Random Forests for Generating Partially Synthetic, Categorical Data

Gregory Caiola[*], **Jerome P. Reiter**[*]

[*]Department of Statistical Science, Duke University, Durham, NC 27708, USA.

E-mail: gregory.caiola@duke.edu; jerry@stat.duke.edu

Abstract. Several national statistical agencies are now releasing partially synthetic, public use microdata. These comprise the units in the original database with sensitive or identifying values replaced with values simulated from statistical models. Specifying synthesis models can be daunting in databases that include many variables of diverse types. These variables may be related in ways that can be difficult to capture with standard parametric tools. In this article, we describe how random forests can be adapted to generate partially synthetic data for categorical variables. Using an empirical study, we illustrate that the random forest synthesizer can preserve relationships reasonably well while providing low disclosure risks. The random forest synthesizer has some appealing features for statistical agencies: it can be applied with minimal tuning, easily incorporates numerical, categorical, and mixed variables as predictors, operates efficiently in high dimensions, and automatically fits non-linear relationships.

Keywords. Confidentiality, Disclosure, Multiple imputation, Nonparametric, Tree.

1 Introduction

Many national statistical agencies, survey organizations, and researchers—henceforth all called agencies—disseminate microdata, i.e. data on individual units, to the public. Wide dissemination of data facilitates advances in science and public policy, enables students to develop skills at data analysis, and helps ordinary citizens learn about their communities. Often, however, agencies cannot release microdata as collected, because doing so could reveal survey respondents' identities or values of sensitive attributes. Stripping unique identifiers like names, tax identification numbers, and exact addresses before releasing data may not suffice to protect confidentiality when quasi-identifiers, such as demographic variables, employment/education histories, or establishment sizes, remain on the file. These quasi-identifiers can be used to match units in the released data to other databases.

Agencies therefore further limit what they release, typically by altering the collected data. Common strategies include recoding variables, such as releasing ages or geographical variables in aggregated categories; reporting exact values only above or below certain thresholds, for example reporting all incomes above 100,000 as "100,000 or more"; swapping data values for selected records, e.g., switch the quasi-identifiers for at-risk records with those for other records to discourage users from matching, since matches may be based on incorrect data; and, adding noise to numerical data values to reduce the possibilities of exact matching on key variables or to distort the values of sensitive variables. See [28] for a general overview of common statistical disclosure limitation methods.

These methods can be applied with varying intensities. Generally, increasing the amount of alteration decreases the risks of disclosures; but, it also decreases the accuracy of inferences obtained from the released data, since these methods distort relationships among the variables. For example, intensive data swapping severely attenuates correlations between the swapped and unswapped variables. These methods also make it difficult for secondary data users to obtain valid inferences that account for the impacts of the disclosure limitation. For example, to analyze properly data that have been perturbed by additive random noise, users should apply measurement error models [12], which are difficult to use for non-standard estimands.

Motivated by these drawbacks of standard disclosure limitation strategies, and particularly when data need to be significantly altered to protect confidentiality, several national statistical agencies have started to use adaptations of multiple imputation—now called synthetic data—to release confidential data, as first suggested by [26] and [17]. In particular, agencies are releasing data comprising the units originally surveyed with some collected values replaced with multiple imputations; these are known as partially synthetic datasets [20]. For example, the U.S. Federal Reserve Board in the Survey of Consumer Finances replaces monetary values at high disclosure risk with multiple imputations, releasing a mixture of these imputed values and the unreplaced, collected values [15]. The U.S. Bureau of the Census has released a partially synthetic, public use file for the Survey of Income and Program Participation that includes imputed values of Social Security benefits information and dozens of other highly sensitive variables [1]. The Census Bureau protects the identities of people in group quarters (e.g., prisons, shelters) in the American Community Survey by replacing quasi-identifiers for records at high disclosure risk with imputations [14]. The Census Bureau also has developed synthesized origin-destination matrices, i.e. where people live and work, available to the public as maps via the web (On The Map, http://lehdmap.did.census.gov/). In the U.S., partially synthetic, public use datasets are in the development stage for the Longitudinal Business Database [16], the Longitudinal Employer-Household Dynamics database, and the American Community Survey veterans and full sample data. Statistical agencies in Germany [8, 7] and New Zealand [13] also are developing synthetic data products. Other examples of partially synthetic data are in [2], [3], and [18].

The key to the success of synthetic data approaches, especially when replacing many values, is the data generation model. Current practice for generating synthetic data typically employs sequential modeling strategies based on parametric or semi-parametric models similar to those for imputation of missing data in [19]. The basic idea is to impute Y_1 from a regression of Y_1 on $(Y_2, Y_3, etc.)$, impute Y_2 from a regression of Y_2 on $(Y_1, Y_3, etc.)$, impute Y_3 from a regression of Y_3 on $(Y_1, Y_2, etc.)$, and so on. An advantage of this strategy is that it is generally easier to specify plausible conditional models than plausible joint distributions. A disadvantage is that the collection of conditional distributions is not guaranteed to correspond to a proper joint distribution, particularly when the models use different conditioning sets.

Specifying these conditional imputation models can be daunting in surveys with many variables available for conditioning. The data frequently include numerical, categorical, and mixed variables, some of which may not be easy to model with standard parametric tools. The relationships among these variables may be non-linear and interactive. Therefore, it may be advantageous to use non-parametric methods to generate imputations.

In this article, we investigate the effectiveness of random forests [5] for data synthesis.

Random forests can capture complex relationships that might not be easily found or estimated with standard parametric techniques, and they avoid stringent parametric assumptions. They can handle diverse data types in high dimensional data. They are computationally fast and easy to implement, with little tuning required by the user. These features suggest that, suitably adapted, random forests have potential as a method for generating model-free synthetic data with high analytic validity. We focus explicitly on models for categorical variables, although we discuss extensions of the random forest synthesizer to continuous variables in the concluding remarks.

