Les Cahiers du GERAD ISSN: 0711–2440

Multivariate Forests with Missing Mixed Outcomes

A. Dine, D. Larocque,
F. Bellavance

G–2013–95

December 2013

Multivariate Forests with Missing Mixed Outcomes

Abdessamad Dine
Denis Larocque
François Bellavance

GERAD & Department of Management Sciences
HEC Montréal
Montréal (Québec) Canada, H3TG 2A7

abdessamad.dine@hec.ca
denis.larocque@hec.ca
francois.bellavance@hec.ca

December 2013

Les Cahiers du GERAD
G–2013–95

Abstract: In this paper, we propose a multivariate random forest method for multiple responses of mixed types with missing responses. Imputation is performed for each bootstrap sample used to build the individual trees that form the forest. The individual trees are built using a weighted splitting rule allowing down–weighting of imputed observations. A simulation study shows the benefits of this approach over complete case analysis when missing responses are MCAR and MAR. In particular, the gain in prediction accuracy of the proposed method is larger in the MAR case and also increases as the proportion of missing increases.

Key Words: Multivariate tree; Random forest; Mixed responses; Multiple Imputation; General location model; GLOM.

1 Introduction

Tree–based ensemble methods like random forests (Breiman, 2001) are well–established powerful and useful prediction tools; see Rokach (2008), Siroky (2009) and Verikas et al. (2011) for recent surveys. However, the literature is more sparse for the case of multivariate responses. When all outcomes are of the same type (e.g. all continuous or all binary), multivariate trees have been proposed by Ciampi et al. (1991), Segal (1992), Zhang (1998), Siciliano and Mola (2000) and Lee (2005). Segal and Xiao (2011) studied multivariate random forests using the approach of Segal (1992). In many studies, multiple outcomes of mixed types (continuous and categorical) are measured. Dine et al. (2009) proposed a multivariate tree that can accommodate multiple outcomes of mixed types. Barcena and Tusell (2004) proposed a method that uses univariate trees, one for each response, to compute multivariate predictions. This approach does not build a single multivariate tree but, could be used to compute predictions with mixed outcomes.

There are many situations, however, where one or more responses are missing for a number of subjects. Many general approaches were proposed to deal with missing data and the monograph by Little and Rubin (2002) provides a good entry point in the literature. So far, the literature on the treatment of missing values with tree–based methods has focused almost entirely on the case of missing covariates and univariate trees. In this case, the classical approach is to use surrogate splits (Breiman et al., 1984). Feelders (1999) found empirically that multiple imputation is preferable to surrogate splits. Kim and Yates (2004) compared seven ways to handle missing values including surrogate splits and mean imputation and concluded that no method outperforms the others in all cases. Saar–Tsechansky and Provost (2007) studied the case where covariates are missing at prediction time and concluded that a reduced feature approach is preferable to imputation. Such a method is based on many different models that use only the covariates available for the case to be predicted. As such, they can be expensive (in terms of computational time and storage) and this is why they also proposed a hybrid method. Twala, Jones and Hand (2008) proposed a method that generalizes the idea of treating missing as a new category which works for categorical and continuous covariates. Their method exhibited a good performance across many scenarios including many missing data patterns. Twala (2009) compared seven approaches and concluded that listwise deletion is inferior to the other six methods and that multiple imputation is preferable. Ding and Simonoff (2010) made a systematic investigation of the different missingness patterns. One of their conclusions is that creating a new class that represents the missing values is the best approach when the test set contains missing values.

The previous works mentioned above were aimed at the case of missing covariates. Segal (1992) was the only one to propose a method to handle missing responses in the case of longitudinal continuous outcomes. He generalized the EM algorithm under some strong assumptions: a first-order autoregressive (AR1) covariance structure, a consecutive sequence of missing values, and a missing completely at random (MCAR) mechanism.

The goal of this paper is to propose and study a general multivariate random forest method that can handle missing responses under less restrictive assumptions in the context of multiple responses of mixed types. This method has a multiple imputation flavor since an imputation is performed for each bootstrap sample used to build the individual trees during the forest construction. The splitting criterion of Dine et al. (2009) is generalized so that observation weights can be incorporated. This allows down–weighting of observations that require imputations. The rest of the paper is organized as follows. The method is described in detail in Section 2. The results of a simulation study is presented in Section 3 followed by concluding remarks in Section 4.

2 Multivariate forest with imputed responses

In this section, we describe a general algorithm for predicting multivariate responses of mixed types when some responses may be missing.

2.1 Data setup

The data setup is the same as in Dine et al. (2009). Data is available for a sample $\mathbf{Z}_i = (\mathbf{y}_i, \mathbf{f}_i, \mathbf{x}_i)$, $i = 1, \ldots, N$, where, for subject i, $\mathbf{y}_i = (y_{1i}, \ldots, y_{pi})$ are the p continuous responses, $\mathbf{f_i} = (f_{1i}, \ldots, f_{ui})$

are the u categorical responses and $\mathbf{x_i}$ are the covariates (of any types). Note that $\min(p, u) \geq 0$ and $\max(p, u) > 0$, i.e., we may have only continuous or categorical responses. The goal is to build a forest of trees to predict the responses using the covariates.

