i|k}(1 - f_{i|k})$. Alternatively, some methods like CHAID, optimize the strength or the statistical significance of the association between the resulting partition (columns of Table 1) and the target variable (rows of Table 1).

Let us recall that CHAID tries, at each step, to merge the columns of crosstables like Table 1 to find the best grouping of values for each candidate attribute, i.e. the grouping that optimizes the criterion. CHAID makes no change, however, on the values of the target variable. *Arbogodaï*, like the Twoing approach in CART, considers merging both columns and rows. Unlike the Twoing rule that looks for the best solution among 2 × 2 tables only, we seek however the best cross-partition without constraints on the number of rows and columns. Section 4 discusses this joint row-column partitioning issue. Before turning to it, we motivate the approach in Section 3.

3 Motivations for a joint n-ary partitioning

Consider the contingency Table 2. The best bi-partition of its columns is $S_{\text{bin}} = \{\{a, b\}, \{d, e\}\}$, whether we use the Gini, the Twoing, the significance of Pearson's Chi-squares or an association measure like the t of Tschuprow. Now, the best 3 way partition is $S_{\text{3way}} = \{\{a\}, \{b, d\}, \{e\}\}$ with any of the criteria except Twoing which is not applicable. Clearly S_{3way} cannot be obtained by splitting the classes of S_{bin}. This proves that multiple binary partitions are not equivalent to n-ary partitions and can sometime miss optimal solutions.

Table 2. A n-ary solution different from that of successive binary splits

	a	b	d	e	Total
c_1	200	100	10	1	311
c_2	10	150	150	10	320
c_3	1	10	100	200	311
Total	211	260	260	211	942

The merging of response values is different in nature from that of the values of a predictive attribute. Indeed, the partition of the response values does not translate into a split of the node. Considering such mergings in the optimization process merits therefore some further justification. This is given by simply extending the argument of Breiman et al. ([6], p. 105) who argue that searching for superclasses (the groups of the partitions of the response values) provides strategic information on the similarities of responses. When two or more responses, red car and blue car for example, are almost equally frequent it may be a better strategy to predict that the customer will buy a red or a blue car than explicitly a red one. Simultaneously, it may be useful to know that yellow and pink colors are much less probable than all other non red and non blue proposed colors. There is thus no reason to limit the argument to two superclasses only. Multi-supermodalities provide a more refined strategic information.

Now, the grouping on one variable (say the row variable) may obviously affect the optimal grouping on the other attribute (the column variable.) For example, grouping first rows c_1 and c_2 in Table 2 we get the reduced table

$$\begin{bmatrix} 210 & 250 & 160 & 1 \\ 1 & 10 & 100 & 200 \end{bmatrix} .$$

The most similar columns in this new table are obviously the first two. Hence, the best columns partitioning would be $\{\{a, b\}, \{d\}, \{e\}\}$ or $\{\{a, b, d\}, \{e\}\}$, but clearly not the one found without grouping the rows. Due to this relationship between the partitions of the rows and the columns, it is then essential to determine them simultaneously.

4 Simultaneous row and column partitioning

In this section, we introduce the method adopted for determining the best simultaneous partition of the rows and the columns. First, we specify the objectives and briefly review related works. We then define the formal setting and describe our merging heuristic.

4.1 Objectives

While the univariate optimal grouping of values has been abundantly studied since the pioneering work of Walter Fisher [8], the literature about the simultaneous grouping of rows and columns of a table is less rich. We can mention the related work by Fisher [9, 10] about the optimal grouping of the unknowns and equations of predictive economic models. The simultaneous partitioning of the cases (rows) and the variables (columns) in a data matrix has been studied among others by Anderberg [1], Bock [5] and Govaert [13]. In [12, 13], Govaert investigates the special case of binary tables. In the framework of contingency tables that we are interested in, the optimal partitioning problem has been studied from different points of view. Benzecri [3] is interested in the partition in a fixed number of groups that maximizes the Pearson Chi-square. A solution to this problem is given in [7] in the form of an iterative heuristic that clusters alternatively the rows and the columns. Gilula and Krieger [11] study how the Pearson Chi-square behaves when the table is reduced by aggregation. Hirotsu [16] and Greenacre [14] are interested in finding the most homogeneous tables. As already mentioned, Breiman et al. [6] have considered with their Twoing approach the joint dichotomization of two variables.

Our objective is to find both the number of groups and the joint partition of rows and columns of a contingency table that maximizes the row-column association. None of the works cited gives a satisfactory solution to this problem. Some, like those done after Benzecri [3] or those by Breiman et al. [6] assume the number of groups fixed a priori. The others either do not consider the case of contingency tables or consider criteria, homogeneity for example, that are hardly transposable in our setting.

Clearly, the exhaustive scanning of all combinations of partitions of each of the variable is not practicable for large tables. We show in Section 7.1 that the search of the optimal solution becomes untractable when the number of values of the predictive and/or target variable exceeds 5 or 6. We consider therefore a heuristic, first introduced in [24], that successively looks for the optimal grouping of two row or column categories. We recall the principle of the algorithm hereafter and will propose an in depth study of its behavior and performance in Section 7.

4.2 Formal framework

Let X be a predictive attribute. From here on we shall drop the subscripts j when there is no ambiguity. Cross-tabulating variable C with X generates a contingency table $\mathbf{T}_{\ell\times m}$ with ℓ rows and m columns.

Let $\theta_{CX} = \theta(\mathbf{T}_{\ell\times m})$ denote a generic association criterion for table $\mathbf{T}_{\ell\times m}$. This criterion θ_{CX} may thus be a Chi-square based association measure like Cramer's v or Tschuprow's t, an asymmetrical PRE measure like Goodman-Kruskal's τ_{CX} or Theil's uncertainty coefficient u_{CX}, or, when both variables are ordinal, an ordinal association index like Kendall's τ_b or Somers' d_{CX}.

