MDPI

Article

Anony*mice*d shareable data: Using *mice* to create and analyze multiply imputed synthetic data sets

* Correspondence:
† These authors contributed equally to this work.

Version October 31, 2021 submitted to Psych

Abstract: Synthetic data sets can greatly improve the dissemination of data and further analysis of private data. Generating and analyzing synthetic data sets is straightforward, yet a synthetic data analysis pipeline is seldom adopted by applied researchers. We outline a simple procedure for generating and analyzing synthetic data sets with `mice` in `R`. We demonstrate using simulations that the analysis results obtained on synthetic data yields unbiased and valid inferences, and leads to synthetic records that cannot be distinguished from the true data records. The ease of use when synthesizing data with `mice`, together with the validity of inferences obtained through this procedure opens up a wealth of possibilities for data dissemination and further research of initially private data.

Keywords: mice; imputation; synthetic data.

1. Introduction

Open science, including open data, has been marked as the future of science [1], and the advantages of publicly available research data are numerous [2,3]. Collecting research data requires an enormous investment both in terms of time and monetary resources. Openly accessible research data bears the potential of increasing the scientific returns for the same data collection effort. Additionally, the fact that public funds are used for data collection results in increasing demand for access to the collected data. Nevertheless, the possibilities to distribute research data directly are often very limited due to restrictions on data privacy and data confidentiality. Although these regulations are much needed, privacy constraints are also ranked among the toughest challenges to overcome in the advancement of modern day social science research [4].

Anonymizing research data might seem a quick and appealing approach to limit the unique identification of participants. However, this approach is not sufficient to fulfil contemporary privacy and confidentiality requirements [5,6]. Over the years, several other techniques have been used to increase the confidentiality of research data, such as categorizing continuous variables, top coding values above an upper bound or adding random noise to the observed values [7]. However, these methods may distort the true data relation between variables, thereby reducing the data quality and the scientific returns for re-using the same data for further research.

An alternative solution has been proposed separately by Rubin [8] and Little [9]. Although their approaches differ to some extent, the overarching procedure is to use bonafide observed data to generate multiply imputed synthetic data sets that can be freely disclosed. While in practice, one could see this as replacing the observed data values by multiple draws from the posterior predictive distribution of the observed data, based on some imputation model, Rubin would argue that these synthetic data values are merely draws from the same true data generating model. In that sense, the observed data is never replaced, but the population is resampled from the information captured in the (incomplete) sample. Using this approach, the researcher could replace the observed data set as a

whole with multiple synthetic versions. Alternatively, the researcher could opt to only replace a subset of the observed data. For example, one can choose to only replace dimensions in the data that could be compared with publicly available data sets or registers. Likewise, synthetisation could be limited to those values that are disclosive, such as high incomes or high turnovers.

Conceptually, the synthetic data framework is based upon the building blocks of multiple imputation of missing data, as proposed by Rubin [10]. Instead of replacing just the missing values with multiple draws from the posterior predictive distribution, one could easily *overimpute* any observed sensitive values. Similarly to multiple imputation of missing data, the multiple synthetic data sets allow for correct statistical inferences, despite the fact that the analyses do not use the "true" value. The analyses over multiple synthetic data sets should be pooled into a single inference, so that the researcher can draw valid conclusions from the pooled results. To that respect, the variance should reflect the added variability that is induced by the imputation procedure.

Potentially, this approach could fulfill the needs for openly accessibly data, without running into barriers with regard to privacy and confidentiality constraints. However, there is no such thing as a free lunch: data collectors have to put effort in creating high-quality synthetic data. The quality of the synthetic data is highly dependent on the imputation models, and using flawed models to generate synthetic data might bias subsequent analyses [11–13]. Conversely, if the models used to create the synthetic data are able to preserve the relationships between the variables as in the original data, the synthetic data can be nearly as informative as the observed data. Thus, to fully exploit the benefits of synthetic data, additional complications to actually create these high-quality data sets should be kept at a minimum.

To mitigate unnecessary challenges related to creating synthetic data sets on behalf of the researcher, software aimed at multiple imputation of missing data can be employed. Especially if researchers acquired familiarity with this software during earlier projects, or used it earlier during the research process, the additional burden of creating synthetic data sets is relatively small. The R-package `mice` [14] implements multiple imputation of missing data in a straightforward and user-friendly manner. However, the functionality of `mice` is not restricted to the imputation of missing data, but allows to impute any value in the data: even observed values. Consequently, `mice` can be utilized for the creation of multiply imputed synthetic data sets.