The remainder of the article is as follows. In Section 2, we review partially synthetic data approaches. In Section 3, we review random forests and present adaptations for generating synthetic data. In Section 4, we apply the random forest synthesizer on data from the U.S. Current Population Survey. We include discussions of data utility and disclosure risk for the resulting partially synthetic datasets. In Section 5, we conclude with a discussion of implementation issues for the random forest synthesizer.

2 Review of Partially Synthetic Data

To illustrate how partially synthetic data might work in practice, we modify the setting described by [21]. Suppose the agency has collected data on a random sample of 10,000 people. The data comprise each person's race, sex, income, and years of education. Suppose the agency wants to replace race and sex for all people in the sample—or possibly just for a subset, such as all people whose income exceeds 100,000—to disguise their identities. The agency generates values of race and sex for these people by randomly simulating values from the joint distribution of race and sex, conditional on their education and income values. These distributions are estimated using the collected data and possibly other relevant information. The result is one partially synthetic data set. The agency repeats this process say ten times, and these ten data sets are released to the public.

To illustrate how a secondary data analyst might utilize these released datasets, suppose that the analyst seeks to fit a regression of income on education and indicator variables for the person's sex and race. The analyst first estimates the regression coefficients and their variances separately in each simulated dataset using standard likelihood-based estimates and standard software. Then, the analyst averages the estimated coefficients and variances across the simulated datasets. These averages are used to form 95% confidence intervals based on the simple formulas developed by [20], described below.

Let $D_{syn} = (D_1, \ldots, D_m)$ be the m partially synthetic datasets created by the agency for sharing with the public. Let Q be the secondary analyst's estimand of interest, such as a regression coefficient or population average. For $l = 1, \ldots, m$, let q_l and u_l be respectively the estimate of Q and the estimate of the variance of q_l in synthetic dataset D_l. Secondary analysts use $\bar{q}_m = \sum_{l=1}^{m} q_l/m$ to estimate Q and $T_m = \bar{u}_m + b_m/m$ to estimate $var(\bar{q}_m)$, where $b_m = \sum_{l=1}^{m}(q_l - \bar{q}_m)^2/(m-1)$ and $\bar{u}_m = \sum_{l=1}^{m} u_l/m$. For large samples, inferences for Q are obtained from the t-distribution, $(\bar{q}_m - Q) \sim t_{\nu_m}(0, T_m)$, where the degrees of freedom $\nu_m = (m-1)\left[1 + m\bar{u}_m/b_m\right]^2$. Derivations of this inferential method are presented in [20] and [25].

When the race and sex values are simulated from their true probability distribution, the synthetic data should have similar characteristics on average as the collected data. There is an analogy here to random sampling. Some true distribution of the variables exists in the population. The collected data are just a random sample from that population distribution. If the agency generates partially synthetic data from that same distribution—which is guaranteed for income and education in this example, since they remain unchanged—it essentially creates different random samples from the population. Hence, the analyst using these synthetic samples essentially analyzes alternative samples from the population.

The on average caveat is important: parameter estimates from any one simulated data set are unlikely to equal exactly those from the collected data. The synthetic parameter estimates are subject to two sources of variation, namely (i) sampling the collected data and (ii) generating synthetic values. It is not possible to estimate both sources of variation from only one released synthetic data set. However, it is possible to do so from multiple synthetic data sets, which explains why the multiple imputation framework applies.

The protection afforded by partially synthetic data depends on the nature of the synthesis. Replacing quasi-identifiers with imputations makes it difficult for users to know the original values of those variables, which reduces the chance of identifications. Replacing values of sensitive attributes makes it difficult for users to learn the exact values of those variables, which can prevent attribute disclosures. Releasing partially synthetic data is subject to inferential disclosure risk when the models used to simulate attributes are "too accurate." For example, when data are simulated from a regression model with a very small mean square error, intruders may be able to estimate outcomes precisely using the model. Or, if all people in a certain demographic group have the same, or even nearly the same, value of an outcome variable, the imputation models likely will generate that value. Agencies can reduce these types of risks by using less precise models when necessary. Nonetheless, there remain disclosure risks in partially synthetic data no matter which values are replaced, because the original records are released on the public use file. Analysts could utilize the released, unaltered values to facilitate disclosure attacks, for example via matching to external databases, or they may be able to estimate genuine values from the synthetic data with reasonable accuracy.

Partially synthetic data sets can have positive data utility features. When the data are simulated from distributions that reflect the distributions of the collected data, frequency-valid inferences can be obtained for wide classes of estimands. This is true even for high fractions of replacement, whereas swapping high percentages of values or adding noise with large variance produces worthless data. The inferences are determined by combining standard likelihood-based or survey-weighted estimates; the analyst need not learn new statistical methods or software to adjust for the effects of the disclosure limitation. The released data can include simulated values in the tails of distributions (no top-coding). Finally, because many quasi-identifiers can be simulated, finer details of geography can be released, facilitating small area estimation.

There is a cost to these benefits: the validity of synthetic data inferences depends on the validity of the models used to generate the synthetic data. The extent of this dependence is driven by the nature of the synthesis. For example, when all of race and sex are synthesized, analyses involving those variables reflect only the relationships included in the data generation models. When the models fail to reflect certain relationships accurately, analysts' inferences also will not reflect those relationships. Similarly, incorrect distributional assumptions built into the models will be passed on to the users' analyses. On the other hand, when replacing only a select fraction of race and sex and leaving many original values on the file, inferences may be relatively insensitive to the assumptions of the synthetic data models. In practice, this dependence means that agencies should release information that helps analysts decide whether or not the synthetic data are reliable for their analyses. For example, agencies might include the code used to generate the synthetic values as attachments to public releases of data. Or, they might include generic statements that describe the imputation models, such as "Main effects and interactions for income, education, and sex are included in the imputation models for race." Another possibility is for the agency to build a verification server, as suggested by [24], that provides data analysts with feedback on the quality of the analysis done on the synthetic data. Analysts who desire finer detail than afforded by the synthetic data may have to apply for restricted access to the collected data.

