frontiers in Big Data

METHODS
published: 28 May 2020
doi: 10.3389/fdata.2020.00016

Variational Autoencoder Modular Bayesian Networks for Simulation of Heterogeneous Clinical Study Data

*Luise Gootjes-Dreesbach[1], Meemansa Sood[2,3], Akrishta Sahay[2], Martin Hofmann-Apitius[2,3] and Holger Fröhlich[2,3,4]**

[1] UCB Pharma (UCB Celltech Ltd.), Slough, United Kingdom, [2] Department of Bioinformatics, Fraunhofer Institute for Algorithms and Scientific Computing (SCAI), Sankt Augustin, Germany, [3] Bonn-Aachen International Center for IT, University of Bonn, Bonn, Germany, [4] UCB Pharma (UCB Biosciences GmbH), Monheim am Rhein, Germany

In the area of Big Data, one of the major obstacles for the progress of biomedical research is the existence of data "silos" because legal and ethical constraints often do not allow for sharing sensitive patient data from clinical studies across institutions. While federated machine learning now allows for building models from scattered data of the same format, there is still the need to investigate, mine, and understand data of separate and very differently designed clinical studies that can only be accessed within each of the data-hosting organizations. Simulation of sufficiently realistic virtual patients based on the data within each individual organization could be a way to fill this gap. In this work, we propose a new machine learning approach [Variational Autoencoder Modular Bayesian Network (VAMBN)] to learn a generative model of longitudinal clinical study data. VAMBN considers typical key aspects of such data, namely limited sample size coupled with comparable many variables of different numerical scales and statistical properties, and many missing values. We show that with VAMBN, we can simulate virtual patients in a sufficiently realistic manner while making theoretical guarantees on data privacy. In addition, VAMBN allows for simulating counterfactual scenarios. Hence, VAMBN could facilitate data sharing as well as design of clinical trials.

Keywords: Bayesian Networks, autoencoders, clinical study simulation, longitudinal data, time series data

Edited by:
Enrico Capobianco, University of Miami, United States

Reviewed by:
Shailesh Tripathi, Tampere University of Technology, Finland
Thomas Hartung, Johns Hopkins University, United States

****Correspondence:***
Holger Fröhlich
holger.froehlich@scai.fraunhofer.de

Specialty section:
This article was submitted to Medicine and Public Health, a section of the journal Frontiers in Big Data

Received: *10 October 2019*
Accepted: *16 April 2020*
Published: *28 May 2020*

Citation:
Gootjes-Dreesbach L, Sood M, Sahay A, Hofmann-Apitius M and Fröhlich H (2020) Variational Autoencoder Modular Bayesian Networks for Simulation of Heterogeneous Clinical Study Data. Front. Big Data 3:16. doi: 10.3389/fdata.2020.00016

INTRODUCTION

Clinical studies are important to increasingly base medical decisions on statistical evidence rather than on personal experience. Within a given area of disease, there can exist many studies, and each of these studies has unavoidably certain biases due to inclusion/exclusion criteria or overrepresentation of specific geographic regions and ethnicities. Moreover, usually, neither the same clinical outcome measures nor the same molecular data are systematically collected in different studies of the same disease. Accordingly, compilation of a comprehensive view of a specific disease requires to analyze and compare multiple studies. However, legal and ethical constraints typically do not allow for sharing sensitive patient data beyond summary statistics outside the organization that is the owner, and even within one and the same organization, the same reasons sometimes prevent data sharing. In consequence, there exist data "silos." This is increasingly becoming an issue, as medicine as a whole is becoming more and more driven by the availability of Big Data and their analysis, including the increasing use of Artificial Intelligence (AI) and, in particular, machine learning methods in precision medicine (Fröhlich et al., 2018). While recent developments of federated machine learning techniques are certainly a major step forward (McMahan et al., 2016; Ghosh et al., 2019), these methods do not permit researchers to unbiasedly investigate, mine, and understand data of differently designed clinical studies located within separate organizations. In particular, it should be noted that the usual assumption behind federated machine learning is that data of the same type/format are spread over different organizations. In contrast, clinical studies of patients with one and the same disease conducted by different hospitals or companies usually vary significantly in their design (including study inclusion and exclusion criteria) and measured variables.

Sufficiently realistic simulations of virtual patient cohorts based on AI models trained *within* each data-hosting organization could not only be a mechanism to break data "silos" but also to allow researchers to conduct counterfactual experiments with patients, e.g., in the context of intensive care units (Knab et al., 2016; Chase et al., 2018) or for better design of clinical trials (Lim et al., 2017; Galbusera et al., 2018). Regarding the latter, we should mention that most existing work on virtual trial simulation focuses on modeling of mechanistically well-understood pharmacokinetic and pharmacodynamic processes (Holford et al., 2010; Pappalardo et al., 2018). In contrast, our focus is here on data-driven, model-based simulations of virtual patients across biological scales and modalities (e.g., clinical, imaging) where no or little mechanistic understanding is available and required.

We suggest a generative modeling framework for simulation of virtual patients, which is specifically designed to address the following key features of clinical study data:

- Limited sample size in the order of a few hundred patients
- Highly heterogeneous data with many variables of different distributions and numerical scales
- Longitudinal data with many missing values.

Our novel proposed method [Variational Autoencoder Modular Bayesian Network (VAMBN)] is a combination of a Bayesian Network (Heckerman, 1997) with modular architecture and Variational Autoencoders (Kingma and Welling, 2013) encoding defined groups of features in the data. Due to its specific design, VAMBN does not only allow for generating virtual patients under certain theoretical guarantees for data privacy (Dwork et al., 2006a) but also for simulating counterfactual interventions within them, e.g., a shift by age. Moreover, we demonstrate that one can "learn" the conditional distribution of a feature in one study to counterfactually add it to another one.

Abbreviations: VAMBN, Variational Autoencoder Modular Bayesian Network; BN, Bayesian Network; MBN, Modular Bayesian Network; VAE, Variational Autoencoder; HI-VAE, Heterogenous and Incomplete Data Variational Autoencoder; DAG, directed acyclic graph; MCAR, missing completely at random; MAR, missing at random; MNAR, missing not at random; BIC, Bayesian Information Score; PPMI, Parkinson's Progression Marker Initiative; PD, Parkinson's disease; UPDRS, Unified Parkinson's Disease Rating Scales; ESS, Epworth Sleepiness Scale; RBD, REM sleep behavior disorder; CSF, cerebrospinal fluid.

We evaluate our VAMBN on the basis of two Parkinson's disease (PD) studies, where we show that marginal distributions, correlation structure, as well as expected effects (treatment effect on motor symptoms and difference of clinical outcome measures to healthy controls, respectively) are largely preserved in simulated patients. Moreover, we demonstrate that counterfactual simulation results match general expectations. Finally, we show that VAMBN models capture expected causal relationships in the data.

METHODS

Motivation and Conceptual Idea Behind VAMBN

Our proposed approach rests on the idea of learning a generative model of longitudinal clinical study data within the data-hosting organization, which can then be used to simulate virtual patients that can be shared with the outside world. Our approach combines two classes of generative modeling techniques: Bayesian Networks (BNs) (Heckerman, 1997) and Variational Autoencoders (VAEs) (Kingma and Welling, 2013). BNs are probabilistic graphical models, which represent a joint statistical distribution of several random variables by factorizing it according to a given directed acyclic graph into local conditional statistical distributions. Attractive properties of BNs are as follows:

- Efficient encoding of multivariate distributions
- Interpretability, because the graph structure can be used to represent causal relationships
- A theoretical framework to simulate interventions via the "do" calculus (Pearl, 2000).

Unfortunately, under general conditions, inference within a BN and learning of the graph structure from data are both NP-hard computational problems (Koller and Friedman, 2009). Computationally efficient parameter and structure learning can only be achieved if all random variables follow multinomial or Gaussian distributions. However, this scenario is, in reality, too restrictive for many applications, including clinical study data, where many variables do not follow any known parametric distribution. In addition, the NP hardness of BN structure learning raises severe concerns because clinical study data have often dozens of variables (measured over time). However, the number of patients is typically only in the order of a few hundreds. Hence, the chance to identify the correct graph from these limited data is questionable.

VAEs are a neural-network-based approach that maps input data to a low dimensional latent distribution (typically a Gaussian) through several sequential encoding steps. VAEs are typically trained via stochastic gradient descent to optimize an evidence/variational lower bound (ELBO) on the log-likelihood of the data. VAEs have recently been extended to deal with heterogeneous multimodal and missing data (Nazabal et al., 2018), which is the common situation in clinical studies. VAEs are generative because drawings from the latent distribution can be decoded again. A limitation of VAEs is that in a situation with comparably small data a dense VAE model with several hidden layers could easily overfit. Moreover, interpretation of the neural network models is far more challenging than for BNs.

