HHS Public Access

Author manuscript

J Algorithm Comput Technol. Author manuscript; available in PMC 2023 January 20.

Published in final edited form as:

J Algorithm Comput Technol. 2022 ; 16: . doi:10.1177/17483026211065379.

DataSifter II: Partially synthetic data sharing of sensitive information containing time-varying correlated observations

Nina Zhou[1,2], **Lu Wang**[2], **Simeone Marino**[1], **Yi Zhao**[2], **Ivo D Dinov**[1,3]

[1]Statistics Online Computational Resource, University of Michigan, USA

[2]Department of Biostatistics, University of Michigan, USA

[3]Michigan Institute for Data Science, University of Michigan, USA

Abstract

There is a significant public demand for rapid data-driven scientific investigations using aggregated sensitive information. However, many technical challenges and regulatory policies hinder efficient data sharing. In this study, we describe a partially synthetic data generation technique for creating anonymized data archives whose joint distributions closely resemble those of the original (sensitive) data. Specifically, we introduce the DataSifter technique for time-varying correlated data (DataSifter II), which relies on an iterative model-based imputation using generalized linear mixed model and random effects-expectation maximization tree. DataSifter II can be used to generate synthetic repeated measures data for testing and validating new analytical techniques. Compared to the multiple imputation method, DataSifter II application on simulated and real clinical data demonstrates that the new method provides extensive reduction of re-identification risk (data privacy) while preserving the analytical value (data utility) in the obfuscated data. The performance of the DataSifter II on a simulation involving 20% artificially missingness in the data, shows at least 80% reduction of the disclosure risk, compared to the multiple imputation method, without a substantial impact on the data analytical value. In a separate clinical data (Medical Information Mart for Intensive Care III) validation, a model-based statistical inference drawn from the original data agrees with an analogous analytical inference obtained using the DataSifter II obfuscated (*sifted*) data. For large time-varying datasets containing sensitive information, the proposed technique provides an automated tool for alleviating the barriers of data sharing and facilitating effective, advanced, and collaborative analytics.

Corresponding author: Ivo D Dinov, Statistics Online Computational Resource, University of Michigan, Ann Arbor, MI 48109, USA. statistics@umich.edu.

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Keywords

Data-sharing; electronic health records; longitudinal imputation; synthetic data generation; time-varying data; DataSifter

Introduction

Open science advocates for broad sharing of data, reproducible research, source code, and software tools. The benefits of open science on many aspects of our everyday lives are well documented.[1–5] Along with the well-known exponential increase of the amount of newly acquired data, there is also an equally striking exponential decay of the value of data that is stored but not processed or shared.[6,7] However, sharing data without loss of privacy is difficult, especially in medical and healthcare settings. In fact, 66% of the participants in the 2017 Health Information National Trends Survey were concerned about data privacy when health information is electronically exchanged.[8] For data sharing involving protected health information, organizations' institutional review boards (IRBs) need to review the research before the required information can be retrieved from existing medical records and processed to extract valuable information. IRB's initial review process may take up to 4 months. This process has significant variability depending on the type of review, for example, expedited, exempt, or full board reviews may take additional time, from 16 to 631 days.[9] In the United States, healthcare systems own the property rights for Electronic Health Records (EHR), and researchers have to bear the costs of data extraction and transfer under data use agreements.[10] Such regulation guarantees the protection of individual privacy rights but delays researchers' ability to gain access to appropriate information, build models, and rapidly validate scientific discovery. This slows the knowledge transfer and impedes the translation of basic science discoveries into clinical practice. As a result, the restricted and significantly delayed access to data may limit the information utility for answering specific scientific questions. For example, in 2015, Keegan et al.[11] examined the relationship between ethnicity and short-term breast cancer survival using 2010 Kaiser Permanente Northern California EHR data. However, to obtain both the demographics and the cancer treatments for patients across facilities, they had to reduce the study cohort time frame from 8 to 3 years to have both datasets available and link them together, which significantly impacted the statistical power of this scientific investigation. Thus, there are enormous benefits in developing new statistical methods to facilitate secure and quick information exchange between data stewards and data science experts.

Three existing strategies provide secure mechanisms for modeling, processing, and interrogating sensitive cross-sectional data. These include secure enclave access, data encryption (e.g. fully homomorphic encryption), and synthetic data publishing. First, secure data enclave environments[12,13] offer a platform for researchers to analyze sensitive data without compromising risks for misuse and other violations. Many health information storage solutions, for example, EHRs, rely on technology that provides managed data access for research in safe and controlled environments.[14] Several possible unmodified database management systems can be utilized to provide secure data enclaves.[15,16] Second, data encryption methods, including fully homomorphic encryption (FHE), encode the data to

allow computations directly on the resulting ciphertext.[17–19] FHE relies on homomorphic computing (result-preserving property) on the ciphertext without exposing the sensitive raw data to independent researchers. The above two mechanisms provide secure channels for data transfer and storage, however, these approaches do not shorten the data sharing process and limit the type of advanced analytics that can be used to interrogate the data.

To address these limitations, the third strategy, synthetic data generation, provides "fake" records that closely resemble the real data for predictive and inference model constructions. Both the computer science and statistical science communities have developed various methods in this field. Most of the methods developed by computer science communities fall under the umbrella of differential privacy.[20,21] For quantitative data types, this class of methods creates fully synthetic data for predictive purposes, where noisy conditional models generate each synthetic record.[22–24] The most recent developments in this class are related to generative adversarial networks (GANs).[25] They generate candidate synthetic records and use a discriminator that accepts only differentially private data to provide privacy protection guarantee.[23,26] To the best of our knowledge, there is less work related to generating data for model-based inference purposes in the differential privacy framework. A recent work[27] proposed HealthGAN for generating fully synthetic data for privacy preservation and inference purposes. They defined data utility and privacy using Euclidean distances between original and fully synthetic records, which is not guaranteed to be differentially private. The authors reported that seven out of 29 confidence intervals estimated using the synthetic data did not actually contain the original data. Moreover, the HealthGAN cannot handle time-varying data elements. In the statistical science community, synthetic data generation was first proposed by Rubin[28] with two classes of generating methods, as summarized by Reiter and Raghunathan.[29] The fully synthetic data sets are created by conditional distributions estimated from sensitive datasets. Popular methods for constructing these conditional distributions include multiple imputation (MI).[28,30] Also, there are Bayesian data augmentation methods like SynSys,[31] which aims to enrich the existing data to train machine learning predictive models. However, since the distributions of subject characteristics are not considered in the data generation procedure, fully synthetic data may not represent the original patient population. The other class of methods creates partially synthetic data to alleviate this issue, which utilizes a set of multiple-imputed data replacing sensitive data value cells with imputations.[32–34] This class of methods treats data obfuscation as a missing data handling problem, where they generate artificial missingness for sensitive values in the data set and impute the value with the remaining untouched data. As a result, partially synthetic data provide valid statistical inference. Yet, combined information from a set of multiple imputed datasets indicates the locations of true and obfuscated cells resulting in no privacy protection for the true cells. In practice, covering all possible sensitive values is barely achievable and selecting the obfuscation location is a subjective and critical step for data privacy protection.

The DataSifter technique (DataSifter I) was recently introduced[35] as an automated procedure for generating partially synthetic data. This approach offers better protection of participant privacy in the case of using high-dimensional sensitive cross-sectional data. The DataSifter framework is designed to help data governors (institutions who possess and manage data) safely share synthetic subsets of their sensitive data archives using customized levels of

statistical obfuscation. It perturbs individual-level records and allows researchers to acquire population-level information that closely approximates the true signal securely. The core DataSifter technique relies on two processes supporting the critical statistical obfuscation of the data. First, it randomly and artificially generates missingness in the data, following the Missing Completely At Random (MCAR) mechanism,[36] and uses robust iterative imputation methods, for example, missForest,[37] to approximate the original information. After the first step, DataSifter I creates a partially synthetic dataset disguising the locations of true values, which provides better privacy protection than the MI method. Second, DataSifter I computed cluster neighborhoods using Euclidean and Gower distances for continuous and categorical variables, respectively. Then, within each neighborhood cluster, DataSifter I randomly swaps a subset of feature values between similar records. This second operation guarantees partial obfuscations within each record while preserving the holistic cohort distribution in the variable space.

The quality of generated partially synthetic data can be quantified by *disclosure risk* and the *deviance of model-based inference*. The disclosure risk describes the privacy-awareness of the synthetic data. There are two classes of disclosure risk criterion, namely ϵ-differential privacy and statistical disclosure risk measurement. The criterion developed by the computer science communities (differential privacy) guarantees strong protection such that intruders do not learn much about a target record even if they have access to the rest of the original records.[38] The confidentiality guarantee provides a stringent threshold for data privacy. However, since the threshold is difficult to satisfy, it is challenging to ensure the analytical usefulness of a deferentially private synthetic dataset. The other class, probabilistic disclosure risk measure, enables the investigation of record-level disclosure risk under different scenarios of intruder's prior knowledge about the original dataset.[39,40] A higher probability is associated with a more vulnerable record. This measurement allows accurate comparisons of disclosure risk for different synthetic datasets. McClure and Reiter[41] discovered that the level of ϵ corresponding to particular levels of statistical disclosure risk depends on the properties of the observed data. In this paper, we propose a statistical disclosure risk measurement to quantify time-specific record-level risk (*data privacy*) for longitudinal variables, assuming that the intruder can access all but the target record. After achieving desirable data privacy, we measure the analytical value of the synthetic data to ensure *data utility*. Since inference is our objective, we define data utility as the deviation of the model inference based on the original and the partially synthetic datasets given a predefined inferential model and some specific clinical or research questions. There is a trade-off between data privacy and data utility. Among the partially synthetic data generating methods, MI methods are designed to minimize the loss of data utility, while the DataSifter framework focuses on maximizing the data privacy protection under acceptable data utility. In this paper, the above two criteria are used to compare different partially synthetic datasets.

Time-varying correlated data, including longitudinal data, are ubiquitous and provide valuable information for many biomedical and health conditions. For example, in EHR databases, patient characteristics and disease progression variables are collected repeatedly across visits. Maintaining or preserving the within-subject covariance structure among time-varying measurements presents another layer of challenges. To the best of our knowledge, to

date, there are no automated procedures that enable secure sharing of sensitive time-varying correlated data using partially synthetic data. The MI methods tilt the balance towards data utility, and the DataSifter I method cannot handle time-varying data elements. We propose a new algorithm, *DataSifter with Time-varying Correlated Data (DataSifter II)*, that extends the DataSifter functionality in the case of dealing with large, cross-sectional, time-varying, and self-correlated data elements. DataSifter II introduces artificial missingness and embeds robust longitudinal imputation methods to handle high-dimensional sensitive data with time-varying measures. As illustrated in Figure 1, our proposed procedure operates on the time-varying data separately from the time-invariant (cross-sectional) data elements and then integrates the two parts to compile the obfuscated *sifted* dataset (output). Our extensive simulation and application studies show that compared to the MI method, the proposed method can better protect data privacy while preserving a proper level of data utility.