Given the synthesis model and estimates of its parameters, sophisticated analysts could generate their own synthetic values. However, as shown by [23], releasing that much information about the synthesis models increases disclosure risks. Furthermore, some analysts are not able to generate synthetic data given the models and parameter estimates; they need agencies to do it for them. Thus, describing the imputation models is necessary, but it is not sufficient. Agencies also should release synthetic data generated from the models.

3 Random Forests for Synthetic Data

The ideal data synthesizer has at least the following properties. First, it preserves as many relationships as possible while protecting confidentiality. Second, it can handle diverse data types. Third, it is computationally feasible for large datasets. And fourth, it is easy for agencies to implement with little tuning required. Random forests have the potential to satisfy these desiderata, as we now explain.

3.1 Background on random forests

Random forests extend the ideas of classification and regression trees (CART), as developed in [6]. Hence, we begin this review of random forests with a discussion of CART.

CART models seek to approximate the conditional distribution of a univariate outcome from multiple predictors. The CART algorithm partitions the predictor space so that subsets of units formed by the partitions have relatively homogeneous outcomes. The partitions are found by recursive binary splits of the predictors. The series of splits can be effectively represented by a tree structure, with leaves corresponding to the subsets of units. The 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.

CART models have potential advantages over parametric models for approximating conditional distributions. First, CART modeling may be more easily applied than parametric modeling, particularly for data with irregular distributions. Second, CART models can capture non-linear relationships and interaction effects that may not be easily revealed in the process of fitting parametric models. Third, CART provides a semi-automatic way to fit the most important relationships in the data, which can be a substantial advantage when there are many potential predictors. Primary disadvantages of CART models relative to parametric models include difficulty of interpretation, discontinuity at partition boundaries, and decreased effectiveness when relationships are accurately described by parametric models [11].

CART models have been adapted for generating partially synthetic data [22]. After fitting a large tree, the agency prunes the tree to satisfy confidentiality criteria, e.g., values in leaves must be sufficiently diverse. Then, the agency generates synthetic data by sampling from the leaves using a Bayesian bootstrap [27]. If it is desired to avoid releasing genuine values from some leaf, as may be the case for sensitive numerical data, the agency approximates a smooth density to the bootstrapped values using a Gaussian kernel density estimator with support over the smallest to the largest value of the outcome in the leaf.

Then, for each unit, the agency samples randomly from the estimated density in that unit's leaf using an inverse-cdf method.

Random forests are collections of CARTs, i.e., usually 500 or more trees grown on the same data. Each tree is based on a different random subset of the original data; usually, the subsets include around 2/3 of the full sample. Each tree is grown using a different random subset of the predictor variables to determine the binary splits; usually, roughly $\sqrt{p}$ variables are selected for each tree, where p is the total number of predictors. Typically each tree is grown to the maximum size, so that each terminal leaf contains one observation. There is no pruning of leaves for random forests. It is possible to force leaves to contain more than one observation, which speeds up the algorithm.

Data analysts can use the forest for a variety of purposes, including determining variable importance, clustering, outlier detection, and prediction of outcomes for new cases. Random forests maintain many of the benefits of CARTs but have some potential advantages. By randomly sampling predictors to grow each tree, the random forest allows more variables to contribute to predictions than CART does. This is potentially beneficial for synthetic data generation when the dimension of the predictor space is large compared to the sample size. Additionally, by randomly sampling observations and predictors, random forests effectively average over many small nearly independent trees, which can reduce bias and variance compared to one large tree. See [4] for further comparisons of random forests and CARTs.

For partial synthesis of categorical data, we use the forest to generate replacement values for the units in the original dataset, as described in the next section.

3.2 Random forest synthesizer for categorical data

Let the observed data D comprise $p+1$ variables, $(Y_1, \ldots, Y_{p+1})$, measured on n observations. To describe the random forest synthesizer, we first presume that the agency seeks to replace values of only one categorical variable, Y_j, given values of all other variables, Y_{-j}. We then extend to multiple variables. For $i = 1, \ldots, n$, let $Z_i = 1$ when record i has its value of Y_j replaced, and let $Z_i = 0$ otherwise.

We initiate the synthesizer by fitting a random forest of Y_j on Y_{-j} using the original data (before any synthesis) for only those records with $Z_i = 1$. The latter restriction is to ensure that the forest is tailored to the data that will be replaced. For example, if Y_j is a binary variable such that $p(Y_{ij} = 1) = .60$ in the entire dataset and $p(Y_{ij} = 1) = .20$ for cases with $Z_i = 1$, restricting the synthesis model to cases with $Z_i = 1$ appropriately ensures that around 20% of replacements equal one rather than possibly 60% of replacements. We create 500 trees such that each terminal leaf contains only one value of Y_{ij}. We use the standard defaults for the tuning parameters in random forests: random samples of around $(2/3)n$ records and random selection of roughly $\sqrt{p}$ predictors. We use the Gini index as the criteria used to determine the binary splits, which is a default criteria for many applications of random forests for categorical outcomes.

For any record i with values of predictors $Y_{i,-j}$, we run it down each tree in the forest to obtain a predicted value of Y_j. That is, we follow the sequence of partitioning for record i until we reach the terminal leaf in the tree. We tabulate the predictions for Y_{ij} to form the data for a multinomial distribution. For example, if the forest generates 500 predictions for a particular Y_{ij} such that 300 predict a race of white, 100 predict a race of black, 75 predict a race of Asian, and 25 predict a race of American Indian, we form a multinomial distribution with $p(white) = .6$, $p(black) = .2$, $p(Asian) = .15$, and $p(Amer.Ind.) = .05$. To generate the synthetic $Y_{i,j}$, we randomly sample one value from the implied multinomial distribution. We repeat this process for each record i with $Z_i = 1$. The result is one synthetic copy of Y_j.