Our suggested approach aims to combine the advantages of BNs and VAEs while mitigating their limitations (**Figure 1**): Following the idea of module networks (Segal et al., 2003, 2005), we first define modules of variables that group together according the design of the study. For example, demographic features, clinical assessment scores, medical history, and treatment might each form such a module. This means that we assume the grouping of variables into modules to be known and defined upfront. Our aim is then to learn a BN between low a dimensional representation of variables in these modules. We call such a structure as Modular Bayesian Network (MBN). In contrast to Segal et al., we do not use regression trees to represent conditional joint distributions of variables within each module, but Variational Autoencoders for Heterogeneous and Incomplete Data (HI-VAEs) (Nazabal et al., 2018) because they are generative. Each HI-VAE is thus only trained on a small subset of variables, hence significantly reducing the number of network weights compared to a full HI-VAE model for the entire dataset and allowing for applying the well-established "do" calculus for simulating interventions (Pearl, 2000). We call our approach Variational Autoencoder Modular Bayesian Network (VAMBN). Due to its generative nature, VAMBN allows for simulating virtual subjects by first drawing a sample from the BN and second by decoding it through the VAE representing the corresponding module.

We validate virtual patient cohorts by comparing against original patients:

- Marginal distributions of individual variables
- Correlation structures
- Expected differences between patient subgroups, e.g., treated vs. placebo patients.

In the following, we explain the individual steps of our method in more detail, and we discuss how data privacy can be theoretically guaranteed.

Modular Bayesian Networks

The starting point of our proposed approach is a BN describing in a longitudinal manner statistical dependencies between low dimensional representations of groups/modules of variables: We assume that each low dimensional representation is the result of a HI-VAE encoding. We identify low dimensional representations with random variables $X = (X_\upsilon)_{\upsilon \epsilon V}$ indexed by nodes in a directed acyclic graph (DAG) $G = (V,E)$. This means that there is a DAG between low dimensional representations of modules. According to the definition of a BN, the joint distribution $p\left(X_1, X_2, \ldots, X_n\right)$ factorizes according to:

$$p\left(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n\right) = \prod_{\upsilon \in V} p\left[X_\upsilon = x_\upsilon | X_{pa(\upsilon)} = x_{pa(\upsilon)}\right]$$

where $pa(\upsilon)$ denotes the parents of node υ and $x_{pa(\upsilon)}$ their joint configuration (Koller and Friedman, 2009). For a given node υ, we summarize the set of associated conditional probabilities into a parameter vector θ_υ, and these parameter vectors are assumed to be statistically independent for different nodes υ, υ'. Since the BN in our case is defined over low dimensional representations of groups of variables, we here call the structure Modular Bayesian Network (MBN). We use this terminology to discriminate against a BN defined over original input variables (which is more conventional).

In our situation, there exists a subset $\widetilde{X} \subset X$ that is time dependent, i.e., $\widetilde{\mathbf{x}} = \left(\widetilde{x}\left(1\right), \ldots, \widetilde{x}\left(T\right)\right)$ with T being the number of visits. Dynamic Bayesian Networks (Ghahramani, 1998) usually deal with this situation by implicitly unfolding the BN structure over time, i.e., introducing for each visit t a separate copy $\widetilde{X}\left(t\right)$ of $\widetilde{X}$ while requiring that edges always point from time slice t to time slice $t+1$ (corresponding to a first order Markov process). This implicit unfolding assumes a stationary Markov process, i.e., parameters θ do not change with time. In our setting, this assumption is most likely wrong because patients change in their disease outcome during the course of a study, i.e., $p\left(\widetilde{X}\left(t\right) | \widetilde{X}\left(t-1\right)\right) \neq p\left(\widetilde{X}\left(t+1\right) | \widetilde{X}\left(t\right)\right)$. Hence, we here use an unfolding strategy, in which we explicitly use different copies $\widetilde{X}\left(t\right)$ for each time point. In addition, unfolding of the BN structure saves us from modeling the dynamical behavior of the data within the VAE framework [e.g., via LSTM units (Hochreiter and Schmidhuber, 1997)], which would require far more parameters.

In our case nodes (i.e., random variable), either follow a Gaussian distribution (we explain the reasons later), or they could be of categorical nature, i.e., follow a multinomial distribution and not be autoencoded. A restriction we impose at this point is that a discrete node cannot be the child of a Gaussian one. Under this assumption, the conditional log-likelihood of the training data $D = \{x_{\upsilon i} \mid i = 1, \ldots, N,\ \upsilon \in V\}$ given G can be calculated analytically (Andrews et al., 2018):

$$\log p\left(D|G\right) = \sum_{\upsilon \in V} \log p\left(X_\upsilon | X_{pa(\upsilon)}\right)$$

$$\log p\left(X_\upsilon | X_{pa(\upsilon)}\right) = \sum_{c \in C} \ell_c\left(Y_c\right)$$

$$\ell_c\left(Y_c\right) = \frac{n_c}{2}\left(\log\left|\Sigma_c\right| + k \log 2\pi + 1\right) + n_c \log \frac{n_c}{N}$$

where C is the set of possible partitionings of Gaussian variable X_υ according to the configuration of its discrete parents, and n_c is the number of patients in partition c. Note that modeling a Gaussian distribution conditional on discrete parents corresponds to a local ANOVA model. The associated design matrix is denoted as Y_c, and k is the number of columns of that matrix. Σ_c is the covariance matrix. In a similar way, the local log-likelihood for a discrete node X_υ with only discrete parents can be computed. We refer to Andrews et al. (2018) for more details. By considering, in addition, the number of parameters of the MBN, we can use the Bayesian Information Criterion (BIC) to score G with respect to data D. In practice, we make

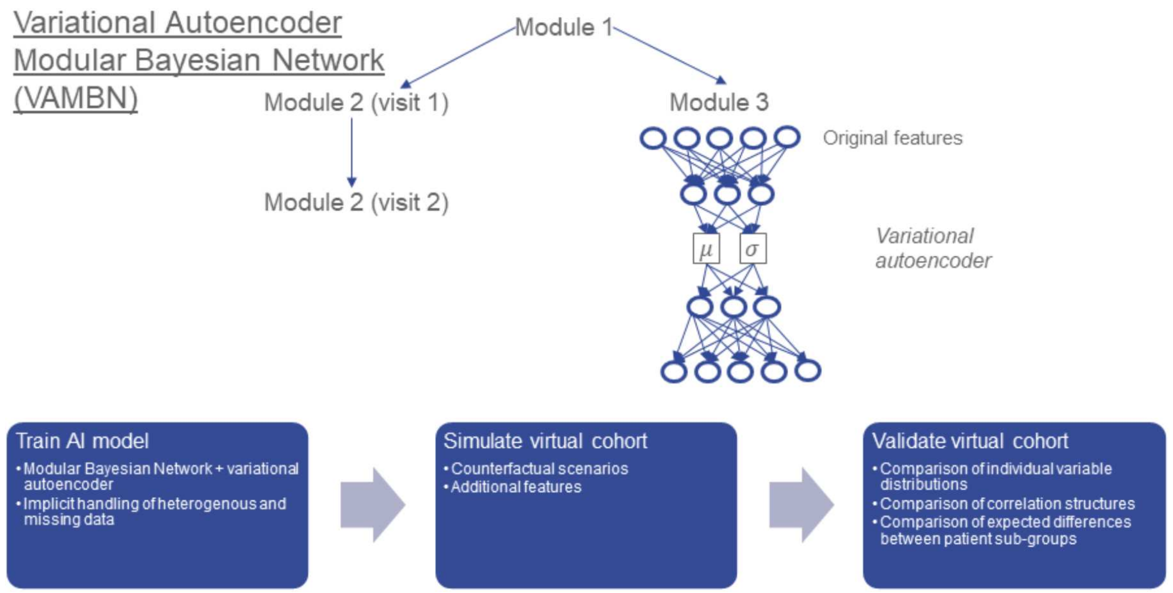


FIGURE 1 | Conceptual overview about Variational Autoencoder Modular Bayesian Network VAMBN) approach: In a first step, a low dimensional representation of known modules of variables is learned via HI-VAEs. The same group of variables (e.g., module 2) may have been assessed at different visits of the study, for example visits 1 and 2. Accordingly, we get a low dimensional representation of module 2 at visit 1 and module 2 at visit 2. In a second step, a Bayesian Network is learned between low dimensional representations of modules, such that the temporal ordering of visits is considered and further constraints explained later are fulfilled. We call the resulting structure a Module Bayesian Network (MBN). The MBN explicitly models missing data at specific visits. Virtual patients can be generated by sampling from the MBN and the HI-VAEs for each module. The "do" calculus allows for simulation of counterfactual interventions into virtual cohorts, such as adding features from another dataset. We carefully validate virtual cohorts by comparison against real patients.

use of the corresponding implementation in R-package bnlearn (Scutari, 2010).

Modeling Missing Data in MBNs

One of the key challenges with longitudinal patient data is missing values, which can result due to different reasons: (a) patients drop out of a study, e.g., due to worsening of symptoms; (b) a certain diagnostic test is not taken at a particular visit (e.g., due to lack of patient agreement), potentially resulting in missing information for entire variable groups; and (c) unclear further reasons, e.g., time constraints, data quality issues, etc. From a statistical point of view, these reasons manifest into different mechanisms of missing data (Rubin, 1976; Kang, 2013):

- Missing completely at random (MCAR): The probability of missing information is not related to either the specific value that is supposed to be obtained or other observed data. Hence, entire patient records could be skipped without introducing any bias. However, this type of missing data mechanism is probably rare in clinical studies.
- Missing at random (MAR): The probability of missing information depends on other observed data but is not related to the specific missing value, which is expected to be obtained. An example would be patient dropout due to worsening of certain symptoms, which are at the same time recorded during the study.
- Missing not at random (MNAR): any reason for missing data, which is neither MCAR or MAR. MNAR is problematic because the only way to obtain unbiased estimates is to model missing data.

