TREE-BASED MULTIPLE IMPUTATION METHODS

Michael Dellermann, Anatol Sluchych, and Jonah Wermter

Freie Universität Berlin

1. Motivation

- Standard MICE approach: conditional models to be specified for *all* variables with missing data
- Still may fail to capture interactive and nonlinear relations among variables as well as non-standard distributions
- Classification and regression trees (CART) *automatically* capture interactions, nonlinear relations, and complex distributions with no parametric assumptions or data transformations needed (Burgette & Reiter 2010)
- Implementation in R: *mice* and *miceRanger* packages

2. Tree-based methods

- seek to approximate conditional distribution of univariate outcome from multiple predictors
- partition the predictor space so that subsets of units formed by the partitions have relatively homogeneous outcomes
- partitions are found by recursive binary splits of the predictors
- series of splits can be effectively represented by a tree structure, with leaves corresponding to the subsets of units
- values in each leaf represent the conditional distribution of the outcome for units in the data with predictors that satisfy the partitioning criteria that define the leaf

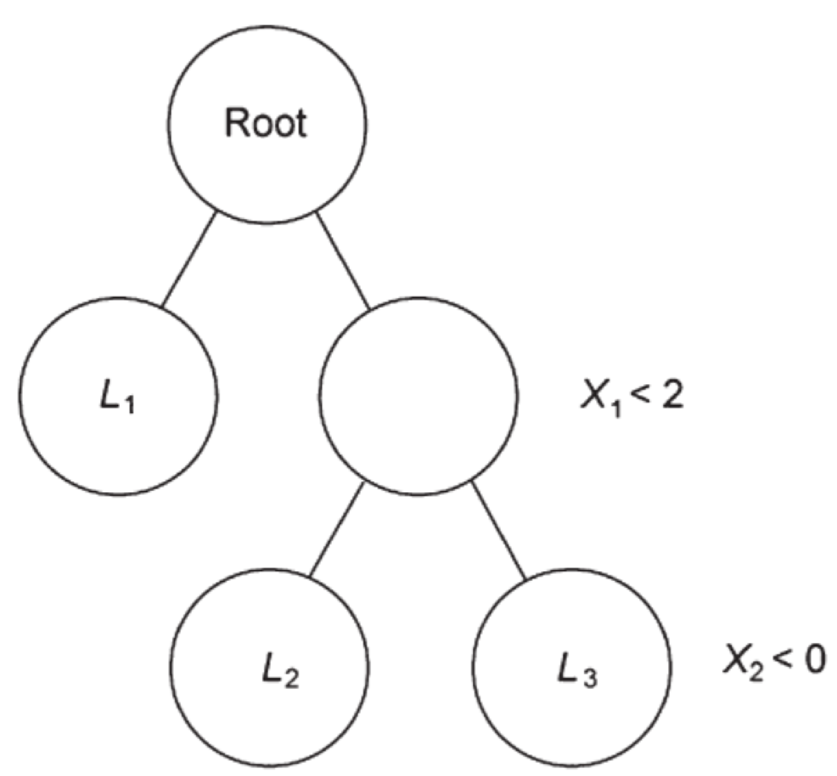

Fig. 1: Example of a tree structure. Source: Burgette & Reiter (2010)

Disadvantages relative to parametric models:

- decreased efficiency when the parametric models are adequate
- discontinuities at partition boundaries
- categorical predictors with many levels can cause computational difficulties

3. Imputation algorithm

Let Y be $n \times p$ the data matrix arranged as $Y = (Y_p, Y_c)$, where

- Y_p consists of p_1 *partially observed* columns, such that moving from left to right, the number of missing elements in each column is nondecreasing
- Y_c remaining completely observed columns
- Y_{obs} set of observed and Y_{mis} set of missing elements

4-steps algorithm:

1. Initial values for the missing values filled in as follows:
   (a) Define a matrix Z equal to Y_c
   (b) Impute missing values in Y_i, where $i = 1, ...p_1$, using CART on Z and append the completed version of Y_i to Z prior to incrementing i
2. Replace the originally missing values of Y_i, where $i = 1, ...p_1$, with CART on Y_{-i}
3. Repeat l times step 2
4. Repeat steps 1–3 m times and obtain m imputed sets.

