Available online at www.sciencedirect.com

SCIENCE DIRECT®

Statistics & Probability Letters 69 (2004) 253–259

STATISTICS & PROBABILITY LETTERS

www.elsevier.com/locate/stapro

Fitting multivariate responses using scalar trees[☆]

María Jesús Barcena, Fernando Tusell*

Departamento de Estadística y Econometría, Facultad de CC.EE. y Empresariales, Avda. Lehendakari Aguirre, 83, 48015 Bilbao, Spain

Received 1 March 2004; received in revised form 4 April 2004

Available online 6 July 2004

Abstract

We describe an algorithm for the fitting of multivariate responses using classification and regression trees, named the intersection-seeking algorithm. Although motivated by problems of record linkage and imputation of missing values in surveys, the algorithm may be used in other contexts.

Keywords: File completion; Imputation; Regression and classification trees

1. Introduction

We consider the case in which a vector response $\boldsymbol{Y} = (Y_1, \ldots, Y_q)$ depends on the vector of predictors $\boldsymbol{X} = (X_1, \ldots, X_p)$. We have a training sample of N_A cases in which both $\boldsymbol{X}$ and $\boldsymbol{Y}$ are observed. For an additional N_B cases, only $\boldsymbol{X}$ is observed and we are required to produce fitted values for $\boldsymbol{Y}$. Thus, we have $N = N_A + N_B$ observations with the structure shown in Fig. 1.

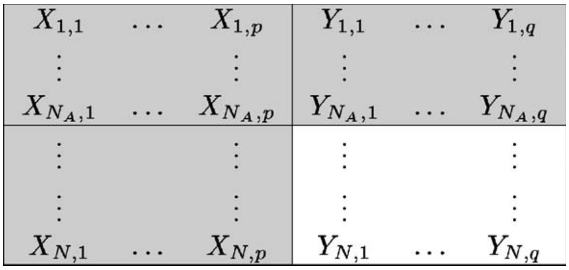
This is a common problem arising when we want to produce a single file with regular structure out of different files with only a common subset of variables observed (for example, two files may contain data from two sample surveys which share a common set of questions). The next section describes an algorithm designed to produce fits using univariate response binary trees.

[☆] We thank, for support, the Spanish MEC (grant PB98-0149) and the UPV/EHU (Grupo Consolidado 9/UPV 00038.321-13631/2001).

*Corresponding author.

E-mail address: etptupaf@bs.ehu.es (F. Tusell).

0167-7152/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.spl.2004.06.012



$X_{1,1}$	$\cdots$	$X_{1,p}$	$Y_{1,1}$	$\cdots$	$Y_{1,q}$
$\vdots$		$\vdots$	$\vdots$		$\vdots$
$X_{N_A,1}$	$\cdots$	$X_{N_A,p}$	$Y_{N_A,1}$	$\cdots$	$Y_{N_A,q}$
$\vdots$		$\vdots$	$\vdots$		$\vdots$
$\vdots$		$\vdots$	$\vdots$		$\vdots$
$X_{N,1}$	$\cdots$	$X_{N,p}$	$Y_{N,1}$	$\cdots$	$Y_{N,q}$

Fig. 1. Structure of the problem. The unshaded area of the table is missing.

2. The intersection-seeking algorithm

Consider the case i, $i \in \{N_A+1, \ldots, N\}$, for which an imputation of $\mathbf{Y}_{\rm i}$ is sought. To simultaneously impute $\mathbf{Y}_{\rm i} = (Y_{i1}, \ldots, Y_{iq})$, we use the univariate trees $\mathscr{Y}_X^{(j)}$ constructed for each of the variables Y_j, $j = 1, \ldots, q$, on the predictors in $\boldsymbol{X}$. They are built with the methodology described by Breiman et al. (1984) as implemented by Therneau and Atkinson (1997); but a different strategy can be used (e.g., Murthy et al., 1994).