Missing values in clinical study data are most likely a combination of MAR and MNAR mechanisms. In general, multiple imputation methods have been proposed to deal with missing data in longitudinal patient data (Kang, 2013). Specifically for MNAR, it has been suggested to explicitly encode systematic missingness of variables or variable groups via dedicated indicator variables (Mustillo and Kwon, 2015). The missing value itself can technically then be filled in by any arbitrary value, e.g., zero.

In our MBN framework, auxiliary variables are fixed parents of all nodes, which contain missing values in a non-random way. There also exist higher level missing data nodes that show whether a participant does not have any data for the entire visit. If the auxiliary variable of a node representing an autoencoded variable group is identical to the missing visit node, the auxiliary variable itself is removed from the network and the node is directly connected to the missing visit node instead. These higher level nodes account for the high correlation between the different auxiliary nodes at a visit. Note that to facilitate modeling in the MBN, auxiliary and missing visit nodes were only introduced for nodes and visits with more than 5 missing data points in total.

MBN Structure and Parameter Learning

Structure Learning

Most edges in the MBN structure are not known and hence need to be deduced from data. Unfortunately, MBN structure learning is an NP hard problem because the number of possible DAGs grows superexponentially with the number of nodes (Chickering et al., 2004). Hence, the search space of possible network structures should *a priori* be restricted as much as possible. We follow two essential strategies for this purpose:

1. We group variables in the raw data into autoencoded modules, as explained above.
2. We impose causal constraints on possible edges between modules.

More specifically, we imposed the following type of constraints:

- Modules of demographic and other clinical baseline features (e.g., age, gender, ethnicity) can only have outgoing edges.
- Modules representing medical history can only depend on the modules mentioned in 1 and biomarkers.
- Modules of imaging features can be related to each other, but they do not influence other modules.
- Modules of clinical outcome measures (e.g., UPDRS) can influence imaging, and they can be mutually correlated with assessment of non-motor symptoms.
- Biomarker modules can influence all modules, except for modules of clinical baseline features.
- Longitudinal measures must follow the right temporal order, i.e., there are no edges pointing backwards in time.
- Empirically proven edges (e.g., the treatment effect on the first maintenance visit in SP513 data) must be reflected in the network structure.
- Auxiliary and missing visit nodes were connected to their respective counterparts at the next time point, accounting for a correlation between these measures over time, e.g., through study dropout.

Accordingly, we blacklisted possible edges that could violate any of these constraints. Structure learning was then conducted via a tabu search (Hong et al., 2016), which is essentially a modified hill climber that is designed to better escape local optima. Each state in the search space represents a candidate MBN structure, which can be scored according to the BIC. This choice was made because score-based search algorithms have empirically found to show a more robust behavior in terms of network reconstruction accuracy than constraint-based methods for mixed discrete/continuous data, specifically for smaller sample sizes (Raghu et al., 2018). In addition, it should be noted that due to the typical small number, variables in the MBN runtime were not a major concern here.

Parameter Learning

Given a graph structure G of a MBN parameters (i.e., conditional probability tables and conditional densities) can be estimated via maximum likelihood. Note that estimation of the conditional Gaussian density for a node V amounts to fitting a linear regression function with parents of V being predictor variables. Conditional probability tables, on the other hand, can be estimated by counting relative frequencies of V taking on a particular value υ.

Variational Autoencoders

VAEs were introduced by Kingma and Welling (2013) and can be interpreted as a special type of Bayesian Network, which has the form $Z \rightarrow X$, where Z is a latent, usually multivariate standard Gaussian, and X a multivariate random variable describing the input data. Moreover, for any sample (x, z), we have $p(x \mid z) = N\left(\mu(z), \sigma(z)\right)$. One of the key ideas behind VAEs is to variationally approximate

$$\log q\left(z|x\right) = \log N\left(z|\mu\left(x\right), \sigma\left(x\right)\right)$$

This means that $\mu(x)$ and $\sigma(x)$ are the mean and standard deviation of the approximate posterior and are outputs of a multilayer perceptron neural network that is trained to minimize for each data point x the ELBO criterion

$$\log\left(x\right) \geq \frac{1}{2}\sum_{j=1}^{D}\left(1 + \log\sigma_j(x)^2 - \mu_j(x)^2 - \sigma_j(x)^2\right) + \sum_{l}\log p\left(x|z^{(l)}\right)$$

where $z = \mu\left(x\right) + \sigma\left(x\right) \odot \in^{(l)}$ with $\in^{(l)} \sim N\left(0, I\right)$. Here, $\odot$ denotes an element-wise multiplication.

Variational Autoencoders for Heterogeneous and Incomplete Data

VAEs were originally developed for homogeneous data without missing values. However, clinical data within one and the same module (e.g., demographics) could contain continuous as well as discrete features of various distributions and numerical ranges, i.e., the data are highly heterogeneous. Moreover, there could be missing values. Recently, Nazabal et al. (2018) extended VAEs to address this situation. Their HI-VAE approach starts from a factorization of the VAE decoder according to

$$p\left(x, z\right) = p(z)\prod_{j} p\left(x_j|z\right)$$

where $x \in \mathbb{R}^D$ denotes a D-dimensional data vector, and $z \in \mathbb{R}^K$ is its K-dimensional latent representation. Furthermore, x_j indicates the jth feature in x. In the factorization, it is further possible to separate observed (O) from missing features (M):

$$p\left(x|z\right) = \prod_{j\in O} p\left(x_j|z\right)\prod_{j\in M} p\left(x_j|z\right)$$

A similar separation is possible in the decoder step. Accordingly, VAE network weights can be optimized by solely considering observed data (input dropout model). Note that the input dropout model is essentially identical to the approach we described earlier for MBNs.

To account for heterogeneous data types, Nazabal et al. suggest to set

$$p\left(x_j|z\right) = p\left(x_j|\gamma_j = h_j\left(z\right)\right)$$

where $h_j\left(\cdot\right)$ is a function learned by the neural network, and γ_j accordingly models data modality specific parameters (e.g., for real-valued data $\gamma_j = \left(\mu_j(z), \sigma_j^2(z)\right)$). Moreover, the authors use batch normalization to account for differences in numerical ranges between different data modalities. Finally, Nazabal et al. do not use a single Gaussian distribution as a prior for z, but a mixture of Gaussians, i.e.:

$$s \sim Categorical(\pi)$$
$$z|s \sim N(\mu(s), I_K)$$

where s is K-dimensional. We refer to Nazabal et al. (2018) for more details about their VAE extension. Importantly, categorical variables s are added to the MBN graph G as parents of variables encoding modules. In practice, we kept K at 1 for all modules, resulting in a single normal distribution for z, with the exception of the demographic data in both studies and the neurological examination in SP513 data. For these modules, K was set to 2. This choice was made after visual inspection of the embeddings for each of the individual variable groups, indicating that for modules containing demographic data and neurological examination, $K = 2$ was the minimal value for which a sufficient fit to the data was possible. This was likely due to the existence of many categorical features among these variables.

VAMBN: Bringing MBNs and HI-VAEs Together

Let $v \in V$ be a node in our MBN and X_v the corresponding random variable. Note that X_υ is a low dimensional embedding/encoding of certain variables in the original input space, A_v. The total likelihood $p(X, A \mid G, \Theta)$ given graph G and model parameters Θ can be written as:

$$p(X, A|G, \Theta) = \prod_{\upsilon \in V} p\left(X_\upsilon|pa(X_\upsilon), \Theta_\upsilon\right) p(A_\upsilon|X_\upsilon, \Theta_\upsilon)$$

where $p(A_v \mid X_v, \Theta_v)$ is the generative model of the data represented by HI-VAE (it is the decoder distribution). Moreover, $pa(X_\upsilon)$ denotes all module nodes plus (in our case, one-dimensional) categorical δ variables, see last section. Hence, $p\left(X_\upsilon|pa(X_\upsilon), \Theta_\upsilon\right)$ is a normal distribution with mean

$$m_\upsilon = \Theta_\upsilon^{(0)} + \sum_{p \in pa(X_\upsilon)} \Theta_\upsilon^{(\rho)} \rho$$

[i.e., modeled via a linear regression with intercept $\Theta_\upsilon^{(0)}$ and slope coefficients $\Theta_\upsilon^{(\rho)}$], and residual variance $\nu_\upsilon = Var(X_\upsilon - m_\upsilon)$.

Our aim is to find parameters Θ maximizing $\log p(X, A \mid G, \Theta)$. Using the factorization of this quantity and the typical assumption of node-wise statistical independence of parameters (Koller and Friedman, 2009), we can optimize the total log-likelihood by the following two steps:

1. For all $\upsilon \in V$: $\widehat{\Theta}_\upsilon^* = \arg\max \log p\left(A_\upsilon|X_\upsilon, \widehat{\Theta}_\upsilon\right)$. This is achieved via training an HI-VAE model for each module X_υ, i.e., optimizing associated network weights $\widehat{\Theta}_\upsilon$.
2. For all $\upsilon \in V: \widetilde{\Theta}_\upsilon^* = \arg\max \log p\left(X_\upsilon|pa(X_\upsilon), \widetilde{\Theta}_\upsilon\right)$. This is achieved by learning the MBN structure G and associated parameters $\widetilde{\Theta}_\upsilon$ based on HI-VAE-encoded modules.

