Trees and Induction Graphs for Multivariate Response

Antonio Ciampi[1], Djamel A. Zighed[2], and Jérémy Clech[1]

[1]International Agency for Research on Cancer
150, Cours Albert Thomas – 69372 Lyon Cedex 08 – France
{ciampi, clech}@iarc.fr
[2]Laboratoire ERIC – Université Lumière – Lyon2
5, Avenue Pierre Mendès-France – 69676 Bron – France
zighed@eric.univ-lyon2.fr

Abstract. We show that induction graphs can be generalized to treat more general prediction problems than those usually treated: prediction of a class variable or of a one dimensional continuous variable. We treat here the case in which the prediction concerns a multivariate continuous response. The approach used, called here GENIND1, is a combination of previous work by two of the authors (RECPAM and SIPINA). We show also that in the GENIND1 framework, clustering (unsupervised learning) as well as prediction (supervised learning) can be treated. The approach is applied to nutritional data.

1 Introduction

Induction graphs, a generalization of trees, are a powerful tool in Knowledge Discovery and Data Mining [3,8]. They can be seen as a structured extraction of predictive rules from a data set, and therefore they provide an appealing summary of the information contained therein.

In this work we initiate a new development which extends the application of both trees and induction graphs to more complex prediction problems than those usually treated by current techniques. In particular, we consider the problem of predicting a multivariate response, characterised by a random vector of which we wish to predict the mean and the variance-covariance matrix. The problem, such as formulated, has an obvious direct interest. Moreover, as we shall see, the generality of the approach will also allow us to develop clustering algorithms of the conceptual clustering type. In particular we will describe two such algorithms inspired, respectively, by the Lance and Williams algorithm [5] (see also [2,4]), and Gower's predictive classification as described in Gordon [6].

The roots of this development are in the authors previous contributions, the tree-growing algorithm RECPAM (RECursive Partition and Amalgamation) [3] and the induction graph construction method SIPINA [8]. RECPAM actually extracts from data an induction graph, but using a constructive approach which is less general than SIPINA : while in RECPAM AMalgamation of children nodes is done only at the end of the tree-growing process, in SIPINA partitioning and merging is alternated in the construction algorithm. On the other hand, RECPAM was originally conceived to predict multidimensional parameters and has been applied to handle outcome

D.A. Zighed, J. Komorowski, and J. Zytkow (Eds.): PKDD 2000, LNAI 1910, pp. 359-366, 2000.

information of complex structure, such as censored survival times, count responses and multivariate normal responses [3]. This work further develops some of the ideas in [4]. We shall call the proposed approach GENIND (GENeral INDuction).

2 GENIND, a General Algorithm for Induction Graph Construction

We describe here a first version of a general induction graph construction algorithm which is essentially a restatement of RECPAM [3]. In the future we plan to develop a SIPINA-like approach as well, and so we give the proposed family of algorithms a new name, GENIND. The RECPAM-like version presented here, will be called GENIND1, the SIPINA-like approach will be called GENIND2, and so on.

The proposed algorithm, GENIND1, consists of three steps, all conditional on a given data matrix ***D***. We suppose that the columns of ***D***, representing, as usual, variables, are partitioned *a priori* into *predictor* and *outcome variables*: ***D***=[***Z***|***Y***] where ***Z*** and ***Y*** are N×n and N×m matrices, and N is the number of individuals.

STEP 1 or *tree-growing,* consists, like CART [1], in building recursively a binary tree by maximising the Information Gain (IG) at each node until the leaves would not be larger than a minimal size fixed by the user. Here, IG is defined as the reduction of the *deviance function* as discussed below. Thus, at he end of this step, a tree structure, composed by nodes and leaves, is obtained: in GENIND language, it is the *large tree*.

STEP 2 or *pruning,* operates on this *large tree*. The pruning sequence is obtained by minimizing the Information Loss (IL) at each step, where IL is the increase in minimized deviance from the current subtree to the smaller subtree obtained by removing a question or, more generally, a branch. Thus, this operation produces a smaller tree, called the *honest tree*. Pruned branches of the *large tree*, will be referred to as *virtual* branches and are drawn in dotted lines on our tree diagrams, see Figure 1. The *honest tree* leaves can be seen as *virtual roots* of *virtual trees* with *virtual nodes* and *leaves*.