Typically the agency will simulate several categorical variables, for example race, sex, and marital status, to protect confidentiality. The random forest synthesizer can be utilized sequentially in a manner akin to chained regressions for multiple imputation of missing data [19]. To illustrate, suppose that the agency seeks to replace $s < (p + 1)$ categorical variables with imputations. For each of these variables, the agency fits a random forest of Y_j on Y_{-j}, resulting in s random forests. Each random forest is fit using only the records that have values of Y_j replaced. Let $Y_{(1)}$ be the variable with the most values to be replaced; let $Y_{(2)}$ be the variable with the second most values to be replaced, and so on until $Y_{(s)}$. For now, we presume that each variable has a distinct number of records to be replaced; we consider ties shortly. The agency then proceeds as follows.

3.1 Replace $Y_{(1)}$ with synthetic replacements using the random forest synthesizer for $Y_{(1)}$. Let $Y_{(1)}^{rep}$ be the replaced values of $Y_{(1)}$.

3.2 Replace $Y_{(2)}$ with synthetic replacements using the random forest synthesizer for $Y_{(2)}$. Use the values of $Y_{(1)}^{rep}$ along with all other data when running observations down trees. Let $Y_{(2)}^{rep}$ be the replaced values of $Y_{(2)}$.

3.3 Replace each $Y_{(j)}$ where $j = 3, \ldots, s$ using the appropriate random forests and the values of previously synthesized variables, $(Y_{(1)}^{rep}, Y_{(2)}^{rep}, \ldots, Y_{(j-1)}^{rep})$, along with all other data when running observations down the trees.

The result is one synthetic dataset. These three steps are repeated for each of the m synthetic datasets, and these datasets are released to the public.

As with other sequential imputation strategies, there is no mathematical theory underpinning the ordering of variables for synthesis. It is possible that different orderings produce different risk and utility profiles. By ordering the variables by decreasing amount of synthesis, we base the largest number of synthetic imputations on the most genuine predictor values. Presumably, this affords the highest data quality for the variable with the largest synthesis. Alternatively, one could order the variables in increasing amount of synthesis, which could result in lower disclosure risks since the protection from synthesizing propagates down the chain.

When two or more variables have the same amount of synthesis, one approach is to select the ordering at random. A second approach, driven by computational concerns, is to impute categorical variables with small numbers of categories early in the sequence and those with large numbers of categories later in the sequence. Saving the variables with many levels until the end can speed up computation, since splitting a categorical variable with many levels is time consuming. A third approach is to experiment with several orderings to determine which produces datasets with the most desirable risk-utility profile. When practical, this is the optimal approach.

Each forest in the sequence is always grown on the genuine data; previously synthesized values are used only when determining each record's appropriate leaves in the trees. Growing trees with the genuine data ensures that the synthesis models capture relationships in the genuine data, and using previously generated synthetic values to determine leaves ensures that synthetic replacements are consistent with those relationships. This process differs slightly from sequential regression imputation for missing data, for which the previously imputed data are used both to estimate the imputation models and to generate imputed values [19]. We also note that, unlike in missing data contexts, there is only one cycle of the synthesis steps. This again stems from using the original data to build the models: there is no need to run several iterations to get away from starting values, i.e., the initial completions of missing data.

We use all trees when forming the multinomial distributions to synthesize each Y_{ij}. When variables that are not synthesized are strong predictors of Y_{ij} and synthesized variables are weak predictors of Y_{ij}, the trees that contain i in the training samples typically will terminate in leaves that contain unit i's original value. Thus, the data for the multinomial distributions can be highly peaked around the original value of Y_{ij}. High likelihoods of synthesizing the original values could represent an increased risk to confidentiality. If disclosure risk evaluations determine that this risk is too high, an alternative is to base the data for the multinomial distribution only on trees for which i does not appear in the training sample. This approach requires additional trees to ensure large samples for the multinomial distributions, which slows down the algorithm.

The algorithm can include prior distributions on the probabilities of the multinomial distribution within the leaves; for example, Dirichlet distributions are convenient, conjugate prior distributions. Including informative prior distributions can be useful for improving confidentiality protection. The prior distributions could ensure that there are non-zero probabilities of generating any supported outcome value in the leaves. This ensures that the random forest synthesizer meets the conditions of finite differential privacy [10], so that no exact disclosures of sensitive data exist. We note that with a large number of trees, noninformative prior distributions are unlikely to impact the results of the synthesis substantially.

A special case arises when all of the variables to be replaced undergo the same amount of synthesis. This includes synthesizing all values of the categorical variables to be replaced. In such cases, for an arbitrary ordering of the variables, the agencies can use 3.1 – 3.3, or they can proceed as follows. Let $Y_{(0)}$ be all variables that are not replaced.

3.1a Fit the random forest of $Y_{(1)}$ on $Y_{(0)}$ only. Replace $Y_{(1)}$ using the random forest synthesizer for $Y_{(1)}$. Let $Y_{(1)}^{rep}$ be the replaced values of $Y_{(1)}$.

3.2a Fit the random forest of $Y_{(2)}$ on $(Y_{(0)}, Y_{(1)})$ only. Replace $Y_{(2)}$ with synthetic replacements using the random forest synthesizer for $Y_{(2)}$. Use the values of $Y_{(1)}^{rep}$ and $Y_{(0)}$ when running observations down trees. Let $Y_{(2)}^{rep}$ be the replaced values of $Y_{(2)}$.

3.3a For each j where $j = 3, \ldots, s$, fit the random forest of $Y_{(j)}$ on $(Y_{(0)}, Y_{(1)}, \ldots, Y_{(j-1)})$. Replace each $Y_{(j)}$ using the appropriate random forests based on the values of previously synthesized variables, $(Y_{(1)}^{rep}, Y_{(2)}^{rep}, \ldots, Y_{(j-1)}^{rep})$ and $Y_{(0)}$, when running observations down the trees.

The result is one synthetic dataset. These three steps are repeated for each of the m synthetic datasets, and these datasets are released to the public. Steps 3.1a – 3.3a utilize the fact that a joint distribution is the product of conditional distributions. This can reduce the amount of computation in high dimensional problems, since fewer variables are in the predictor space for all forests except the one for $Y_{(s)}$. We note that $f(Y_{(j)}|Y_{(0)}, \ldots, Y_{(j-1)})$, for any $j > 0$, is generally not the true distribution for $Y_{(j)}$ given all other variables in the dataset. However, the goal of the synthesis models is not to capture each true conditional distribution; rather, it is to simulate effectively from the entire joint distribution.