Let P_c be a partition of the values of the row variable C, and P_x a partition of the states of X. Each couple (P_c, P_x) defines then a contingency table $\mathbf{T}(P_c, P_x)$. The optimization problem considered is then the maximization of the association θ_{CX} among the set of couples (P_c, P_x) of partitions :

$$\max_{P_c, P_x} \theta\big(\mathbf{T}(P_c, P_x)\big) \ . \tag{2}$$

For ordinal variables, hence for interval or ratio variables, only partitions obtained by merging adjacent categories make sense. We consider then the restricted program

$$\begin{cases} \max\limits_{P_c, P_x} \ \theta\big(\mathbf{T}(P_c, P_x)\big) \\ \text{u.c.} \quad P_c \in \mathcal{A}_c \text{ and } P_x \in \mathcal{A}_y \end{cases} \tag{3}$$

where $\mathcal{A}_c$ and $\mathcal{A}_x$ stand for the sets of partitions obtained by grouping adjacent values of C and X. Letting $\mathcal{P}_c$ and $\mathcal{P}_x$ be the unrestricted sets of partitions, we have for $\ell, m > 2$, $\mathcal{A}_c \subset \mathcal{P}_c$ and $\mathcal{A}_x \subset \mathcal{P}_x$. Finally, note that ordinal association measures may take negative values. Then, for maximizing the strength of the association, the objective function $\theta\big(\mathbf{T}(P_c, P_x)\big)$ should be the absolute value of the ordinal association measure.

4.3 The heuristic

The heuristic is an iterative greedy process that successively merges the two rows or columns that most improve the association criteria $\theta(\mathbf{T})$.

Such a heuristic may indeed not end up with the optimal solution, but perhaps only with a quasi-optimal solution. See Section 7.3 for empirical insights on this sub-optimality. Formally, the configuration (P_c^k, P_x^k) obtained at step k

is the solution of

$$\begin{cases} \max\limits_{P_c,P_x} \; \theta\big(\mathbf{T}(P_c,P_x)\big) \\ \text{u.c. } P_c = P_c^{(k-1)} \text{ and } P_x \in \mathcal{P}_x^{(k-1)} \\ \qquad \text{or} \\ \qquad P_c \in \mathcal{P}_c^{(k-1)} \text{ and } P_x = P_x^{(k-1)} \end{cases} , \tag{4}$$

where $\mathcal{P}_c^{(k-1)}$ stands for the set of partitions on C resulting from the grouping of two classes of the partition $P_c^{(k-1)}$.

For ordinal variables, $\mathcal{P}_c^{(k-1)}$ and $\mathcal{P}_x^{(k-1)}$ should be replaced by the sets $\mathcal{A}_c^{(k-1)}$ and $\mathcal{A}_x^{(k-1)}$ of partitions resulting from the aggregation of two adjacent elements.

Let $\mathbf{T}^0 = \mathbf{T}_{\ell\times m}$ be the table associated with the finest partition of the categories of C and X. Starting with $\mathbf{T}^0$, the algorithm successively determines the tables $\mathbf{T}^k, k = 1, 2, \ldots$ corresponding to the partitions solution of (4). The process continues while $\theta(\mathbf{T}^k) \geq \theta(\mathbf{T}^{(k-1)})$ and is stopped when the best grouping of two categories leads to a reduction of the criteria.

The *quasi-optimal grouping* is the couple (P_c^k, P_x^k) solution of (4) at the step k where

$$\theta\big(\mathbf{T}^{(k+1)}\big) - \theta\big(\mathbf{T}^k\big) < 0 \quad \text{and} \quad \theta\big(\mathbf{T}^k\big) - \theta\big(\mathbf{T}^{(k-1)}\big) \geq 0 \ .$$

By convention, we set the value of the association criteria $\theta(\mathbf{T})$ to zero for any table with a single row or column. The algorithm then ends up with such a single value table if and only if all rows (columns) are equivalently distributed.

5 Arbogodaï : a new decision tree approach

We now introduce the new *Arbogodaï* tree growing method. We first explain the principle of the *Arbogodaï* algorithm and, then, describe how it works on an example.

5.1 Principle of the algorithm

Arbogodaï follows the general principle of tree growing presented in Section 2. Its specificity is an additional preparatory step before testing the attributes at a node. This step consists in optimally reducing the size of the table that crosses the target variable with the tested attribute. The splitting criterion is then computed using the found partitions of both the attribute and the target variable values. The splitting of the selected node is done according to the found classes of values of the selected predictive attribute.

This additional step plays a role similar to discretization. The merging of values can indeed be assimilated to some sort of discretization that works also on nominal variables. Remember, however, that the merging is done here simultaneously at each step on the target and the predictive attribute.

The reduction of the table is that for which the row-column association θ is maximized. Indeed we use the heuristic of Section 4.3 and measure the association θ with the t of Tschuprow: $t = \{n^{-1}[(\ell-1)(m-1)]^{-1/2}\,\chi^2\}^{1/2}$, where $\chi^2 = \sum_{i=1}^{\ell}\sum_{k=1}^{m}(nn_{i.}n_{.k})^{-1}(nn_{ik} - n_{i.}n_{.k})^2$ is the Pearson Chi-square statistic. Unlike some other association measures, the t of Tschuprow always increases with the merging of equivalently distributed rows or columns (see [25] and Section 7.2.)

The splitting criterion is the reduction in uncertainty (gain in purity) achieved with the columns of the reduced table as compared to its margin. The uncertainty after the split is computed for every X_j by applying formula (1) on the optimal reduced table for X_j at the considered node s.

More specifically, we use Laplace estimates of the column distributions after the split, i.e. the proportion of cases that take the ith value of C in column k is estimated by:

$$f^{*(\lambda)}_{i|k} = \frac{n^*_{ik} + \lambda}{n^*_{.k} + \ell^*\lambda} \tag{5}$$

where the * denotes values for the reduced table.

Using the quadratic (Gini) entropy, the gain in uncertainty considered by *Arbogodaï* then reads,

$$\begin{aligned} h(C^*) - h(C^*|X_j^*) &= 1 - \sum_i f^{*2}_{i.} - \sum_k f^*_{.k}\big(1 - \sum_i (f^{*(\lambda)}_{i|k})^2\big) \\ &= \sum_i \big[\big(\sum_k f^*_{.k}(f^{*(\lambda)}_{i|k})^2\big) - f^{*2}_{i.}\big] \end{aligned} \tag{6}$$

The use of Laplace estimates penalizes the gain of uncertainty obtained at nodes with small counts. With very small counts, i.e. when λ represents a significant proportion of the count, a split may even deteriorate the uncertainty criterion (see [28] p.76.).

5.2 Example

We now describe the *Arbogodaï* algorithm through an example. We consider the *Flags* dataset from the UCI repository [4]. The response variable C takes 6 nominal values $\mathcal{C} = \{c_1, c_2, c_3, c_3, c_4, c_5, c_6\}$ and there are 29 mixed categorical and quantitative predictive attributes $X_1, \ldots, X_{29}$. The dataset contains 194 cases. Figure 2 shows an extract of the two first levels of the *Arbogodaï* tree for these data.