STEP 3 or *amalgamation*, operates on *partitions* (the first one is composed by the honest tree leaves) and produces an induction graph, GENIND1 Graph. It is proper to RECPAM and is similar to STEP 2. It recursively constructs a superpartition from a (super)partition by amalgamating two sets of the given (super)partitions and each update being obtained from the current partition by merging the two sets resulting in minimum IL. The purpose of this step is to simplify the prediction and further increase generalizability.

More details of the steps are described in [3]. These three steps can be seen as a suboptimal but ‘greedy’ construction of a predictive model best fitting the data. The novelty in our approach is twofold. Firstly, the merging step, which leads from a tree to an induction graph, is new. Secondly, and perhaps most importantly, the approach presented here can naturally be applied to more complex response structures than the simple ones found in classification and (univariate) regression. This generality is embodied in the definition of deviance: we will show in the next two sections 3 and 4 how a proper definition of the deviance function permits the development of genuinely new algorithms.

The result of GENIND1 when applied to a data matrix ***D*** is both a GENIND1-*predictive structure* and a GENIND1-*predictor*. The *predictive structure* is the functional form of the dependence of the *parameter* $\theta = (\theta_1, \theta_2, \ldots, \theta_p)$ on the predictor variables *z*, see (1a). In general, in order to specify it, we need a function which associates to each individual of the population, a value for each of the components of θ. The parameters are features of the process generating the outcome variable vector ***y***: for example if we can specify a statistical model for ***y***, θ may represent the parameter of the associated probability distribution.

In order to define this *predictive structure*, let the vector of the predictors be partitioned as: $(\boldsymbol{z}|\boldsymbol{x})$, where the *z*'s are called *tree predictors* or t-predictors. We will denote by *I* all 'dummy' variables attached to GENIND1 elements. Thus a particular *I* indicates, for every individual of the population, whether or not he belongs to the GENIND1 element associated to it. More explicitely, I_g, g = 1,…, G, will denote the 'dummy' variables of the GENIND1 classes, I_l, l = 1,…, L, those of the leaves of the tree, and $I_{l(v)}$, v = 1,…,V, those of the virtual leaves. Clearly, the I_g's and the I_l's can be expressed as sums of the $I_{l(v)}$'s. In the equations below we use the slightly imprecise notation '$v \in l$' and '$v \in g$' to denote an index running, respectively, over the virtual leaves belonging to leaf *l* and to GENIND1-class *g*. We now can write the predictive structures associated to the honest tree and to the GENIND1-classification as follows:

Honest Tree:

$$\theta_k(z; x^{(r(k))}, x^{(lf(k))}, x^{(v(k))}) = x^{(r(k))} \cdot \beta^{(k)} + \sum_{l=1}^{L} (x^{(lf(k))} \cdot \gamma_l^{(k)}) I_l(z) + \sum_{l=1}^{L} \sum_{v \in l} (x^{(v(k))} \cdot \alpha_{l(v)}^{(k)}) I_{l(v)}(z) \tag{1a}$$

GENIND1-classification:

$$\theta_k(z; x^{(r(k))}, x^{(lf(k))}, x^{(v(k))}) = x^{(r(k))} \cdot \beta^{(k)} + \sum_{l=1}^{G} (x^{(lf(k))} \cdot \gamma_l^{(k)}) I_l(z) + \sum_{l=1}^{L} \sum_{v \in g} (x^{(v(k))} \cdot \alpha_{l(v)}^{(k)}) I_{l(v)}(z) \tag{1b}$$

for k = 1,…,p. Notice that for each component of θ there are three subsets of x, denoted x(r(k)), x(lf(k)) and x(v(k)). These are the root-, leaf- and virtual-leaf-predictors for component k, abbreviated as r-, lf- and v-predictors, to be specified by the user. Notice also that if the user specify that there are no v-variables, then the third term in the above equation is not present, and similarly for r-variables.

