Easily Implemented, Nonparametric Synthesizers Based on Algorithmic Methods from Computer Science

Jerry Reiter
Department of Statistical Science
Duke University
jerry@stat.duke.edu

General setting: Challenging synthesis task

-- Thousands of units, dozens of variables.
-- Numerical and categorical data.
-- Skewed or multi-modal distributions.
-- Complicated relationships.
-- Many public uses.
-- Intense synthesis required.

Aside: these are not necessary for synthetic data approaches to be useful.

Key feature of challenging synthesis tasks

-- Analyses primarily reflect features built into agency's synthesis models.

-- Agency works hard to generate synthetic data that are inference-valid for many analyses.

A pie-in-the-sky vision for synthetic data generators

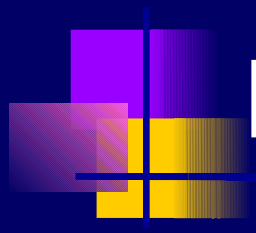
An ideal synthetic data generator would

-- preserve as many relationships as possible while protecting confidentiality,

-- handle diverse data types,

-- be computationally feasible for large data,

-- be easy to implement with little tuning by the agency.



Research proposal

Build synthesizers using algorithmic methods from computer science.

Two approaches considered here:

-- Regression trees (CART).
-- Random forests (RF).

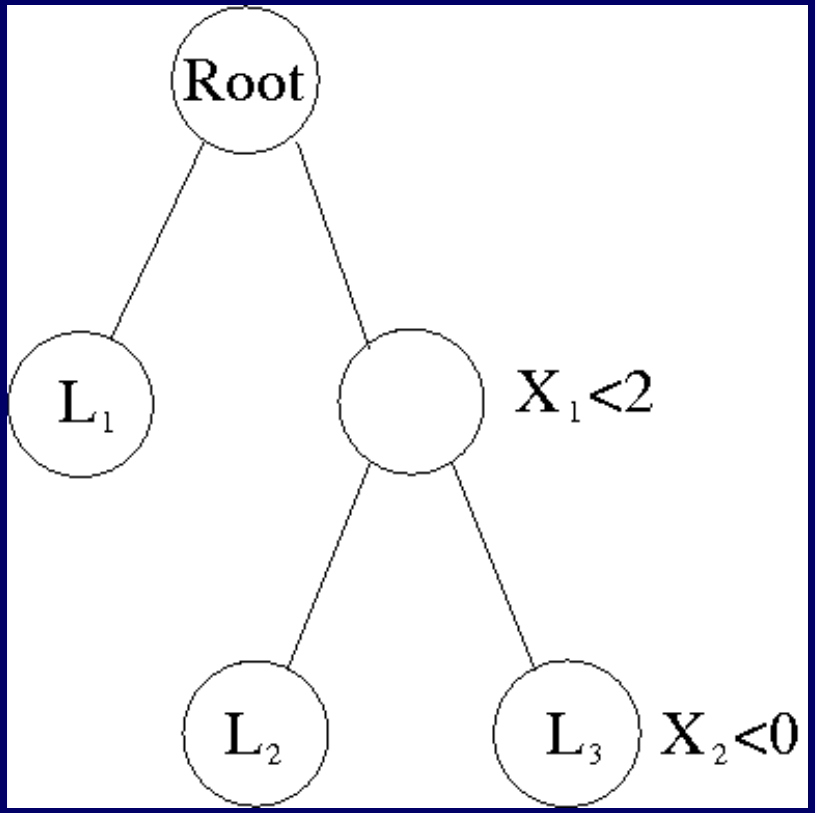
Overview of CART

Goal: Describe f (Y | X).

-- Partition X space so that subsets of units formed by partitions have relatively homogenous Y.

-- Partitions from recursive binary splits of X.

-- Free routines in R.