Step 1. At the root of the tree, we have the distribution of all 194 cases among the 6 values of the response C. The 29 predictive attributes are successively tested. For every attribute, we first determine the optimal reduced crosstable with the target variable. We then select the attribute for which the gain in uncertainty computed on the reduced table is maximal. The winner is X_7, which takes 8 values: $\mathcal{X}_7 = \{a, b, c, d, e, f, g, h\}$. The two simultaneous groupings found

Table 3. Step 1 optimal crosstable and Laplace estimates of column distributions

C / X_7	$\{c,d,e,h\}$	$\{a,b,g\}$	$\{f\}$
$\{c_1\}$	33	6	0
$\{c_2,c_4,c_5,c_6\}$	2	100	1
$\{c_3\}$	17	9	26
Total	52	115	27

$f^{*(\lambda)}_{i\|k}$	$\{c,d,e,h\}$	$\{a,b,g\}$	$\{f\}$	$f^*_{i.}$
$\{c_1\}$	0.618	0.059	0.033	0.201
$\{c_2,c_4,c_5,c_6\}$	0.055	0.855	0.067	0.530
$\{c_3\}$	0.327	0.084	0.900	0.268
$f^*_{.k}$	0.268	0.592	0.139	1

by the heuristic of Section 4.3 are $\mathcal{X}^*_7 = \{\{c,d,e,h\};\{a,b,g\};\{f\}\}$ and $\mathcal{C}^* = \{\{c_1\};\{c_2,c_4,c_5,c_6\};\{c_3\}\}$. The corresponding crosstable is shown in Table 3 together with the table of the derived conditional frequencies $f^{*(\lambda)}_{i|k}$. The latter have been computed by setting $\lambda = 1$. The marginal uncertainty is $h(C^*) = 1 - .201^2 - .530^2 - .268^2 = .605$ and the uncertainty after the split, which is the weighted average of the uncertainty of each column, is $h(C^*|X^*_3) = .314$. The gained information is thus .291. This is the maximal value achievable with any of the 29 attributes.

Step 2. The process is repeated on every terminal node of the previously obtained tree. Notice that we try to merge the original set of values $\mathcal{X}$ and $\mathcal{C}$ and not the set of previously merged classes. In our example, the next best split occurs at the middle node ($X_7 \in \{a,b,g\}$). The attribute selected for splitting this node is X_3. The 6 values of the target C were merged to form 4 target classes. However, no merging of the attribute values could improve the association between X_3 and the target C. The node is therefore split in 4 new classes corresponding to the 4 values of X_3. This leads to the tree with 6 leaves shown in Figure 2.

Following steps. In our example, the tree growing process is stopped after step 2. Without explicit stopping rules, the growing continues until the criterion can no

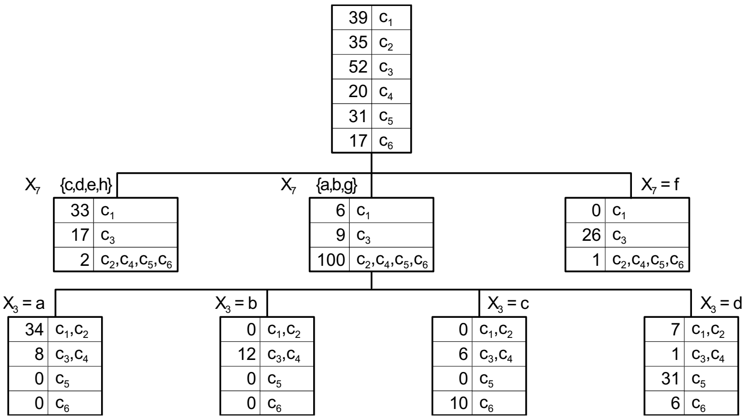

Fig. 2. Example of an Arbogodaï tree

Table 4. Crosstable for splitting the two leftmost leaves with X_{29}

	$\{a, b, d, e\}$	$\{f\}$
$\{c_1, c_2, c_4\}$	38	1
c_3	0	3
other	0	0

	$\{a, d, e\}$	$\{b, f\}$
c_3	2	2
c_4	8	0
other	0	0

longer be improved. At step 3, *Arbogodaï* would scan the 6 leaves of the previously grown tree.

Two further remarks should be made: (i) At a same level, nodes that do not result from a same parent may have different partitions of the set $\mathcal{C}$ of response values. (ii) When the same attribute is used as the splitting variable at more than one node, its values are not necessarily partitioned the same way for each split. For example, growing the tree of Figure 2 one level further leads to split each of the two left most leaves of level 2 with the same attribute X_{29}. The corresponding crosstables are given in Table 4. It can be seen that the values of C are once partitioned as $\{\{c_1, c_2, c_4\}, \{c_3\}, \{c_5, c_6\}\}$ and once as $\{\{c_3\}, \{c_4\}, \{c_1, c_2, c_5, c_6\}\}$. Likewise, attribute X_{29} is used once with the partition $\{\{a, b, d, e\}, \{f\}\}$ and once with $\{\{a, e, d\}, \{b, f\}\}$.

6 The induced rules and their accuracy

Arbogodaï can generate two types of classification rules: (i) Classical rules by disregarding the merged classes of response values in the final leaves. (ii) Multiple conclusion rules for leaves with merged response values. This Section specifies the nature of these rules, defines error rates adapted for them and presents experimentation results.

We give hereafter the multiple conclusion rules generated by the tree of Figure 2. Each path joining the root to a leaf defines the premise of a rule. The conclusion is drawn from the distribution in the leaf, i.e., for cases falling in the leaf, the rule predicts the modal value in the leaf, or modal class of values when some are merged. The tree has 6 leaves giving rise to the 6 following rules (the value between parentheses is the confidence of the rule for the training data). Clearly, when the majority class contains only one value we get classical rules. Here, only R_3 and R_4 provide multiple conclusions in the form of "either c_1 or c_2."

R_1 : **If** $X_7 \in \{c, d, e, h\}$ **then** $C = c_1$ (33/52)
R_2 : **If** $X_7 = f$ **then** $C = c_3$ (26/27)
R_3 : **If** $X_7 \in \{a, b, g\}$ **and** $X_3 = a$ **then** $C \in \{c_1, c_2\}$ (34/42)
R_4 : **If** $X_7 \in \{a, b, g\}$ **and** $X_3 = b$ **then** $C \in \{c_3, c_4\}$ (12/12)
R_5 : **If** $X_7 \in \{a, b, g\}$ **and** $X_3 = c$ **then** $C = c_6$ (10/16)
R_6 : **If** $X_7 \in \{a, b, g\}$ **and** $X_3 = d$ **then** $C = c_5$ (31/45)

6.1 Error rates

The accuracy of the learned rules is usually assessed with the misclassification error rate or equivalently with the classification success rate. For classical rules, the misclassification rate reads $err = 1 - \sum_{s \in S} f_s f_{\max|s}$ where f_s is the proportion of cases in leaf s and $f_{\max|s} = \max_i f_{i|s}$ is the frequency of the modal response in leaf s.