The simplest specification, which could be the default, is that there is only one lf-variable and that this is the constant term (*i.e.* the other two sets of variables are empty). Then the above equations become simply:

Honest Tree:

$$\theta_k(z) = \sum_{l=1}^{L} \gamma_l^{(k)} I_l(z) \tag{2a}$$

GENIND1-classification:

$$\theta_k(z) = \sum_{g=1}^{G} \gamma_g^{(k)} I_g(z) \tag{2b}$$

for k = 1,…,p. When the (θ1, θ2,…,θp) are defined as class probabilities, then the honest tree equation represents a predictive structure identical to that given by CART, AID, ID3, etc. for classification; similarly, the GENIND1-classification equation reduces to the RECPAM predictive structure for multinomial probabilities.

By GENIND1-predictor we mean the GENIND1-predictive structure, with values of the parameters which appear in the structure specified by minimization of the deviance function. We will use ‘hats’ in order to distinguish the predictive structure from the predictor, *e.g.* the predictor corresponding to the predictive structure (2b) will be denoted by:

$$\hat{\theta}_k(z)=\sum_{g=1}^{G}\hat{\gamma}_g^{(k)}I_g(z) \tag{3}$$

3 GENIND1 for Multivariate Outcome

Suppose we want to discover the relationship between a vector of multivariate outcomes y and a vector of covariates (predictors) z. Suppose also that we may assume that z only affects the mean vector and the variance-covariance matrix of y. Then: θ =(μ1,…,μp, Σ) =(μ, Σ). The goal is to discover a predictive structure of the GENIND1 type which is an adequate representation of the relationship between z and y. Therefore our approach explicitely takes care of the associations among components of y, in contrast with the approach consisting in growing distinct decision trees for each component of y. To do so, we dispose of a data matrix D=[Z|Y]. For simplicity we are restricting ourselves to the case in which there are no x-variables other than the constant, i.e. for each component of θ one and only one of the three sets of r-, lf-, and v- variables is non-empty and this may only contain the constant term. We also limit ourselves to the situation in which the specification of where the constant term belongs is the same for all components of μ and for all components of Σ (though it may be different for μ and for Σ). A more general structure will be described elsewhere.

3.1 Deviance Function and Information Content

First, suppose that the trivial partition (root node) is an adequate representation of the data. Then for any given (μ, Σ) a natural definition for the deviance function is the sum of the Malahanobis distances of each vector from the mean:

$$dev(\boldsymbol{D},\theta)=\sum_{i=1}^{\mathrm{N}}(y^{(i)}-\mu)^{\mathrm{T}}\Sigma^{-1}(y^{(i)}-\mu)$$

If we assume multivariate normality, which is not necessary in this context, an even more natural definition of deviance is twice the negative log-likelihood:

$$dev(\boldsymbol{D},\theta)=p\log|\Sigma|+\sum_{i=1}^{\mathrm{N}}(y^{(i)}-\mu)^{\mathrm{T}}\Sigma^{-1}(y^{(i)}-\mu)$$

As described in the previous section, the algorithm repeatedly computes differences of *minimized deviances*. For instance the IG of a tree *T* with respect to the trivial tree T_0 given the data matrix ***D*** is defined, with obvious meaning of symbols, as:

$$IG(T : T_0 \mid \boldsymbol{D}) = dev(\boldsymbol{D}, \hat{\theta}_{T_0}) - dev(\boldsymbol{D}, \hat{\theta}_T)$$

3.2 GENIND1-Predictor

Now, depending on the specific assumptions we want to introduce, different predictive structures may be specified. We will consider the following three cases:

1. The variance-covariance matrix is assumed known. (In practice the variance-covariance matrix is never known, but assuming it constant, it can be estimated once and for all as the sample variance-covariance matrix at the root). Then the only component of interest is the vector of the means. The predictor associated to the GENIND1-classification is simply:

$$\hat{\mu}(z) = \sum\nolimits_{g=1}^{G} \hat{\mu}_g I_g \tag{4}$$

Deviance minimization is trivial and the values of the parameters in (4) are given by the sample means at the GENIND1-classes of the components of *y*, with obvious meaning of symbols:

$$\hat{\mu}_g(z) = \frac{1}{N_g} \sum_{i \in g} y^{(i)}$$

2. The variance-covariance matrix is assumed to be unknown but constant. Then in the language of this work, the μ-component has constant term as lf-variable, while for the Σ-component has constant term as r-variable. This corresponds to the predictor:

$$\hat{\mu}(z) = \sum_{g=1}^{G} \hat{\mu}_g I_g(z) \tag{5a}$$

$$\hat{\Sigma}(z) = \hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (y^{(i)} - \hat{\mu}(\mathrm{z}^{(\mathrm{i})}))^T (y^{(i)} - \hat{\mu}(\mathrm{z}^{(\mathrm{i})})) \tag{5b}$$