After creating multiply imputed synthetic data sets, the goal is to obtain valid statistical inferences in the spirit of Rubin [10] and Neyman [15]. In the missing data framework, this is done by performing statistical analyses on all imputed data sets, and pooling the results of the analyses according to Rubin's rules [10, p. 76]. In the synthetic data framework, the same procedure is followed, but with a slight twist: there are no values that remain constant over the synthetic data sets. The procedure of drawing valid inferences from multiple synthetic data sets is therefore slightly different.

In this manuscript we detail a workflow for synthesizing data with `mice`. First, the `mice` algorithm for the creation of synthetic data will be shortly explained. The aim is to generate synthetic sets that reassure the privacy and confidentiality of the participants. Second, a straightforward workflow for imputation of synthetic data with `mice` will be demonstrated. Third, we demonstrate the validity of the procedure through statistical simulation.

2. Generating synthetic data with `mice`

The `mice` package [14] in `R` [16] has been developed for multiple imputation to overcome problems related to nonresponse. In that context, the aim is to replace missing values with plausible values from the posterior predictive distribution of that variable. In `mice`, this is accomplished using fully conditional specification (FCS) [17], which breaks down the multivariate distribution of the data $\mathbf{Y} = (\mathbf{Y}_{obs}, \mathbf{Y}_{mis})$ into $j = 1, 2, \ldots, k$ univariate conditional densities, where k denotes the number of columns in the data. Using FCS, a model is constructed for every incomplete variable and the missing values $Y_{j,mis}$ are then imputed with draws from the posterior predictive distribution of $P(Y_{j,mis}|\mathbf{Y}_{obs}, \theta)$ on a variable-by-variable basis. Note that the predictor matrix Y_{-j} may contain yet imputed values from

an earlier imputation step, and thus will be updated after every iteration. This procedure is applied m times, resulting in m completed data sets $\mathbf{D} = (\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, \ldots, \mathbf{D}^{(m)})$, with $\mathbf{D}^{(l)} = (\mathbf{Y}_{obs}, Y_{mis}^{(l)})$.

In `mice`, the generation of multiply imputed data sets to solve for unobserved values is straightforward. The following pseudocode details the multiple imputation of the `mice::boys` data set [18] into the object `imp` with `m = 10` imputated sets and `maxit = 7` iterations for the algorithm to converge, using the default imputations methods for each column data class.

```
library(mice)
imp <- mice(boys,
            m = 10,
            maxit = 7)
```

It is straightforward to extend the imputation approach to generate synthetic values. Rather than imputing missing data, the observed values are then replaced by synthetic draws from the posterior predictive distribution. For simplicity, assume that the data is completely observed (i.e., $\mathbf{Y} = \mathbf{Y}_{obs}$). Following the notation of Reiter and Raghunathan [19], let for n units denote $Z_i = 1$ if any of the values of unit $i = 1, 2, \ldots, n$ are to be replaced by imputations, and $Z_i = 0$ otherwise, with $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_n)$. Accordingly, the data consists of values that are to be replaced and values that are to be kept (i.e., $\mathbf{Y} = (\mathbf{Y}_{rep}, \mathbf{Y}_{nrep})$. Now, instead of imputing $\mathbf{Y}_{mis}$ with draws from the posterior predictive distribution of $P(Y_{j,mis}|\mathbf{Y}_{obs}, \theta)$ as in the missing data case, $\mathbf{Y}_{rep}$ is imputed from the posterior distribution of $P(Y_{j,rep}^{(l)}|\mathbf{Y}_{-j}^{(l)}, \mathbf{Z}, \theta)$, where l is an indicator for the synthetic data set ($l = 1, 2, \ldots, m$). Note that synthetic values that are imputed at an earlier step can be used for imputing variable j. This process results in the synthetic data $\mathbf{D} = (\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, \ldots, \mathbf{D}^{(m)})$.