CART for synthesis

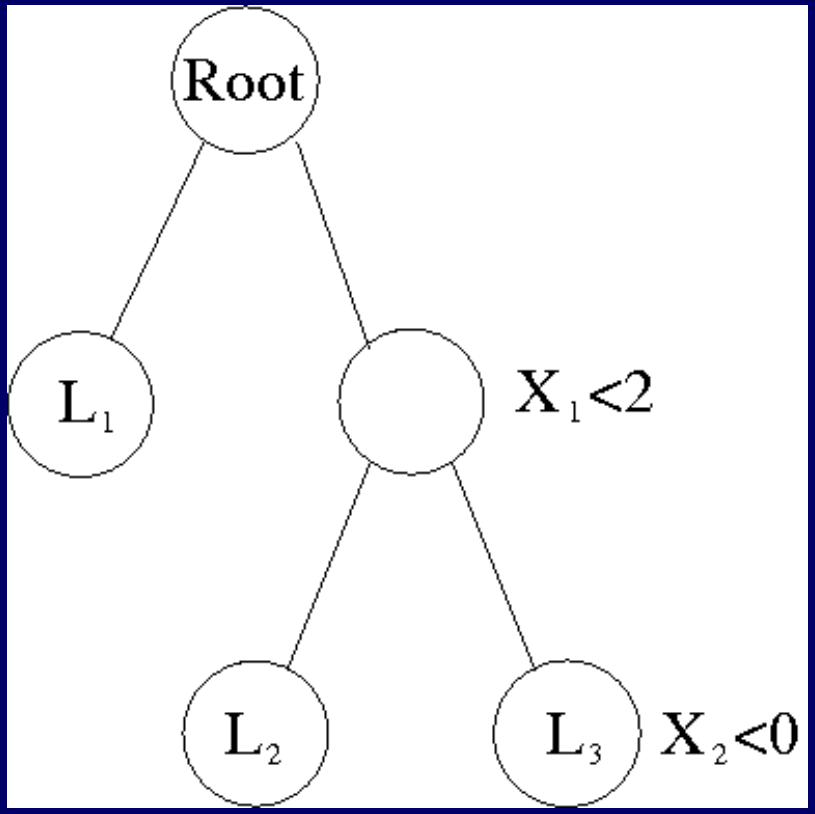
Goal: Synthesize Y | X.

-- Grow maximum tree.

-- Prune for confidentiality.

-- For any X, trace down tree until reach appropriate leaf.

-- Draw Y from Bayes bootstrap (or smoothed density estimate).



CART for synthesis

Synthesize with chained imputations akin to Raghunathan *et al.* (2001, *Surv. Methodol.*).

a) using genuine data, run CART for each variable conditional on others as appropriate.

b) generate new values for each variable using already synthesized data to trace down trees.

Reiter (2005, *JOS*) discusses order of synthesis.

Illustration of CART synthesis

- 10,000 household heads, March 2000 CPS.
- Age, race, sex, marital status, education, alimony payments, child support payments, SS payments, income, property taxes.
- Synthesize all values of marital status, race, and sex. Leave other variables at original values.

Illustration of CART synthesis

- Make 5 synthetic datasets using CART.
- Obtain confidence intervals using methods in Reiter (2003, *Surv. Methodol.*).
- Compare inferences for regression coefficients in original and synthetic data.
- Table 1: overall reasonable inferences. Problems arise with small sub-pop's.

Random forests: Overview

Based on collections of CARTs

a) Pick random subset of data.
b) Pick random subset of variables.
c) Grow maximum tree.
d) Store leaves of tree for each variable.
e) Repeat a – d many times.

Free routines in R (not computationally efficient).

Some studies show RF has better predictive performance than CART.

RF for synthesis (categorical data only so far)

Goal: Synthesize Y | X.

-- Grow large forest (500 trees).

-- For any X, trace down each tree until reach appropriate terminal leaves.

-- Draw Y from multinomial (possibly with Dirichlet prior) using terminal values as the "data."

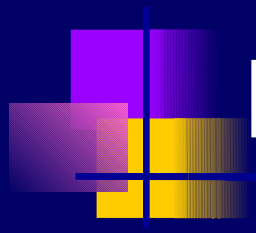
Use a chained imputation algorithm for synthesis.

Illustration of RF for synthesis

-- Same design as in CART illustration.

-- Table 2: good performance overall with some exceptions.

-- Table 3: Hard to distinguish CART or RF as more effective than the other.



Future work

-- Examine performance in larger data sets.

-- Develop RF methods for numerical data.

-- Examine other algorithmic approaches like support vector machines.

Extensions of synthetic data approaches

-- Synthetic PUMS for census or large sample microdata (Drechsler and Reiter, in progress).

-- Two stage synthetic data for reducing risks (Reiter and Drechsler, *Stat. Sinica*, in press).

-- Synthetic data approaches for combining data owned by two agencies (Kohnen and Reiter, 2009, *JRSSA;* Reiter 2009, *ISR*).