Assume that when case i is dropped down the trees built for each of the variables in Y, it ends in the leaves $\mathscr{Y}_{i_1}^{(1)}, \ldots, \mathscr{Y}_{i_q}^{(q)}$ and hence belongs to the intersection:

$$\mathscr{C}_{i_1,\ldots,i_q} = \mathscr{Y}_{i_1}^{(1)} \cap \mathscr{Y}_{i_2}^{(2)} \cap \cdots \cap \mathscr{Y}_{i_q}^{(q)}. \tag{1}$$

The simple idea in our method is to impute $\mathbf{Y}_{\rm i}$ as a function of the values $\mathbf{Y}$ from cases in the training sample (file A) which also belong to $\mathscr{C}_{i_1,\ldots,i_q}$. Those cases have values for each variable $Y_1, \ldots, Y_q$ which, as far as the relevant trees can ascertain, are indistinguishable from the ones of the case to impute. We can impute using one $\mathbf{Y}$ sampled randomly from $\mathscr{C}_{i_1,\ldots,i_q}$ or several if multiple imputation is desired.

For instance, let $q = 2$ and let the trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$ have the simple form depicted in Fig. 2. Let $\mathscr{X}$ be the space of all possible values of X. A tree of Y on the X induces a partition of $\mathscr{X}$ such that in each class we have like values of Y. In Fig. 3 the partitions of the $\mathscr{X}$ space induced by trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$ are shown.

Consider a case to impute i such that $a' < X_1 < a$ and $X_2 < b''$; it will end in leaves $\mathscr{Y}_2^{(1)}$ and $\mathscr{Y}_7^{(2)}$ when dropped down the trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$. The intersection of those leaves,

$$\mathscr{C}_{2,7} = \mathscr{Y}_2^{(1)} \cap \mathscr{Y}_7^{(2)} \tag{2}$$

is shown in Fig. 4. We propose to impute $\mathbf{Y}_{\rm i}$ using the values of $\mathbf{Y}$ observed for cases in the training sample that also fall in $\mathscr{C}_{2,7}$. The partition of $\mathscr{X}$ made of all intersections is the coarsest one such that each intersection contains cases falling in the same leaves when dropped down the trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$.

A problem may arise if no cases in the training sample belong to a particular intersection $\mathscr{C}_{i_1,\ldots,i_q}$—not one of the subjects in the training sample ended in exactly the same leaves as the subject to impute. When this happens, the intersection needs to be gradually enlarged to a nonempty set: starting from the leaves $\mathscr{Y}_{i_1}^{(1)}, \ldots, \mathscr{Y}_{i_q}^{(q)}$ where i ended, our algorithm "climbs" the trees, replacing one node at a time by its "father". In doing so, we have at each step a choice of q

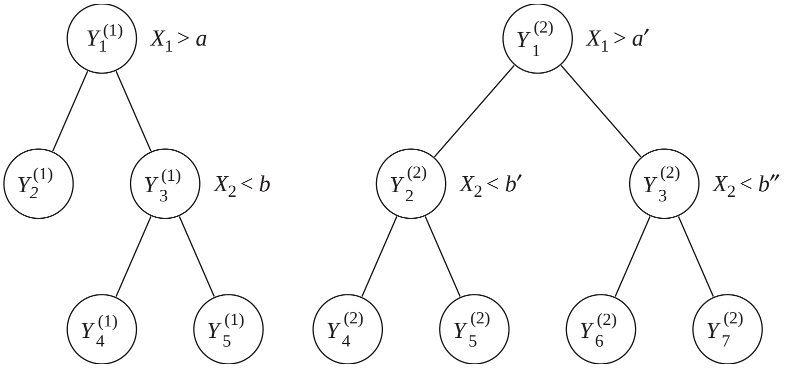


Fig. 2. Trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$. Next to each nonterminal node is the condition whose fulfillment sends a case through the right son.