Overall, the training of the proposed VAMBN approach thus consists of the following steps:

1. Definition of modules of variable
2. Training of HI-VAEs for each module. In practice, the training procedure included a hyperparameter optimization over
   a. Learning rate $\in \{0.01, 0.001\}$
   b. Minibatch size $\in \{16, 32\}$
   Each candidate parameter set was evaluated via a 3-fold cross-validation using the reconstruction loss as objective function.
3. Definition of constraints for possible edges in the MBN
4. Structure and parameter learning of the MBN using encoded values for each module: Note that by construction of our model each variable, X_υ follows a mixture of Gaussian distributions. Let $s \sim Categorical(\pi)$ indicate the mixture component. Hence, $X_v \mid s$ is Gaussian. Introducing s into the MBN thus yields a network with only Gaussian and discrete nodes, and parameter and structure learning can accordingly performed computationally efficiently, as explained before.

We also considered to use $N(m_\upsilon, \nu_\upsilon)$ as a prior for X_υ instead of the original Gaussian mixture prior for training of HI-VAE models in a second iteration of the entire VAMBN training procedure. In reality, we could not observe a significant increase in the total model likelihood $p(X, A \mid G, \Theta)$ due to this computationally more costly procedure, see Section A of the **Supplementary Materials**. Reported results hence only refer to the original VAMBN approach without any further continued training using a modified prior.

Simulating Virtual Patients and Counterfactual Scenarios

The trained VAMBN model can be used to create a virtual patient cohort. Virtual patients are simulated as follows:

1. Draw samples from the MBN. This can be achieved by following the topological order of nodes in the DAG. This means that we first sample from the conditional distribution of parent nodes before we do the same for their children while conditioning on the already drawn values each of the parents.
2. Decode MBN samples through HI-VAE. Note that a sample drawn from the MBN represents a vector of latent codes. Decoding maps these codes back into the original input space.

To perform a simulation of a counterfactual situation, we rely on the ideal intervention scheme established by Pearl (2000) via the "do" calculus: This means that rather than sampling from a joint distribution $p(X_1, X_2, \ldots X_n)$ we draw from $p\left(X_1, X_2, \ldots, X_{p-1}, X_{p+1}, \ldots, X_n|do\left(X_p = x\right)\right)$ where $do(X_p = x)$ denotes the scenario that variable X_p in the MBN has been (counterfactually) fixed to value x. Practically, this can be achieved by deleting all incoming edges to X_p in the MBN structure, setting $X_p = x$ and then drawing from the modified

MBN. Subsequently, the variables can be decoded through the HI-VAE, as described before.

Using VAMBN for Counterfactually Adding Features to a Dataset

A special case of the counterfactual simulation described in the last section is the addition of features to a dataset, which have not been observed within a particular study A, but within another study B: Let Y be a (module of) variables in study B not observed in A. We assume the existence of MBNs B_A and B_B for both datasets. Moreover, we suppose $pa(Y) \subset B_A$, i.e., parents of Y are also in A. Hence, we can draw from the interventional distribution

$$p\left(Y|do\left(pa\left(Y\right)=\mathbf{a}\right)\right)$$

where **a** denotes a configuration of parent nodes of Y observed in dataset A. Therefore, we can counterfactually add for any patient in dataset A possible values for Y by considering his/her observed features that may impact Y.

Differential Privacy Respecting Model Training

One of our motivations for developing VAMBN was to enable a mechanism for sharing data across organizations that addresses data privacy concerns. Practically, this could be achieved by sharing either simulated datasets or ready trained VAMBN models. However, specifically in the latter case, there is the concern that by systematically feeding inputs and observing corresponding model outputs, it might be possible to reidentify patients that were used to train VAMBN models. This is particularly true for HI-VAEs, which encode groups of raw features.

Differential privacy is a concept developed in cryptography that poses guarantees on the probability to compromise a person's privacy by a release of aggregate statistics from a dataset (Dwork et al., 2006b): Let A be a randomized algorithm and $0 < \varepsilon$, $0 < \delta < 1$. According to Dwork et al. (2006a) A: $D \rightarrow R$ is said to respect (ε, δ) differential privacy, if for any two datasets D_1, D_2 ε D that differ only in one single patient and for any output of the randomized algorithm $S \subseteq R$, we have

$$\Pr\left(A\left(D_1\right) \in S\right) \leq e^{\epsilon} \Pr\left(A\left(D_2\right) \in S\right) + \delta$$

Abadi et al. (2016) showed that it is possible to directly incorporate (ϵ, δ) differential privacy guarantees into the training of a neural network by clipping the norm of the gradient and adding a defined amount of noise to it.

It is straightforward to incorporate this approach into the training of each of the VAE models within VAMBN. Hence, we are able to provide guarantees on (ϵ, δ) differential privacy for the entire VAMBN model because (ϵ, δ) differential privacy is composable. This means that the property for a system of several components is fulfilled if all of its components fulfill (ϵ, δ) differential privacy (Dwork et al., 2006a).

DATA

SP513

SP513 was a randomized, double-blinded, and placebo-controlled study to compare two PD drugs within an early disease population (Giladi et al., 2007). We here examine 557 patients of the final analysis set, which had received treatment. Out of these patients, 117 received placebo, 227 ropinirole, and 213 another dopamine agonist. Both drugs were first uptitrated within a 13 week time period and then followed up for 24 weeks. We model the screening and baseline visits as well as three visits in the maintenance phase. Clinical variables captured during the trial comprised baseline demographics, disease duration, UPDRS scores, Epworth Sleepiness Scale (ESS), Hoehn and Yahr stage, and standard blood biomarkers for safety assessment (e.g., hemoglobin, creatinine, etc.).

PPMI

The Parkinson's Progression Markers Initiative (PPMI) (www.ppmi-info.org/data) consists of multiple cohorts from a network of clinical sites with the aim to identify and verify progression markers in PD. It is a multimodal, longitudinal observation study with data collected using standardized protocols (Parkinson Progression Marker Initiative, 2011). PPMI comprises of eight cohorts with different clinical and genetic characteristics. Here, we used data of 362 *de novo* PD patients and 198 healthy controls. All PD patients were initially untreated and diagnosed with the disease for 2 years or less. They showed signs of resting tremor, bradykinesia, and rigidity. We used 266 clinical variables measured at 11 visits during 96 months comprising demographics, patient PD history, DaTSCAN imaging, non-motor symptoms, cerebrospinal fluid (CSF) biomarkers (A–β, α-synuclein, dopamine, phospho-tau, total tau) and UPDRS scores.

RESULTS

VAMBN Reflects Expected Causal Relationships in Data

As outlined in *Methods*, our proposed VAMBN approach results into a Modular Bayesian Network that describes conditional statistical dependencies between groups of variables that are encoded via HI-VAEs. An obvious initial question is whether learned dependencies between modules reflect expected causal relationships and, if yes, how statistically stable these can be detected. To address this point, we performed a non-parametric bootstrap of the MBN structure learning (Davison and Hinkley, 1997). This means that, for each study, we resampled the existing N patients 1,000 times with replacement. For each bootstrap dataset, we ran a complete MBN structure learning, and we counted the fraction of times that each edge was included in the model. We overlayed this bootstrapped network with the MBN learned from the complete data to get an overall impression of the learned VAMBN model as well as the stability of inferred conditional statistical dependencies.

Figure 2 highlight that, in both SP513 as well as PPMI, inferred edges agree well with expected causal dependencies: For