3. The variance-covariance is assumed to vary the same way as the mean vector. Then, the μ-component has the same expression than (5a), whereas the Σ-component has the constant term as lf-variable, which gives:

$$\hat{\Sigma}(z) = \sum_{g=1}^{G} \hat{\Sigma}_g I_g(z) \quad \text{with} \quad \hat{\Sigma}_g = \frac{1}{N_g} \sum_{i \in g} (y^{(i)} - \hat{\mu}_g(z^{(i)}))^T (y^{(i)} - \hat{\mu}_g(\mathrm{z}^{(\mathrm{i})})) \tag{6}$$

4 Conceptual Clustering Algorithm Based on GENIND

In the previous section, we have shown how to learn, from a data matrix D =[Z|Y], a prediction rule for y given z . This implies an a priori distinction between z and y, a natural one when y can be regarded as outcome. Although this situation, or supervised learning, has an obvious intrinsic interest (see example in the next section), GENIND can also serve as basis for an unsupervised learning approach.

Suppose then that no clear distinction can be made among variables, so ***D****=[X]* is considered as a set of measurements of the vector *x*. We are interested in discovering from ***D*** a structure of homogeneous and distinct classes of individuals. We also require that these classes be defined by simple statements involving some of the components of *x*, in other words we are interested in developing *conceptual clustering* algorithms. We propose here a GENIND based algorithm, called 'Factor Supervision', but of course, some other is possible like the Gower's predictive clustering approach (see also Gordon [6]).

Our approach is based on an earlier proposal [2]. It consists in transforming the unsupervised learning problem into a supervised one, with supervision provided by the first few factors extracted from [X]. Specifically, if y = (y1,y2,…,yp) denotes a vector having as components the first p principal components of X, let us consider the augmented matrix D′ =[X|Y]. Then the GENIND approach can be applied to the augmented matrix to discover clusters. Discovering such clusters means discovering clusters in a subspace where most of the 'interesting' dispersion of the original variables takes place, yet these clusters are described in terms of the original variables.

5 An Example from Nutritional Epidemiology

To illustrate the flexibility of GENIND1 both as a method for constructing predictors and as a method of conceptual clustering, we report briefly on a preliminary analysis of a data set which is a subset from a much larger epidemiological study called EPIC. It is no more than an illustration; a full report will appear elsewhere. EPIC is a multi-centre prospective cohort study designed to investigate the effect of dietary, metabolic and other life-style factors on the risk of cancer [7]. The study started in 1990 and includes now 23 centres from 10 European countries. By now, dietary data are available on almost 500,000 subjects. Here we will consider only data from a subsample of 1,201 women from the centre 'Ile-de-France'. Also, we limit ourselves to an analysis of data from a 24-hour recall questionnaire concerning intake of 16 food-groups and four energy-producing nutrients : carbohydrates, lipids, proteins and alcohol. The food-group variables, are expressed in grams of intake, and the nutrient variables are measured in Kcal.

5.1 Predicting nutrieNts from Food-Group Variables

In a first analysis, we constructed a GENIND1 predictor for nutrients using the food-group variables as predictor variables. The induction graph in Figure 1, actually a tree, was obtained by the approach of section 3, with a likelihood-based deviance and assuming that both mean vector and variance-covariance matrix vary across the leaves, see equations (5a) and (6). We obtained a large tree with 5 leaves, which reduced to an honest tree of 3 leaves after pruning; no amalgamation was possible. Two food-group variables define the tree structure : Alcohol and Meat. For leaf 1 we have the following energy consumption pattern (standard deviation in parenthesis): 201.101 (76.142) Kcal for carbohydrates, 77.047 (33.864) Kcal for lipids, 75.139 (25.261) Kcal for proteins and 0.006 (0.130) Kcal for alcohol. For leaf 2 the pattern is: 208.681 (84.166) Kcal for carbohydrates, 84.995 (38.957) Kcal for lipids, 73.494 (23.879) Kcal for proteins and 17.681 (13.701) Kcal for alcohol. Finally for leaf 3 we have 201.044 (81.174) Kcal for carbohydrates, 104.155 (43.092) Kcal for lipids, 99.647 (30.547) Kcal for proteins and 23.885 (21.198) Kcal for alcohol.