For multiple conclusion rules, we can indeed simply compute the classical error by ignoring the multiple conclusion and focusing on the modal value in each leaf. For taking the multiple conclusion into account, we define however tow additional kinds of error rates:

$$\begin{array}{ll} \text{superclass error} & serr = 1 - \sum_{s \in S} f_s f^*_{\max|s} \\ \text{weighted superclass error} & werr = 1 - \sum_{s \in S} f_s f^*_{\max|s} \big(\sum_{i \in \mathcal{C}_{\max,s}} \hat{p}_{i|\max,s} \, f_{i|\max,s} \big) \end{array}$$

where $\mathcal{C}_{\max,s}$ is the set of response values in the modal superclass at leaf s, $f_{i|\max,s}$ the frequency of response c_i in that superclass and $\hat{p}_{i|\max,s}$ an estimation of the probability of c_i in the superclass. We get resubstitution error rates when the frequencies are those of the learning sample and generalization error rates when the frequencies are obtained from validation data. The estimations $\hat{p}_{i|\max,s}$'s are in any case computed on the training data. To get more reliable estimates, we use the marginal distribution at the parent node. This can be justified as follows. Values are merged when their distributions among the values of the split attribute are similar. Hence, their distributions inside the superclass are similar too and therefore similar to the marginal distribution.

The *superclass error*, $serr$, is computed as for classical rules but with the superclass frequencies $f^*_{i|s}$ instead of the single response frequencies $f_{i|s}$. Doing so, we do not care indeed of classification error inside the modal superclasses. This may have sense independently for each rule. We cannot compare, however, the error rate of a rule that predicts for instance c_1 with that of a rule that predicts c_1 or c_2. Hence, the global superclass error does not make much sense.

The *weighted superclass error*, $werr$, takes the uncertainty inside the majority class into account. It assumes that each case falling in a leaf is randomly assigned to a value in the modal superclass. The supposed random assignment is done according to the learned distribution inside $\mathcal{C}_{\max,f}$. For instance for our example tree, a case $(X_7 = a, X_3 = a, C = c_2)$ is correctly classified in the modal superclass of leaf 3. In that leaf, the estimated proportion of cases taking $C = c_2$ in the superclass is 85%. Thus, we weight this correct classification down and count it as a .85 correct classification. In resubstitution, if we use $\hat{p}_{i|\max,s} = f_{i|\max,s}$, this is equivalent to weighting down the success rates with the Gini uncertainty of the distribution inside the superclass: $werr = \sum_s [1 - (1 - serr_s) \text{Gini}(C_{\max,s})]$, where $serr_s$ is the superclass error for rule s.

It is well known that the learning error rate suffers from an optimistic bias. It underestimates the generalization error rate. For validation, it is then common to compute the classification error rate on a separate dataset not used for learning. Alternatively, and perhaps more frequently, a cross-validation error rate is computed. A 10 folds cross-validation (10CV), for instance, consists in splitting the learning sample into 10 approximately equally sized parts. Dropping each time a different part we get 10 learning datasets from which 10 trees are induced. For each of them we compute the error rate on the dropped out data. The cross-validation error rate is the mean values of the 10 resulting error rates.

6.2 Experimentation

We have experimented our approach on 8 benchmark datasets. Table 5 gives the cross-validation success rates obtained for each dataset with *Arbogodaï* and, for the sake of comparison, with CART and CHAID. For *Arbogodaï*, we give the rate derived from both the classical and the weighted superclass error. *Arbogodaï* ranks first for 5 of the 8 datasets whatever error is considered. Unsurprisingly, its superiority is mostly significant when the number of values of the target variable is large.

Table 5. Cross-Validation classification success rates (in percents)

	CART		ChAID		Arbogodaï			
Dataset	$1-err$	stdev	$1-err$	stdev	$1-err$	stdev	$1-werr$	stdev
Iris (3 cl.)	95.11	0.08	94.81	0.08	98.35	0.11	95.50	0.08
Flags (6 cl.)	75.14	0.40	75.21	0.40	78.83	0.41	83.37	0.34
Breast (2 cl.)	97.54	0.17	97.19	0.15	98.17	0.13	98.08	0.17
Car (4 cl.)	83.47	0.32	93.62	0.23	86.75	0.32	87.81	0.31
Ionosphere (2 cl.)	92.10	0.19	89.68	0.20	89.34	0.3	93.36	0.25
Pima (2 cl.)	84.44	0.38	83.55	0.38	81.39	0.38	81.20	0.40
Wine (3 cl.)	97.71	0.19	97.99	0.19	98.09	0.07	95.21	0.20
Zoo (7 cl.)	87.57	0.22	85.99	0.26	88.61	0.12	94.04	0.16

7 Advanced study of the merging heuristic

The joint response and predictive attribute partitioning is done with the heuristic described in Section 4.3. We propose here an in depth study of this greedy algorithm that successively seeks the best merge of two row or column categories. First, we examine its complexity and compare it with the exhaustive scanning of all partitions. To acquire some knowledge about possible merging criteria, the effect of the merging of two categories on a large choice of association measures has been examined analytically in [25]. We recall here the main findings of this theoretical analysis. Then, we investigate the efficacy of the algorithm, by providing details of simulation results shortly presented in [23, 25]. Finally we discuss the generalization of the simultaneous merging process to more than two variables and evoke some alternative merging strategies.

7.1 Complexity of the simultaneous merging heuristic

This subsection compares the complexity of the heuristic to that of the exhaustive exploration of all possible couples (P_c, P_x) of row and column partitions.

For the exhaustive scanning, the number of cases to explore is given by $\#\mathcal{P}_c\#\mathcal{P}_x$, i.e. the number of row partitions times the number of column partitions.