The tree method developed in Dine et al. (2009) does not allow missing data in the responses. In this paper, we allow that an observation may have one or more missing response(s). However, an observation with no observed response at all would be discarded right from the start.

2.2 Multivariate random forest with responses imputation

The proposed generic random forest algorithm can be described in the following steps:

1. For $t = 1, \ldots, T$
   (a) Draw a bootstrap sample $\mathbf{Z}^*_{t1}, \ldots, \mathbf{Z}^*_{tN}$ from the original data.
   (b) Impute the missing responses in the bootstrap sample with a strategy that uses only the responses.
   (c) Assign a weight w_{ti} to $\mathbf{Z}^*_{ti}$; $i = 1, \ldots, N$.
   (d) Grow a multivariate tree with the weighted GLOM splitting rule (see Section 2.3 below) using the assigned weights.
2. Compute predictions by aggregating the predictions of the T individual trees (see Section 2.4 below).

This algorithm is very general. The weights in step c) can be defined in many ways. A natural way would be to give less weights to observations that required imputations. The weighted GLOM splitting rule would then give less weight to these incomplete cases in the tree building process but also in the calculation of the predictions as described in Sections 2.3 and 2.4.

The specific implementation of this algorithm used for the simulation study in this paper is described in Section 3.2.

2.3 Weighted GLOM splitting rule and tree

Assume we have N_F observations in the current father node and wish to split it into two children nodes, the left and right nodes. In the GLOM approach, the vector $\mathbf{f}_i$ of categorical responses is casted into a single variable c_i that can take $S_F \leq \prod_{j=1}^{u} s_j$ possible values, called states, where s_j is the number of different values of the categorical outcome f_j. Only the states observed in the node to be splitted are used and, consequently, the number of states S_F may vary from node to node. The state indicator c_i then takes any integer values from 1 to S_F.

We modify the two–node GLOM splitting criterion of Dine et al. (2009) to account for imputed responses. This is done through the incorporation of weights associated to each data point. We only present the final splitting criterion here but more details are given in the Appendix. Let J_F denote the subset of indices from $\{1, \ldots, N\}$ of the observations that are in the father node. We first standardize the weights such that they sum to N_F, i.e., $\sum_{i \in J_F} w_i = N_F$. In the following, the superscripts "L" and "R" will indicate quantities computed for the observations that are in the left and right nodes for a given candidate split. Let J^L_k (J^R_k) denote the subset of indices that are in the left (right) node and in state k. Then

$$n^L_{wk} = \sum_{i \in J^L_k} w_i \quad \text{and} \quad n^R_{wk} = \sum_{i \in J^R_k} w_i \quad ; \quad k = 1, \ldots, S_F$$

are the weighted total number of observations (i.e., the sum of the weights) that are in each state for each children nodes. Let also N^L_w (N^R_w) be the sum of the weights in the left (right) node. The splitting criterion is given by:

$$ll_w = \sum_{k=1}^{S_F} (n^L_{wk} \ln(\hat{\pi}^L_{wk}) + n^R_{wk} \ln(\hat{\pi}^R_{wk})) - \frac{N_F}{2} \ln |\hat{\mathbf{\Sigma}}_{w2}| - \frac{pN_F}{2} \ln(2\pi) - \frac{pN_F}{2} \tag{1}$$

where

$$\hat{\pi}^L_{wk} = \frac{n^L_{wk}}{N^L_w} \quad \text{and} \quad \hat{\pi}^R_{wk} = \frac{n^R_{wk}}{N^R_w} \quad \text{for} \quad k = 1, \ldots, S_F, \tag{2}$$

$$\hat{\boldsymbol{\mu}}^L_{wk} = \frac{1}{n^L_{wk}} \sum_{i \in J^L_k} w_i \mathbf{y}_i \quad \text{and} \quad \hat{\boldsymbol{\mu}}^R_{wk} = \frac{1}{n^R_{wk}} \sum_{i \in J^R_k} w_i \mathbf{y}_i \quad \text{for} \quad k = 1, \ldots, S_F \tag{3}$$

and

$$\hat{\boldsymbol{\Sigma}}_{w2} = \frac{1}{N_F} \sum_{k=1}^{S_F} \{ \sum_{i \in J^L_k} w_i (\mathbf{y}_i - \hat{\boldsymbol{\mu}}^L_{wk})(\mathbf{y}_i - \hat{\boldsymbol{\mu}}^L_{wk})' + \sum_{i \in J^R_k} w_i (\mathbf{y}_i - \hat{\boldsymbol{\mu}}^R_{wk})(\mathbf{y}_i - \hat{\boldsymbol{\mu}}^R_{wk})' \}. \tag{4}$$

$\hat{\pi}^L_{wk}$ ($\hat{\pi}^R_{wk}$) is the weighted maximum likelihood estimate (WMLE) of the probability that an observation in the left (right) node is in state k. $\hat{\boldsymbol{\mu}}^L_{wk}$ ($\hat{\boldsymbol{\mu}}^R_{wk}$) is the WMLE of the mean vector of the continuous responses in the left (right) node for state k. Finally, $\hat{\boldsymbol{\Sigma}}_{w2}$ is the WMLE of the common, across the two nodes and the S_F states, covariance matrix of the continuous responses. In fact, this criterion is the observed likelihood of a weighted two–node GLOM model. See the Appendix for more details. In practice, the last two terms $(-pN_F \ln(2\pi)/2 - pN_F/2)$ are constant for all candidate splits and can be omitted. With equal weights, i.e., when $w_i \equiv 1$, the splitting criterion (1) reduces to the one given in equation (4) of Dine et al. (2009).