The contribution of our paper is twofold: (1) we propose an automated algorithm for time-varying partially synthetic data generation allowing user-specified secure level and protecting the original within-subject covariance structure, and (2) we formally define practical data privacy and data utility measurements to validate synthetic data before release. The rest of the manuscript is organized as follows: In section “Privacy and utility measurement for partially synthetic data,” we define the data privacy and utility measurement for partially synthetic data evaluation. Specifically, in section “Data structure and notations,” we define the disclosure risk and show that partially synthetic datasets generated by the DataSifter framework provide better privacy protection than that of the MI method. Section “DataSifter II technique” describes the DataSifter II protocol. Section “Simulation studies” validates the data utility preservation and privacy protection of the proposed algorithm under different simulation settings and compares the performance against the MI method. In section “Biomedical application,” we apply DataSifter II to the Medical Information Mart for Intensive Care III (MIMIC-III) clinical data and demonstrate its performance in maintaining a careful balance between protecting sensitive information and preservation of the data utility. We summarize the findings and discuss the expected impact and future developments in section “Discussion.”

Privacy and utility measurement for partially synthetic data

Data structure and notations

Time-varying correlated data are common in most biomedical and epidemiology studies. For example, in multi-center studies, we typically measure the target variables across all subjects at a single time point, but the subjects may be correlated within each center. In longitudinal data, the target variables are measured repeatedly at baseline and during follow-up, and thus we have intrinsic within-subject correlations. In this case, researchers take repeated measurements on the same subjects to reduce measurement errors, which may also involve within-subject correlations. The DataSifter II framework can be applied to any correlated data. For illustration purposes in this study, we investigate the use of DataSifter II on longitudinal data.

Consider a longitudinal EHR dataset with n patients, each recorded until J_i^{th} visit, where the time intervals between visits are similar across patients. We collect m_l longitudinal variables at each visit and m_s static variables for patient characteristics. For simplicity, we denote time-varying variables as Y's and time-invariant variables as X's. In the following sections, we use $i = 1, \ldots, n$ to denote patients; $j = 1, \ldots, J_i$ to denote the visit time, which allow different visit times among patients; and k to index the variables (columns) in the dataset such that for static variable $k = 1, \ldots, m_s$, and for longitudinal variable $k = 1, \ldots, m_l$. Dummy variables are created for all categorical longitudinal variables. Then the longitudinal measurements for subject i are

$$\mathbf{Y}_i = \begin{bmatrix} Y_{i,1,1} & \cdots & Y_{i,1,m_l} \\ Y_{i,2,1} & \cdots & Y_{i,2,m_l} \\ \ldots & \ldots & \ldots \\ Y_{i,J_i,1} & \cdots & Y_{i,J_i,m_l} \end{bmatrix}$$

with patient i's, time j's record of longitudinal variable k denoted as $Y_{i,j,k}$. The time-invariant variables of subject i are denoted as $\mathbf{X}_i = \left(X_{i1}, \ldots, X_{im_s}\right)$, which can also be represented as $\mathbf{X}_i = \left(\mathbf{X}_{i1}, \ldots, \mathbf{X}_{im_s}\right)$ to match the dimension of $\mathbf{Y}_i$ where $\mathbf{X}_{ik} = \left(X_{i,1,k}, \ldots, X_{i,J_i,k}\right)^{\mathrm{T}}$, and $k = 1, \ldots, m_s$ repeats the static variable for J_i times.

Missing data occurs often in longitudinal observations. Missingness can come from a completely missing record or partially missing record, where patient i does not have all data available for some visits. In this case, we denote the missing indices for kth outcome as $mis_k = \{(i, j) | Y_{i,j,k} = \text{NA}\}$, and observed indices as $obs_k = \{(i, j) | Y_{i,j,k} \neq \text{NA}\}$. Fully observed long format data $\mathbf{Y}_{(\sum_i J_i) \times m_l} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_n)^{\mathrm{T}}$ has $\sum_i J_i$ rows and m_l columns. Similarly, the static variables are denoted as $\mathbf{X}_{(\sum_i J_i) \times m_s} = (\mathbf{X}_1, \ldots, \mathbf{X}_n)^{\mathrm{T}}$. We use $\mathbf{D} = [\mathbf{X}, \mathbf{Y}]$ to denote the observed dataset.

While generating partially synthetic data, we view data obfuscation as an artificial missing creation and imputation procedure. Here we focus on obfuscating time-varying data and further denote each row in partially synthetic time-varying data as $(\mathbf{Y}_{i,j,\text{nrep}_{ij}}, \widehat{\mathbf{Y}}_{i,j,\text{rep}_{ij}})$, where $\text{nrep}_{i,j}$ is a set of indexes for unreplaced variables, $\text{rep}_{i,j}$ is a set of indexes for replaced variables compared with the original row such that $\text{nrep}_{i,j} \cup \text{rep}_{i,j} = \{1, \ldots, m_l\}$, and $\widehat{\mathbf{Y}}_{i,j,\text{rep}_{ij}}$ is a vector of the synthetic values created for patient i's record at time j. Similarly we denote one row in the synthetic static data as $(\mathbf{X}_{i,\text{neep}_i}, \widehat{\mathbf{X}}_{i,\text{rep}_i})$, where nrep_i and rep_i are indexes for unreplaced and replaced variables, and $\widehat{\mathbf{X}}_{i,\text{rep}_i}$ denotes a vector of synthetic values for patient i's static characteristics. Finally, we use $\mathbf{Z}$ to denote a synthetic dataset that is composed of the time-varying and static synthetic data components.

Data privacy measurement

In this section, we formally define data privacy in the form of disclosure risk and compare the disclosure risk between MI and DataSifter methods. Assume we have a m_I-dimensional partially synthetic time-varying data vector $\mathbf{y}_{i,j} = (\mathbf{y}_{i,j,\text{nrep}_{i,j}}, \mathbf{y}_{i,j,\text{rep}_{i,j}})$ corresponding to each individual i at time j in the original dataset $\mathbf{D}$ and the static portion of data is complete. We have a partially synthetic dataset $\mathbf{Z}$ that follows a similar joint distribution as the original data with unchanged static variables (unobfuscated). Specifically, we denote the partially synthetic dataset generated by DataSifter as $\mathbf{Z}_{\text{sift}}$ and $U (U \geq 2)$ multiply imputed (MI) datasets as $\mathbf{Z}_{\text{MI}} = (\mathbf{Z}_{\text{MI}}^{(1)}, \ldots, \mathbf{Z}_{\text{MI}}^{(U)})$. To compare the *disclosure risk* between the *sifted* and MI partially synthetic datasets, we closely follow the Bayesian risk approach described by Reiter et al.[40] Suppose an intruder is interested in learning some of the true values in the $\mathbf{y}_{i,j}$ vector. Let A represent the intruder's prior knowledge about the original dataset $\mathbf{D}$, which is often referred to a subset of records in $\mathbf{D}_{-(i,j)} = \mathbf{D} \setminus \{\mathbf{y}_{i,j}\}$. Let S denote any information known by the intruder about the synthetic data generation procedure. Then, define the disclosure risk for $y_{i,j,k}$ as the conditional distribution

$$\underbrace{p\left(Y_{i,j,k} = y_{i,j,k} \mid \mathbf{Z}, A, S\right)}_{disclosure\ risk},$$

where $j \in \text{nrep}_{i,j} \cup \text{rep}_{i,j}$. The intruder cannot infer the location (indices) of unchanged ($\text{nrep}_{i,j}$) and changed ($\text{rep}_{i,j}$) cells in $\mathbf{Z}_{\text{sift}}$, whereas the unchanged cells in a set of U multiple-imputed datasets would imply the index locations for $\text{nrep}_{i,j}$. Hence, for the output *sifted* dataset, we have the disclosure risk:

$$p\left(Y_{i,j,k} = y_{i,j,k} \mid \mathbf{Z}_{\text{sift}}, A, S\right).$$

For $\mathbf{Z}_{\text{MI}}$ we have

$$\begin{aligned} &p\left(Y_{i,j,k} = y_{i,j,k} \mid \mathbf{Z}_{\text{MI}}, A, S\right) \\ &\quad = \left\{1 - I\left(k \in \text{nrep}_{i,j}\right)\right\} p\left(Y_{i,j,k} = y_{i,j,k} \mid \mathbf{Z}_{\text{MI}}, \right. \\ &\left.\mathbf{Y}_{i,j,\text{nep}_{i,j}} = \mathbf{y}_{i,j,\text{nrep}_{i,j}}, A, S\right) + I\left(k \in \text{nrep}_{i,j}\right), \end{aligned}$$

where $I(k \in \text{nrep}_{i,j})$ is an indicator that takes the value of 1 when the data cell (i, j, k) of the multiple imputed datasets are not replaced with obfuscated value. When $k \in \text{nrep}_{i,j}$, the intruder knows that $\mathbf{Z}_{\text{MI}}$ contains true value at cell (i, j, k) so that the discourse risk is 1. On the other hand, when $k \notin \text{nrep}_{i,j}$, it is appropriate to assume that knowing the locations of true covariate values in the record for patient i's jth visit (knowing $\mathbf{Y}_{i,j,\text{nrep}_{i,j}} = \mathbf{y}_{i,j,\text{nrep}_{i,j}}$) yields similar or higher disclosure risk compared to not knowing the locations. When the information contained in $\mathbf{Z}_{\text{sift}}$ and $\mathbf{Z}_{\text{MI}}$ is similar regarding inferring the distribution of $Y_{i,j,k}$, we have

$$p\left(Y_{i,j,k} = y_{i,j,k} \mid \mathbf{Z}_{\text{MI}}, A, S\right) \geq p\left(Y_{i,j,k} = y_{i,j,k} \mid \mathbf{Z}_{\text{sift}}, A, S\right).$$

Therefore, when both synthetic datasets contain comparable information, the DataSifter output has smaller, or rarely similar, disclosure risk compared to the MI method.

Data governors can quantify the privacy protection level of the synthetic longitudinal data using our disclosure risk defined above. Specifically, when calculating the maximal privacy loss for each record, we assume the intruder knows all other records in the raw dataset, that is, $A = \mathbf{D}_{-(i,j)}$, and the imputation model for the synthetic data is known. The proposed data privacy measurement (PM) for cell (i, j, k) is defined as the difference between the expected and observed values

$$\begin{aligned}\mathbf{PM}_{i,j,k} &= E\left(Y_{i,j,k} \mid \mathbf{Z}, \mathbf{D}_{-(i,j)}, S\right) - y_{i,j,k} \\ &= \left\{\int y \cdot p\left(Y_{i,j,k} = y \mid \mathbf{Z}, \mathbf{D}_{-(i,j)}, S\right) dy\right\} - y_{i,j,k}.\end{aligned}$$

The PM provides statistical disclosure risk for every time point in a longitudinal record. In practice, the conditional model for $Y_{i,j,k}$ is constructed using $\mathbf{D}_{-(i,j)}$ with the identical model specification as the missing imputation model, assuming the intruder has maximal prior knowledge about the original data. For $Y_{i,j,k}$ in $\mathbf{Z}_{\text{sift}}$ or replaced cells in $\mathbf{Z}_{\text{MI}}$, we calculate the expected difference between the model prediction given other covariates in the synthetic data and the true value. For $\mathbf{Z}_{\text{MI}}$, the PMs of $Y_{i,j,k}$ for unchanged cells are 0. Assume we introduce a percent of artificial missingness to the original data and $\mathbf{D}_{-(i,j)}$ provides sufficient information for accurate $Y_{i,j,k}$ predictions. For MI, there are $(1 - a)$ of $Y_{i,j,k}$ with PM = 0 and the remaining cells have equal or slightly smaller PM. Thus, DataSifter is expected to improve PM by at least $1/a$ times compared to MI. We examine the average PM for every time-varying variable in the simulation and application sections below.