Table 1: Description of variables used in the empirical studies

Variable	Label	Range
Sex	X	male, female
Race	R	white, black, American Indian, Asian
Marital status	M	7 categories, coded 1–7
Age (years)	G	0 – 90
Household income ($)	I	-21,010 – 768,700
Social security payments ($)	S	0, 1 – 50,000
Child support payments ($)	C	0, 1 – 23,917
Highest attained education level	E	16 categories, coded 31–46
Number of people in household	N	1 – 16
Number of people in household under age 18	Y	0, 1 – 11
Household property taxes ($)	P	0, 1 – 100,000
Household alimony payments ($)	A	0, 1 – 54,008

4 Empirical Illustration

We now illustrate the random forest synthesizer for categorical variables with a subset of public release data from the March 2000 U.S. Current Population Survey. The data comprise twelve variables measured on 10,000 heads of households; see Table 1.

We consider age, race, marital status, and sex to be quasi-identifiers that intruders can know precisely. To reduce the risk of identifications, we simulate all records' values of sex, race, and marital status; we do not replace age as this has too many values to treat as a categorical outcome in a random forest. Simulating all data is arguably more synthesis than necessary. There are 484 records with unique combinations of age, race, marital status, and sex; and, there are 394 combinations of the four variables with two cases. To protect confidentiality it may be sufficient for the agency to simulate the quasi-identifiers for only these subsets of the full sample. Nonetheless, we simulate all values to illustrate heavy synthesis. We use the software package "randomForest" in R to fit the random forests with the default specifications of tuning parameters, i.e., 500 trees, $2/3n$ sampled records per tree, $\sqrt{p}$ predictors per tree, and the Gini index for the splitting criterion.

4.1 Analytic usefulness

We first illustrate the analytic usefulness of the resulting synthetic datasets by comparing the synthetic and original data inferences for the coefficients in three regressions: a regression of the logarithm of income on a function of all the predictors, including non-linear effects in age and interactions among marital status and sex; a regression of the square root of positive social security payments on several predictors for a subset of individuals; and, a regression of the square root of positive child support payments on a small number of predictors. The first regression is based on 9938 people; the second is based on 2725 people; and, the third is based on 331 people. The regressions examined here are not necessarily the best possible for these variables. However, they do illustrate the performance of the random forest synthesizer for a variety of analyses.

We use the synthesizer in steps 3.1 – 3.3 to generate two independent sets of $m = 5$ synthetic datasets. Of course, in genuine contexts, only one set of m datasets would be released to the public for analysis. We create two sets to give a sense of the variability that can arise from the entire synthesis process with $m = 5$. We present results based on a synthesis ordering of marital status, race, and sex. Table 2 summarizes the inferences for the coefficients when fitting the models on the observed data and on the two sets of $m = 5$ synthetic datasets. In general, the estimated coefficients and 95% confidence intervals are similar across all three sources of data, even for the interactions and non-linear effects. Every one of the observed data coefficients is covered by the synthetic data confidence intervals. The least accurate synthetic coefficients involve people who are married in the armed forces and, for the regression of child support, people who are non-white. The numbers in these categories are relatively small: there are only 63 men and 57 women who are married in the armed forces, and there are only 53 non-whites who receive positive child support payments.

To examine the effect of ordering, we also evaluated results when reversing the order of synthesis, i.e., using the order $X - R - M$. The resulting synthetic datasets provided more accurate point estimates (in the sense of being close to the observed data point estimate) for some the coefficients and less accurate for others, but on the whole the differences were minor. Compared to the point estimates in the first replication in Table 2, the point estimates for two runs of the $X - R - M$ strategy were closer to the observed data point estimates for about half of the estimands.

We also examined using steps 3.1a – 3.3a to generate the synthetic datasets for the ordering $M - R - X$. On the whole, the results had slightly lower analytic validity than when using steps 3.1 – 3.3. Compared to the point estimates in the first replication in Table 2, the point estimates for two runs of the strategy based on steps 3.1a – 3.3a were closer to the observed data point estimates for about 40% of the estimands. Similar results are obtained when we used the point estimates in the second replication in Table 2 as the basis of comparison; this is also true for the the effect of ordering.

Finally, we compared the performance of the random forest synthesizer to an analogous CART synthesizer. We require a minimum of five records in each leaf of the trees and do not otherwise prune the trees; see [22] for discussion of specifying tree parameters. All CART models are fit in R using the "tree" function. We created ten replications of the partial synthesis for both random forests and CART. We averaged the point estimates across the ten replicates and compared these averages to the observed data point estimates. We were not able to distinguish the two procedures: for some coefficients the forest was more reliable, and for others the CART was more reliable. The CART resulted in smaller differences between synthetic and observed point estimates in slightly more than half of the replicates. Although this is a crude method of comparison, it suggests that there is not a clear winner between random forests and CART for these data. This result could be due to the modest number of variables in the dataset, since random forests tend to have comparative advantages over CART for prediction in high dimensions.

4.2 Evaluating disclosure risks

To evaluate disclosure risks associated with the random forest synthesizer, we compute probabilities of identification using methods developed by [23] for partially synthetic data. We summarize these methods below when the intruder knows that particular target records are in the sample; see [9] for modifications when this is not the case.

Suppose the intruder has a vector of information, t, on a particular target unit in the sample, D. Let t_0 be the unique identifier of the target, and let D_{i0} be the (not released) unique identifier for record i in D_{syn}, where $i = 1, \ldots, n$. Let K be any information released about the synthesis models.