For example, overimputing synthetic values for both the observed and missing cells in the `mice::boys` data set into the object `syn`, given the same imputation parameters as before, can be realized by the following code execution.

```
syn <- mice(boys,
            m = 10,
            maxit = 7,
            where = matrix(TRUE,
                           nrow = nrow(boys),
                           ncol = ncol(boys)))
```

where the argument `where` requires a matrix of the same dimensions as the data, (i.e., a $n \times k$ matrix) containing logicals z_{ij} that indicate which cells are selected to have their values replaced by draws from the posterior predictive distribution. This approach allows to overimpute a subset of the observed data, or as in the above example, the observed data as a whole, resulting in a data set that partially or completely consists of synthetic data values.

Choosing an adequate imputation model to impute the data is paramount, as a flawed imputation model may drastically impact the validity of inferences [12,13]. Imputation models should be as flexible as possible to capture most of the patterns in the data, and to model possibly unanticipated data characteristics [20,21]. Parametric methods, albeit easy to implement in practice, may be too restrictive to capture generally complex patterns in the data, especially in the case of nonlinear relations and interactions between multiple variables. Classification and regression trees [CART; 22] allow to model more complex patterns in the data, and have therefore been suggested as an appropriate imputation method [23–25]. Loosely speaking, CART sequentially splits the predictor space into non-overlapping regions in such a way that the within-region variance is as small as possible after every split. As such, CART does not impose any parametric distribution on the data, making it a widely applicable method that allows for a large variety of relationships within the data [26]. Given these appealing characteristics and the call for the use of flexible methods when multiply imputing

data, we will focus our illustrations and evaluations of `mice` to method `mice.impute.cart()`, realized by:

```
syn <- mice(boys,
            m = 10,
            maxit = 7,
            method = "cart",
            where = matrix(TRUE,
                           nrow = nrow(boys),
                           ncol = ncol(boys)))
```

In a nutshell, the above code shows the simplicity of creating $m = 10$ synthetic data sets using `mice`. In practice, however, one should take some additional complicating factors into account. For example, one should account for deterministic relations in the data. Additionally, relations between variables may be described best using a different model than `CART`. Such factors are data dependent and should be considered by the imputer. In the next section, we will describe how the `boys` data can be adequately imputed. Additionally, we will show through simulations that this approach yields valid inferences.

3. Materials and Methods

We demonstrate the suitability of using `mice` for synthetisation using a simulation study on the `mice::boys` data set. This data set consists of the values of 748 Dutch boys on the following 9 variables:

Table 1. Description of the features in the `mice::boys` data set.

column	description
age	age in years
hgt	height (cm)
wgt	weight (kg)
bmi	body mass index
hc	head circumference (cm)
gen	genital Tanner stage G1-G5
phb	pubic hair Tanner P1-P6
tv	testicular volume (ml)
reg	region

Unfortunately, this data set does not differ from the vast majority of collected data sets, in the sense that it suffers from missing data. For simplicity, we complete the missing values using the default `mice` imputation model for all predictors except `bmi`, which is passively imputed using its deterministic relation with `wgt` and `hgt`. Specifically, the imputed values are used to calculate the exact `bmi` values that correspond with `hgt` and `wgt`.

```
# create a single imputed, completely observed `boys` data set
set.seed(123)

meth <- make.method(boys)
meth["bmi"] <- "~ I(wgt / (hgt / 100)^2)"
pred <- make.predictorMatrix(boys)
pred[c("hgt", "wgt"), "bmi"] <- 0

imp <- mice(boys,
            m = 1,
            maxit = 10,
```

```
            method = meth,
            predictorMatrix = pred)

data <- complete(imp)
```

3.1. Simulation methods

Usually, one would draw samples from a population that can be synthesized to evaluate the performance of the synthesization methods. As we only have access to a single sample, 1000 bootstrap samples of the `boys` data have been synthesized with $m = 5$ imputations for every data cell to induce an appropriate amount of sampling variance. Synthetic values are generated using the `CART` imputation method for all columns, except for `bmi`. The deterministic relation `bmi` which will be synthesized passively based on the synthetic values for `hgt` and `wgt` to preserve the relation in the synthetic data. Additional parameters that come with the use of `mice.impute.cart()` are the complexity parameter `cp` and the minimum number of observations in any terminal node `minbucket`, that both constrain the flexibility of the imputation model. The values of the parameters `cp` and `minbucket` ought to adhere to the call for imputation models that are as flexible as possible. Appropriate values for these parameters, as well as the input for the `predictorMatrix`, depend on the data at hand. In the current example, the complexity parameter is specified at `cp = 1e-08` rather than the default value `1e-04`, and the minimum number of observations in each terminal node is set at `minbucket` $= 3$ rather than the default value 5. By allowing for more complexity in the imputation model, bias in the estimates from the synthetic data set is reduced. Additionally, since the synthesis pattern is monotone, the number of iterations can be set to `maxit = 1` [e.g., 7, Ch. 3].