Data utility measurement

Given a pre-specified model, we can obtain the desired utility of the partially synthetic data by comparing the model fitted with the original and synthetic data in terms of model inference and prediction accuracy. The data governor can consider a feasible parametric model regressing a summary variable on other covariates. For example, in EHR data, we can predict patients' comorbidity scores over time, representing summary scores for the patients' medical conditions. For model inference, to test the data utility based on a regression coefficient β, we first fit the desired model with the original dataset and obtain its confidence interval. Then, we generate L partially synthetic datasets under the same target parameter setting, where L is a large positive integer. By fitting the desired model on each synthetic data, we obtain a set of $\hat{\beta}_l$, $l = 1, \ldots, L$ and corresponding confidence intervals $(\text{LB}_{\hat{\beta}_l}, \text{UB}_{\hat{\beta}_l})$. In the ideal case, where the true coefficient β^* is known, we can directly use the confidence interval coverage, $\sum_{l=1}^{L} I\{\beta^* \in (\text{LB}_{\hat{\beta}_l}, \text{UB}_{\hat{\beta}_l})\}/L$, as our utility measurement. In practice, we obtain the empirical confidence interval for $\hat{\beta}_l$ from the synthetic datasets and measure if it overlaps with the confidence interval provided by the original dataset. In terms of prediction accuracy, we can set aside a randomly selected test set and compare the prediction error between models constructed with the remaining original and synthetic data records.

Many alternative performance metrics can be defined to track the utility of the DataSifter-generated synthetic datasets. For example, the Akaike information criterion, which is commonly used to contrast the relative quality of statistical models for a given set of data, provides one strategy for comparing the overall quality of the models constructed from original and synthetic datasets. We will focus on the parameter-level utility measurement for the rest of the paper.

Evaluating the performance of partially synthetic datasets

We can assess the performance of multiple partially synthetic datasets or select the desired parameters for a synthetic data generation process based on the data privacy and utility measurements defined above. Practically, we can first define a minimal average PM level for selected or all data cells as the threshold for data privacy. Then, we compare the data utility among the qualified datasets in terms of the confidence interval coverage or the confidence interval overlap probabilities for different model parameters and model prediction accuracy. Datasets with higher PM, higher confidence interval coverage or confidence interval overlap, and lower model prediction accuracy reduction are preferable.

DataSifter II technique

The proposed DataSifter II procedure operates on static variables **X** and time-varying variables **Y** separately and merges the two components back together to form the final partially synthetic data. The static variables are obfuscated with DataSifter I algorithm. The DataSifter II requires complete static variables as candidate predictors while obfuscating **Y**. We handle possible missingness for time-invariant variables by the missForest technique.[37] When obfuscating **Y**, we first impute the original missingness in **Y** with inverse probability-weighted imputation models. Then, we randomly introduce missingness to the working time-varying data and impute back with a proposed robust imputation method.

The main assumptions of the DataSifter II include (A1) the possible missingness mechanisms in the original data include missing at random (MAR) or missing completely at random (MCAR); and (A2) the sensitivity of each variable is equally important. There are three possible missing mechanisms. MCAR assumes that missingness is unrelated to any observed or unobserved variables. Under this missing mechanism, the subset of complete records without missingness is representative of the study population without selection bias. However, MCAR happens rarely. MAR is more plausible to occur such that missingness can be accounted for with observed data. Otherwise, missing not at random may occur, where analysis performed on the complete portion of data can suffer from selection bias. We consider the (A1) assumption to hold in the original dataset to provide sufficient data utility for synthetic data generation. Under this assumption, MAR or MCAR guarantees unbiased missing imputation, ensuring the obfuscation quality of DataSifter II. (A2) allows indistinguishable obfuscation for each variable and greatly simplifies the obfuscation procedure. We can further adjust the procedure of missingness assignment for specific scenarios where the data governors believe a pre-defined set of variables can be more sensitive than the rest.

Sifting static variables with DataSifter I

We apply DataSifter I to obfuscate the static portion **X**. The DataSifter I protocol includes the following four steps: (1) imputation of potential missingness in the original data using the *missForest*[37] algorithm, (2) introducing artificial missingness to the working data, (3) imputing the missing cells back, and (4) swapping partial information for similar records. The resulting *sifted* data has complete records in the same format as the original data and with minimal distortion of the original joint probability distribution.

The imputation procedures in DataSifter aim to create a single complete dataset disguising the original or artificial missing positions. We use the missForest technique that outputs a single imputed dataset and is proven to have smaller imputation errors than common methods, including MI.[37,42] This non-parametric imputation technique sequentially imputes and updates the data by variable. In the first iteration, when imputing for the first target variable, it fills in the missing cells among other predictor variables with mean imputation. Then, it constructs random forest models (target variable versus all other variables) to provide imputations. In subsequent iterations, while imputing and updating by variable, the imputation accuracy for each target variable improves as the missing cells in all other variables are replaced with better predictions.

When imputing the original missing data, we employ the original stopping criterion in missForest. It stops to iterate when the difference between the latest and prior imputed data matrix is at least as great as the previous difference measured or the maximal iteration limit has been achieved. The difference between matrices in sets of continuous (**N**) and categorical (**F**) variables are defined as

$$\Delta_{\mathbf{N}} = \frac{\sum_{k \in \mathbf{N}} (\widehat{\mathbf{X}}_k^{(r)} - \widehat{\mathbf{X}}_k^{(r-1)})^2}{\sum_{k \in \mathbf{N}} (\widehat{\mathbf{X}}_k^{(r)})^2}$$

and

$$\Delta_{\mathbf{F}} = \frac{\sum_{k \in \mathbf{F}} \sum_{i=1}^{n} I(\widehat{\mathbf{X}}_{ik}^{(r)} \neq \widehat{\mathbf{X}}_{ik}^{(r-1)})}{\text{Number of missing cells in categorical variables}},$$

where $\widehat{\mathbf{X}}_k^{(r)}$ is the imputed vector and $\widehat{X}_{ik}^{(r)}$ is the imputed value for subject i of the kth variable in the rth iteration.

When imputing artificial missingness, the true missing value is known. We define the stopping criterion under a tolerance level ϵ as

$$\frac{\|X^*_{mis_k,k} - \widehat{X}^{(r)}_{mis_k,k}\|_1}{\|X^*_{mis_k,k}\|_1} < \epsilon,$$

where $X^{*}_{mis_k,\,k}$ is the true values in the working data after imputing original missing data and $\widehat{X}^{(r)}_{mis_k,\,k}$ is the imputed values.

Iterative imputation algorithm for time-varying data

Although DataSifter I applies robust nonparametric imputation methods like missForest to impute static missing variables, effective missing imputation for time-varying data can be challenging. In this paper, we propose an iterative imputation algorithm for longitudinal data similar to the missForest algorithm. The proposed algorithm considers two types of missing mechanisms (MAR and MCAR) and two modeling options (linear mixed model and RE-EM tree). It handles missingness in time-varying variables **Y** with complete static variables **X** as potential predictors.

Before the imputation, we initiate all missing cells with the closest value from the same subject (last value carry forward or next value carry backward). If the subject has no observations of certain variables, we initialize such missing cells with mean imputations. Then, we sort the variables ascendingly based on missing percentage so that $Y_{\cdot,\cdot,1}$ has the smallest missing percentage and $Y_{\cdot,\cdot,m_l}$ has the most missing. Next, we start our iterative imputation procedure. Within an iteration, we impute from the first to the last variable with missing. While imputing a target variable $Y_{\cdot,\cdot,k}$, we separate the working data into four groups: the observed values of the target variable $Y_{obs_k,\,k}$, variables other than the target among the observed rows $\left[\mathbf{Y}_{obs_k,\,-k},\ X_{obs_k,\,\cdot}\right]$, the missing cells of the target variable with current imputation values $Y_{mis_k,\,k}$, and variables other than the target among the missing rows $\left[\mathbf{Y}_{mis_k,\,-k},\ \mathbf{X}_{mis_k,\,\cdot}\right]$, where obs_k and mis_k are the patient and visit index sets (i, j) with observed and missing variable k, respectively. Imputation models for the target variable is constructed by regressing $Y_{obs_k,\,k}$ on $\left[\mathbf{Y}_{obs_k,\,-k},\ \mathbf{X}_{obs_k,\,\cdot}\right]$ and we update $Y_{mis_k,\,k}$ based on the imputation model. The imputation of a following missing variable k' $(k < k' \leq m_l)$ will benefit from this update because we have better estimates of the missing values in $Y_{\cdot,\cdot,k}$ for constructing the imputation model or providing covariates when predicting $Y_{mis_{k'},\,k'}$. The algorithm finalizes the imputation result of a target variable when the imputation model predictions for the observed values are close to the true values after multiple iterations. For artificial missing, we directly compare the true missing values with its predictions. The algorithm stops when fewer than two variables need to be finalized, or the maximal iteration is achieved.

Imputation model under missing at random assumption.—Under different missing patterns, the algorithm utilizes different imputation models. When we have MAR, the missingness depends on observed data and the complete observations might be biased. We utilize inverse probability weighting to obtain an unbiased pseudo sample for imputation model fitting. By pseudo sample, we mean the weighted sample that creates balance by up-weighting the underrepresented population and down-weighting the over-represented population in the complete observations, where the weights can be calculated at the subject-level, or subject-and-time-level, to allow better imputation under different situations. For

subject-level, the probability of missing for each subject denoted as $P\{I(\mathbf{Y}_i \text{ contains NA})\}$ is modeled with the corresponding logistic regression using all working complete static variables and a LASSO penalty is applied for variable selection. Weights are calculated by the estimated inverse probability of being observed

$$w_i = \frac{1}{1 - \hat{P}(\mathbf{Y}_i \text{ contains NA})}.$$

In observational data like EHR, missingness at the subject-and-time-level happens sporadically under usual circumstances; that is, missingness can happen at any time point for a patient. Similarly, subject i at time j will be weighted by $w_{i,j} = 1/[1 - \hat{P}(Y_{i,j} \text{ is missing})]$, where $\hat{P}(Y_{i,j} \text{is missing})$ is estimated by a generalized linear mixed model (GLMM) and LASSO penalty is applied for variables selection. After estimating the weights for the observed records, we construct the imputation model for each target longitudinal variable with a weighted linear mixed model. The linear mixed model with random intercept follows:

$$Y_{\cdot,\cdot,k} = \mathbf{X}^{*\mathrm{T}}\boldsymbol{\beta} + \mathbf{Z}^{\mathrm{T}}\mathbf{b} + \boldsymbol{\epsilon},$$

where $Y_{\cdot,\cdot,k}$, $k \in \{1, \ldots, m_l\}$ is the target longitudinal outcome, $\mathbf{X}^*$ are the selected significant predictors, $\mathbf{Z}$ is the design matrix for random effects, $b_i \sim N(0, \sigma_b^2)$, and $\epsilon_i \sim N(0, \sigma^2)$.