Consider first the case of a nominal variable. The number $B(a) = \#\mathcal{P}$ of possible partitions of its a categories can be computed iteratively by means of

Table 6. Number of configurations explored

	nominal case		ordinal case	
$\ell = m$	exhaustive	heuristic	exhaustive	heuristic
2	4	4	4	4
3	25	15	16	11
4	225	39	64	22
5	2704	81	256	37
6	41209	146	1024	56
7	769129	239	4096	79
8	17139600	365	16384	106
9	447195609	529	65536	137
10	$1.345{\cdot}10^{10}$	736	262144	172
20	$2.675{\cdot}10^{27}$	6271	$2.749{\cdot}10^{11}$	466
50	$3.449{\cdot}10^{94}$	101676	$3.169{\cdot}10^{29}$	4852
100	$2.264{\cdot}10^{231}$	823351	$4.017{\cdot}10^{59}$	19702

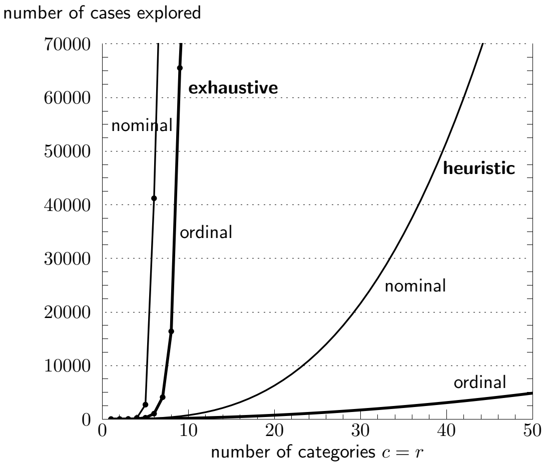

Fig. 3. Complexity versus size of the square table (for heuristic, values reported are upper bounds.)

Bell's [2] formula $B(a) = \sum_{k=0}^{a-1} \binom{a-1}{k} B(k)$ with $B(0) = 1$. Hence the number of cases to browse is $B(\ell)B(m)$ in the nominal case, which is for instance about $1.3 \cdot 10^{10}$ for $\ell = m = 10$.

For ordinal variables, hence for discretization issues, only adjacent groupings are considered. This reduces the number of cases to browse. The number $G(a) = \#\mathcal{A}$ of adjacent groupings of a values is $G(a) = \sum_{k=0}^{a-1} \binom{a-1}{k} = 2^{(a-1)}$. Thus, the number of cases to explore is $G(\ell)G(m)$ in the ordinal case. For $\ell = m = 10$ this is for example 262144.

For the heuristic, we can only give the maximal number of couples (P_c, P_x) we may have to scan. The actual number of couples explored depends indeed on when the stop criterion is reached. Assuming $\ell \leq m$, the upper bound is given, in the nominal case, by $\sum_{j=2}^{m} \left[\binom{j}{2} + \binom{\ell}{2}\right] + \sum_{i=2}^{\ell} \binom{i}{2} + 1$, that is

$$\frac{\ell(\ell^2 - 1) + m(m^2 - 1)}{6} + \frac{(m-1)\ell(\ell-1)}{2} + 1$$

In the ordinal case, it reads

$$\frac{(\ell + m - 1)(\ell + m - 2)}{2} + 1 \ .$$

For $m = \ell = 10$, these bounds are respectively 736 and 172.

Figure 3 and Table 6 show how the relative efficacy of the heuristic increases with the number of initial categories. The values reported concern square tables. It is worth mentioning that the seemingly exponential increase in the number of cases reported for the heuristic concerns the upper bound. Practically, the effective number of cases browsed will be much lower.

7.2 Summary of analytical results

Since the heuristic merges at each step two categories only, we studied in [25] the effect of such a grouping on a choice of association measures, namely Pearson's X^2 and the Likelihood Ratio G^2 Chi-square statistics, Tschuprow's t, Cramer's v, Goodman & Kruskal's τ, Theil's u, Goodman and Kruskal's γ, Kendall's τ_b and Somers' d. The latter three are ordinal measures and apply therefore only to ordinal variables. The formula of the indexes are recalled in the appendix.

Table 7 summaries the results established in [25]. For the ordinal measures that take their values in $[-1, 1]$, we report effects on the absolute value of the measure and consider, indeed, only the merging of two adjacent categories.

For our purpose, we expect an improve of the criteria when two equivalently distributed values are merged. We do not recommend therefore criteria that can remain unchanged after such a merge. Thus, Chi-squares statistics that cannot be increased with a merge, but also the Cramer v and the asymmetrical PRE measures (the Goodman-Kruskal τ and the Theil u) are not suited for our needs. Among the ordinal measures, only the Goodman-Kruskal γ and the Kendall τ_b satisfy the requested condition. Among the measures studied, the t of Tschuprow that fits the condition and can be used with both ordered and unordered values is our preferred choice.

Table 7. Effect of a merge of two categories on a choice of association measures

		merge of response values		merge of predictor values	
Criteria	asym.	not eq. distrib	equiv. distrib.	not eq. distrib.	equiv. distrib
Chi-square Statistics					
X^2, G^2		$-$	$=$	$-$	$=$
Nominal Association Measures					
Cramer's v		$+/-$	$+/=$	$+/-$	$+/=$
Tschuprow's t		$+/-$	$+$	$+/-$	$+$
G-K τ	*	$+/-$	$+$	$-$	$=$
Theil's u	*	$+/-$	$+$	$-$	$=$
Ordinal Association Measures					
G-K γ		$+/-$	$+$	$+/-$	$+$
Kendall's τ_b		$+/-$	$+$	$+/-$	$+$
Kendall's τ_c		$+/-$	$+/=$	$+/-$	$+/=$
Somers' d	*	$+/-$	$=$	$+/-$	$+$

7.3 Reliability of the joint merging heuristic

The purpose of this section is to assess the reliability of the results provided by the heuristic. A series of simulation studies have been run to investigate two aspects: (i) the proportion of global optima missed by the heuristic and (ii) how far the solution of the heuristic is from the global optimum.

Several association measures have been examined. We report outcomes for the t of Tschuprow, the τ of Goodman and Kruskal and the τ_b of Kendall. Among the measures considered (simulations have been run for all the measures listed in Table 7) the t of Tschuprow has been retained because it provides the worse scores for both the proportion of missed optima and the deviations from the global optima. The τ of Goodman and Kruskal has been selected as a representative of the asymmetrical PRE (proportion of reduction in error of prediction) measures. Likewise, the τ_b of Kendall has been selected to represent the ordinal measures.

The comparison between quasi and global optima is done for square tables of size 4, 5 and 6. Above 6, the global optimum can no longer be obtained in a reasonable time.

For the t of Tschuprow and the τ of Goodman and Kruskal, we report respectively in Tables 8 and 9 results for the nominal case as well as for the ordinal case. The τ_b of Kendall being an ordinal measure, Table 10 exhibits only figures for the ordinal case.