A tree can be built by maximizing the splitting criterion from the root node (with all the observations) and by repeating the process recursively for the children nodes until a stopping criterion is reached.

2.4 Final prediction from a forest

Assume we need a prediction for a new observation. Let $\hat{\mathbf{y}}_t$ denote the prediction of the continuous responses for the t^{th} tree in the forest as given by the weighted mean (3) in the terminal node in which the new observation falls. Then the final forest prediction is given by

$$\hat{y}_{RF} = \sum_{t=1}^{T} \hat{\mathbf{y}}_t / T,$$

that is the unweighted mean of the individual weighted mean predictions. For the j^{th} categorical response f_j, $j = 1, \ldots, u$, let $\hat{\pi}_{jkt}$ denote the estimated probability that $P(f_j = k)$ for the t^{th} tree in the forest. This probability is estimated as the weighted proportion of observations for which $f_j = k$ in the terminal node in which the new observation falls. The final forest estimated probabilities are given by

$$\hat{\pi}_{jk} = \sum_{t=1}^{T} \hat{\pi}_{jkt} / T.$$

The final forest prediction is then given by the class with the largest estimated probability, i.e.,

$$\hat{f}_{jRF} = \underset{k}{\text{argmax}} \ \hat{\pi}_{jk} \quad \text{for} \quad j = 1, \ldots, u.$$

3 Simulation study

In this section, we study empirically the performance of the multivariate random forest algorithm with responses imputed under two types of missing data generation processes, missing completely at random (MCAR) and missing at random (MAR), and under three levels of missingness, 25%, 50% and 75%. This produces six different scenarios. Four different weighting strategies are investigated. They are described in Section 3.2. A forest using only the complete cases, i.e., only the observations with no missing responses, will serve as the benchmark.

3.1 Simulation design

The simulation design has six responses in all, three continuous and three binary. The responses are related to the covariates through the tree structure defined in Figure 1. Seven covariates are generated independently according to the uniform distribution in the interval (0,10) but only four are actually related to the responses. The three others are noise covariates. To generate the responses, an observation is sent down the tree according to the values of the covariates. Then, six equicorrelated ($\rho = 0.3$) variates are first generated as multivariate normal with mean vectors (node specific) given in Table 1 and unit variance. The first three variates form the three continuous responses (Y_1, Y_2, Y_3). The last three variates are dichotomized according to whether they are greater (1) or less (0) than 0 and form the three binary responses (Y_4, Y_5, Y_6).

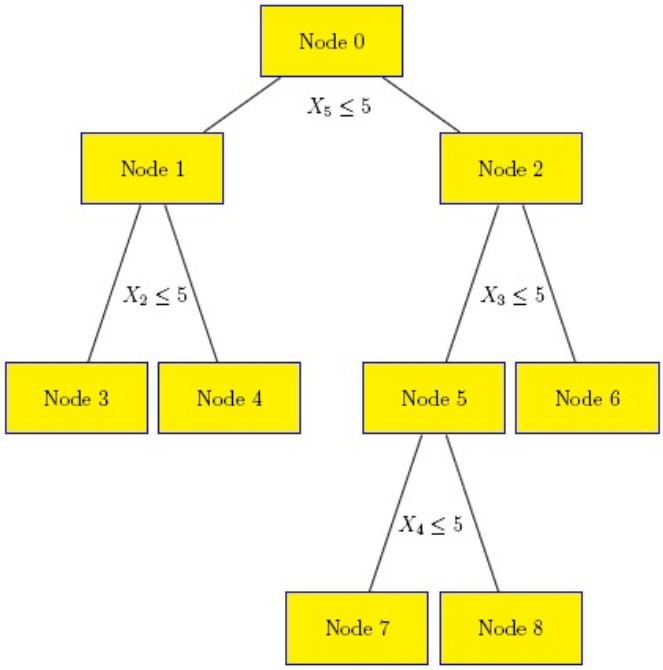

Figure 1: Tree structure used for generating data in the simulations

Table 1: Mean vector of the terminal nodes of the tree in Figure 1

Node	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
3	1	2	3	-0.25	-0.25	-0.52
4	1.5	2.5	4	0.67	0.39	0.52
6	1	2.4	3	-0.52	-0.25	-0.67
7	0.5	3.5	2.5	-0.52	-0.25	-0.67
8	2.5	3	3.5	0.39	0.13	-0.67

A training data set of size 500 was generated to build the forests and the prediction accuracies are estimated with an independent test set of size 10,000. This whole process was repeated for 500 simulation runs.