Accordingly, $\mathrm{Var}(Y_{\cdot,\cdot,k}) = \sigma^2(\mathbf{Z}\mathcal{T}\mathbf{Z}^{\mathrm{T}} + I) = \sigma^2\mathbf{H}$, with $\mathcal{T} = (\sigma_b^2/\sigma^2)I_{n\times n}$. We estimate the imputation model by optimizing the weighted log likelihood for complete cases:

$$L^w = C - \frac{1}{2}\log(|\mathbf{H}|) - \frac{1}{2}n\log(\sigma^2) - \frac{1}{2\sigma^2}(Y_{\cdot,\cdot,k} - \mathbf{X}^*\beta)^{\mathrm{T}}\mathbf{W}^{*\mathrm{T}}\mathbf{H}^{-1}\mathbf{W}^*(Y_{\cdot,\cdot,k} - \mathbf{X}^*\beta),$$

where C is a constant and

$$W^* = \begin{bmatrix} \sqrt{w_{1,1}} & 0 & \cdots & 0 \\ 0 & \sqrt{w_{1,2}} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \sqrt{w_{n,J_n}} \end{bmatrix}$$

is a $(\sum_{i=1}^n J_i) \times (\sum_{i=1}^n J_i)$ diagonal matrix. We obtain each stochastic imputation with $X_i^*\hat{\boldsymbol{\beta}} + \hat{b}_i$, where $\hat{b}_i$ is randomly sampled from $N(0, \hat{\sigma}_b{}^2)$.

Imputation model under MCAR.—Under MCAR, we propose to employ two modeling options GLMM[43] or RE-EM tree[44] as imputation models within the iterative procedure. The two procedures are referred to as DataSifter II GLMM and DataSifter II RE-EM. Note that GLMM can handle various data types, including continuous, binary, and count data,

whereas the RE-EM tree is an effective algorithm for continuous measurements. For the DataSifter II GLMM, variable selection is conducted separately with GLMM LASSO. Here we perform a grid search for the regularization parameter and use Bayesian information criterion to select the best model. Since an appropriate starting point is crucial for model convergence, DataSifter II incorporates two methods to initiate the parameters when fitting GLMM LASSO. The first method estimates all the parameters by glmmPQL, which is using pseudo-likelihood. GlmmPQL estimates the mean and variance parameters iteratively with maximum likelihood assuming normality.[45] It approximates the true likelihood with a strong normality assumption, but provides a computationally efficient way of estimating initial regression parameters. When the signal is sparse and the GLMM algorithm does not converge with the glmmPQL initial values, we may consider initialization using zeros or another user-specified initialization.

Selected variables are denoted as $\mathbf{X}^* = (\mathbf{X}_1^*, \ldots, \mathbf{X}_s^*)$, which may come from longitudinal variables other than the target and static variables. We fit the following prediction model for every missing longitudinal variable

$$\eta_{i,j} = g\{E(Y_{i,j})\} = \mathbf{X}_{i,j}^{*\mathrm{T}}\boldsymbol{\beta} + \mathbf{Z}_{i,j}^{\mathrm{T}}\boldsymbol{\gamma}_i,$$

where $g(\cdot)$ is a known link function, $\mathbf{Z}_{i,j}$ is the designed matrix for random effects $\boldsymbol{\gamma}_i$, $\boldsymbol{\gamma}_i \sim N(0, D)$, $i = 1, \ldots, n$ represents subjects, and $j = 1, \ldots, J_i$ are different time points.

After estimating $\boldsymbol{\beta}$ and D using observed data, we impute the missing values by randomly sampling $\hat{\boldsymbol{\gamma}}_i \sim N(0, \hat{D})$ and then obtain a *Best Linear Unbiased Prediction* imputation prediction $g^{-1}(\mathbf{X}_{i,j}^{*\mathrm{T}}\hat{\boldsymbol{\beta}} + \mathbf{Z}_{i,j}^{\mathrm{T}}\hat{\boldsymbol{\gamma}}_i)$ for $Y_{i,j,k}$ with missing values.

DataSifter II RE-EM provides an alternative robust obfuscation for continuous time-varying measurements. RE-EM tree model combines the tree-based estimation for fixed effects and parametric estimation for random effects.[44] RE-EM tree is a semi-parametric generalization of the linear mixed effect model

$$Y_{i,j} = f(X_{i,j,1}, \ldots, X_{i,j,m_s}) + \mathbf{Z}_{\mathbf{i},j}{}^{\mathrm{T}}\boldsymbol{\gamma}_i + \epsilon_{i,j},$$

where $(\epsilon_{i,1}, \ldots, \epsilon_{i,J_i})^{\mathrm{T}} \sim N(0, R_i)$, $\boldsymbol{\gamma}_i \sim N(0, D)$, $f(\cdot)$ function denotes a regression tree in the previous model, and R_i records the variance-covariance structure for the ith error term. RE-EM tree enjoys the capability of modeling the non-linear trend for fixed effects so that variable selection can be avoided. Parameter estimation for the RE-EM tree follows a two-step procedure: First, when estimating $f(\cdot)$, RE-EM tree adapts the CART tree algorithm. Assuming that $\boldsymbol{\gamma}_\mathbf{i}'\mathbf{s}$ are known and equal to the current estimate $\boldsymbol{\gamma}_i^{(r)}$, the outcome of $f(\cdot)$ is $Y_{i,j} - \mathbf{Z}_{\mathbf{i},j}{}^{\mathrm{T}}\boldsymbol{\gamma}_i^{(r)}$. Fitting the tree is a binary recursive procedure that splits the whole population into similar subgroups. The default minimum number of subjects in the terminal node is set to 20. Also, a new split will be made when the reduction in sum of squares between the individual outcome and group average is <1%. In other words, we

set the *complexity parameter* (cp) to be 0.01. To avoid overfitting, pruning is done by 10-fold cross-validation after the initial fitting. The final tree is selected with the largest *cp* value within one standard error above the minimized 10-fold cross-validated error. Second, extract the random effect estimates $\hat{\boldsymbol{\gamma}}_i$ from a linear mixed model that regress Y_{ij} on $\hat{f}\left(X_{i,j,1},\ldots,X_{i,j,m_s}\right)$.

Create partially synthetic time-varying data

Using the proposed iterative imputation tool, we can create partially synthetic time-varying data by handling potential missing in **Y**, generate artificial missingness and impute back.

Similar to the preprocessing step for static variables, we intend to initiate the process with a complete dataset containing both static and time-varying data. The working complete static data **X** can be obtained from missForest imputation. If missing values exist in the time-varying variables, we pre-process the data using the proposed imputation algorithm under MAR assumption. In practice, we can choose the subject-level or subject-and-time-level propensity model for missing based on different missing patterns, that is, we model $P\{I(\mathbf{Y}_i \text{ contains NA})\}$ if missingness usually happens at subject-level and $P\{I(\mathbf{Y}_{ij} = \text{NA})\}$ if missingness happens sporadically. Since the true missing values are unknown, we finalize the imputation result for a target variable k when $\|\mathbf{Y}_{obs_k,k} - \widehat{\mathbf{Y}}^{(r)}_{obs_k,k}\|_1 / \|\mathbf{Y}_{obs_k,k}\|_1 < \epsilon$ at current iteration r for a pre-specified tolerance level ϵ, where $k \in \{1,\ldots,m_l\}$.

Following the initial imputation, we start Sifting by introducing artificial random missing values to the longitudinal variables in the working complete dataset. Such randomly generated missingness follows an MCAR missing mechanism, which guarantees that the unweighted complete-case analysis is bias-free. We then impute the missing variables one by one with the proposed imputation procedure under MCAR with a data-driven choice of either the parametric or semi-parametric imputation model.

Implementation of DataSifter II

We use Algorithm 1 to summarize the proposed imputation method for time-varying variables. The algorithm terminates the imputation for variable $Y_{\cdot,\cdot,k}$ at the rth iteration when

$$\frac{\|\mathbf{Y}_{obs_k,k} - \widehat{\mathbf{Y}}^{(r)}_{obs_k,k}\|_1}{\|\mathbf{Y}_{obs_k,k}\|_1} < \epsilon$$

at a tolerance level ϵ. When imputing artificial missing data, the original missing values are given. Hence, we have an alternative criterion for determining if the imputation results are finalized:

$$\frac{\|\mathbf{Y}_{mis_k,k} - \widehat{\mathbf{Y}}^{(r)}_{mis_k,k}\|_1}{\|\mathbf{Y}_{mis_k,k}\|_1} < \epsilon$$

Algorithm 1.

Time-varying data missing imputation algorithm.

Input: complete static variables $\mathbf{X} \in \mathbb{R}^{(\sum_i \mathbf{J_i}) \times \mathbf{m_s}}$, time-varying variables $\mathbf{Y} \in \mathbb{R}^{(\sum_i \mathbf{J_i}) \times (\mathbf{m_l})}$, missing mechanism (MAR or MCAR), imputation model (GLMM or RE-EM tree), and tolerance level ϵ.

Initially impute the missing cells in Y with a combination of last value carry forward and next value carry backward.

Sort the m_l variables in **Y** based on missing rate so that the first variable in **Y** has the least missing and the last variable in **Y** has the most missing.

Create a list of missing variable indexes vlist = {k_m, ..., m_l} where missingness appear from the knth variable.

Sort the m_l variables in **Y** based on missing rate so that the first variable in **Y** has the least missing and the last variable in **Y** has the most missing.

Create a list of missing variable indexes vlist = {k_m, ..., m_l} where missingness appear from the k_{th} variable.

repeat

  for $k \in$ vlist

    Separate data in four groups with

$$\left[\frac{Y_{obs_k,k}}{Y_{mis_k,k}} \middle| \frac{Y_{obs_k,-k} X_{obs_k,.}}{Y_{mis_k,-k} X_{mis_k,.}}\right]$$

    if missing mechanism = MAR

      Construct propensity score model for missingness and calculate the inverse probability of missing for records with row indexes obs_k.

      Perform variable selection with LMM with LASSO using $Y_{obs_k,k}$ as outcome and $\left[\mathbf{Y}_{obs_k,-k}, \mathbf{X}_{obs_k,.}\right]$ as potential predictors.

      Fit inverse probability weighted LMM with $Y_{obs_k,k}$ as outcome and selected predictors as covariates.

      Impute missing values $Y_{mis_k,k}$ using the weighted LMM.

    else

      if imputation model = GLMM

        Fit GLMM with LASSO regularization regressing $Y_{obs_k,k}$ on $\left[\mathbf{Y}_{obs_k,-k}, \mathbf{X}_{obs_k,.}\right]$.

      else

        Fit RE-EMtree regressing $Y_{obs_k,k}$ on $\left[\mathbf{Y}_{obs_k,-k}, \mathbf{X}_{obs_k,.}\right]$.

      end if

      Impute missing values $Y_{mis_k,k}$ using the fitted imputation model.

    end if

  end for

  iteration = iteration + 1

  if Imputation finalizing criteria at tolerance level ϵ has met

    Exclude k from vlist.

  end if

until iteration > maxit or the length of vlist ≤ 1.

Output *Sifted* time-varying variables $\mathbf{Y^s}$.

Algorithm 2.

Outlines the complete DataSifter II implementation method.