To assess the performance of `mice` for synthesizing data, we compare the bootstrapped samples with the synthetic versions of these bootstrapped samples. Specifically, univariate descriptive statistics, the correlation matrix, and two linear regression models as well as one ordered logistic regression model will be considered. Subsequently, the bias in the parameters and the 95% confidence interval coverage of the synthetic data will be examined. Similar to multiple imputation of missing data, correct inferences from synthetic data require correct pooling over the multiply imputed data sets.

Obtaining a final point estimate of the parameter of interest Q after imputation is fairly easy and no different from pooling in the case of missing data [10]. One can calculate the average of the m point estimates $q^{(l)}$

$$\bar{q}_m = \sum_{l=1}^{m} \frac{q^{(l)}}{m},$$

with $l = 1, \ldots, m$.

Also similarly to the missing data case, variances, and subsequently confidence intervals, should incorporate the increase in variance that is due to imputation [7,27]. Yet, the increase in variance due to imputation differs according to whether missing values are imputed or observed data is overimputed with missing values. Whereas the variance estimate after imputation of missing data needs to account for the fact that a certain amount of information in the data is missing, variance estimation from synthetic data does not suffer from this issue. The adjusted variance estimate that follows from using multiple synthetic data sets only suffers from the fact that a finite number of m synthetic data sets are used to resemble the observed data. Hence, the according variance estimate for synthetic data as developed by Reiter [27] yields

$$T = \bar{u}_m + \frac{b_m}{m},$$

with between-imputation variance

$$b_m = \sum_{l=1}^{m} \frac{(q^{(l)} - \bar{q}_m)^2}{(m-1)},$$

and sampling variance

$$\bar{u}_m = \sum_{l=1} \frac{u^{(l)}}{m},$$

where $u^{(l)}$ denotes the variance estimate in the lth synthetic data set.

4. Results

We evaluate the synthetic data with respect to the *true* data set on the basis of three aims. We believe that every reliable and valid data synthetisation effort in statistical data analysis should be able to yield 1) unbiased univariate statistical properties, 2) unbiased bivariate properties, 3) unbiased and valid multivariate inferences and 4) synthetic data that cannot be distinguished from real data. We consider the evaluation of the synthetic data simulations in the above order.

4.1. Univariate estimates

The univariate descriptives for the original data and the synthetic data can be found in Table 2.

Table 2. Univariate descriptives for the true data and $m = 5$ pooled univariate descriptives for the synthetic data over 1000 simulations. Variable names followed by a * are categorical.

	n	mean	sd	median	min	max	skew	kurtosis
original age	748	9.16	6.89	10.50	0.04	21.18	-0.03	-1.56
synthetic age	748	9.15	6.89	10.49	0.04	20.96	-0.03	-1.55
original hgt	748	131.10	46.52	145.75	50.00	198.00	-0.30	-1.47
synthetic hgt	748	131.06	46.50	145.32	50.69	197.16	-0.30	-1.47
original wgt	748	37.12	26.03	34.55	3.14	117.40	0.38	-1.03
synthetic wgt	748	37.09	26.00	34.44	3.35	112.26	0.38	-1.03
original bmi	748	18.04	3.04	17.45	11.73	31.74	1.14	1.79
synthetic bmi	748	18.05	3.08	17.48	11.49	32.37	1.11	1.85
original hc	748	51.62	5.86	53.10	33.70	65.00	-0.91	0.12
synthetic hc	748	51.61	5.86	53.18	34.38	62.85	-0.91	0.12
original gen*	748	2.53	1.59	2.00	1.00	5.00	0.52	-1.36
synthetic gen*	748	2.53	1.59	2.00	1.00	5.00	0.52	-1.35
original phb*	748	2.75	1.86	2.00	1.00	6.00	0.56	-1.25
synthetic phb*	748	2.75	1.86	2.00	1.00	6.00	0.56	-1.24
original tv	748	8.43	8.12	3.00	1.00	25.00	0.85	-0.78
synthetic tv	748	8.42	8.11	3.19	1.00	25.00	0.85	-0.77
original reg*	748	3.02	1.14	3.00	1.00	5.00	-0.08	-0.77
synthetic reg*	748	3.02	1.14	3.00	1.00	5.00	-0.08	-0.76

We see from Table 1 that the synthetic data estimates closely resemble the true data estimates. All sample statistics of interest show negligible bias over the 1000 synthetic data sets. Hence, univariately the imputation model proves adequate.