We introduce missing responses by removing some of the responses in the training data set. Three levels of missingness are used: 25%, 50% and 75%. For instance, 75% of missingness means that the probability that an observation has at least one missing response is 75%. Hence, only about 25% of the observations are complete. The benchmark method would use only these complete observations to build the forest.

To create the MCAR setting, we first choose randomly the number of missing responses in each observation. We limited this number to a maximum of three for each observation. Once the number of missing responses is chosen for an observation, then the missing responses are chosen randomly among the six original responses. To obtain levels of missingness p=25%, 50% and 75%, the number of missing responses, 0, 1, 2 or 3, are chosen randomly with probabilities $1-p$, $p/3$, $p/3$ and $p/3$.

To create a MAR setting, we used a model in which the probability that a response (continuous or binary) is missing depends on the values of the other responses (continues and binary) for the same observation. More precisely, the probability that a response Y_j, $j = 1, \ldots, 6$, is set to be missing is given by the following logistic model:

$$\text{logit}[P(Y_j \text{ is missing})] = \beta_0 + \beta_1 \left[\sum_{i \neq j} Y_i \right]$$

The parameters β_0 and β_1 are set to get the desired level of missingness. We set, $\beta_1 = 0.2$ in all cases and $\beta_0 = 3.10$, 2.14 and 1.32 to obtain levels of missingness of 25%, 50% and 75%, respectively. Table 2 presents the observed proportion, across the 250,000 observations from all training data sets (500 training data sets of size 500), of observations with different numbers of missing responses.

Table 2: Proportion of observations with different numbers of missing responses in the MAR scenario

	Level of missingness		
# Missing	25%	50%	75%
0	74.4%	50.4 %	24.2 %
1	23 %	34.8 %	36.2 %
2	2.2 %	12.2 %	23.4 %
3	0.4 %	2.2 %	12.6 %
4	0 %	0.4 %	2.8 %
5	0 %	0 %	0.8 %

3.2 Forest implementation

The tree building and forest aggregation of Section 2.2 were coded in Ox (Doornik, 2002) and the data generation and imputation were implemented in SAS (SAS Institute, 2010). Specifically, we used the default settings of the MI procedure in SAS to impute the incomplete cases, which is based on the Markov Chain Monte Carlo (MCMC) algorithm. This method produces continuous imputed values. Since three of the six outcomes are binary, we followed an approach proposed by Schafer (1997) that treats a binary variable as continuous in the imputation process and subsequently round each imputed value. More precisely, each binary outcome was treated as a continuous variable by using the default options of MI. They were then rounded by assigning a "1" if the imputed value is greater or equal to 0.5 and a "0" otherwise.

In this paper, we investigate four weighting schemes.

Scheme 1: We assign a weight of one for complete and imputed observations. Thus $w_{ti} = 1$ for all observations regardless of the missing values.

Scheme 2: We assign a weight of 1 to an observation without any missing response and a weight uniformly chosen between 0 and 1 is assigned to an observation with at least one missing response. More precisely, for a given bootstrap sample, $w_{ti} = 1$ for complete response cases and $w_{ti} = w_t$ for incomplete response cases where $w_t \sim U(0, 1)$. Hence, all incomplete cases receive the same weight regardless of the number of missing responses. However, w_t itself varies with the bootstrap sample.

Scheme 3: We assign a weight equal to the proportion of complete responses for the observation. Thus, a complete observation receives a weight of 1 and, for example, an observation with two missing responses out of six responses receives a weight of 2/3.

Scheme 4: We assign a weight of 1 to complete observations. An incomplete observation receives a weight uniformly chosen between 0 and the proportion of complete responses for the observation. This weight varies also with the bootstrap sample.

The weighting schemes 1 and 3 are deterministic. Schemes 2 and 4 add a random element in order to investigate if this can improve the predictions.

3.3 Results

We assess the predictive performance with the predictive mean squared error ($PMSE$) for the continuous outcomes and the predictive misclassification rate (PMR) for the binary outcomes. Figures 2 to 4 summarize the results for the 500 simulation runs. In Figures 2 (MCAR) and 3 (MAR), each plot represent one of the outcome. The upper plots are for the continuous outcomes and the lower plots are for the binary outcomes. Each plot contains the results for both the benchmark complete cases forest and the imputed forests (four weighting schemes) as the level of missingness varies. Looking at the MCAR setting in Figure 2 we see that the performance of the two methods, with and without imputed responses, is negatively correlated with the level of missingness. As the level of missingness increases, the box plots shift higher meaning that the performance of the learning method depends on the quantity of data available. We also see that for any given level of missingness, the imputed forest is better than the complete case forest for all responses, and regardless of the weighting scheme. The performances of the four weighting schemes are very similar. Schemes 1 and 3 seem slightly better but not enough to declare a clear winner.