1: **Input:** static variables $\mathbf{X} \in \mathbb{R}^{\mathbf{n} \times \mathbf{m_S}}$, time-varying variables $\mathbf{Y} \in \mathbb{R}^{(\sum_{\mathbf{i}} \mathbf{J_i}) \times (\mathbf{m_I})}$, imputation model *I,* DataSifter I obfuscation level *L,* percent of artificial missing to introduce *a,* and tolerance level ϵ.

2: Operate DataSifter I on the static variables under obfuscation level *L* and obtain complete static variables. For patient *i* create J_i, replicates of the working complete static record to create $\mathbf{X}^s \in \mathbb{R}^{\sum_{\mathbf{i}} \mathbf{J_i} \times \mathbf{m_S}}$.

3: Operate Algorithm 1 ($\mathbf{X^S}$, **Y,**MAR,GLMM,ϵ) to impute possible original missingness in **Y**.

4: Introduce random missingness to a\% of data values for data obfuscation purpose among the m_I time-varying variables in the working data and obtain data with artificial missingness denoted as **Y***. Record real values of the missing cells.

5: Operate Algorithm 1 ($\mathbf{X^S}$, **Y***,MCAR,I,ϵ) to obtain sifted time-varying variables $\mathbf{Y^S}$.

6: **Output:** A single complete and *sifted* dataset $\left[\mathbf{X^S},\ \mathbf{Y^S}\right] \in \mathbb{R}^{(\sum_{\mathbf{i}} \mathbf{J_i}) \times (\mathbf{m_S} + \mathbf{m_I})}$.

Residual diagnostics

The residuals or errors introduced by DataSifter II obfuscated values follow a mixture distribution. When the final imputation model for variable $Y_{\cdot,\cdot,k}$ at its last iteration r_k satisfies $\|Y_{mis_k,k} - \hat{Y}^{(r_k)}_{mis_k,k}\|_1 / \|Y_{mis_k,k}\|_1 < \epsilon$, the summation of absolute residuals is controlled by ϵ and the original observed values. On the other hand, the residual is $Y_{mis_k,k} - \hat{Y}^{(\text{maxit})}_{mis_k,k}$. Thus, the model fitting can be assessed with the observed versus predicted values diagnostic plot. First, we subset the obfuscated cells. Then, we plot the observed values on the vertical axis and the predicted values on the horizontal axis. When the two values are only different by a small error term, points in the diagnostic plot will scatter around the diagonal line. If significant outliers are detected, we may consider alternative strategies to remedy these atypical cases, for example, removing outliers from the final shareable dataset to better protect the data utility.

Simulation studies

Simulation setup

In this section, we conduct controlled simulation studies to evaluate the data privacy and utility protection of DataSifter and MI methods. The original simulation data is generated with $n = 500$, or $n = 1000$ subjects, each with J_i time points, where J_i varies from 1 to 10 with equal probability, two longitudinal variables (Y_1 and Y_2), five static independent true predictors (X_1, X_2, . . ., X_5), and $w = 5$ or $w = 20$ white noise variables. The true static predictors are generated by normal distributions with different means and unit variance. The white noise variables are also generated by normal distributions but with different means and larger variances. The longitudinal variable Y_1 is associated with static variables only (X_1, X_2, X_3), and Y_2 is associated with both static (X_4, X_5) and longitudinal (Y_1) variables. We consider linear and non-linear associations when generating Y_1 and Y_2, respectively.

Under the linear association, Y_1 is generated by the following Linear Mixed Model:

$$Y_{i,j,1} = 1 - X_{1,i} - 0.5X_{2,i} - 0.3X_{3,i} + 0.8\text{Visit}_{i,j} + b_{0i} + \epsilon_{i,j},$$

where $i = 1, \ldots, n$ is the indicator for patients, and $j = 1, \ldots, J_i$ is the indicator for time. Here J_i is the total visit number for patient i and $J_i \in \{1, 2, \ldots, 10\}$, b_{0i} is a subject specific random intercept that follows a $N(0, 1)$ distribution, and $\epsilon_{i,j}$ are independent for different time points and follows $N(0, 4)$. Variable Y_2 depends on two static variables and Y_1.

$$Y_{i,j,2} = -15 + 0.2Y_{i,j,1} - X_{4,i} + 0.2X_{5,i} + 2\text{Visit}_{i,j} + b_{1i} + \epsilon_{i,j}.$$

Similarly, $b_{1i} \sim N(0, 1)$ and $\epsilon_{i,j} \sim N(0, 4)$. We know that under random intercept $V(Y_{i,\cdot,1}) = Z_i D Z_i^{\mathrm{T}} + \sigma^2 I_{J_i}$ where $Z_i = 1_{J_i \times 1}$, $D = \text{Var}(b_{0i})$ and $\sigma^2 = \text{Var}(\epsilon_{i,j})$. Thus, $Cov(Y_{i,j,1}, Y_{i,j',1}) = \text{Var}(b_{0i})$ and $\text{Corr}(Y_{i,j,1}, Y_{i,j',1}) = Cov(Y_{i,j,1}, Y_{i,j',1}) / \sqrt{\text{Var}(Y_{i,j,1})\text{Var}(Y_{i,j',1})} = \text{Var}(b_{0i}) / [\text{Var}(\epsilon_{i,j}) + \text{Var}(b_{0i})]$. After some calculations, $\text{Corr}(Y_{i,j,1}, Y_{i,j',1}) = 0.2$ for all i and $j \neq j'$. Similarly, $\text{Corr}(Y_{i,j,2}, Y_{i,j',2}) = 0.2$.

We also consider cases with non-linear relationships. Similar to the linear setting, we construct models with a compound symmetry correlation structure. Our two longitudinal variables are derived by

$$Y_{i,j,1} = 10 + 3\sin(X_{1,i}) - 0.2X_{2,i}^2 - 0.1X_{1,i} \cdot |X_{3,i}| + \text{Visit}_{i,j} + b'_{0i} + \epsilon'_{i,j},$$

and

$$Y_{i,j,2} = 2 + 0.05\sin(Y_{i,j,1}) + 0.4\exp\{\cos(X_{4,i})\} - 0.02Y_{i,j,1} \cdot |X_{5,i}| + 2\text{Visit}_{i,j} + b'_{1,i} + \epsilon'_{i,j},$$

Where $b'_{0i} \sim N(0, 9)$, $b'_{1i} \sim N(0, 16)$, and $\epsilon'_{i,j} \sim N(0, 64)$. We have $\text{Var}(Y_{i,j,1}) = 73$, $\text{Corr}(Y_{i,j,1}, Y_{i,j',1}) = 0.12$, $\text{Var}(Y_{i,j,2}) = 80$, and $\text{Corr}(Y_{i,j,2}, Y_{i,j',2}) = 0.2$, where $j \neq j'$.

The complete data generated by the above procedure will be used to examine different data obfuscation methods. To mimic real-world data, we also consider a scenario where some observations in Y_1 and Y_2 contains missing values, which follow the MAR missing data mechanism. First we define the missing indicator for variable Y_1 to be $M(Y_{i,j,1}) = I(Y_{i,j,1} = NA)$, $\forall i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, J_i\}$ and similarly for Y_2. The original missingness $M(Y_{i,j,1})$ and $M(Y_{i,j,2})$ is generated from the two sets of logistic regressions. The first set considers different probabilities of missingness at individual and time levels. Under this model, we allow partially missing subjects.

$$\text{logit}\left[P\left\{M\left(Y_{i,j,1}\right)=1\right\}\right] = -2+10X_{1,i}-10\text{Visit}_{i,j}+b_i.$$
$$\text{logit}\left[P\left\{M\left(Y_{i,j,2}\right)=1\right\}\right] = 3-4X_{5,i}-12\text{Visit}_{i,j}+b_i',$$

where b_i, $b_i' \sim N(0,\ 1)$. The two models will provide around 20% to 30% missingness for each longitudinal variable.

We compared the performance of four types of synthetic datasets: DataSifter with GLMM generated on complete original data, DataSifter with RE-EM tree generated on complete original data, multiple imputed synthetic data generated on complete original data, and DataSifter with RE-EM tree generated on original data that contains missing. All the *sifted* data are generated without static data obfuscation (L =no obfuscation). We applied the MI method using two-level normal models with homogeneous within-group variances as the imputation model to create multiply imputed partially synthetic datasets, which is implemented in *mice* R package.[46,47] We compared the first three types of synthetic data to assess the obfuscation performance. To demonstrate that DataSifter can successfully handle missingness in the original data, we further show that the decrease in data utility preservation is small after bringing in original missingness under the RE-EM tree imputation method. All synthetic data are generated by randomly introducing 20% artificial missingness in Y_1 and Y_2 and impute the missing cells back. Static variables, including the white noise variables, are not obfuscated. One hundred replications are constructed for each type of synthetic dataset under every simulation setting.

Simulation results

Using the proposed data utility and data PM, we evaluate synthetic datasets generated by DataSifter and MI. Data utility is measured in terms of prediction accuracy and inference based on models trained on synthetic datasets. For prediction accuracy, we construct test sets with identical sample size as the training dataset ($n = 500$ or $n = 1000$ and $J_i \in \{1, \ldots, 10\}$). We then use the predictive models constructed on the synthetic datasets to predict the target longitudinal variables in the test set. Absolute deviance in predicted and observed values of Y_1 and Y_2 are calculated as the prediction error. The model inference is measured by the 95% confidence interval coverage of the true parameter value among the 100 replications under linear association scenario. Data utility is examined using the average PM for the first 100 records of Y_1 and Y_2. As defined in section “DataSifter II technique,” for an obfuscated (replaced) target cell $Y_{i,j,k}$, we use the conditional model fitted by $\mathbf{D}_{-(i,j)}$ to represent intruder’s prior knowledge and $\text{PM}_{i,j,k}$ is the difference between the conditional mean and the observed $y_{i,j,k}$.

The average prediction errors on test data are summarized in Table 1. Based on 100 replications, under most simulation settings, the average test error on Y_1 and Y_2 are similar across different synthetic data generation methods, and these results are indistinguishable from the original data. For Y_2, under the linear association, the MI method provides slightly better prediction accuracy. This result indicates that the parameter estimations with synthetic data generated by DataSifter are relatively accurate. Note that the DataSifter RE-EM tree provides similarly accurate coefficient estimates when the raw datasets have original missing

values, demonstrating the algorithm's ability to handle partially missing sensitive data. Moreover, stable results are observed under linear and nonlinear associations, training sample sizes, and noise levels, which suggests that our proposed imputation method is robust.

Since the proposed imputation method is aimed at minimizing imputation error rather than accounting for the uncertainty of the missing values, the 95% confidence intervals constructed on *sifted* datasets are narrower than the original data. As shown in Table 2, for variable Y_1, the MI method achieves desired 95% coverage while the DataSifter GLMM achieves 89–98% accuracy. Due to the slower convergence rate of non-parametric estimations and non-linear model specification, the synthetic datasets generated by DataSifter RE-EM have smaller CI coverage compared to the previous two methods, ranging from 76% to 94%. For Y_2, the CI coverage for X_4 (one of the static predictors) is relatively small under the DataSifter methods (43–68%). Nevertheless, we observe that the DataSifter GLMM method (87% and 84%) provides much better CI coverage for predictor Y_1 than the MI method (39% and 21%). Y_1 is a special predictor for Y_2, as it includes artificial missing. This result validates the utility benefit of the proposed iterative imputation method for the cases where important predictors have extensive missingness. DataSifter II better preserves the outcome and predictor relationship because it updates the missing predictor information during each iteration. In contrast, the MI method provides imputations solely based on complete records and may suffer from missing predictors, that is, a smaller number of complete records. Therefore, the CI coverages from the DataSifter RE-EM tree method are expected to be smaller than the MI method for longitudinal predictors that contain artificial missing. However, given the same percentage of artificial missingness, the proposed algorithms' benefit in CI coverage diminishes under larger sample sizes.