For each measure, size and variable type, 200 contingency tables have been randomly generated. Each table was obtained by distributing 10000 cases among its $\ell \times m$ cells with a random uniform process. This differs from the solution used to generate the results given in [25], which were obtained by distributing the cases with nested conditional uniform distributions: first a random percentage of the cases is attributed to the first row, then a random percentage of the remaining cases is affected to the second row and so on until the last row; the total of each row is then likewise distributed among the columns. The solution retained here

Table 8. Simulations: t of Tschuprow

Tschuprow		nominal			ordinal	
Size	4×4	5×5	6×6	4×4	5×5	6×6
Non zero deviations	39.5%	62.5%	74.5%	23.5%	36%	46.5%
maximum	0.073	0.074	0.077	0.077	0.063	0.108
mean	0.025	0.023	0.028	0.019	0.019	0.012
standard deviation	0.015	0.014	0.016	0.014	0.016	0.015
skewness	0.986	0.979	0.598	1.674	0.972	3.394
With zero deviations						
mean	0.010	0.015	0.021	0.005	0.007	0.006
standard deviation	0.016	0.016	0.018	0.011	0.013	0.012
skewness	1.677	1.062	0.615	3.168	2.211	4.457
Relative deviations						
maximum	0.168	0.198	0.221	0.179	0.194	0.307
mean	0.079	0.077	0.093	0.063	0.066	0.046
Mean initial association	0.260	0.240	0.226	0.263	0.244	0.228
Mean global optimum	0.340	0.316	0.303	0.301	0.275	0.250

Table 9. Simulations: τ of Goodman and Kruskal

G&K τ		nominal			ordinal	
Size	4×4	5×5	6×6	4×4	5×5	6×6
Non zero deviations	5%	6.5%	12%	6%	19%	32.5%
maximum	0.013	0.031	0.029	0.076	0.077	0.059
mean	0.007	0.010	0.008	0.025	0.016	0.013
standard deviation	0.004	0.009	0.009	0.021	0.016	0.012
skewness	-0.308	1.004	1.181	1.107	2.361	1.908
With zero deviations						
mean	0.0004	0.0007	0.0010	0.0015	0.003	0.004
standard deviation	0.0018	0.0033	0.004	0.008	0.009	0.009
skewness	5.323	6.471	5.137	6.685	5.040	3.255
Relative deviations						
maximum	0.142	0.296	0.318	0.420	0.518	0.401
mean	0.062	0.091	0.079	0.216	0.168	0.149
Mean initial association	0.074	0.060	0.048	0.073	0.060	0.050
Mean global optimum	0.148	0.128	0.113	0.118	0.098	0.084

generates indeed tables that are closer to the uniform distribution and should therefore exhibit lower association. As will be shown, low association are the less favorable situations for the heuristic. Thus, we can expect the results obtained with this random uniform generating process to provide some upper bounds for the deviations from the global optima.

Tables 8 to 10 exhibit, for each series of tables generated, the proportion of optima missed and characteristic values (maximum, mean value, standard

Table 10. Simulations: τ_b of Kendall

Kendall τ_b	ordinal		
Size	4×4	5×5	6×6
Non zero deviations	19%	24.5%	32%
maximum	0.596	0.597	0.542
mean	0.235	0.182	0.140
standard deviation	0.195	0.190	0.157
skewness	0.076	0.598	0.652
With zero deviations			
mean	0.045	0.045	0.045
standard deviation	0.125	0.123	0.111
skewness	2.775	2.849	2.445
Relative deviations			
maximum	1.954	1.970	1.982
mean	0.355	0.259	0.074
Mean initial association (abs value)	0.094	0.078	0.064
Mean global optimum (abs value)	0.256	0.236	0.215

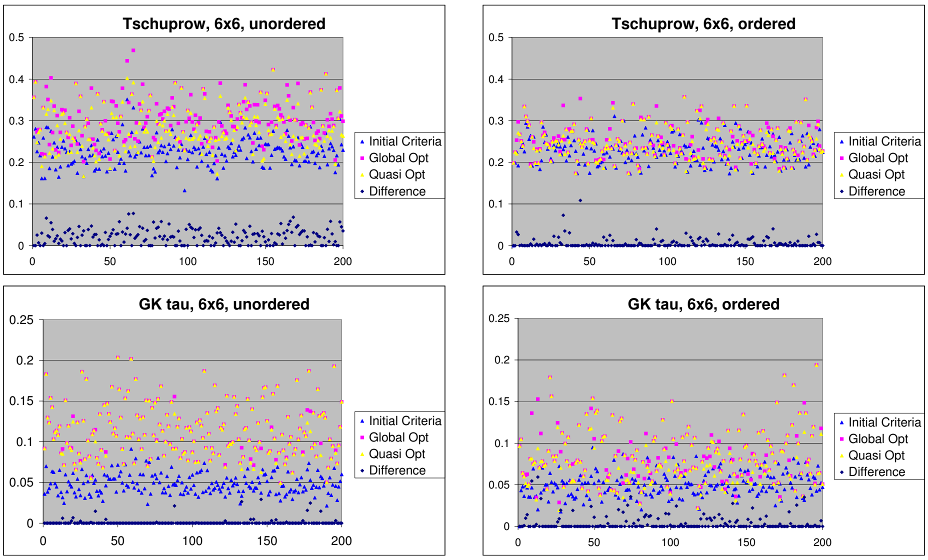

Fig. 4. Initial, quasi and global otpima

deviation, skewness) of the distribution of the deviations between global and quasi optima. Relative deviations, of which the maximum and the mean value are reported, are the ratios between deviations and global optima. The last two rows give respectively the average of the initial values of the criterion and the mean value of the global optima. In Table 10, these two last figures are means of absolute values since the τ_b's may be negative.

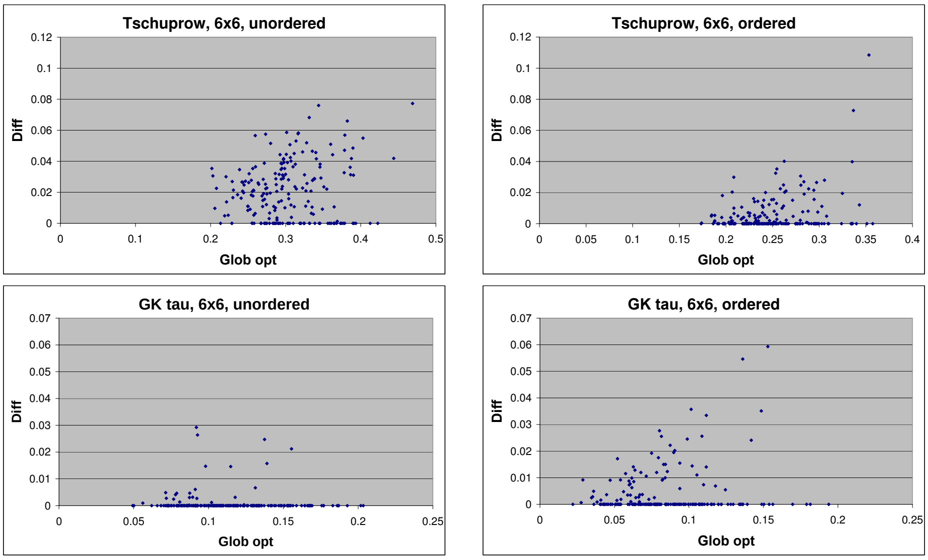

Fig. 5. Deviations versus global otpima

Additional insight for the Tschuprow's t and Goodman and Kruskal's τ cases is provided by Figures 4 and 5. Figure 4 shows plots of the 200 initial values, quasi optima and global optima for 6×6 cases. Figure 5 plots the 200 deviations against the global optima.