It is also apparent that the gap between the two methods (with and without imputed responses) increases as the level of missingness increases. All these remarks also apply for the MAR setting in Figure 3. Consequently, as the level of missingness increases, the proposed imputed forest loses its predictive accuracies at a slower pace compared to the complete cases forest. It seems however that the advantage of the imputed forest is even larger in the MAR setting. To see this more clearly, Figure 4 depicts the differences between the two approaches, using weighting scheme 3, as the level of missingness varies. We can see even more clearly that the gap between the two approaches increases as the level of missingness increases and that the increase rate is more pronounced in the MAR setting.

4 Concluding remarks

In this paper, we introduced a very general method to handle missing values in the responses when multiple responses of possible mixed types are observed. The method is very general and could be used with different imputation strategies and any multivariate tree algorithm but, we selected the one of Dine et al. (2009) because it can accommodate responses of mixed types. One particular implementation of the method along with four different weighting schemes was investigated with a simulation study with MCAR and MAR missing responses. Compared to a complete case strategy, the results indicate that fitting a multivariate random forest with imputed responses enhances predictions regardless of the type of responses and weighting strategy. Moreover, the gap between the two approaches increases as the percentage of incomplete observations increases and is more pronounced in the MAR setting. Two of the weighting strategies produced slightly better predictions. They are: i) giving an equal weight to all observations regardless of the missing values, and ii) giving a weight equal to the proportion of complete responses for the observation.

The goal of this paper was to introduce the method and show its potential. Many aspects deserve future investigations. One possibility would be to study more thoroughly the weighting scheme. For instance, investigating other weighting schemes and the sensitivity of the method to the weighting scheme could be of interest. Moreover, missing could occur in both responses and covariates and missing covariates could also be present at prediction time. Investigating this more general framework is also worthy of future work.

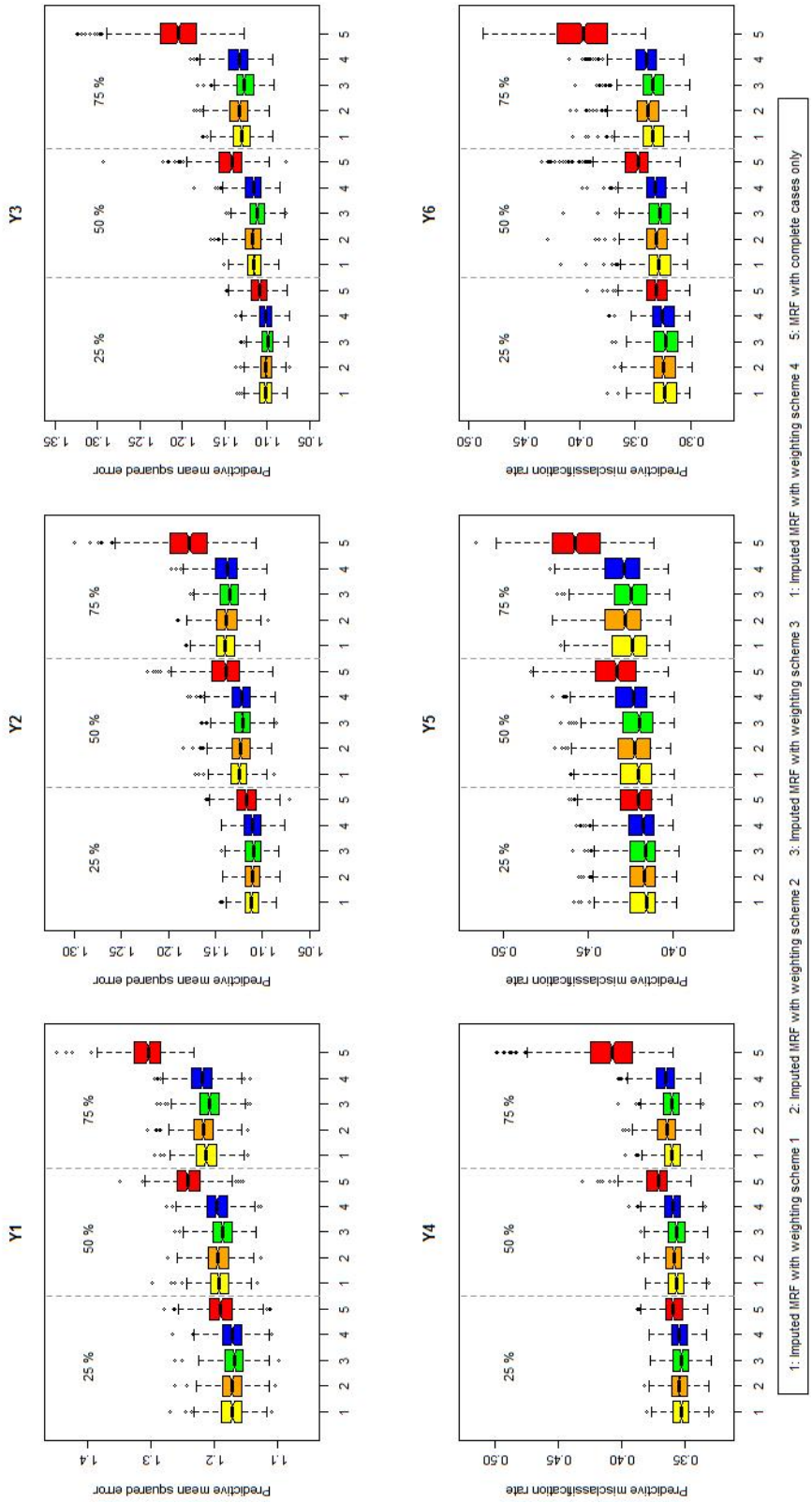

Figure 2: MCAR setting: Distribution of ($PMSE$) and (PMR) for 3 continuous and 3 categorical outcomes for 3 missing levels: 25, 50 and 75%