The average PM for different synthetic datasets is illustrated in Figure 2. Each boxplot records the distribution of average PM for Y_1 and Y_2 over 100 replications among the first 100 records. Under all scenarios, the two DataSifter methods have similar PM values and distributions. The MI method offers significantly lower PM, see Figure 2. In fact, when introducing 20% artificial missingness to the longitudinal variables, the average mean PM is around 5.25 times higher in *sifted* datasets compared to multiply imputed datasets. This result provides empirical evidence for the PM improvement derivation, see section "DataSifter II technique." Compared to multiple imputed datasets, *sifted* datasets have at least 1/a times higher average PM, where a is the percent of the introduced artificial missing values in the data.

Biomedical application

The MIMIC-III represents a sizable single-center database that provides patients' medical records in a large tertiary care hospital between 2001 and 2012. MIMIC-III data stores information related to patients' admission, including vital signs, medications, laboratory measurements, length of stay, survival data, and more.[48] We consider a subset of 7080 patients who had at least two visits to the hospital who contributed 17,594 hospital admission records with demographic variables, including insurance type, gender, race, age, marital status, and death after admission. Admission information such as insurance type,

admission type, and month of admission is also available. MIMIC-III contains de-identified or coded data that is considered free of protected health information. However, the data request process, including taking an online course and submitting an application with specific research topics and requested information, can still take more than three weeks. Using the data for any rigorous scientific investigation requires the researcher to go through a time-consuming data request procedure, while in the end, the investigation may find no significant results. DataSifter II allows a quicker turnaround for checking the potentials of research hypotheses. For example, we want to investigate the association between length of stay in tertiary care hospitals and Medicaid insurance type controlling other patient demographic variables using the MIMIC-III data. We illustrate how to use the *sifted* data to answer our initial research question and evaluate both utility and privacy protection performance in the *sifted* MIMIC-III data. We also compare the synthetic MIMIC-III datasets generated by DataSifter II with that of the MI method. A linear mixed effect model is used to regress the length of hospital stay on patient characteristics. The privacy protection effort is measured by the *privacy measurement* (PM) for age and length of hospital stay among the first 100 records.

First, we obfuscate the following longitudinal variables: (1) length of stay, (2) month of admission, (3) death after visit, and (4) age at visit. Next, we consider generating two types of Sifted data: with (L =medium) or without (L =no obfuscation) static data obfuscation using DataSifter I. By using the DataSifter II protocol, we introduce 20% missingness in the longitudinal variables specified above to obtain the first type of *sifted* data without static obfuscation. The RE-EM tree model is used as the imputation model because of its more flexible mean structure. Then, we generate the second type of *sifted* data with further obfuscation on the static variables using DataSifter I under the medium level of obfuscation, which entails two rounds of artificial missing introduction and imputation, each one randomly obfuscating 25% of the cells. The other setting for the medium obfuscation level defines neighbors as the cases with the closest top 5% distance and swaps 60% of the features with a neighboring case. As a comparison, we also create partially synthetic data with MI based on 20% random, artificial missingness on the four longitudinal variables. The MI dataset and the Sifted dataset without static obfuscation have the same amount of data cells being replaced in each replication. The Sifted dataset with static obfuscation has the highest level of privacy protection among the three by altering an extra 25% of the cells in the static data. Fifty replications are generated for each type of partially synthetic data.

Next, we compare the model parameter estimates between models fitted on the original data and on the three different types of synthetic data, assuming the following linear mixed effect model:

$$\begin{aligned}\text{Length of stay}_{i,j} = {} & \beta_0 + \beta_1 \text{Age} + \beta_2 \text{Medicaid} \\ & + \beta_3 \text{Private insurance} + \beta_4 \text{White} \\ & + \beta_5 \text{Black} + \beta_6 \text{Male} \\ & + \beta_7 \text{Emergency admission} \\ & + \beta_8 \text{Urgent admission} \\ & + \beta_9 \text{Single} + \beta_{10} \text{English language} \\ & + \beta_{11} \text{Visit}_{i,j} + b_{1i} \text{Visit}_{i,j} + \epsilon_{i,j},\end{aligned}$$

where $b_{1i} \sim N\left(0, \sigma_{b1}^2\right)$, and $\epsilon_{i,j} \sim N(0, \sigma^2)$.

Results in Figure 3 show that the DataSifter II provides much better privacy protection than the MI method with a small loss in data utility. Medicaid is not associated with the length of stay in any synthetic and original data fitted models. According to Figure 3(A), most of the mean PM by row (record) are below five among the 50 multiply imputed synthetic datasets. The average PM is 0.33 for age at admission and 1.53 for the length of hospital stay. The DataSifter method provides a significant improvement in terms of PM with 14.87 and 8.17 as the average PMs for age and hospital stay, respectively. We can also infer from Figure 3(A) that the mean PMs can vary considerably from row to row in *sifted* datasets without static obfuscation.

Figure 3(B) illustrates the deviance of parameter estimates between the model fitted with three types of synthetic dataset and the original linear mixed model. Only significant parameter estimates are shown in the plot. The box plots represent the empirical CIs or the distribution of $\hat{\beta}$'s among 50 replicates. The black dots and purple intervals illustrate the coefficient estimates and CIs from the original model. According to Figure 3(B), all the empirical CIs from MI and DataSifter without static obfuscation overlap with the CIs acquired from original data. The MI-created synthetic datasets provide the most accurate $\hat{\beta}$'s that align closely with the original estimates and a small estimation bias is observed for the *sifted* data without static obfuscation. Five out of seven empirical CIs from the model constructed from the *sifted* data with static obfuscation have overlapped with the original CIs. The results suggest that the data utility is well preserved in *sifted* datasets after intensive obfuscation.

Since none of the fitted models obtained from the *sifted* datasets show that β_2 is significantly different from zero, researchers who are interested in the relationship between Medicaid and length of hospital stay may conclude "no statistical association" from anyone of the *sifted* datasets presented in the simulation study.

Discussion

The reported results shown above demonstrate that the DataSifter with Time-varying Correlated Data (DataSifter II) technique balances between maintaining the energy of the original data (preserves information or data utility) while simultaneously introducing a level of privacy protection safeguarding against re-identification of sensitive information contained in the archive. The simulation results based on introducing 20% artificial missingness suggest that data utility is better preserved for longitudinal variables that depend only on static variables (Y_1) compared to variables that depend on both static and longitudinal variables (Y_2). The two imputation method options, GLMM and RE-EM tree, provide accurate and computationally efficient imputations for Y_1 and Y_2 under linear and nonlinear generative models. The RE-EM tree method is efficient computationally when the number of longitudinal variables is large and the number of subjects is small compared to the number of variables in the data. According to both simulation and application results, DataSifter II provides extended privacy protection with moderate utility loss in terms of CI coverage compared to the MI method.

The goal of the proposed algorithm is to support preliminary hypothesis testing on privacy-aware partially synthetic datasets. DataSifter II does not provide the best data privacy protection like differentially private algorithms or the best data utility preservation like MI. We aim to balance the level of perturbation introduced to the original data and the quality to provide statistical inference. This is to ensure maximal data utility under different levels of privacy protection needs. For example, a synthetic dataset for internal use requested by a trusted research institute might have a weaker privacy protection requirement than a synthetic dataset for educational purposes. Accurate quantification of both criteria is critical for achieving the goal. Hence, we proposed and applied the PM in this paper that provides record-level disclosure risk of the longitudinal synthetic data, assuming the intruder has maximal prior knowledge about the original data. Data utility corresponding to each level of obfuscation is then examined by the deviance in inference based on a pre-defined model.

Section “Data utility measurement” mentioned that the data governor could consider using a pre-defined parametric model for utility purposes, such as regressing summary variables like comorbidity score on other variables. One might argue that one model does not cover all inference-related use cases. In practice, we can propose multiple parametric models for better inference testing. It could be challenging to propose a comprehensive set of testing models for general EHR related to numerous diseases. However, when the data request is explicit about a specific disease, the data governor may consider models related to the context.

The data obfuscation for static variables was done by the original DataSifter I method, which includes the per record swapping operation ensuring record-level perturbation (some of the data cells among the static variables are altered) for any non-isolated subject. Our application shows that this extra level of protection on static variables only introduces a small bias in model-based statistical inference. It is almost impossible to apply the swapping operation on time-varying variables without damaging the within-subject correlation. As a result, some of the timevarying variables for a specific record might be untouched in the synthetic dataset due to chance when the percent of artificial missingness is small. We can consider multiple iterations of the DataSifter II operation and adaptively assign artificial missing data to guarantee obfuscation in each record’s time-varying variables.

Depending on the specific data characteristics, the DataSifter II performance may be impacted in terms of its efficiency and balance between privacy protection and utility reservation. We employed a linear (GLMM) model and a tree (RE-EM tree) structured model to approximate the distribution for each longitudinal variable conditional on other variables in the data. The utility preservation for each longitudinal variable may be affected by (1) the complexity of the relationship, (2) empirical variance of the target time-varying variable, (3) the data type of the predictors, and (4) alternative within-subject covariance structures.

The DataSifter II algorithm provides data governors and researchers with a semi-automated and reliable framework for sensible information exchange. Despite perturbing individual-level records, the overall *sifted* time-varying data shares similar population-level information with the original process. DataSifting allows data owners to create pseudo populations with

a custom level of obfuscation, meeting different data sharing needs. This method provides a rapid and effective information exchange process facilitating research hypotheses testing (confirmatory analytics) and data-driven discovery (exploratory analytics). Many biomedical research and development partnerships may benefit from the DataSifter II technology to conduct advanced trans-disciplinary research and translate fundamental science advances into clinical practice. The proposed algorithm can also be useful beyond the health and biomedical domains, which in this study represent the primary target application areas. For instance, statistical obfuscation and generation of longitudinal synthetic data could be very useful in studies of insurance policies and claims, census records, geopolitical information, social justice, and environmental data. Many dynamic processes might contain protected personal, regulated governmental, or IP organizational information, which require guarding against inappropriate (mis)use. In such cases, DataSifter II can be utilized to simulate realistic records to meet different requirements for data privacy, security, and utility in various applications. We have shared the DataSifter II R package on our GitHub repository https://github.com/SOCR/DataSifterII.

Acknowledgements

We thank the Colleagues at the Michigan Institute for Data Science (MIDAS), the Michigan Neuroscience Graduate Program (NGP), and the Statistics Online Computational Resource (SOCR) who provided valuable suggestions and constructive recommendations. Special thanks to Lu Wei and Brandon Cummings at the University of Michigan for productive discussions, ideas, and contributions.