The verifications of our assumptions are in progress, but we think that if this GENIND graph is very small, it is probably du to some similar diets in the centre 'Ile-de-France'.

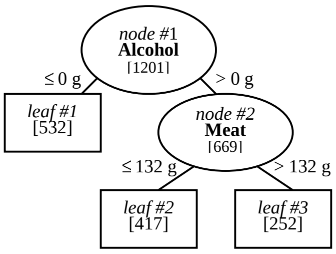

Fig. 1. GENIND1 graph for predictive clustering.

5.2 Clustering Based on Food-Group Variables: Factor Supervision

The graph in Figure 2 was obtained by applying the 'factor supervision' approach of section 4. A preliminary principal component analysis of the sixteen food-group variables yielded five principal components which explain 80% of the dispersion. These components were used as 'response variables' in a GENIND1 construction. A 7-leaves large tree was pruned to a 3-leaves honest tree which, after the AMalgamation step, yielded two GENIND1 classes. One class includes 371 subjects characterized by no consumption of 'Soups and bouillon' and 'Alcohol'. The other class includes 830 subjects which consume at least one of 'Alcohol' and 'Soups and bouillon'.

Acknowledgements

We thank Dr. F. Clavel for having made the data available and Dr. E. Riboli for general assistance in becoming familiar with the EPIC project.

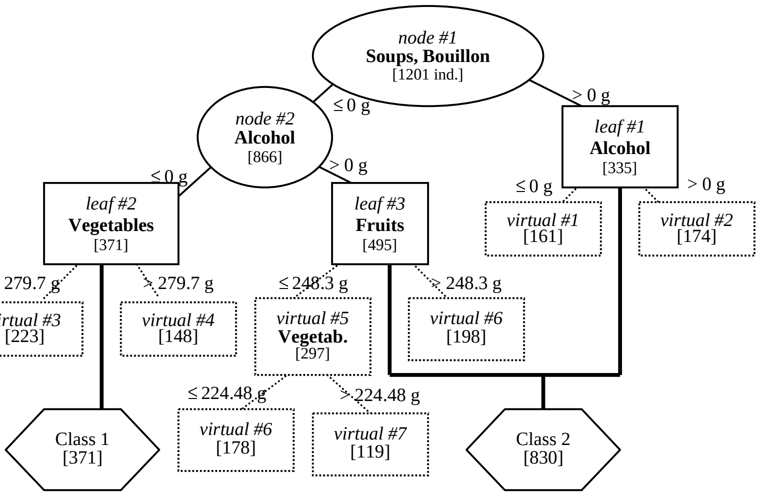

Fig. 2. GENIND1 graph for factor supervision.

References

1. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification And Regression Trees. The Wadsworth Statistics/Probability Series (1984).
2. Chavent, M., Guinot, C., Lechevallier, Y., Tenenhaus, M.: Méthodes divises de classification et segmentation non supervisée: recherche d'une typologie de la peau humaine saine. Rev. Statistique Appliquée, XLVII (1999) 87-99.
3. Ciampi, A.: Constructing prediction trees from data: the RECPAM approach. Proceedings of the Prague '91 Summer School of Computational Aspects of Model Choice, Physica-Verlag, Heidelberg, (1992) 105-152.
4. Ciampi, A.: Classification and discrimination: the RECPAM approach. COMPSTAT 94, Physica Verlag, Heidelberg, (1994) 129-147.
5. Lance, G.N., Williams, W.T.: A general theory of classificatory sorting strategies. Computer Journal, (1967) 9: 373-380.
6. Gordon, A.D.: Classification. CRC Press 2nd Edition (1999).
7. Riboli E, Kaaks R.: The EPIC Project: rationale and study design. European Prospective Investigation into Cancer and Nutrition. Int J Epidemiol. (1997) 26 Suppl : S6-14.
8. Zighed, D.A., Rakotomala, R.: Graphes d'induction - Apprentissage et Data Mining. HERMES (2000).