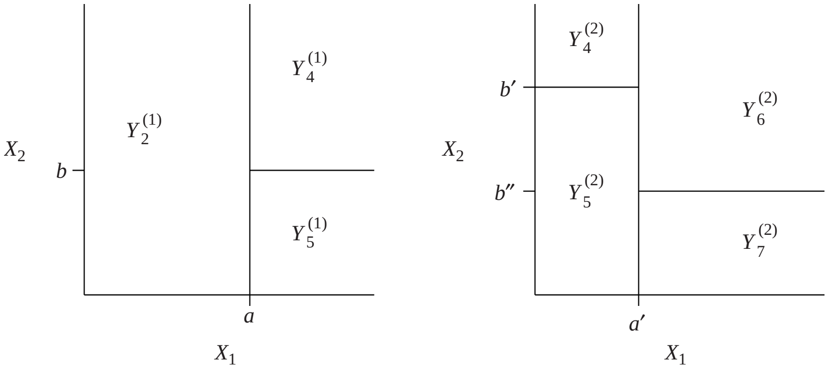


Fig. 3. Partitions of the $\mathscr{X}$ space induced by trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$.

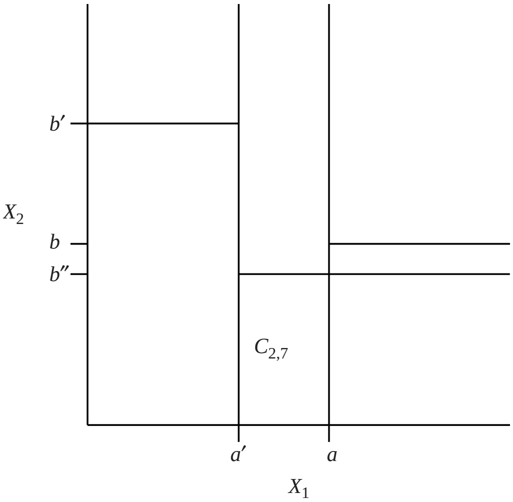


Fig. 4. Overlay of partitions of $\mathscr{X}$ induced by trees $\mathscr{Y}_X^{(1)}$ and $\mathscr{Y}_X^{(2)}$, and intersection $\mathscr{C}_{2,7}$.

trees that we may climb. The goal is to choose each step in such a way that the quality of the imputation suffers least. Thus, the order of climbing is governed by the deviance—we first climb the tree where the replacement of a node by its ancestor leads to the least possible increase in deviance.

3. Simulations

We considered situations with categorical and continuous predictors and responses that were always continuous. The p-dimensional vector $(\boldsymbol{W}, \boldsymbol{Z})$ of fully observed variables (the X columns in Fig. 1) is made of p_{CAT} categorical variables ($\boldsymbol{W}$) and p_{CON} continuous variables ($\boldsymbol{Z}$). The q-dimensional vector $\boldsymbol{Y}$ of responses (the Y columns in Fig. 1) were generated as functions of the $p = p_{\mathrm{CAT}} + p_{\mathrm{CON}}$ predictors plus noise: $\boldsymbol{Y} = f(\boldsymbol{W}, \boldsymbol{Z}) + \varepsilon$, with $f()$ linear or quadratic.

The $\boldsymbol{X} = (\boldsymbol{W}, \boldsymbol{Z})$ are generated according to the following scheme (the general location model, briefly described next: see also Olkin and Tate, 1961). The vector $\boldsymbol{W}$ is drawn from a multinomial distribution. Let d_ℓ be the number of levels of variable W_ℓ in $\boldsymbol{W}$ ($\ell = 1, \ldots, p_{\mathrm{CAT}}$); then, $\boldsymbol{W}$ can take one of $D = \prod_{\ell=1}^{p_{\mathrm{CAT}}} d_\ell$ states. If $\boldsymbol{W}$ takes state d, $(d \in 1, \ldots, D)$, then $\boldsymbol{Z}|\boldsymbol{W} \sim N(\boldsymbol{\mu}_d, \Sigma)$. Therefore, $\boldsymbol{Z}$ is multivariate normal with common covariance matrix and possibly a different mean vector $\boldsymbol{\mu}_{\mathrm{i}}$ for each combination of levels of the categorical variables.