4.2. Bivariate estimates

An often used bivariate statistic is Pearson's correlation coefficient. When evaluating this correlation coefficient on the numeric columns in the `boys` data set, we find that biases are very small. The results are displayed in Table 3.

Table 3. Bivariate correlations of the numerical columns in the true data with in parentheses the corresponding bias of the $m = 5$ pooled synthetic correlations over 1000 simulations. All estimates are rounded to 3 decimal places.

	age	hgt	wgt	bmi	hc	tv
age	1	0.976 (0.001)	0.950 (0.000)	0.627 (0.009)	0.853 (0.000)	0.810 (0.002)
hgt	0.976 (0.001)	1	0.944 (0.001)	0.596 (0.013)	0.907 (0.000)	0.754 (0.000)
wgt	0.950 (0.000)	0.944 (0.001)	1	0.791 (0.009)	0.834 (0.000)	0.817 (0.000)
bmi	0.627 (0.009)	0.596 (0.013)	0.791 (0.009)	1	0.588 (0.009)	0.610 (0.007)
hc	0.853 (0.000)	0.907 (0.000)	0.834 (0.000)	0.588 (0.009)	1	0.623 (0.000)
tv	0.810 (0.002)	0.754 (0.000)	0.817 (0.000)	0.610 (0.007)	0.623 (0.000)	1

We see that the correlations obtained from synthetic data are unbiased with respect to the true data set. The largest absolute bias over 1000 simulations equals 0.013, indicating that the imputation model is capable of preserving the bivariate relations in the data.

4.3. Multivariate model inferences

First, we evaluate the performance of our synthetic simulation set on a linear model where `hgt` is modeled by a continuous predictor `age` and an ordered categorical predictor `phb`. The results for this simulation can be found in Table 4.

Table 4. Simulation results for a linear regression model with continuous and ordered categorical predictors. The model evaluated is `hgt` $\sim$ `age + phb`. Depicted are the true data estimate and the bias from the true data estimate and the coverage rate of the 95% confidence interval for the bootstrap and synthetic data sets.

		Bootstrap		Synthetic	
term	estimate	bias	cov	bias	cov
(Intercept)	63.087	-0.001	0.970	0.405	0.958
age	7.174	0.000	0.958	-0.033	0.947
phb.L	-12.250	0.008	0.950	0.582	0.927
phb.Q	-1.376	-0.022	0.926	0.112	0.934
phb.C	-3.564	0.051	0.915	0.301	0.912
phb^4	-0.431	0.016	0.930	0.106	0.940
phb^5	2.064	0.060	0.941	0.077	0.943

We see that the finite nature of the true data set together with the design-based simulation setup yields slight undercoverage for the dummy variables of `phb`. This finding is observed in both the bootstrap coverages (i.e. the fraction of 95% confidence intervals that cover the true data parameters) and the synthetic data coverages. Hence, it is likely that this undercoverage stems from the simulation setup, rather than the imputation procedure. Besides the undercoverage, there is a tiny bit of bias in the estimated coefficients of the variable `phb` that occurs in the synthetic estimates, but not in the observed estimates. Yet, since the bias is relatively small and does not result in confidence invalidity, it seems fair to assume that the introduced bias is not that problematic.

Second, we evaluate a proportional odds logistic regression model wherein ordered categorical column `gen` is modeled by continuous predictors `age` and `hc`, and categorical predictor `reg`. The results for this model evaluation are shown in Table 5.

Table 5. Simulation results for a proportional odds logistic regression model with continuous and ordered categorical predictors. The model evaluated is `gen` $\sim$ `age + hc + reg`. Depicted are the true data estimate and the bias from the true data estimate and the coverage rate of the 95% confidence interval for the bootstrap and synthetic data sets.