The intruder's goal is to match unit i in D_{syn} to the target when $D_{i0} = t_0$. Let I be a random variable that equals i when $D_{i0} = t_0$ for $i \in D_{syn}$. The intruder thus seeks to calculate $Pr(I = i|t, D_{syn}, K)$ for $i = 1, \ldots, n$. Because the intruder does not know the actual values in Y^{rep}, he or she should integrate over its possible values when computing the match probabilities. Hence, for each record we compute

$$Pr(I = i|t, D_{syn}, K) = \int Pr(I = i|t, D_{syn}, Y^{rep}, K) Pr(Y^{rep}|t, D_{syn}, K) dY^{rep}. \tag{1}$$

This construction suggests a Monte Carlo approach to estimating each $Pr(I = i|t, D_{syn}, K)$. First, sample a value of Y^{rep} from $Pr(Y^{rep}|t, D_{syn}, K)$. Let Y^{new} represent one set of simulated values. Second, compute $Pr(I = i|t, D_{syn}, Y^{rep} = Y^{new}, K)$ using exact matching assuming Y^{new} are collected values. This two-step process is iterated h times, where ideally h is large, and (1) is estimated as the average of the resultant h values of $Pr(I = i|t, D_{syn}, Y^{rep} = Y^{new}, K)$. When K has no information, the intruder treats the simulated values as plausible draws of Y^{rep}, so that $h = m$. In the disclosure risk evaluations below, we assume that K is empty.

Following [9], we quantify disclosure risk with summaries of these identification probabilities. It is reasonable to assume that the intruder selects as a match for t the record i with the highest value of $Pr(I = i|t, D_{syn}, K)$, if a unique maximum exists. We consider two risk measures: the true match rate and the false match rate. Let c_i be the number of records with the highest match probability for the target t_i where $i = 1, ..., n$; let $d_i = 1$ if the true match is among the c_i units and $d_i = 0$ otherwise. Let $k_i = 1$ when $c_i d_i = 1$ and $k_i = 0$ otherwise. The true match rate equals $\sum k_i/n$. Let $f_i = 1$ when $c_i(1 - d_i) = 1$ and $f_i = 0$ otherwise; and, let g equal the number of records with $c_i = 1$. The false match rate equals $\sum f_i/g$. We note that the true match rate is computed over all n records, and the false match rate is computed only for records for which a unique match exists.

We assume that the intruder knows each target's age, marital status, race, and sex; the intruder does not know the other variables. The latter assumption is convenient for the empirical illustration, since we focus on synthesis of categorical variables. In general, however, intruders might know some of the numerical values, in which case agencies should synthesize or otherwise perturb those values. We also assume that the intruder knows who participated in the sample.

Under these assumptions, the true match rate among the 10,000 records in both runs of synthetic data used in Table 2 is around 3.0%. The false match rate is around 91%. Hence, intruders are not likely to make correct matches and are very likely to make mistakes when they find unique matches. Out of the 484 records with unique combinations of (M, R, X, G), approximately 16% can be correctly identified using the synthetic data. However, 68% of all apparent matches in this set are incorrect. There are no obvious patterns distinguishing the incorrect and correct matches; it would be difficult for intruders to determine when their matches are correct based on the synthetic data. Based on these risk measures, the random forest synthesizer has reduced disclosure risks.

When using synthesis based on the ordering $X - R - M$, the true match rate for the two runs is around 4%, and the false match rate is around 89%. Approximately 30% of the 484 cases are correctly identified, and approximately 45% of matches are incorrect. Hence, the synthesis based on the ordering $X - R - M$ results in higher disclosure risks than the synthesis based on the ordering $M - R - X$. Given that the analytic validity results for the two orderings were not much different, the agency should prefer the datasets based on the ordering marital status, race, and sex.

When using synthesis based on steps 3.1a – 3.3a for the ordering $M-R-X$, the true match rate is around 2.8% and the false match rate is around 91%. Hence, the confidentiality risks are slightly lower when using steps 3.1a – 3.3a as compared to using steps 3.1 – 3.3.

5 Concluding Remarks

The random forest synthesizer did reasonably well at preserving inferences—even for non-linear relationships, interactions, and sub-group analyses—under an intense synthesis setting of 100% replacement of three categorical variables. The synthesis was done with minimal tuning: we only selected the order of synthesis and used default specifications for the random forest algorithms. Disclosure risk assessments indicate that the data are altered sufficiently to reduce probabilities of correct identifications. Thus, it appears that random forests have potential to serve as convenient and effective data synthesizers.

The empirical illustration highlights some key issues in implementing the random forest synthesizer. First, agencies should evaluate different orderings for sequential synthesis, since analytic validity and disclosure risks may differ depending on the ordering. The role of ordering in sequential imputation strategies is not well understood even in missing data contexts, where sequential strategies have been widely used for almost a decade. Fortunately, it is easier to evaluate the impact of different orderings for synthetic data generation than for missing data imputation, because the real data provide a gold standard by which to judge the orderings. The agency can compare analytic validity and disclosure risk results for different orderings, and select the one that provides the most desirable balance. Global measures of data utility are particularly appropriate for analytic validity comparisons; for example, agencies can apply the propensity score approach described by [29] on each synthetic replicate and average the resulting measures. It is conceivable that releasing a set of synthetic replicates generated from several orderings provides an optimal risk and utility balance; this is an area for future research.

Second, it appears that relationships in small groups are more likely to be distorted by the synthesizer than relationships in large groups. It may be possible to improve analytic validity by synthesizing within groups rather than using only one model for each variable. For example, when synthesizing sex (or race), the agency can fit and synthesize from separate forests for each martial status category. This could preserve the relationships involving interactions between marital status and sex more effectively than a single model. Or, the agency can separate records by zero and non-zero child support payments and synthesize within these two groups. This could preserve the relationships between the synthesized variables and child support payments more effectively. A disadvantage of synthesizing within groups is that it requires additional effort by the agency.