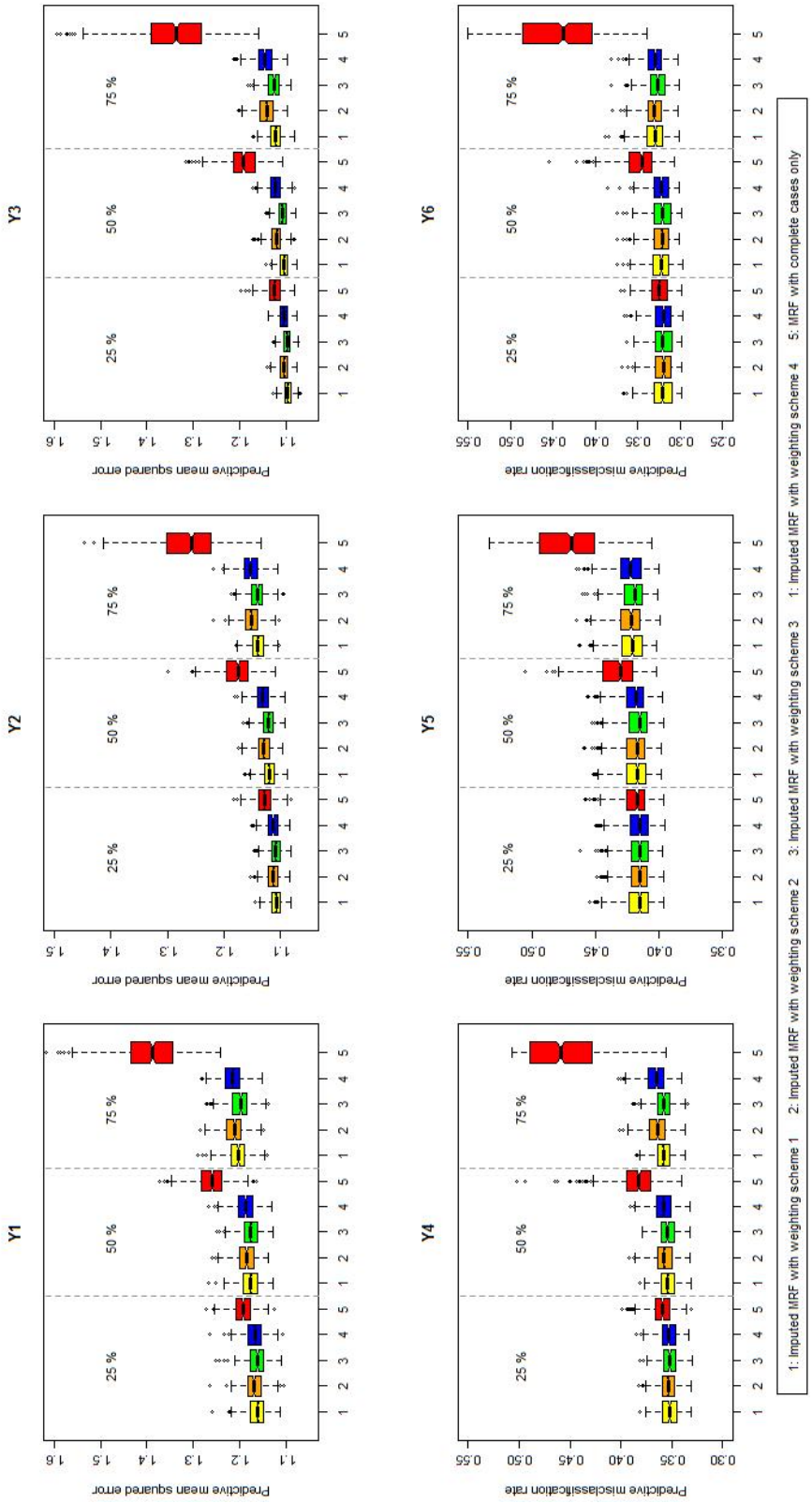

Figure 3: MAR setting: Distribution of $(PMSE)$ and (PMR) for 3 continuous and 3 categorical outcomes for 3 missing levels: 25, 50 and 75%