		Bootstrap		Synthetic	
term	estimate	bias	cov	bias	cov
age	0.461	0.004	0.942	0.002	0.939
hc	-0.188	-0.000	0.929	-0.004	0.945
regeast	-0.339	0.012	0.960	0.092	0.957
regwest	0.486	0.009	0.952	-0.122	0.944
regsouth	0.646	0.012	0.966	-0.152	0.943
regcity	-0.069	0.012	0.940	0.001	0.972
G1|G2	-6.322	0.032	0.934	-0.254	0.946
G2|G3	-4.501	0.052	0.936	-0.246	0.945
G3|G4	-3.842	0.058	0.937	-0.244	0.948
G4|G5	-2.639	0.064	0.936	-0.253	0.947

These results demonstrate that the synthetic data analysis yields inferences that are on par with inferences from the analyses directly on the bootstrapped datasets. Hence, for the regression coefficients as well as the intercepts, the analyses on the synthetic data yield valid results. Nevertheless, a small amount of bias is introduced in the estimated intercepts of the synthetic data. However, the corresponding confidence interval coverage rates are actually somewhat higher than the confidence interval coverage rates of the bootstrapped data. Therefore, the corresponding inferences do not seem to be affected by this small bias.

4.4. Data discrimination

When we combine the original and synthetic data, can we predict which rows come from the synthetic data set? If so, then our synthetic data procedure would be redundant, since the synthetic set differs from the observed set. To evaluate whether we can distinguish between the true data and the synthetic data, we combine the rows from each simulation synthetic data set with the rows from the true data. We then run a logistic regression model to predict group membership: i.e. does a row belong to the true data or synthetic data. As predictors we take all columns in the data. The pooled parameter estimates over all simulations can be found in Table 6.

Table 6. Simulation results for a logistic regression model aimed at discriminating between synthetic records and true records.

term	estimate	std.error	statistic	df	p.value
(Intercept)	0.22	1.15	0.19	521.93	0.60
wgt	0.00	0.02	0.25	420.19	0.60
hgt	-0.00	0.01	-0.18	359.73	0.60
age	-0.00	0.05	-0.00	345.73	0.62
hc	0.00	0.03	0.11	415.44	0.61
gen.L	-0.00	0.42	-0.00	164.76	0.65
gen.Q	-0.01	0.21	-0.04	203.38	0.63
gen.C	-0.00	0.16	-0.02	239.63	0.64
gen^4	0.01	0.21	0.01	237.41	0.63
phb.L	-0.02	0.44	-0.04	156.54	0.64
phb.Q	-0.01	0.22	-0.02	198.20	0.64
phb.C	0.00	0.18	0.00	211.17	0.62
phb^4	0.00	0.18	0.02	228.51	0.64
phb^5	0.00	0.20	0.02	248.57	0.63
tv	0.00	0.02	0.01	264.26	0.63
regeast	-0.00	0.23	0.01	210.90	0.64
regwest	-0.00	0.22	-0.00	221.14	0.63
regsouth	-0.01	0.22	-0.02	226.10	0.64
regcity	0.01	0.27	0.03	204.15	0.64
bmi	-0.02	0.06	-0.26	320.18	0.61

From these pooled results we can see that the effects for all predictors are close to zero and non-significant. When we take the predicted values from the simulated models and compare them with the *real* values, we obtain the summary statistics in Table 7.

Table 7. Confusion statistics for a prediction model aimed at discriminating between synthetic records and true records.

	estimate
Accuracy	0.50381
Balanced Accuracy	0.50381
Kappa	0.00762
McnemarPValue	0.63187
Sensitivity	0.50368
Specificity	0.50394
Prevalence	0.50000

The accuracy of the predictive modeling effort is not better than random selection and the Kappa coefficient indicates that a perfect prediction model is far from realized. The accuracy is quite balanced as there is no skewness over sensitivity and specificity. These findings indicate that the synthetic data is indistinguishable from the true data.

5. Discussion

We demonstrate that generating synthetic data sets with `mice` in `R` is a straightforward process that fits well in a data analysis pipeline. The approach is hardly different from using multiple imputation to solve problems related to missing data, and hence can be expected to be familiar to applied researchers. The multiple synthetic sets yield valid inferences on the true underlying data generating mechanism, thereby capturing the nature of the original data. This makes the multiple synthetisation procedure with `mice` suitable for further dissemination of synthetic data sets.