When $f()$ in $\boldsymbol{Y} = f(\boldsymbol{W}, \boldsymbol{Z}) + \varepsilon$ is linear, the full vector $(\boldsymbol{X}, \boldsymbol{Y})$ is generated according to the general location model; when $f()$ is quadratic, the general location model is only an approximation. We have investigated both cases. The noise vector ε has covariance matrix $\boldsymbol{I}$ (correlated noise made little difference and is not reported). The function $f()$ was scaled so that the signal-to-noise ratio (SNR)—the square root of the ratio of variances of the signal $f(\boldsymbol{W}, \boldsymbol{Z})$ and the noise ε—could be set at different levels.

Each combination of parameters, picked from each column of Table 1, was used to generate $n = 500$ artificial samples. From each sample of size N, the q responses from the last $s = 50$ observations were deleted and then reconstructed once using two methods:

- The intersection-seeking algorithm (Inter). We imputed once by drawing a single observation from the intersection.
- The EM algorithm plus data augmentation based on the general location model, as implemented in the mix library by J. Schafer (available at http://www.stat.psu.edu/~jls/). With this method, the imputations are values drawn from the distribution of the missing data, given the observed data and the parameters set at their maximum likelihood estimated values:

Table 1
Summary of simulation setup. Parameters in curly brackets represent alternatives

p_{CAT}	p_{CON}	q	Levels of $\boldsymbol{W}$	SNR	N
3	2	3	(3,2,3)	{3,5,10}	{200,500,1000}
5	0	5	(8,4,4,3,2)	{3,5,10}	{200,500,1000}
5	5	5	(8,4,4,3,2)	{3,5,10}	{200,500,1000}

$f(\mathbf{X}_{\text{mis}}|\mathbf{X}_{\text{obs}}, \hat{\theta}_{\text{ML}})$. We are not taking into account the variability of $\hat{\theta}_{\text{ML}}$ (obtained with the EM algorithm). Hence, these are not "bayesianly proper" imputations (see Schafer, 1997, p. 105).

The first method is non parametric and the second is parametric and assumes data generated by the general location model. Our objective was to benchmark the intersection-seeking algorithm and see how much one loses in exchange for the flexibility and relative generality of a non parametric method, both when the parametric competitor is "right" ($f(\boldsymbol{W}, \boldsymbol{Z})$ linear) and when it is not ($f(\boldsymbol{W}, \boldsymbol{Z})$ quadratic). Each of the methods was trained on $N - s$ observations, then used to impute the remaining $s = 50$. The criterion to assess the quality of imputation was the square root of the average square error of imputation (RMSE), defined as

$$\text{RMSE} = \sqrt{\frac{1}{sq} \sum\nolimits_{i=N-s+1}^{N} \sum\nolimits_{j=1}^{q} (Y_{ij} - \hat{Y}_{ij})^2}.$$

Since the variance of the variables in $\boldsymbol{Y}$ is $\text{SNR}^2 + 1$, the theoretical RMSE, while imputing each observation by another one randomly chosen from the sample, is $\sqrt{2(\text{SNR}^2 + 1)}$. The larger the drop below that achieved by an imputation method, the better.

We used a port to R (described in Ihaka and Gentleman, 1996) of the library mix and we have also written our own functions in R for the intersection-seeking method. We made extensive use of library rpart (described in Therneau and Atkinson (1997) available from CRAN, http://cran.at.r-project.org).

Here we report on a subset of results which convey the essential of what we found. Table 2 lists the average square root of RMSE defined above for different combinations of SNR, sample size $N, p_{\text{CAT}}, p_{\text{CON}}, q$ and different types of functional relationship $f()$.