Looking at Tables 8 to 10, we see that the proportion of optima missed by the heuristic is relatively important and tends to increase with the size of the table. The proportion is somewhat lower for PRE measures (the τ of Goodman and Kruskal). This is probably due to the fact that PRE measures cannot be improved by merging values of the predictor (see table 7), which means that the groupings are in this case almost exclusively made on one (the target) variable. Curiously however, the percentages of missed optima are, for PRE measures, larger in the ordinal case than in the nominal one.

This high percentage of missed optima is luckily balanced by the small deviation between the quasi and global optima. The mean value of the non zero deviations is roughly less than half the difference between the initial value of the criterion and the global optimum. In the case of stronger initial associations than those generated here with a uniform random distribution, this ratio becomes largely more favorable, i.e. smaller. The level and dispersion of the non zero deviations seems to remain stable when the size of the table increases. These deviations tend naturally to be larger when the association measure provides larger values. Inversely, the relative deviations take larger values when the association measure tends to zero.

Finally, let us recall that the τ_b of Kendall takes its values in $[-1, 1]$. The deviations may thus exceed the absolute value of the global optimum when the quasi and global optima are of opposite signs. This explains why some maximal relative deviations are greater than one.

Globally, the outcomes of these simulation studies show that the cost in terms of reliability of the heuristic remains moderate when compared with the dramatic increase of performance.

7.4 Multidimensional grouping

In supervised learning we are interested in the best way of using the predictors to discriminate between the values of the response variable. Arbogodaï, like other tree algorithms, proceeds by partitioning the predictor values in order to reduce as much as possible the uncertainty on future responses in each class. At each step of the growing process, a node is split according to a single predictor. Interaction effects are introduced by successive splits along a stem. This has the advantage to generate an easily described partition. Some interactions, however, are not representable by trees. Hence, to allow for additional interaction effects, it may make sense to consider splits defined simultaneously on several predictors. Generalizing Arbogodaï in this way would indeed require to extend the simultaneous row-column merging process to the more general multidimensional joint merging case.

At the limit, if we consider all predictors simultaneously with the response, the multidimensional merging process, assuming it is practicable, would provide some optimal partition without resorting to a tree.

Our heuristic is intended for the simultaneous partitioning of two variables only. There is no straightforward way to extend it to the general multivariate case with more than two variables. On the one hand, it would require the definition of a suitable multivariate association measure, i.e. an index for a multiway contingency table. Coefficients like the multiple correlation measure the association between one (target) variable and the set of predictors. Hence, they do not measure globally the association between all variables. On the other hand, multiplying the dimensions of the table would dramatically increase the complexity of the heuristic and hence render it unusable.

A solution seems practicable, nevertheless, when we are in presence of one target variable and a set of predictors. In the spirit of the multiple correlation, the multivariate case can in this setting be handled by taking as column variable the composite variable defined by the crossing of the predictors. The optimal grouping of the row target variable and the composite predictor provides then simultaneously the optimal conditional partitions of the predictors and the target variable.

Let us illustrate with an example. The target variable y is dresses quality (high, poor) and the predictors are x_1 the type of dresses (W=women, M=men, C=children) and x_2 the family income (L=low, M=medium, H=high). An optimal solution may then look out as depicted in Table 11.

In this example, we see that medium and low family income are grouped together for men and children while medium and high family income are grouped for women. Likewise, all three categories women, men and children are grouped

Table 11. An aggregated multivariate table

	type	W	W	M	C	W	M	M	C	C
quality	income	M	H	H	H	L	L	M	L	M
high		50				10				
poor		5				100				

together for either high income or low income. The interactions between type and income that define these two classes clearly cannot be represented in tree form. This demonstrates the usefulness of such a multivariate approach.

7.5 Alternative merging strategies

The heuristic discussed aims at finding the optimal way to merge row and column categories in a contingency table. the adopted strategy focused on the maximization of the association. Other criteria may obviously be considered and should be investigated. For instance, when the data collected in the table are a sample of a larger population, the association computed is an estimate and one should then also care about its standard error or its significance level. Beside this aspect we are presently working on a strategy to find an optimal aggregation under some constraints. Indeed, the objective of the reduction of the size of the table is to avoid cells with low frequencies that provide unreliable information. Therefore, it is worth to be able to maximize for instance the association under a constraint on the minimal cell frequency. On the algorithmic side, we are presently working on a top-down divisive approach in which, starting from the completely aggregated table we would iteratively split rows or columns. We expect such a top-down approach to be more efficient when the number of row and column categories becomes large.

8 Conclusion

To conclude, we would like to point out that the *Arbogodaï* method is well suited for mixed nominal and ordinal multi-valued attributes since the merging of any or only adjacent values can be set on the fly. It is also able to handle similarly nominal and ordinal, hence quantitative, target variables. Thus, *Arbogodaï* could be seen as some sort of regression tree. The originality is that, unlike for instance CART that generates point predictions for each leaf, *Arbogodaï* would provide interval predictions. The multi-conclusion of an *Arbogodaï* rule can hence be seen as a generalized interval for qualitative responses. Finally, let us mention that we are presently designing further experiments for comparing *Arbogodaï* with other tree methods and especially CHAID and CART. This aspect requires a careful investigation. Indeed, the parameterization of the trees (depth, pruning, stopping rules,...) plays a crucial role on the classification performance. We are trying, therefore, to set up rigorous conditions that would ensure more fair, hence more useful, comparison results. We also plan to investigate the relationship to the minimal description length (MDL) principle [22], as the optimally reduced tables can be seen as theories that best describe, locally at each node, the relevant knowledge about the relationship between each attribute and the target variable.