To some, the procedure of generating multiple synthetic sets may seem overly complicated. We would like to emphasize that analyzing a single synthesized set, while perhaps unbiased, would underestimate the variance properties that are so important in drawing statistical inferences from finite data sets. After all, we are often not interested in the sample at hand, but aim to make inferences

about the underlying data generating mechanism as reflected in the population. Properly capturing the uncertainty of synthetic data sets, just like with incomplete data sets, is therefore paramount.

Besides capturing uncertainty of synthetic data, it is important that imputers pay close attention to disclosure risks that remain after creating synthetic data sets. Creating synthetic data, unless generated from a completely parametric distribution, does not remove all potential disclosure risks. For example, if the values that ought to be replaced get the exact same value imputed, the synthetisation procedure has no use.

Additionally, if not all variables in the data are synthesized, but the variables that are synthesized can be linked to open access data, a disclosure risk may remain [28]. If the open access data allows for identification, and the corresponding observations in the synthetic data can be identified, the variables that are not synthesized may provide information that should not have been disseminated. The associated problems generally decrease when the sensitive variables are synthesized as well. Still, it is important to remain critical regarding the extent to which the synthetic data might be disclosive. The practical development of easy to use software to identify which observations are at risk of disclosure is an area of future work that can improve these issues. Subsequently, the implementation of ways to overcome such problems, for example by record swapping as suggested by Jiang *et al.* [13], once detected is welcomed.

That said, while the fields of data disclosure control and data synthetisation originated decades ago, it is more relevant today than ever. In our study we outline a standard research workflow where the goal is to synthesize a complete data set. To simplify the simulation we adopted a bootstrapping scheme to induce sampling variance in order to conform to the combination procedure defined by Reiter [27]. Ideally, one would like to omit the bootstrap from the synthetization to adopt a scheme where the sampled data itself serves as a reference: much like the procedure outlined in Vink and van Buuren [29] for incomplete data simulation. The corresponding pooling rules have not been derived yet and the incorporation in data analysis workflows would require proper attention from developers alike.

It is important to note that in our simulations we used a single iteration. A single iteration is sufficient only when the true data is completely observed, or when the missingness pattern is monotone [7]. If both observed and unobserved values are to be synthesized, then more iterations and a careful investigation into the convergence of the algorithm are in order. Synthetic data generation with `mice` is therein no different than multiple imputation with `mice`.

There are other developments that can generate synthetic data sets. For example, the `synthpop` [30] package in `R` would yield valid synthetic data sets, but requires the true data to be completely observed. To avoid this, one could adopt a two-step approach wherein the incomplete values are multiply imputed before synthetization. Given m multiple imputations and r synthetizations, at least $m \times r$ synthetic data sets are then in order. However, this approach requires that researchers first impute the missingness using a different package, and then use `synthpop` to create synthetic data sets, while `mice` allows to do both. Additionally, the flexibility with `mice` is that both unobserved and observed data values could be synthesized at once, without the need for a two-step approach. Then, using `mice`, m synthetic sets are sufficient. As of today, no pooling rules for one-step imputation of missingness and synthetisation have been developed, but the derivation of those would further reduce the burden of creating synthetic data sets.

The ease of use when synthesizing data with `mice` in `R`, together with the validity of inferences obtained through this procedure opens up a wealth of possibilities for data dissemination and further research of initially private data.