Third, we built the synthesizer with the randomForest package in R, primarily because it is simple to use and freely available. Using a standard desk top computer, it took roughly ten minutes of computer time to generate the five synthetic datasets. However, using the randomForest package for synthesis has a drawback: we had to store the terminal leaves from all 500 trees to obtain the data for the multinomial distributions. With large datasets, storing all the terminal leaves from large trees can cause memory problems in R. For example, in experiments with a dataset comprising 50,000 records and twelve variables, R ran out of memory when implementing the random forest synthesizer. Thus, agencies that wish to use the randomForest package for large amounts of synthesis may need to synthesize in smaller groups. Undoubtedly, memory problems can be avoided by more clever coding than we implemented, for example modifications of Breiman's random forest programs in Fortran; we leave that to future research.

The random forest synthesizer can be used for more (or less) intense synthesis settings than the one illustrated in the empirical study. Increasing the number of variables to be synthesized increases computing time at a roughly linear rate. The synthesizer can be used on categorical variables with more than seven levels; for example, in separate investigations, synthesizing all 10,000 values of the sixteen-category education variable took less than one minute of running time on a standard desktop computer. Undoubtedly, there is a level at which the algorithm operates too slowly on the machine used for synthesis; however, our experience suggests that the synthesizer will run in reasonable time for categorical variables with typical numbers of levels.

An important research topic is to develop random forest synthesizers for numerical data. One possibility is to fit random forest regressions, run observations down the trees to get terminal values, and sample from these terminal values (perhaps after a Bayesian bootstrap) to get a synthetic dataset. Here, the agency may want to fit a smooth to the values in the terminal leaves to avoid releasing exact numerical data, as described in Section 2 for CART. The agency also may want to require a minimum terminal leaf size greater than one to provide more data to estimate the smooth. Selecting the optimal size involves some trade offs in bias and variance, as well as confidentiality risks.

Acknowledgments

This research was supported by a grant from the U.S. National Science Foundation (NSF-SES-0751671).

References

[1] Abowd, J. M., Stinson, M., and Benedetto, G. (2006) Final report to the Social Security Administration on the SIPP/SSA/IRS public use file project, U.S. Census Bureau Longitudinal Employer-Household Dynamics Program, Available at http://www.bls.census.gov/sipp/synth_data.html.

[2] Abowd, J. M. and Woodcock, S. D. (2001) Disclosure limitation in longitudinal linked data, in P. Doyle, J. Lane, L. Zayatz, and J. Theeuwes, eds., Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, Amsterdam: North-Holland, 215–277.

[3] Abowd, J. M. and Woodcock, S. D. (2004) Multiply-imputing confidential characteristics and file links in longitudinal linked data, in J. Domingo-Ferrer and V. Torra, eds., Privacy in Statistical Databases, New York: Springer-Verlag, 290–297.

[4] Berk, R. A. (2008) Statistical Learning From a Regression Perspective, New York: Springer.

[5] Breiman, L. (2001) Random forests, Machine Learning 45 5–32.

[6] Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984) Classification and Regression Trees, Belmont, CA: Wadsworth, Inc.

[7] Drechsler, J., Bender, S., and Rässler, S. (2008) Comparing fully and partially synthetic datasets for statistical disclosure control in the German IAB Establishment Panel, Transactions on Data Privacy 1 105–130.

[8] Drechsler, J., Dundler, A., Bender, S., and Rässler, S. and Zwick, T. (2008) A new approach for disclosure control in the IAB Establishment Panel–Multiple imputation for a better data access, Advances in Statistical Analysis 92 439 – 458.

[9] Drechsler, J. and Reiter, J. P. (2008) Accounting for intruder uncertainty due to sampling when estimating identification disclosure risks in partially synthetic data, in J. Domingo-Ferrer and Y. Saygin, eds., Privacy in Statistical Databases, New York: Springer-Verlag, 227–238.

[10] Dwork, C. (2006) Differential privacy, in M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, eds., Automata, Languages, and Programming, Berlin: Springer-Verlag, 1–12.

[11] Friedman, J. H. (1991) Multivariate adaptive regression splines (with discussion), The Annals of Statistics 19 1–141.

[12] Fuller, W. A. (1993) Masking procedures for microdata disclosure limitation, Journal of Official Statistics 9 383–406.

[13] Graham, P. and Penny, R. (2005) Multiply imputed synthetic data files, University of Otago, http://www.uoc.otago.ac.nz/departments/pubhealth/pgrahpub.htm.

[14] Hawala, S. (2008) Producing partially synthetic data to avoid disclosure, Proceedings of the Joint Statistical Meetings, Alexandria, VA: American Statistical Association.

[15] Kennickell, A. B. (1997) Multiple imputation and disclosure protection: The case of the 1995 Survey of Consumer Finances, in W. Alvey and B. Jamerson, eds., Record Linkage Techniques 1997, Washington, D.C.: National Academy Press, 248–267.

[16] Kinney, S. K. and Reiter, J. P. (2007) Making public use, synthetic files of the Longitudinal Business Database, Proceedings of the Joint Statistical Meetings, Alexandria, VA: American Statistical Association.

[17] Little, R. J. A. (1993), Statistical analysis of masked data, Journal of Official Statistics 9 407–426.

[18] Little, R. J. A., Liu, F., and Raghunathan, T. E. (2004) Statistical disclosure techniques based on multiple imputation, in A. Gelman and X. L. Meng, eds., Applied Bayesian Modeling and Causal Inference from Incomplete-Data Perspectives, New York: John Wiley & Sons, 141–152.

[19] Raghunathan, T. E., Lepkowski, J. M., van Hoewyk, J., and Solenberger, P. (2001) A multivariate technique for multiply imputing missing values using a series of regression models, Survey Methodology 27 85–96.

[20] Reiter, J. P. (2003) Inference for partially synthetic, public use microdata sets, Survey Methodology 29 181–189.

[21] Reiter, J. P. (2004) New approaches to data dissemination: A glimpse into the future (?), Chance 17:3 12–16.

[22] Reiter, J. P. (2005) Using CART to generate partially synthetic, public use microdata, Journal of Official Statistics 21 441–462.

[23] Reiter, J. P. and Mitra, R. (2009) Estimating risks of identification disclosure in partially synthetic data, Journal of Privacy and Confidentiality 1 99–110.