4. Comparison mice/miceRanger packages

- both implement Stef van Buuren's Multivariate Imputation by Chained Equations
- *mice* supports variety of imputation methods, *miceRanger* only randomForest
- *mice* uses common R packages "rpart" and "randomForest" to implement tree based imputation methods (van Buuren 2023)
- *miceRanger* uses the "ranger"-package instead, which claims to be faster and more efficient with medium and large data sets (Wilson 2022)
  - ⇒ core functions written in C++ (faster than R, compiled vs. interpreted code) (Wright und Ziegler 2017)
  - ⇒ supports parallel computing (Wright und Ziegler 2017; Wright 2023)

5. Empirical simulation study

Objective:

- evaluate efficacy of tree-based imputation methods on missing data
- compare *mice* package methods against the extended methods in *miceRanger*

Empirical data set:

- RAND's Health Insurance Experiment

Monte Carlo simulation:

- simulations (R): 1000 cycles to ensure robustness
- multiple imputations (M): 5 imputations to estimate variability
- sample size (n): 2000 cases to ensure generalizability.
- iterations (niter): 10 iterations for chained equations.
- random forest trees (nrtree): 10 trees in each RandomForest model for depth

Comparison Metrics: determine the most accurate and efficient imputation method that best reconstructs true values while minimizing systematic errors

- bias: deviation of imputed values from true values.
- mean squared error: average squared difference between the imputed and true values.
- coverage: proportion of times true values fall within the calculated confidence intervals.

6. Results

Metric	Method	Bias	MSE	Coverage
mean(age)	BD	NA	NA	NA
mean(age)	CC	NA	NA	NA
mean(age)	CART	NA	NA	NA
mean(age)	RandomForest	NA	NA	NA
mean(age)	miceRanger	NA	NA	NA
mean(educ)	BD	NA	NA	NA
mean(educ)	CC	NA	NA	NA
mean(educ)	CART	NA	NA	NA
mean(educ)	RandomForest	NA	NA	NA
mean(educ)	miceRanger	NA	NA	NA
ρ(mdvis, hltg)	BD	NA	NA	NA
ρ(mdvis, hltg)	CC	NA	NA	NA
ρ(mdvis, hltg)	CART	NA	NA	NA
ρ(mdvis, hltg)	RandomForest	NA	NA	NA
ρ(mdvis, hltg)	miceRanger	NA	NA	NA
reg.(mhi)	BD	NA	NA	NA
reg.(mhi)	CC	NA	NA	NA
reg.(mhi)	CART	NA	NA	NA
reg.(mhi)	RandomForest	NA	NA	NA
reg.(mhi)	miceRanger	NA	NA	NA

7. Conclusion

References

Burgette, Lane F, and Jerome P Reiter. "Multiple imputation for missing data via sequential regression trees". *American journal of epidemiology* 172, no. 9 (2010): 1070–1076.

Hastie, Trevor, et al. *The elements of statistical learning: data mining, inference, and prediction.* Vol. 2. Springer, 2009.

Marvin, N. Wright, and Andreas Ziegler. "ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R". *Journal of Statistical Software* 77, no. 1 (2017). `https://doi.org/10.18637/jss.v077.i01`.

van Buuren, Stef. *Package 'mice'.* Last updated on 05.06.2023, last checked on 28.12.2023. `https://cran.r-project.org/web/packages/mice/mice.pdf`.

Wilson, Sam. *Package 'miceRanger'.* Last updated on 13.10.2022, last checked on 28.12.2023. `https://cran.r-project.org/web/packages/miceRanger/miceRanger.pdf`.

Wright, Marvin N. *Package 'ranger'.* Last updated on 12.11.2023, last checked on 23.12.2023. `https:`