Author Contributions: The authors contributed equally to this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gewin, V. Data sharing: An open mind on open data. *Nature* **2016**, *529*, 117–119. Number: 7584 Publisher: Nature Publishing Group, doi:10.1038/nj7584-117a.
2. Molloy, J.C. The open knowledge foundation: open data means better science. *PLoS Biol* **2011**, *9*, e1001195.
3. Walport, M.; Brest, P. Sharing research data to improve public health. *The Lancet* **2011**, *377*, 537–539.
4. Lazer, D.; Pentland, A.S.; Adamic, L.; Aral, S.; Barabasi, A.L.; Brewer, D.; Christakis, N.; Contractor, N.; Fowler, J.; Gutmann, M.; Jebara, T.; King, G.; Macy, M.; Roy, D.; Van Alstyne, M. Life in the network: the coming age of computational social science. *Science* **2009**, *323*, 721.
5. Ohm, P. Broken Promises of Privacy: Responding to the Surprising Failure of Anonymization. *UCLA Law Review* **2009**, *57*, 1701–1778.
6. Council, N.R. *Putting People on the Map: Protecting Confidentiality with Linked Social-Spatial Data*; The National Academies Press: Washington, DC, 2007. doi:10.17226/11865.
7. Drechsler, J. *Synthetic datasets for statistical disclosure control: theory and implementation*; Vol. 201, Springer Science & Business Media, 2011.
8. Rubin, D.B. Statistical disclosure limitation. *Journal of official Statistics* **1993**, *9*, 461–468.
9. Little, R.J. Statistical analysis of masked data. *Journal of Official statistics* **1993**, *9*, 407–426.
10. Rubin, D.B. *Multiple Imputation for Nonresponse in Surveys*; Wiley, 1987.
11. Reiter, J.P. Simultaneous use of multiple imputation for missing data and disclosure limitation. *Survey Methodology* **2004**, *30*, 235–242.
12. Grund, S.; Lüdtke, O.; Robitzsch, A. Using synthetic data to improve the reproducibility of statistical results in psychological research **2021**.
13. Jiang, B.; Raftery, A.E.; Steele, R.J.; Wang, N. Balancing Inferential Integrity and Disclosure Risk via Model Targeted Masking and Multiple Imputation. *Journal of the American Statistical Association* **2021**, pp. 1–15.
14. Van Buuren, S.; Groothuis-Oudshoorn, K. mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software* **2011**, *45*, 1–67.
15. Neyman, J. On the two different aspects of the representative method: the method of stratified sampling and the method of purposive selection. *Journal of the Royal Statistical Society* **1934**, *97*, 123–150.
16. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2021.
17. Van Buuren, S.; Brand, J.P.L.; Groothuis-Oudshoorn, C.G.M.; Rubin, D.B. Fully conditional specification in multivariate imputation. *Journal of statistical computation and simulation* **2006**, *76*, 1049–1064.
18. Fredriks, A.M.; Van Buuren, S.; Burgmeijer, R.J.; Meulmeester, J.F.; Beuker, R.J.; Brugman, E.; Roede, M.J.; Verloove-Vanhorick, S.P.; Wit, J.M. Continuing positive secular growth change in The Netherlands 1955–1997. *Pediatric research* **2000**, *47*, 316–323.
19. Reiter, J.P.; Raghunathan, T.E. The Multiple Adaptations of Multiple Imputation. *Journal of the American Statistical Association* **2007**, *102*, 1462–1471.
20. Murray, J.S. Multiple imputation: a review of practical and theoretical findings. *Statistical Science* **2018**, *33*, 142–159.
21. Rubin, D.B. Multiple imputation after 18+ years. *Journal of the American statistical Association* **1996**, *91*, 473–489.
22. Breiman, L.; Friedman, J.; Stone, C.J.; Olshen, R.A. *Classification and regression trees*; CRC press, 1984.
23. Reiter, J.P. Using CART to generate partially synthetic public use microdata. *Journal of Official Statistics* **2005**, *21*, 441.
24. Burgette, L.F.; Reiter, J.P. Multiple imputation for missing data via sequential regression trees. *American journal of epidemiology* **2010**, *172*, 1070–1076.
25. Doove, L.L.; Van Buuren, S.; Dusseldorp, E. Recursive partitioning for missing data imputation in the presence of interaction effects. *Computational statistics & data analysis* **2014**, *72*, 92–104.
26. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An introduction to statistical learning*; Vol. 112, Springer, 2013.
27. Reiter, J.P. Inference for partially synthetic, public use microdata sets. *Survey Methodology* **2003**, *29*, 181–188.
28. Drechsler, J.; Reiter, J.P. An empirical evaluation of easily implemented, nonparametric methods for generating synthetic datasets. *Computational Statistics & Data Analysis* **2011**, *55*, 3232–3243.

29. Vink, G.; van Buuren, S. Pooling multiple imputations when the sample happens to be the population. *arXiv preprint arXiv:1409.8542* **2014**.
30. Nowok, B.; Raab, G.M.; Dibben, C. synthpop: Bespoke Creation of Synthetic Data in R. *Journal of Statistical Software* **2016**, *74*, 1–26. doi:10.18637/jss.v074.i11.

Sample Availability: A full simulation archive is available from the zip-enclosed RMD file