[24] Reiter, J. P., Oganian, A., and Karr, A. F. (2009) Verification servers: Enabling analysts to assess the quality of inferences from public use data, Computational Statistics and Data Analysis 53 1475–1482.

[25] Reiter, J. P. and Raghunathan, T. E. (2007) The multiple adaptations of multiple imputation, Journal of the American Statistical Association 102 1462–1471.

[26] Rubin, D. B. (1993) Discussion: Statistical disclosure limitation, Journal of Official Statistics 9 462–468.

[27] Rubin, D. B. (1981) The Bayesian bootstrap, The Annals of Statistics 9 130–134.

[28] Willenborg, L. and de Waal, T. (2001) Elements of Statistical Disclosure Control, New York: Springer-Verlag.

[29] Woo, M. J., Reiter, J. P., Oganian, A. and Karr, A. F. (2009) Global measures of data utility for microdata masked for disclosure limitation, Journal of Privacy and Confidentiality 1 111–124.

Table 2: Point estimates and 95% confidence intervals for coefficients in several regressions using observed data and two independent runs of random forest algorithm, each with $m = 5$ synthetic datasets. Synthesized values include marital status, sex, and race. Income regression uses only people with positive incomes. Social security regression uses only people with positive social security and age greater than 54. Child support regression uses only people with positive child support payments.

Estimand	Observed Data	Synthetic Data Run 1	Synthetic Data Run 2
Regression of $\log(I)$ on			
Intercept	4.9 (4.6, 5.1)	5.0 (4.7, 5.2)	5.0 (4.8, 5.3)
Black	-.19 (-.24, -.14)	-.20 (-.26, -.13)	-.21 (-.27, -.15)
American Indian	-.36 (-.53, -.19)	-.30 (-.51, -.09)	-.37 (-.60, -.15)
Asian	-.06 (-.17, .04)	-.01 (-.14, .11)	-.01 (-.14, .12)
Female	-.02 (-.07, .02)	-.04 (-.09, .01)	-.18 (-.07, .03)
Married in armed forces	.05 (-.14, .23)	-.10 (-.50, .30)	-.05 (-.31, .21)
Widowed	.04 (-.07, .15)	-.08 (-.21, .05)	-.06 (-.18, .07)
Divorced	-.19 (-.26, -.11)	-.22 (-.30, -.14)	-.24 (-.32, -.15)
Separated	-.32 (-.49, -.16)	-.42 (-.69, -.15)	-.36 (-.58, -.15)
Single	-.17 (-.23, -.09)	-.20 (-.27, -.12)	-.21 (-.28, -.13)
Education	.11 (.108, .120)	.11 (.103, .119)	.11 (.107, .118)
Household size > 1	.49 (.45, .54)	.46 (.41, .51)	.45 (.40, .50)
Fem., marr. in arm. forces	-.61 (-.89, -.34)	-.46 (-.96, .04)	-.38 (-.71, -.05)
Widowed females	-.34 (-.46, -.23)	-.28 (-.42, -.15)	-.33 (-.46, -.20)
Divorced females	-.29 (-.39, -.20)	-.27 (-.37, -.17)	-.27 (-.37, -.17)
Separated females	-.38 (-.58, -.18)	-.31 (-.60, -.14)	-.43 (-.68, -.18)
Single females	-.28 (-.37, -.20)	-.25 (-.34, -.16)	-.27 (-.37, -.18)
Age $\times 10$	.41 (.41, .46)	.38 (.33, .44)	.38 (.32, .43)
Age^2 $\times 1000$	-.42 (-.47, -.42)	-.39 (-.44, -.34)	-.39 (-.44, -.34)
Property tax $\times 10000$	.41 (.34, .40)	.42 (.36, .49)	.43 (.36, .49)
Regression of $\sqrt{S}$ on			
Intercept	79.2 (71.4, 87.1)	77.2 (69.2, 85.1)	76.8 (68.8, 84.8)
Black	-4.7 (-7.4, -2.0)	-5.5 (-8.5, -2.5)	-5.2 (-8.2, -2.2)
Native American	-20.3 (-30.0, -10.6)	-17.4 (-29.2, -5.6)	-15.8 (-27.8, -3.9)
Asian	-21.2 (-30.6, -11.7)	-18.6 (-30.7, -6.5)	-16.0 (-27.8, -4.2)
Female	-13.5 (-15.3, -11.6)	-12.9 (-15.0, -10.8)	-13.1 (-15.4, -10.9)
Widowed	8.3 (6.3, 10.4)	7.8 (5.5, 10.1)	7.7 (5.2, 10.1)
Divorced	-.57 (-3.4, 2.3)	-.92 (-4.0, 2.2)	-.10 (-3.3, 3.1)
Single	-2.3 (-6.0, 1.4)	-2.8 (-7.3, 1.6)	-3.1 (-7.2, 10.1)
High school	6.0 (4.1, 7.9)	5.6 (3.7, 7.5)	5.6 (3.7, 7.5)
Some college	6.6 (4.4, 8.8)	6.2 (4.0, 8.4)	6.3 (4.1, 8.5)
Completed college	8.7 (5.6, 11.6)	8.1 (5.1, 11.2)	7.5 (4.4, 10.5)
Post-college	10.8 (7.1, 14.4)	10.2 (6.3, 13.7)	10.1 (6.4, 13.7)
Age	.22 (.11, .32)	.25 (.14, .36)	.26 (.15, .36)
Regression of $\sqrt{C}$ on			
Intercept	-105 (-164, -46)	-102 (-162, -43)	-97 (-156, -39)
Non-white	-9.6 (-18.5, -.62)	-4.7 (-13.7, 4.4)	-5.2 (-15.4, 5.1)
Female	14.8 (2.8, 26.8)	13.5 (.23, 26.8)	12.7 (.87, 24.6)
Education	3.7 (2.3, 5.1)	3.7 (2.2, 5.1)	3.6 (2.2, 5.0)
Number w/age < 18	1.1 (-2.1, 4.4)	.95 (-2.3, 4.2)	.72 (-2.5, 3.9)