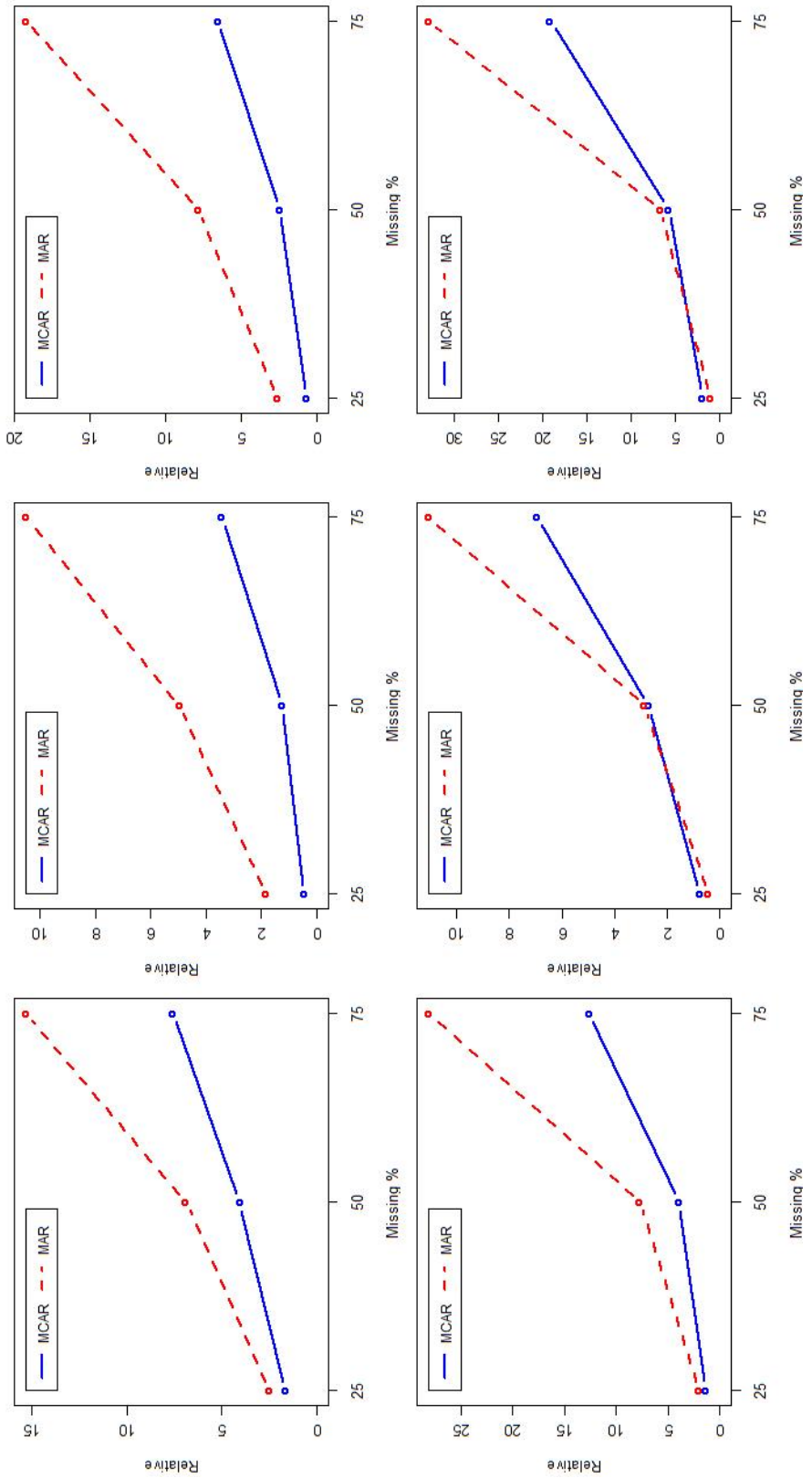

Figure 4: Median differential, in percentage, between the MRF with imputed responses (weighting scheme 3) and the MRF with complete cases in the MCAR and MAR settings

Appendix

Description of the weighted two nodes GLOM behind the splitting rule

In this appendix, we describe the model behind the splitting rule used to build the multivariate trees with weights. This model is an extension of the two nodes general location model (GLOM) used in Dine et al. (2009). See Olkin and Tate (1961) for a description of the original GLOM model. Refer to Sections 2.1 and 2.3 for the notation. We begin by describing the basic GLOM model. Let $(\mathbf{y}, \mathbf{f})$, where $\mathbf{y} = (y_1, \ldots, y_p)$ are the p continuous responses, $\mathbf{f} = (f_1, \ldots, f_u)$ are the u categorical responses. The vector $\mathbf{f}$ is casted into a single variable c that can take S possible values called states such that each state corresponds to one unique combination of the categorical responses. The random variable c can take any integer values from 1 to S. The vector $(\mathbf{y}, \mathbf{f})$ is said to come from a GLOM if

1. c is a multinomial random variable with probability vector $(\pi_1, \ldots, \pi_S)$ ($\pi_k \geq 0$ and $\sum_{k=1}^{S} \pi_k = 1$).
2. The conditional distribution of $\mathbf{y}$ given $c = k$ is $N_p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$.

Thus, in the basic GLOM, $\boldsymbol{\Sigma}$ is the common (across states) covariance matrix and the $\boldsymbol{\mu}_k$ are the state–specific mean vectors of the continuous responses.