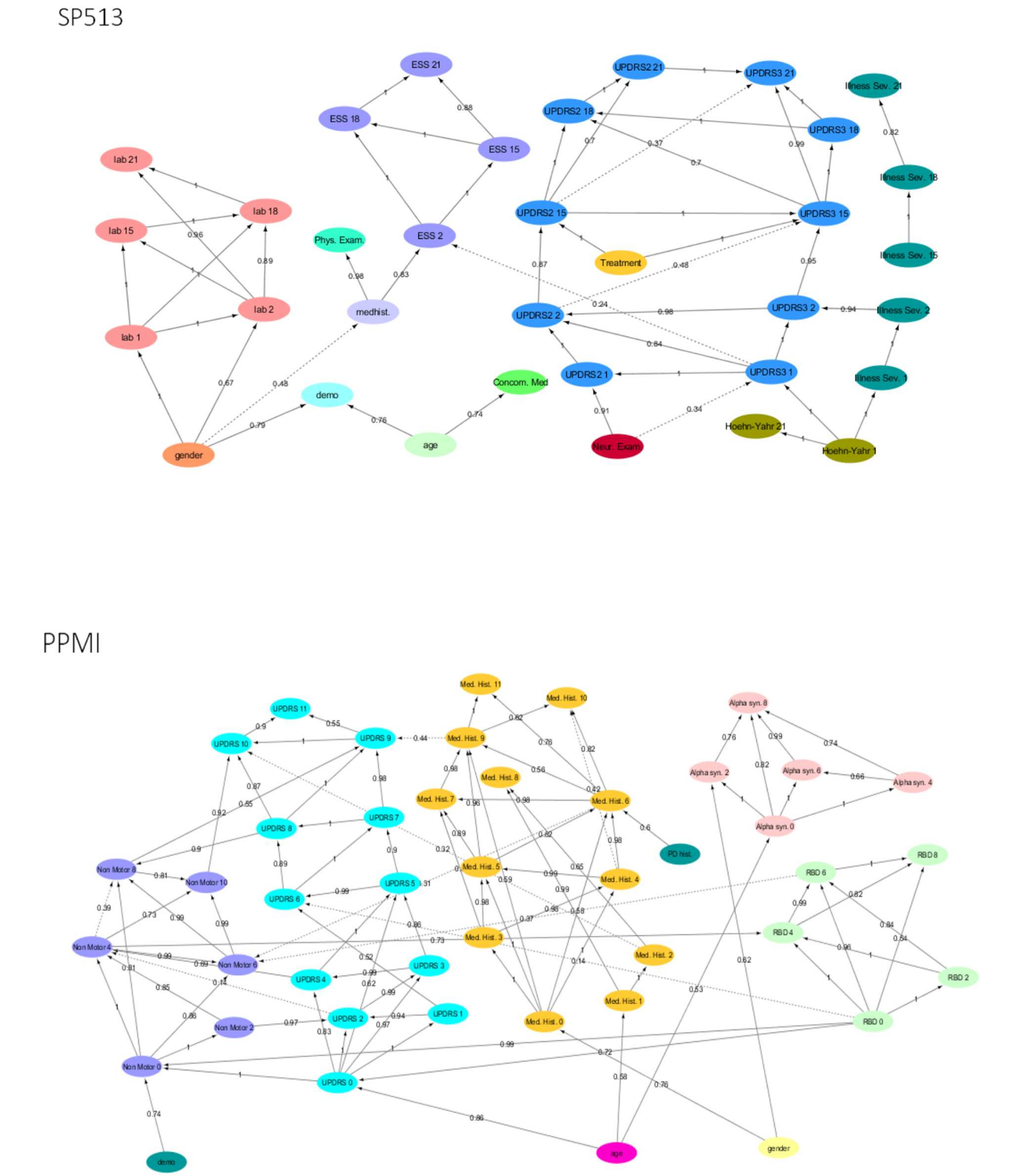


FIGURE 2 | Final Modular Bayesian Networks (MBNs) learned by Variational Autoencoder MBN (VAMBN) based on SP513 and PPMI data. The edges are labeled with the bootstrap frequencies of each connection. For readability, auxiliary variables and missing visit nodes were removed for the visualization. Figures are also available as Cytoscape files in the Supplements for better convenience.

example, in SP513 (**Figure 2**), UPDRS scores of subsequent visits are connected with each other and impact sleepiness (ESS). ESS itself is dependent on medical history. UPDRS scores are, during the titration phase, influenced by Hoehn and Yahr stages and the illness severity score defined in SP513. Safety biomarkers depend on gender, but otherwise have no impact.

In PPMI (**Figure 2**), the RBD sleepiness score and non-motor symptoms mutually influence each other, and the same holds true for UPDRS. UPDRS is dependent on age, medical history, and α-synuclein levels in CSF.

Altogether, these examples underline that VAMBN models permit a certain level of interpretation.

Simulated Patients are Realistic

Simulated patient trajectories generated by VAMBN are only useful if they are sufficiently similar to real ones. On the other hand, we clearly do not want VAMBN to simply regenerate the data it was trained on (which would trivially maximize similarity to real patients). It is therefore not straightforward to come up with a criterion or interpretable index to measure the quality of a virtual patient simulation.

From our point of view, simulated patients should mainly fulfill the following criteria:

(a) Summary statistics (e.g., mean, variance, median, lower quartile, upper quartile) over individual variables should look similar to real ones.
(b) Correlations between variables in simulated patients should be close to the ones observed in real ones.
(c) MBN structures learned from simulated patients should be close to the ones learned from real one.
(d) Treatment effects or other expected outcomes should be similar in simulations, also in terms of effect size.

To assess VAMBN with respect to these criteria, we simulated the same number of virtual patients as real ones in each of the two PD studies. **Figure 3** demonstrates that marginal distributions for individual variables were, in general, sufficiently similar (but not identical) to the empirical distributions of real data in both PD studies. For additional plots, see Section B of the **Supplementary Materials**. In addition, the empirical distributions of Pearson correlations in simulated and real data were close to each other (**Figure 4**). Interestingly, in both cases (marginal distributions and correlations), largest differences were observed between HI-VAE-decoded features of real patients and original features of the same patients. Hence, the majority of the "simulation error" can be attributed to an imperfect fit of HI-VAE models.

As further assessment of the quality of the sampled data, we compared the edges in the graph of the real PPMI patients' data (RP graph) with the edges in graphs of different virtual patient sets (VP graphs). Since the virtual patients are sampled using the RP graph, we would expect to see strong overlap between the graphs, but we would also expect the sample size (in this case, the number of virtual patients used to train the different VP graphs) to affect this similarity. If an edge is present in the RP graph and also a given VP graph, we consider this a hit; if an edge is only present in the VP graph, it is a false positive and so on. Using this logic, we can compute and plot the sensitivity and specificity of the RP-VP comparisons at different VP sample sizes (**Figure 5**). This indicated an overall rapid convergence of sensitivity and specificity of MBNs learned from simulated data toward 1. Hence, simulated data reflect the same or at least very similar patterns than real data.

As a final assessment for the quality of virtual patients, we compared known patient subgroups in simulated and real data. **Figure 6** (right) demonstrates that, in PPMI, UPDRS3 scores of simulated PD patients showed similar differences to healthy controls than in real PD patients. Moreover, the ropinirole treatment effect in simulated and real SP513 patients demonstrated a comparable effect size and *p*-value (**Figure 6**, left).

Altogether, we thus concluded that VAMBN allows for a sufficiently realistic simulation of virtual subjects with respect to our three defined criteria. At the same time, we could confirm that indeed none of the simulated patients were a simple regeneration of one of the patients in the training data.

Generalizability of VAMBN Models

A relevant question is how generalizable VAMBN models are, i.e., whether they are purely overfitted or whether they can sufficiently describe data in an independent test set. To address this point, we randomly split data in SP513 and PPMI into 80% training and 20% test. VAMBN models were only fitted to the training set. We then recorded the log-likelihood of patients in the training and test sets, indicating a sufficiently good agreement (**Figure 7**). We thus concluded that VAMBN models are generally not overfitted. This means that the previously reported agreement of virtual and real patients cannot just be the result of overfitting the data with an overly complex model.

Simulation of Counterfactual Scenarios Match Expectations

Due to its nature as a hybrid of a BN and a generative neural network, VAMBN allows for simulation of counterfactual scenarios via the "do" calculus, as explained in *Methods*. **Figure 8A** demonstrates the effect of counterfactually altering UPDRS2 and UPDRS3 baseline scores of all patients in SP513 to the mean observed in PPMI, i.e., toward lower disease severity. As expected, this resulted into a likewise shift of UPDRS3 scores (reflecting motor symptoms) at end of study.

In PPMI, making all patients 20 years younger shifts the distribution of UPDRS3 scores to the left (fewer motor symptoms), whereas making them 20 years older has the opposite effect (**Figure 8B**). Again, this effect matches expectations.

As a final example, we demonstrate the possibility to counterfactually add a feature to PPMI via the approach described in *Methods*: We used the VAMBN model for SP513 to simulate the shift of the UPDRS3 scores at visit 15 under ropinirole treatment conditional on age, gender, height, weight, as well as UPDRS2 and UPDRS3 baseline scores of patients observed in PPMI. This means that there was only a simulated intervention into these features. By subtracting