Appendix

Formula of the association criteria considered. See for example [19] for more details. We denote by y the response row variable and by x the predictor column variable.

Chi-square Statistics

Pearson $$X^2 = \sum_i \sum_j \frac{(n\, n_{ij} - n_{i\cdot} n_{\cdot j})^2}{(n n_{i\cdot} n_{\cdot j})}$$

Likelihood Ratio $$G^2 = 2 \sum_i \sum_j n_{ij} \, \log\left(\frac{n\, n_{ij}}{n_{i\cdot} n_{\cdot j}}\right)$$

Association Measures Based on Pearson Chi-square

Tschuprow's t $$t = \sqrt{\frac{X^2}{n\sqrt{(\ell - 1)(m - 1)}}}$$

Cramer's v $$v = \sqrt{\frac{X^2}{n(\min\{\ell, m\} - 1)}}$$

Nominal PRE Measures

Goodman-Kruskal τ $$\tau_{y \leftarrow x} = \frac{n \sum_i \sum_j \frac{n_{ij}^2}{n_{\cdot j}} - \sum_i n_{i\cdot}^2}{n^2 - \sum_i n_{i\cdot}^2}$$

Theil's Uncertainty u $$u_{y \leftarrow x} = \frac{n \log_2 n - \sum_i \sum_j n_{ij} \log_2\left(\frac{n_{i\cdot} n_{\cdot j}}{n_{ij}}\right)}{n \log_2 n - \sum_i n_{i\cdot} \log_2 n_{i\cdot}}$$

Ordinal Association Measures

Let η^c, η^d, η_x and η_y be respectively the number of pairs $\{(x_i, y_i), (x_j, y_j)\}$ with a concordant ranking, i.e. $x_i > x_j$ and $y_i > y_j$, with a discordant ranking, i.e. $x_i > x_j$ and $y_i < y_j$, with a tie on x only and with a tie on y only.

Goodman-Kruskal γ $$\gamma = \frac{\eta^c - \eta^d}{\eta^c + \eta^d}$$

Somers' d $$d_{y \leftarrow x} = \frac{\eta^c - \eta^d}{\eta^c + \eta^d + \eta_y}$$

Kendall's τ_b $$\tau_b = \frac{\eta^c - \eta^d}{\sqrt{(\eta^c + \eta^d + \eta_x)(\eta^c + \eta^d + \eta_y))}}$$

Kendall's τ_c $$\tau_c = \frac{\eta^c - \eta^d}{\eta_{\text{tot}}} \left(\frac{\min\{\ell, m\}}{\min\{\ell, m\} - 1}\right)$$

References

1. Anderberg, M.: *Cluster Analysis for Application.* Academic Press, New York (1973)
2. Bell, E.T.: The iterated exponential numbers. *Ann. Math.* **39** (1938) 539–557
3. Benzécri, J.P.: *Analyse des données. Tome 2: Analyse des correspondances.* Dunod, Paris (1973)
4. Blake, C., Merz, C.: UCI repository of machine learning databases. http://www.ics.uci.edu/∼mlearn/MLRepository.html (1998)
5. Bock, H.: Simultaneous clustering of objects and variables. In Diday, E., Lebart, L., Pages, J., Tomassone, R., eds.: *Data Analysis and Informatics*, Amsterdam, North Holland (1979) 187–203
6. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification And Regression Trees.* Chapman and Hall, New York (1984)
7. Celeux, G., Diday, E., Govaert, G., Lechevallier, Y., Ralambondrainy, H.: *Classification automatique des données.* Informatique. Dunod, Paris (1988)
8. Fisher, W.D.: On grouping for maximum of homogeneity. *Journal of the American Statistical Association* **53** (1958) 789–798
9. Fisher, W.D.: Optimal aggregation in multi-equation prediction models. *Econometrica* **30** (1962) 744–769
10. Fisher, W.D.: *Clustering and Aggregation in Economics.* The John Hopkins Press, Baltimore (1969)
11. Gilula, Z., Krieger, A.M.: The decomposability and monotonicity of Pearson's chi-square for collapsed contingency tables with applications. *Journal of the American Statistical Association* **78** (1983) 176–180
12. Govaert, G.: Classification simultanée de tableaux binaires. In Diday, E., Jambu, M., Lebart, L., Pages, J., Tomassone, R., eds.: *Data Analysis and Informatics 3*, Amsterdam, North Holland (1984) 223–236
13. Govaert, G.: Simultaneous clustering of rows and columns. *Control and Cybernetics* **24** (1995) 438–458
14. Greenacre, M.: Clustering the rows and columns of a contingency table. *Journal of Classification* **5** (1988) 39–51
15. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning.* Springer, New York (2001)
16. Hirotsu, C.: Defining the pattern of association in two-way contingency tables. *Biometrica* **70** (1983) 579–589
17. Kass, G.V.: An exploratory technique for investigating large quantities of categorical data. *Applied Statistics* **29** (1980) 119–127
18. Morgan, J.N., Sonquist, J.A.: Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association* **58** (1963) 415–434
19. Olszak, M., Ritschard, G.: The behaviour of nominal and ordinal partial association measures. *The Statistician* **44** (1995) 195–212
20. Quinlan, J.R.: Induction of decision trees. *Machine Learning* (1986) 81–106
21. Quinlan, J.R.: *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo (1993)
22. Rissanen, J.: A universal prior for integers and estimation by minimum description length. *The Annals of Statistics* **11** (1983) 416–431
23. Ritschard, G.: Performance d'une heuristique d'agrégation optimale bidimensionnelle. *Extraction des connaissances et apprentissage* **1** (2001) 185–196

24. Ritschard, G., Nicoloyannis, N.: Aggregation and association in cross tables. In Zighed, Komorowski, Zytkow, eds.: *Principles of Data Mining and Knowledge Discovery.* Springer-Verlag, Berlin (2000) 593–598
25. Ritschard, G., Zighed, D.A., Nicoloyannis, N.: Maximisation de l'association par regroupement de lignes ou colonnes d'un tableau croisé. *Revue Mathématiques Sciences Humaines* **39** (2001) 81–97
26. Zighed, D., Rakotomalala, R., Feschet, F.: Optimal multiple intervals discretization of continuous attributes for supervised learning. In: *Proceedings of the 3rd International Conference in Knowledge Discovery in Databases.* (1997)
27. Zighed, D.A., Auray, J.P., Duru, G.: *SIPINA : Méthode et logiciel.* Editions A. Lacassagne, Lyon (1992)
28. Zighed, D.A., Rakotomalala, R.: *Graphes d'induction: apprentissage et data mining.* Hermes Science Publications, Paris (2000)