The two nodes GLOM arises after when the observations are partitioned into two groups. In the present context, this occurs when a candidate split partitions the observations belonging to a father node into two children nodes and this is why we will use the term "node" instead of "group" and denote the two nodes by using the superscripts "L" and "R" for left and right. With this model, we have node–specific probability vectors $(\pi_1^g, \ldots, \pi_S^g)$ and mean vectors $\boldsymbol{\mu}_k^g$, $g = L, R$, but a common across states and nodes covariance matrix $\boldsymbol{\Sigma}_2$. The observed log–likelihood for a sample of the two nodes GLOM is used as the splitting rule in Dine et al. (2009); see (4) on page 3798 of that article. In this paper, we add observation weights to account for imputations. This produces the weighted two nodes GLOM. It is then straightforward to see that the weighted maximum likelihood estimates and observed log–likelihood of a sample of the weighted two nodes GLOM are given in (1) to (4).

References

Barcena, M.J. and Tusell, F. (2004). Fitting Multivariate Responses Using Scalar Trees. *Statistics & Probability Letters*, **69**, 253–259.

Breiman, L. (2001). Random Forests. *Machine Learning*, **45**, 5–32.

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and Regression Trees*. Belmont, CA: Wadsworth International Group.

Ciampi, A., du Berger, R., Taylor, H.G. and Thiffault, J. (1991). RECPAM: a Computer Program for Recursive Partition and Amalgamation for Survival Data and Other Situations Frequently Occurring in Biostatistics. III. Classification According to a Multivariate Construct. Application to Data on Haemophilus Influenzae Type b Meningitis. *Computer Methods and Programs in Biomedicine*, **36**, 51–61.

Dine, A., Larocque, D., Bellavance, F. (2009). Multivariate Trees for Mixed Outcomes. *Computational Statistics & Data Analysis*, **53**, 3795–3804.

Ding, Y. and Simonoff, J.S. (2010). An Investigation of Missing Data Methods for Classification Trees Applied to Binary Response Data. *Journal of Machine Learning Research*, **11**, 131–170.

Doornik, J.A. (2002). *Object-Oriented Matrix Programming Using Ox*. London: Timberlake Consultants Press and Oxford: www.doornik.com.

Feelders, A. (1999). Handling Missing Data in Trees: Surrogate Splits or Statistical Imputation? *Principles of Data Mining and Knowledge Discovery*, **1704**, 329–334.

Kim, H. and Yates, S. (2004). Missing Value Algorithms in Decision Trees. In H. Bozdogan, editor, *Statistical Data Mining and Knowledge Discovery*, Chapman & Hall/CRC, pages 155–172.

Lee, S.K. (2005). On Generalized Multivariate Decision Tree by Using GEE. *Computational Statistics & Data Analysis*, **49**, 1105–1119.

Little, R.J.A., Rubin, D.B. (2002). *Statistical Analysis with Missing Data*. New Jersey: John Wiley.

Olkin, I. and Tate, R.F. (1961). Multivariate Correlation Models with Mixed Discrete and Continuous Variables. *Annals of Mathematical Statistics*, **32**, 448–465.

Rokach, L. (2009). Taxonomy for Characterizing Ensemble Methods in Classification Tasks: A Review and Annotated Bibliography. *Computational Statistics & Data Analysis*, **53**, 4046–4072.

Saar–Tsechansky, M. and Provost, F. (2007). Handling Missing Values when Applying Classification Models. *Journal of Machine Learning Research*, **8**, 1625–1657.

SAS Institute Inc. (2010). Version 9.2 of the SAS System for Windows. Cary, NC: USA.

Schafer, J.L. (1997). *Analysis of Incomplete Multivariate Data*. London: Chapman and Hall.

Segal, M.R. (1992). Tree-Structured Methods for Longitudinal Data. *Journal of the American Statistical Association*, **87**, 407–418.

Segal, M. and Xiao, Y. (2011). Multivariate Random Forests. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, **1**, 80–87.

Siciliano, R. and Mola, F. (2000). Multivariate Data Analysis and Modeling Through Classification and Regression Trees. *Computational Statistics & Data Analysis*, **32**, 285–301.

Siroky, D.S. (2009). Navigating Random Forests and Related Advances in Algorithmic Modeling. *Statistics Surveys*, **3**, 147–163.

Twala, B. (2009). An Empirical Comparison of Techniques for Handling Incomplete Data Using Decision Trees. *Applied Artificial Intelligence*, **23**, 373405.

Twala, B., Jones, M. C. and Hand, D.J. (2008). Good Methods for Coping with Missing Data in Decision Trees. *Pattern Recognition Letters*, **29**, 950–956.

Verikas, A., Gelzinis, A. and Bacauskiene, M. (2011). Mining Data With Random Forests: A Survey and Results of New Tests. *Pattern Recognition*, **44**, 330–349.

Zhang, H. (1998). Classification Trees for Multiple Binary Responses. *Journal of the American Statistical Association*, **93**, 180–193.