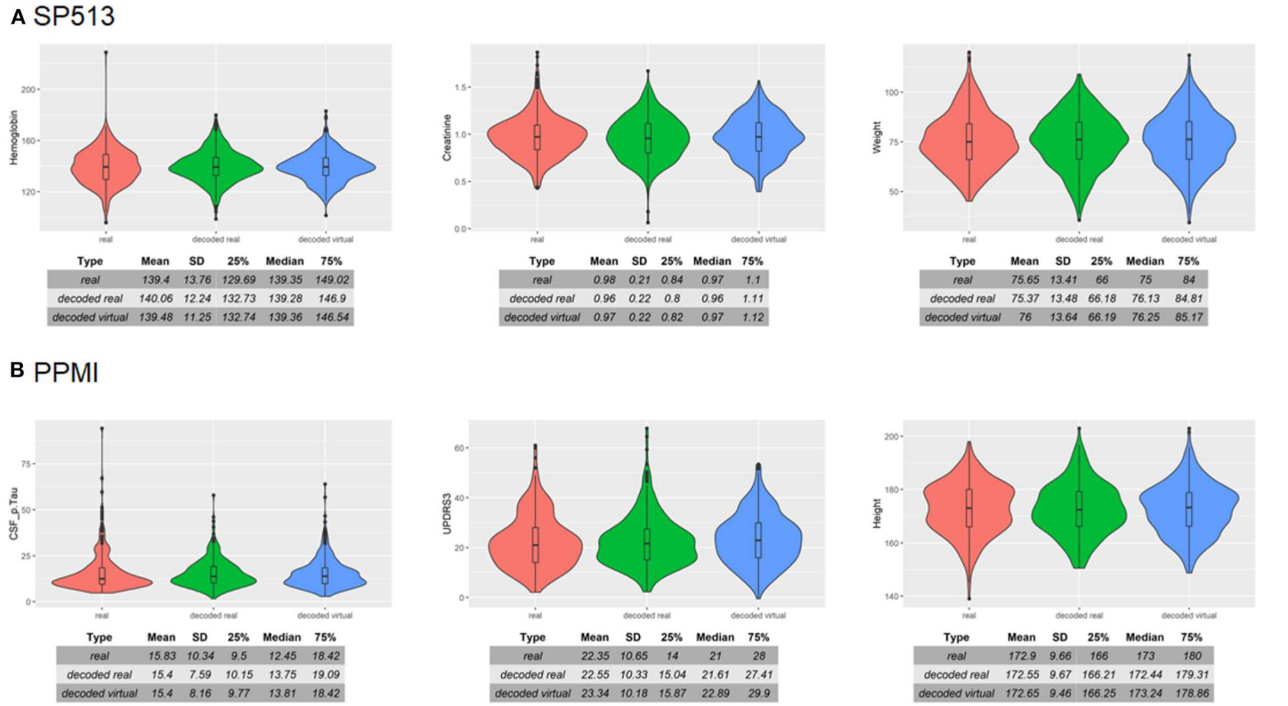


FIGURE 3 | Examples of real and simulated/virtual patients for **(A)** SP513 and **(B)** Parkinson's Progression Markers Initiative (PPMI) datasets. The figure compares the marginal distributions of selected variables for real patients (red), virtual patients (blue), and real patients decoded via the Variational Autoencoders for Heterogeneous and Incomplete Data (HI-VAE) model (green). Tables show summary statistics of the distributions. Plots and tables of further variables can be found in Section B of the **Supplementary Materials**.

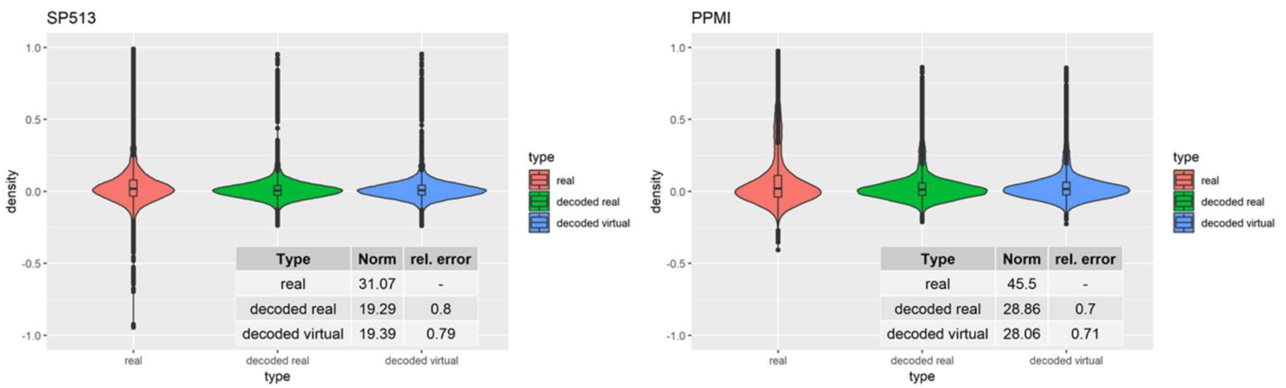


FIGURE 4 | Distribution of Pearson correlation coefficients between variables in real patients (red), virtual patients (blue), and decoded real patients (green). Tables show the Frobenius norm of the correlation matrices as well as the relative error, which consists of the norm of the matrix that is the difference between the decoded real or virtual correlation matrix divided by the norm of the original correlation matrix.

the simulated shift from the observed UPDRS3 off scores in PPMI, we obtained a counterfactual treatment simulation with ropinirole. **Figure 8C** compares the observed UPDRS3 off and on scores (under L-DOPA treatment) to those simulated by VAMBN for ropinirole treatment. Further plots showing the effect at different PPMI visits are shown in Section D of the **Supplementary Materials**. As expected, UPDRS3 scores simulated for ropinirole treatment are significantly shifted compared to observed UPDRS3 off scores but are slightly higher than UPDRS3 on scores under L-DOPA. Indeed, it has been suggested that efficacy of ropinirole is slightly lower than that of L-DOPA (Zhuo et al., 2017).

Overall, these counterfactual simulations exemplify the possibilities of VAMBN and at the same time reconfirm that the model has learned the expected variable dependencies from data because the simulation effects match expectations.

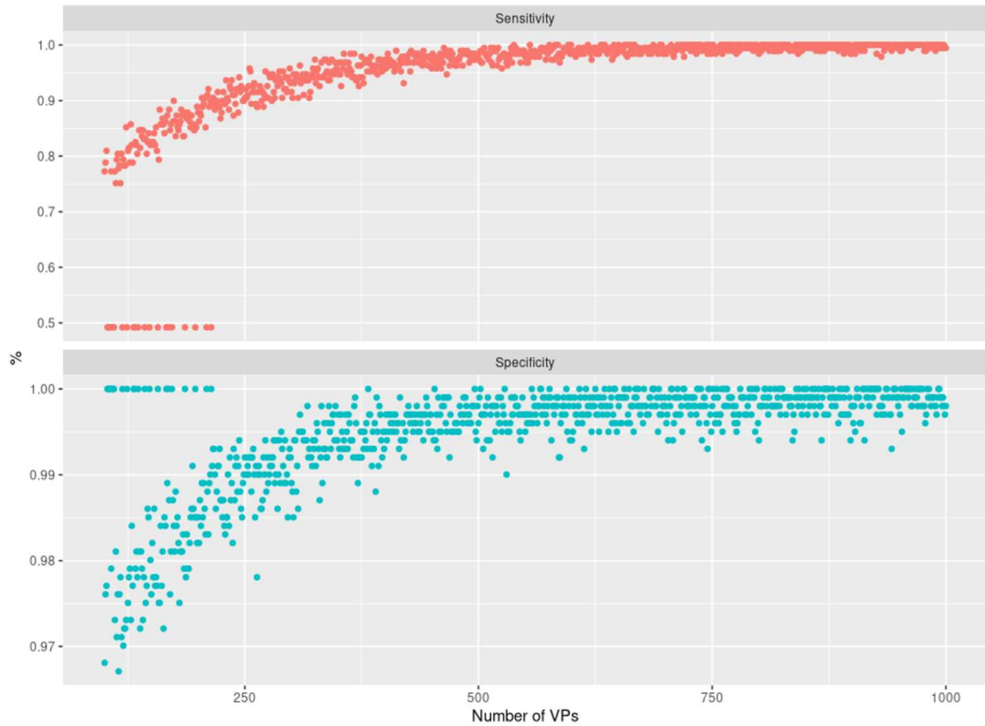
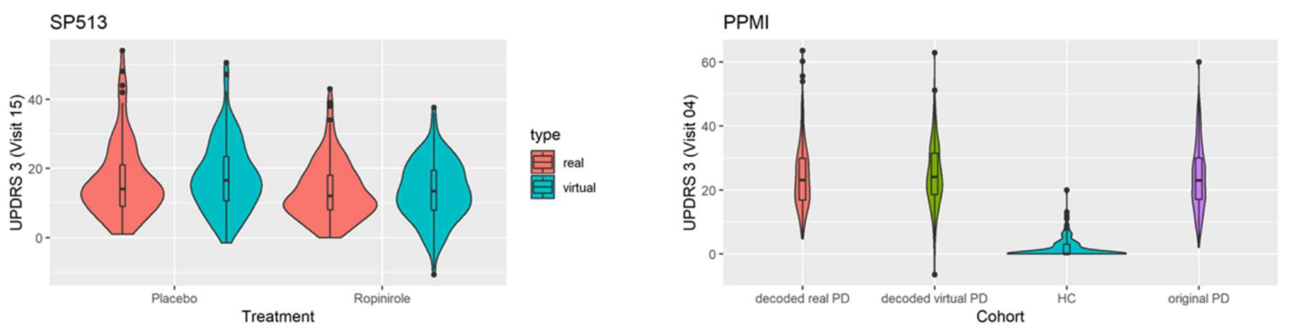


FIGURE 5 | Illustration of the sensitivity (top panel) and specificity (bottom panel) achieved when comparing Modular Bayesian Network (MBN) structures learned from real Parkinson's Progression Markers Initiative (PPMI) data with the ones learned from virtual patients. The *x*-axis shows an increasing number of sampled virtual patients (with increments of 1). For each of those virtual cohorts, one MBN structure was learned and compared against the MBN learned from real PPMI data. The cases where sensitivity is 50% and specificity is 100% correspond to networks that have no false positives and no true positives except whitelisted edges, which we did not count as true positives here. Note that a corresponding situation can always occur by chance (specifically for small sample sizes) because synthetic data are randomly generated by the VAMBN learned model.