When the general location model is adequate (there is a small number of parameters involved and the dependency among predictors and responses is linear), the EM estimation plus imputation by data augmentation (in mix) performs quite well. This happens with Case 1 in Table 2. The intersection-seeking algorithm does not perform nearly as well.

When the general location model is not adequate (like in Case 2, where the functional relationship linking responses and predictors is quadratic), the parametric method (mix) is hardly better than random deck imputation. The performance of the intersection-seeking algorithm also suffers, but is still better than random imputation, except for the smallest sample size: the trees are flexible enough to capture at least partially the relationship among predictors and responses.

We have found that even when responses and predictors are related as the general location model assumes, the intersection-seeking algorithm may be best when N is not large relative to $D = \prod_{\ell=1}^{p_{\text{CAT}}} d_\ell$ (d_ℓ is the number of levels of W_ℓ). The last two columns of Table 2 epitomize two such situations.

Case 3 considers the case with ten predictors (five categorical, five multivariate normal) and five multivariate normal responses. The smallest sample size ($N = 200$) has been dropped from the simulation as it was insufficient to use either method. *Even though the observations have been generated according to the general location model*, the non-parametric, intersection-seeking algorithm does nearly as well for the largest sample size ($N = 1000$) and looks even slightly better for $N = 500$. The reason for this seemingly counter-intuitive result is the following: $p_{\text{CAT}} = 5$ categorical predictors with 8, 4, 4, 3 and 2 levels give a total of $D = \prod_{\ell=1}^{5} d_\ell = 768$ cells. The

Table 2
Each figure is the average RMSE of imputation on $n = 500$ replications

SNR	N	Case 1		Case 2		Case 3		Case 4	
		Inter	mix	Inter	mix	Inter	mix	Inter	mix
3	200	2.14	1.41	4.95	4.58	—	—	2.14	2.97
3	500	1.94	1.42	3.84	4.61	2.32	2.57	1.90	2.60
3	1000	1.92	1.43	3.40	4.56	2.32	2.13	1.92	2.15
$\sqrt{2(\mathrm{SNR}^2+1)}$		4.47		4.47		4.47		4.47	
5	200	3.05	1.43	8.00	7.40	—	—	3.06	4.66
5	500	2.67	1.42	5.90	7.25	3.39	3.94	2.59	3.96
5	1000	2.59	1.42	5.25	7.40	3.38	3.04	2.61	3.02
$\sqrt{2(\mathrm{SNR}^2+1)}$		7.21		7.21		7.21		7.21	
10	200	5.67	1.41	16.10	14.80	—	—	5.63	9.13
10	500	4.77	1.42	11.90	14.60	6.44	7.60	4.58	7.60
10	1000	4.57	1.42	10.40	14.60	6.35	5.57	4.59	5.56
$\sqrt{2(\mathrm{SNR}^2+1)}$		14.21		14.21		14.21		14.21	
p_{CAT}		3		3		5		5	
With levels:		(3, 2, 3)		(3, 2, 3)		(8, 4, 4, 3, 2)		(8, 4, 4, 3, 2)	
p_{CON}		2		2		5		0	
q		3		3		5		5	
Dependency:		Linear		Quadratic		Linear		Linear	

The parameters $p_{\mathrm{CAT}}, p_{\mathrm{CON}}$ and q and the type of dependency among predictors and responses is given below each column. $\sqrt{2(\mathrm{SNR}^2+1)}$ is the RMSE achieved imputing with a case randomly chosen among those completely observed.

general location model prescribes one mean vector $\boldsymbol{\mu}_d$ for each cell, totally unrelated to each other. Clearly, with $N = 500$ observations, a large portion of those mean vectors cannot be estimated. The mix library replaces the global mean vector $\boldsymbol{\mu}$ when there is need to impute a case with a combination of the p_{CAT} levels not seen in the training sample. No advantage is taken of the fact that, perhaps, a "similar" though not equal combination of levels is present in the training sample.