Funding

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was supported in part by the National Science Foundation (NSF) grant nos. 1916425, 1734853, 1636840, 1416953, 0716055, and 1023115, and by the National Institutes of Health (NIH) grant nos. P20 NR015331, U54 EB020406, P50 NS091856, P30 DK089503, UL1TR002240, R01CA233487, R01MH121079, R01MH126137, and T32GM141746.

References

1. Caulfield T, Harmon SH and Joly Y. Open science versus commercialization: A modern research conflict? Genome Med 2012; 4: 17. [PubMed: 22369790]
2. Davis PM. Open access, readership, citations: A randomized controlled trial of scientific journal publishing. FASEB J 2011; 25: 2129–2134. [PubMed: 21450907]
3. McKiernan EC, Bourne PE, Brown CT et al. Point of view: How open science helps researchers succeed. Elife 2016; 5: e16800. [PubMed: 27387362]
4. Piwowar HA, Day RS and Fridsma DB. Sharing detailed research data is associated with increased citation rate. PLoS ONE 2007; 2: e308. [PubMed: 17375194]
5. Tennant JP, Waldner F, Jacques DC et al. The academic, economic and societal impacts of open access: An evidence-based review. F1000Research 2016; 5: 632. [PubMed: 27158456]
6. Dinov ID. Volume and value of big healthcare data. J Med Stat Inf 2016; 4: 1–7.
7. Simon P Analytics: The agile way. UK: John Wiley & Sons, 2017.
8. Office of the National Coordinator for Health Information Technology. Individuals' perceptions of the privacy and security of medical records and health information exchange. 2019. https://dashboard.healthit.gov/quickstats/pages/consumers-privacy-security-medical-record-information-exchange.php.
9. Hall DE, Hanusa BH, Stone RA, et al. Time required for institutional review board review at one veterans affairs medical center. JAMA Surg 2015; 150: 103–109. [PubMed: 25494359]

10. Casey JA, Schwartz BS, Stewart WF, et al. Using electronic health records for population health research: A review of methods and applications. Annu Rev Public Health 2016; 37: 61–81. [PubMed: 26667605]
11. Keegan TH, Kurian AW, Gali K, et al. Racial/ethnic and socioeconomic differences in short-term breast cancer survival among women in an integrated health system. Am J Public Health 2015; 105: 938–946. [PubMed: 25790426]
12. Ayoade G, El-Ghamry A, Karande V, et al. Secure data processing for IoT middleware systems. J Supercomput 2019; 75: 4684–4709.
13. Hong Z, Li Z and Xia Y. Sdvisor: Secure debug enclave with hypervisor. In: 2019 IEEE international conference on service-oriented system engineering (SOSE). IEEE, San Francisco East Bay, CA, USA: IEEE, 2019, pp. 209–2095.
14. Rosenbaum S Data governance and stewardship: Designing data stewardship entities and advancing data access. Health Serv Res 2010; 45: 1442–1455. [PubMed: 21054365]
15. Bajaj S and Sion R. Trusteddb: A trusted hardware-based database with privacy and data confidentiality. IEEE Trans Knowl Data Eng 2013; 26: 752–765.
16. Baumann A, Peinado M and Hunt G. Shielding applications from an untrusted cloud with haven. ACM Trans Comput Syst 2015; 33: 8.
17. Gentry C and Boneh D. A fully homomorphic encryption scheme, vol. 20, Stanford: Stanford University Stanford, 2009.
18. Kanna GP and Vasudevan V. A fully homomorphic–elliptic curve cryptography based encryption algorithm for ensuring the privacy preservation of the cloud data. Cluster Comput 2019; 22: 9561–9569.
19. Wood A, Shpilrain V, Najarian K, et al. Private-key fully homomorphic encryption for private classification. In: International congress on mathematical software. South Bend, IN, USA: Springer, 2018, pp. 475–481.
20. Dwork C Differential privacy: A survey of results. In: 5th International Conference, TAMC 2008, Xi'an, China, April 25–29, 2008. Berlin, Heidelberg: Springer, pp. 1–19.
21. Dwork C and Roth A (2014) The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science 9(3–4): 211–407.
22. Chen R, Xiao Q, Zhang Y, et al. Differentially private high-dimensional data publication via sampling-based inference. In: Proceedings of the 21st ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, NY, USA: Association for Computing Machinery, 2015, pp. 129–138.
23. Jordon J, Yoon J and Van Der Schaar M. Pate-gan: Generating synthetic data with differential privacy guarantees. In: 7th international conference on learning representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019. OpenReview.net, 2019, pp. 1–21.
24. Zhang J, Cormode G, Procopiuc CM, et al. Privbayes: Private data release via bayesian networks. ACM Trans Database Syst 2017; 42: 25.
25. Goodfellow IJ, Pouget-Abadie J and Mirza M. and Bengio Y Generative adversarial networks. arXiv preprint arXiv: 1406.2661.
26. Xie L, Lin K, Wang S, et al. Differentially private generative adversarial network. arXiv preprint arXiv:1802.06739.
27. Yale A, Dash S, Dutta R, et al. Generation and evaluation of privacy preserving synthetic health data. Neurocomputing 2020; 416: 244–255.
28. Rubin DB. Statistical disclosure limitation. J Off Stat 1993; 9: 461–468.
29. Reiter JP and Raghunathan TE. The multiple adaptations of multiple imputation. J Am Stat Assoc 2007; 102: 1462–1471.
30. Raghunathan TE, Reiter JP and Rubin DB. Multiple imputation for statistical disclosure limitation. J Off Stat 2003; 19: 1.
31. Dahmen J and Cook D. Synsys: A synthetic data generation system for healthcare applications. Sensors 2019; 19: 1181.
32. Little RJ. Statistical analysis of masked data. J Off Stat 1993; 9: 407.

33. Reiter JP. Inference for partially synthetic, public use micro-data sets. Surv Methodol 2003; 29: 181–188.
34. Reiter JP and Kinney SK. Inferentially valid, partially synthetic data: Generating from posterior predictive distributions not necessary. J Off Stat 2012; 28: 583.
35. Marino S, Zhou N, Zhao Y, et al. Hdda: Datasifter: Statistical obfuscation of electronic health records and other sensitive datasets. J Stat Comput Simul 2019; 89: 249–271.
36. Rubin DB. Inference and missing data. Biometrika 1976; 63: 581–592.
37. Stekhoven DJ and Bühlmann P. Missforest—non-parametric missing value imputation for mixed-type data. Bioinformatics 2011; 28: 112–118. [PubMed: 22039212]
38. Machanavajjhala A, Kifer D and Abowd J, et al. Privacy: Theory meets practice on the map. In: 2008 IEEE 24th international conference on data engineering. IEEE, 2008, pp. 277–286.
39. Drechsler J and Reiter JP. Accounting for intruder uncertainty due to sampling when estimating identification disclosure risks in partially synthetic data. In: UNESCO chair in data privacy international conference, PSD 2008, Istanbul, Turkey, September 24–26, 2008. Berlin, Heidelberg: Springer, 2008, pp. 227–238.
40. Reiter JP, Wang Q and Zhang B. Bayesian estimation of disclosure risks for multiply imputed, synthetic data. J Priv Confident 2014; 6: 17–33.
41. McClure D and Reiter JP. Differential privacy and statistical disclosure risk measures: An investigation with binary synthetic data. Trans Data Priv 2012; 5: 535–552.
42. Waljee AK, Mukherjee A and Singal AG, et al. Comparison of imputation methods for missing laboratory data in medicine. BMJ Open 2013; 3: 1–7.
43. McCulloch CE and Neuhaus JM. Generalized linear mixed models. In: Armitage P and Colton T (eds), Encyclopedia of biostatistics, vol. 4. Chichester, UK: John Wiley & Sons, Ltd, 2005, pp. 1–5.
44. Sela RJ and Simonoff JS. Re-em trees: A data mining approach for longitudinal and clustered data. Mach Learn 2012; 86: 169–207.
45. Wolfinger R and O'Connell M. Generalized linear mixed models a pseudo-likelihood approach. J Stat Comput Simul 1993; 48: 233–243.
46. Buuren SV and Groothuis-Oudshoorn K. Mice: Multivariate imputation by chained equations in R. J Stat Softw 2010; 45: 1–67.
47. Schafer JL and Yucel RM. Computational strategies for multivariate linear mixed-effects models with missing values. J Comput Graph Stat 2002; 11: 437–457.
48. Johnson AE, Pollard TJ, Shen L, et al. Mimic-III, a freely accessible critical care database. Sci Data 2016; 3: 160035. [PubMed: 27219127]

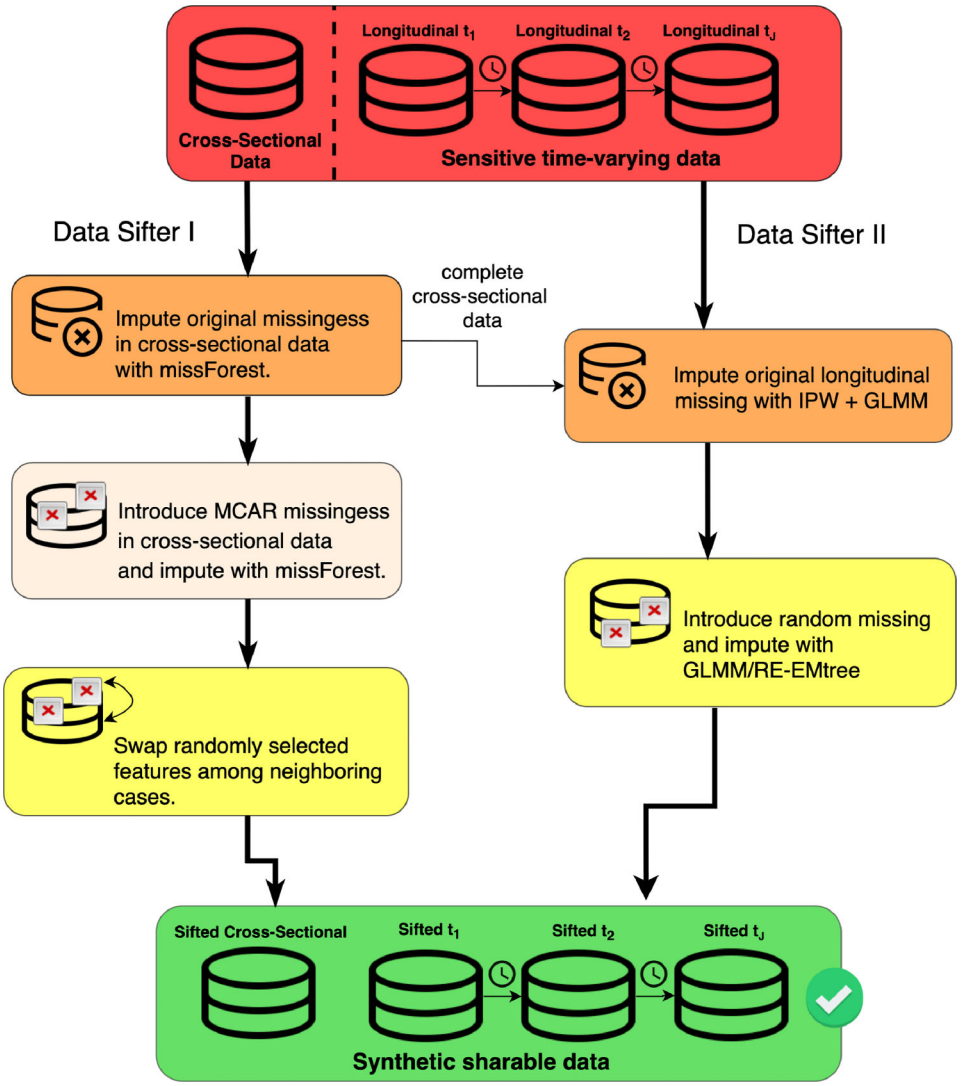

Figure 1.
Graphical workflow depicting the organization of the DataSifter Time-varying Measurements (DataSifter II).

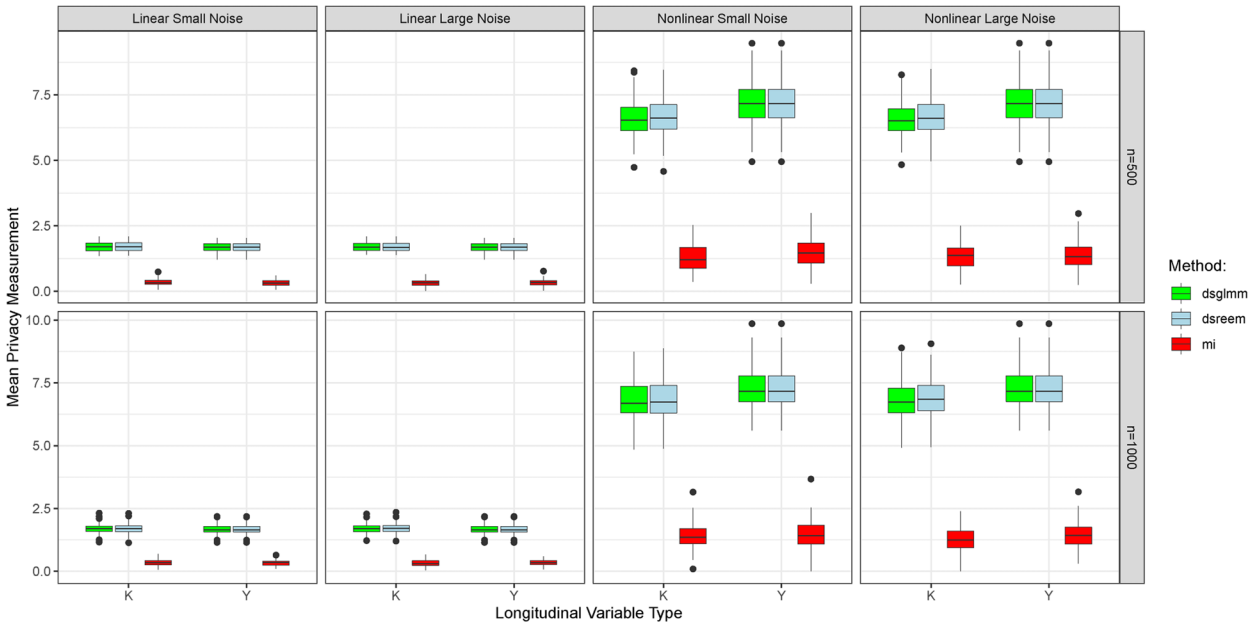

Figure 2.

Average privacy measurement among first 100 rows in the synthetic datasets. The scenario with small noise level contains $w = 5$ and large noise level contains $w = 20$ white noise variables.

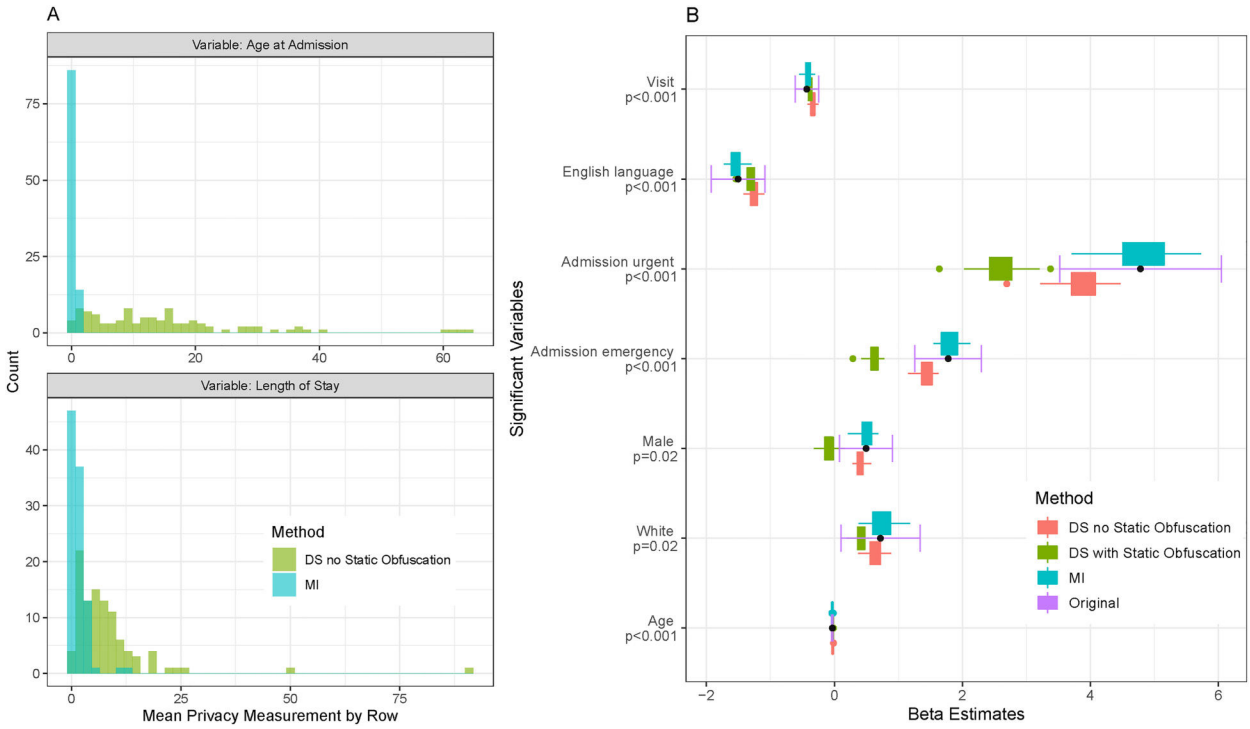

Figure 3.

MIMIC III synthetic data privacy (A) and utility (B) evaluation. Plot A summaries the distribution of mean privacy measurement for age and length of hospital stay for the first 100 rows across 50 synthetic datasets generated by DataSifter (without static obfuscation using DataSifter I) and multiple imputation. Plot B compares the significant coefficient estimates (p-value < 0.05) among the models fitted with original data, and synthetic datasets generated by DataSifter II (with or without static obfuscation) and multiple imputation. The boxes illustrate the distribution of coefficients estimated on 50 synthetic datasets. The black dots and purple intervals are the parameter estimates and confidence intervals from the linear mixed model fitted by the original dataset.

Table 1.

Mean absolute deviation (prediction error) for test dataset based on the model fitted on original and synthetic datasets. The test datasets are generated separately with the same sample size as the training sets.

Training sample	$n = 500$							
Variable	Y_1				Y_2			
Association, Noise level	Linear, $w = 5$	Linear, $w = 20$	Nonlinear, $w = 5$	Nonlinear, $w = 20$	Linear, $w = 5$	Linear, $w = 20$	Nonlinear, $w = 5$	Nonlinear, $w = 20$
Original	1.858	1.858	20.471	20.471	1.903	1.903	7.649	7.649
Multiple imputation	1.851	1.867	20.295	20.289	1.903	1.922	7.693	7.818
DataSifter GLMM	1.903	1.901	20.538	20.525	2.151	2.144	7.949	7.949
DataSifter RE-EM	1.896	1.896	20.503	20.516	2.153	2.149	7.705	7.710
DataSifter RE-EM with original missing	1.871	1.873	20.121	20.101	2.184	2.205	7.728	7.730
Training sample	$n = 1{,}000$							
Variable	Y_1				Y_2			
Association, Noise level	Linear, $w = 5$	Linear, $w = 20$	Nonlinear, $w = 5$	Nonlinear, $w = 20$	Linear, $w = 5$	Linear, $w = 20$	Nonlinear, $w = 5$	Nonlinear, $w = 20$
Original	1.870	1.870	20.746	20.746	1.861	1.861	7.410	7.410
Multiple imputation	1.863	1.883	20.563	20.596	1.863	1.891	7.420	7.520
DataSifter GLMM	1.911	1.911	20.817	20.801	2.100	2.111	7.738	7.741
DataSifter RE-EM	1.911	1.914	20.806	20.797	2.109	2.122	7.456	7.459
DataSifter RE-EM with original missing	1.864	1.864	19.985	19.998	2.245	2.248	7.729	7.732

GLMM: generalized linear mixed model; RE-EM: random effects-expectation maximization.

J Algorithm Comput Technol. Author manuscript; available in PMC 2023 January 20.

Table 2.

Model confidence interval (95%) coverage among 100 replicates. The coverage records the percent of times that CIs from models trained on original and synthetic datasets cover the true parameter estimate.

	n = 500											
Training sample Variable	Y_1						Y_2					
Association, Noise level	Linear, w = 5			Linear, w = 20			Linear, w = 5			Linear, w = 20		
Covariate	X_1	X_2	X_3	X_1	X_2	X_3	X_4	X_5	Y_1	X_4	X_5	Y_1
Original	0.940	0.950	0.960	0.940	0.950	0.960	0.970	1.000	0.960	0.970	1.000	0.960
Multiple imputation	0.940	0.970	0.950	1.000	1.000	1.000	0.960	1.000	0.860	0.980	1.000	0.390
DataSifter GLMM	0.900	0.920	0.930	0.940	0.950	0.930	0.680	0.930	0.890	0.590	0.940	0.870
DataSifter RE-EM	0.870	0.860	0.870	0.900	0.940	0.860	0.670	0.850	0.740	0.600	0.850	0.720
DataSifter RE-EM with original missing	0.690	0.850	0.690	0.660	0.840	0.700	0.490	0.840	0.760	0.440	0.880	0.730
Training sample	n = 1,000											
Variable	Y_1						Y_2					
Association, Noise level	Linear, w = 5			Linear, w = 20			Linear, w = 5			Linear, w = 20		
Covariate	X_1	X_2	X_3	X_1	X_2	X_3	X_4	X_5	Y_1	X_4	X_5	Y_1
Original	0.950	0.960	0.960	0.950	0.960	0.960	0.960	0.970	0.880	0.960	0.970	0.880
Multiple imputation	0.960	0.980	0.960	0.990	1.000	1.000	0.960	0.940	0.660	0.990	0.990	0.210
DataSifter GLMM	0.930	0.960	0.960	0.890	0.980	0.960	0.430	0.860	0.790	0.440	0.910	0.840
DataSifter RE-EM	0.930	0.890	0.760	0.880	0.940	0.770	0.440	0.730	0.510	0.430	0.740	0.550
DataSifter RE-EM with original missing	0.730	0.850	0.760	0.730	0.840	0.770	0.260	0.870	0.520	0.270	0.840	0.530

CI: confidence interval; GLMM: generalized linear mixed model; RE-EM: random effects-expectation maximization.