As compared to this, the intersection-seeking algorithm imputes from a pool of similar cases in an intersection of leaves. If the intersection is empty in the first instance, it will be enlarged gradually and a case will be drawn from the first nonempty intersection, rather than from the whole training sample; this accounts for its superiority when N is not large relative to $D = \prod_{\ell=1}^{p_{\mathrm{CAT}}} d_\ell$.

This superiority is all the more noticeable when there are no continuous predictors (Case 4). The general location model has a clear advantage at capturing linear relationships among continuous variables; trees can only give a coarser, step-like approximation to those relations. But when there are no continuous predictors and N is not large relative to D (Case 4 in Table 2), the intersection-seeking algorithm performs at its best although the general location model recovers some ground as the sample size increases.

One of the referees pointed to us that binary trees are known to be biased towards splitting on categorical variables with many levels (see for instance Loh, 2002). Our predictors in $\boldsymbol{W}$ have always a modest number d_ℓ of levels: increasing those would no doubt make the job of the trees harder, but would also make $D = \prod_{\ell=1}^{p_{\mathrm{CAT}}} d_\ell$ much larger. Short of using a huge sample size, the contingency table generated by $\boldsymbol{W}$ would have most of its cells empty, and most of the mean vectors $\boldsymbol{\mu}_d$ would not be estimable. Hence, the mix library would be predicting most of the time with the global mean vector $\boldsymbol{\mu}$, and the comparison would no longer be fair.

4. Summary and conclusions

A method for multivariate approximation has been presented. It can cope with a large variety of problems, because of the generality of the tool used for approximation—classification or regression trees. It makes few assumptions, is computationally feasible, and appears to give good results: in simulated data, the method works well whenever the common variables X are good predictors for the Ys (see Fig. 1) and the functional relationship among predictors and responses can be approximated reasonably well by a tree.

The method has been tested on simulated and real data sets of relatively large size (see Bárcena and Tusell, 1999, 2000) and can also be extended to cope with irregular patterns of missingness in the data (see Bárcena, 2001).

Acknowledgements

We gratefully acknowledge comments from Vicente Nuñez, Eva Ferreira, Karmele Fernández, and participants of the IWSM' 15 and COMPSTAT'2000 meetings. Comments by one referee led to a much improved revision. Any errors and obscurities that remain are our own.

References

Bárcena, M., 2001. Técnicas multivariantes para el enlace de encuestas. Ph.D. Thesis, Universidad del País Vasco.

Bárcena, M., Tusell, F., 1999. Enlace de encuestas: una propuesta metodológica y aplicación a la Encuesta de Presupuestos de Tiempo. Qüestiió 23, 297–320.

Bárcena, M., Tusell, F., 2000. Tree-based algorithms for missing-data imputation. In: Bethlehem, J., van der Heijden, P. (Eds.), COMPSTAT'2000. Proceedings in Computational Statistics. Physica-Verlag, Heidelberg, pp. 193–198.

Breiman, L., Friedman, J., Olshen, R., Stone, C., 1984. Classification and Regression Trees. Wadsworth, Belmont, CA.

Ihaka, R., Gentleman, R., 1996. R: a language for data analysis and graphics. J. Comput. Graph. Statist. 5, 299–314.

Loh, W.-Y., 2002. Regression trees with unbiased variables selection and interaction detection. Statist. Sinica 12, 361–368.

Murthy, S., Kasif, S., Salzberg, S., 1994. A system for induction of oblique decision trees. J. Artif. Intell. Res. 2, 1–32.

Olkin, I., Tate, R., 1961. Multivariate correlation models with mixed discrete and continuous variables. Ann. Math. Statist. 32, 448–465.

Schafer, J., 1997. Analysis of Incomplete Multivariate Data. Chapman & Hall, London.

Therneau, T., Atkinson, E., 1997. An introduction to recursive partitioning using the RPART routines. Technical Report, Mayo Foundation.