Type	p value	ges
real	*0.006*	*0.03*
virtual	*<0.001*	*0.05*

Type	p value	r
real PD- HC	*<0.001*	*0.771*
decoded real PD - HC	*<0.001*	*0.796*
decoded virtual PD - HC	*<0.001*	*0.783*

FIGURE 6 | Distribution of UPDRS3 scores in SP513 **(left)** and Parkinson's Progression Markers Initiative (PPMI) **(right)**. **(Left)** The plot depicts in red UPDRS3 scores of real SP513 patients under placebo and ropinirole at visit 15 (i.e., during treatment phase), respectively. In blue, the distribution of the UPDRS3 score in the same number of virtual patients is shown. Effect sizes and corresponding *p*-values obtained from two one-way ANOVAs comparing placebo and drug treatment in the real and virtual patients are shown in the tables at the bottom. Similar plots at other visits can be found in Section C of the **Supplementary Materials (Right)** Distribution of original (purple) and decoded (red) UPDRS3 scores of real PPMI de novo Parkinson's disease (PD) patients at visit 4 in comparison to PPMI healthy controls (blue). UPDRS3 scores of virtual PD patients are shown in yellow. The table at the bottom shows differences in UPDRS3 scores between original PD, decoded real PD, and virtual PD patients compared to PPMI healthy controls, showing p-value and effect size from three Mann-Whitney U-tests. Similar plots at other visits can be found in Section C of the **Supplementary Materials**.

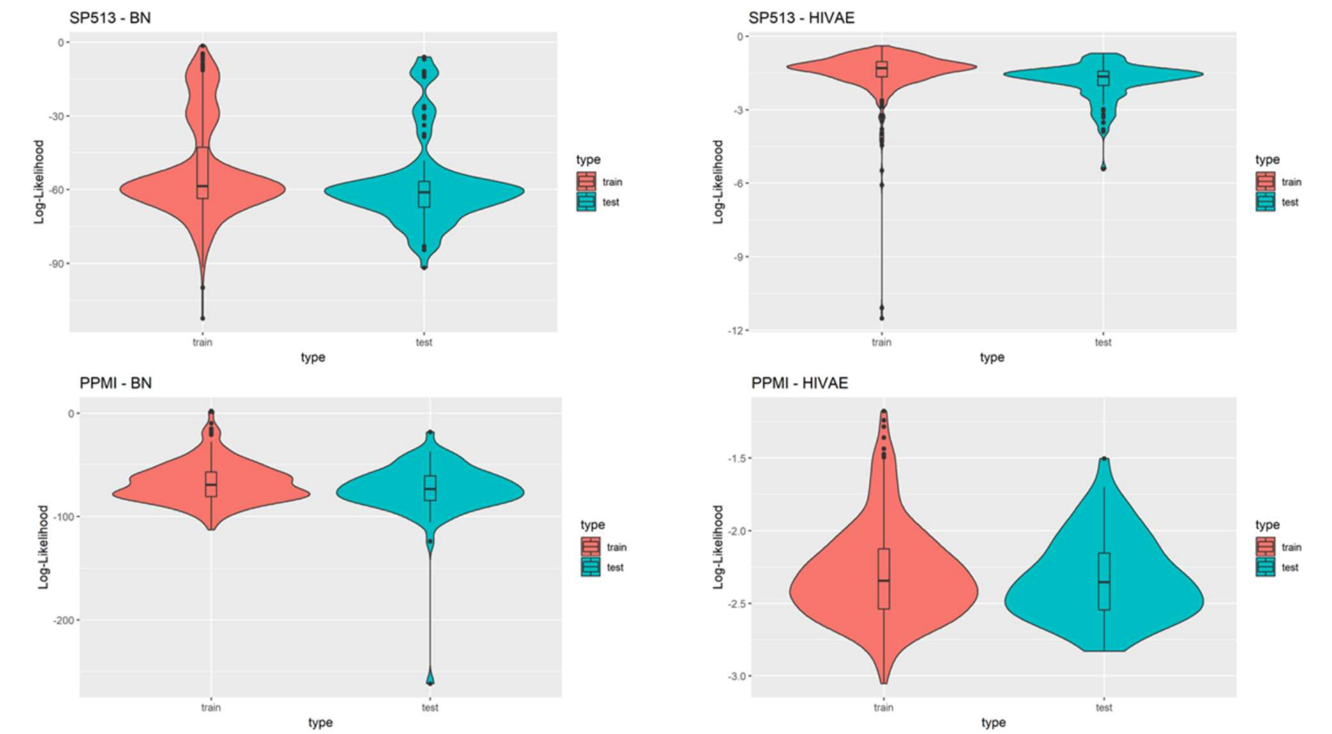


FIGURE 7 | This figure compares the log-likelihoods of real patients in a training set (red) and a test set (blue) of the SP513 (top row) and Parkinson's Progression Markers Initiative (PPMI) datasets (bottom row) for the Modular Bayesian Network (MBN) and the Variational Autoencoders for Heterogeneous and Incomplete Data (HI-VAE) models. The HI-VAE log-likelihoods are based on the participants included in the respective sets after averaging across all separate HI-VAE models (9 for SP513, 34 for PPMI).

Differential Privacy Respecting Modeling Training

As a last point, we investigated differential privacy respecting model training of VAMBN. As indicated in *Methods*, this can be realized by defining a certain privacy loss via constants (ϵ, δ) for each HI-VAE model trained within VAMBN. Smaller values for these constants generally impose stronger privacy guarantees but make model training harder. To investigate this effect more quantitatively, **Figure 9** shows the reconstruction errors of the HI-VAE models for the SP513 laboratory data at the first visit as a function of number of training epochs and in dependence on different values for ϵ, δ. For similar figures of the other visits, see Section E of the **Supplementary Materials**. It can be observed that in dependence on these constants, longer trainings and more data are required to achieve the same level of reconstruction error than for conventional model training without differential privacy.

CONCLUSION

Sensitive patient data require high standards for protection as, e.g., reinforced by the European Union through the General Data Protection Regulation (GDPR—https://eur-lex.europa.eu/eli/reg/2016/679/oj). However, at the same time, these data are instrumental for biomedical research in the entire healthcare sector. Establishing a mechanism for sharing data across organizations without violating data privacy is therefore of utmost relevance for scientific progress. In this paper, we build on the idea of developing generative models to simulate virtual patients based on data from clinical studies. A recent publication proposed to train Generative Adversarial Networks (GANs) based on few variables recorded from more than 6,000 patients in the Systolic Blood Pressure Trial (Beaulieu-Jones et al., 2018). In contrast, our work focuses on the realistic situation regarding much smaller sample size coupled with significantly higher number of variables, which is common in many other medical fields, such as neurology. Our results demonstrate that VAMBN models generally do not overfit and allow for a sufficiently realistic simulation of virtual patients. In contrast to GANs, our VAMBN method relies on an explicit modeling of time dependencies, as well as missing and heterogenous data. Moreover, VAMBN models can be interpreted via the MBN structure. As demonstrated in this work, Bayesian Networks also open the door to simulating counterfactual scenarios (including treatments with drugs) within a well-established theoretical framework, which could, e.g., help in the design of clinical trials. Moreover, we have shown that simulated data could themselves be used to learn complex AI models, such as a Bayesian Network structure, which can subsequently be compared to real data. In addition, we demonstrated that data privacy respecting model training is in principle possible with VAMBN.

From a user perspective, we see two important aspects for the successful application of our approach:

1. A careful understanding of the data and its structure, including the ability to define variable groups

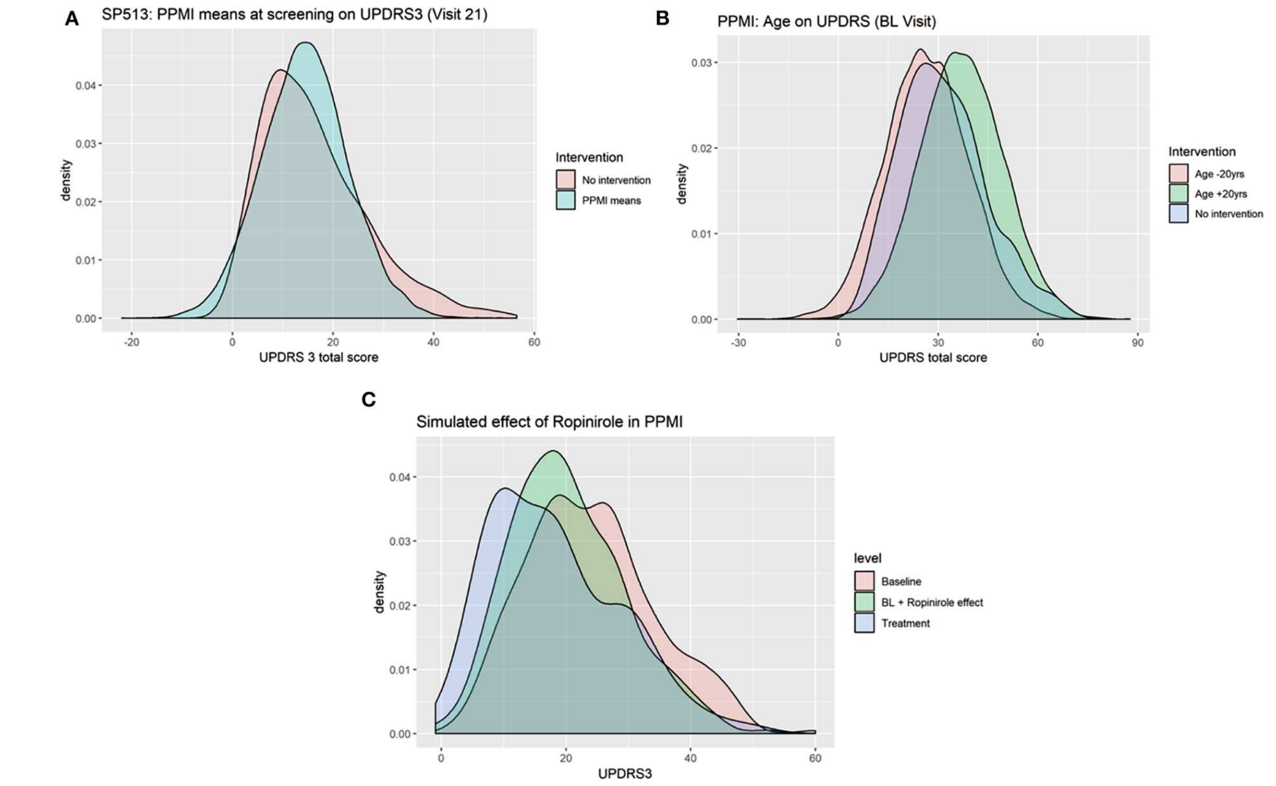


FIGURE 8 | Counterfactual simulation of **(A)** lower disease severity in SP513 (shift of UPDRS3 scores at baseline to mean observed in Parkinson's Progression Markers Initiative (PPMI) at baseline), **(B)** shift of age 20 years up and down, and **(C)** treatment effect of ropinirole in PPMI.

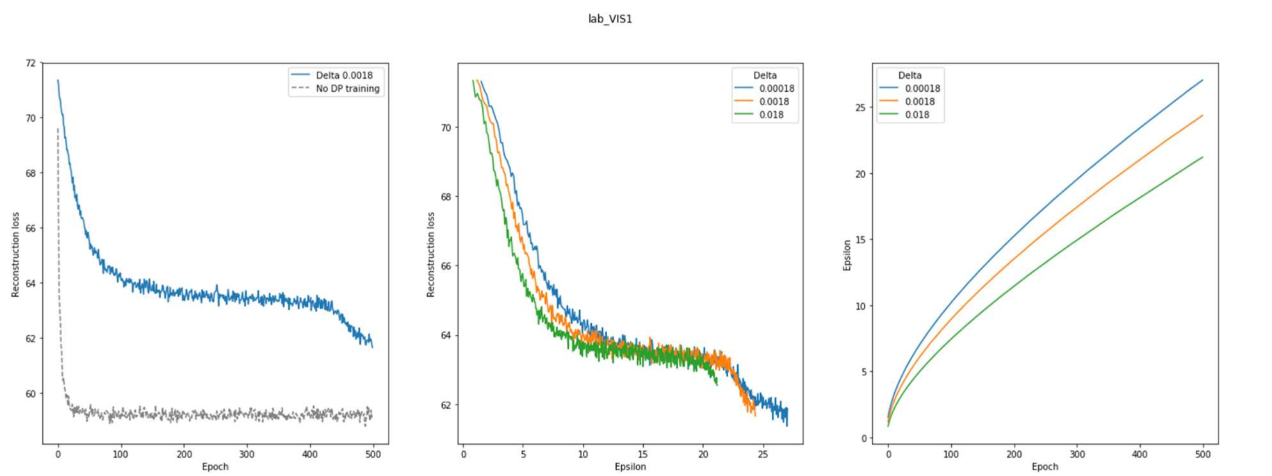


FIGURE 9 | This figure shows the effects of differentially privacy respecting (DP) Variational Autoencoders for Heterogeneous and Incomplete Data (HI-VAE) training on the HI-VAE step of the model. **(Left)** reconstruction loss change between DP and conventional model training for laboratory data at visit 1 for the SP513 study; **(middle)** epsilon plotted against reconstruction loss for different delta values; **(right)** epsilon over 500 epochs, given different deltas. A noise multiplier of 1.1, norm clipping at 1.6, and a learning rate of 0.01 were used. Further plots can be found in Section E of the **Supplementary Materials**.

2. A careful check of the quality of synthetic data, using the approaches suggested in this paper.

Taken together, VAMBN is a new method for simulation of virtual cohorts for which we see a number of interesting future use cases in healthcare:

- Simulation of counterfactual scenarios to help the design of clinical trials
- Privacy preserving sharing of data across organizations to help data scientists understand the structure of sensitive patient data, judge their utility for modeling purposes, and derive

statistical hypotheses that can be verified or falsified with available real data
- Training of AI models that can subsequently be tested with available real data
- Merging of different virtual cohorts from the same indication area into a global virtual meta-cohort based on overlapping variables. This global virtual meta-cohort could be used to
  - identify for a specific real patient within the overall distribution a best matching virtual avatar
  - efficiently generate control arms for clinical trials.

Of course, our work is not without limitations: Building VAMBN models requires (in contrast to GANs) a relatively detailed understanding of data and careful handling of missing values in particular. Our examples have shown that VAMBN models can in practice already be learned from datasets with comparably small sample size and many variables. Nonetheless, our method, as any AI-based approach, is principally dependent on sample size and signal-to-noise ratio in data. In the extreme case of more variables than samples (high dimensional setting), we expect VAMBN to become statistically unstable and overfit. From a technical side, VAMBN implies to train multiple neural networks, which usually requires a modern parallel computing architecture. It thus remains a subject of future research to investigate how VAMBN models could be made better accessible to practitioners in order to facilitate their use in a widespread manner. In the meantime, we have made our python and R code available as part of the **Supplementary Material**.

Overall, we see our work as a useful complement to federated machine learning techniques, which, together with virtual patient simulation tools, could help to break data silos and thus enhance progress in biomedical research.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: See www.ppmi-info.org and Giladi et al. Movement Disorders (2007).

ETHICS STATEMENT

No human studies are presented in this manuscript.

AUTHOR CONTRIBUTIONS

HF and MH-A designed the project. HF guided the work scientifically. HF and LG-D invented the method. LG-D implemented and tested the approach. AS and MS helped with the analysis of the data. All authors have read and approved the manuscript.

FUNDING

The research leading to these results has received partial support from the Innovative Medicines Initiative Joint Undertaking under grant agreement #115568, resources of which are composed of financial contribution from the European Union's Seventh Framework Programme (FP7/2007-2013) and EFPIA companies' in kind contribution.

ACKNOWLEDGMENTS

Data used in the preparation of this article were obtained from the Parkinson's Progression Markers Initiative (PPMI) database (www.ppmi-info.org/data). For up-to-date information on the study, visit www.ppmi-info.org. PPMI—a public-private partnership—was funded by the Michael J. Fox Foundation for Parkinson's Research and funding partners. A list of names of all of the PPMI funding partners can be found at www.ppmi-info.org/about-ppmi/who-we-are/study-sponsors/. This manuscript has been released as a preprint at https://www.biorxiv.org/content/10.1101/760744v1.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fdata.2020.00016/full#supplementary-material

REFERENCES

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., et al. (2016). Deep learning with differential privacy. *Machine Learn.* 2016, 308–318. doi: 10.1145/2976749.2978318

Andrews, B., Ramsey, J., and Cooper, G. F. (2018). Scoring bayesian networks of mixed variables. *Int. J. Data Sci. Anal.* 6, 3–18. doi: 10.1007/s41060-017-0085-7

Beaulieu-Jones, B. K., Wu, Z. S., Williams, C., Lee, R., Bhavnani, S. P., Byrd, J. B., et al. (2018). Privacy-preserving generative deep neural networks support clinical data sharing. *Circ. Cardiovasc. Qual. Outcomes* 12:e005122. doi: 10.1161/CIRCOUTCOMES.118.005122

Chase, J. G., Preiser, J.-C., Dickson, J. L., Pironet, A., Chiew, Y. S., Pretty, C. G., et al. (2018). Next-generation, personalised, model-based critical care medicine: a state-of-the art review of *in silico* virtual patient models, methods, and cohorts, and how to validation them. *Biomed. Eng. Online* 17:24. doi: 10.1186/s12938-018-0455-y

Chickering, D. M., Heckerman, D., and Meek, C. (2004). Large-sample learning of bayesian networks is NP-hard. *J. Mach. Learn. Res*. 5, 1287–1330.

Davison, A. C., and Hinkley, D. V. (1997). *Bootstrap Methods and Their Application.* Cambdrige: Cambridge University Press.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. (2006a). "Our data, ourselves: privacy via distributed noise generation," in *Advances in Cryptology - EUROCRYPT 2006,* ed S. Vaudenay (Berlin; Heidelberg: Springer), 486–503.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006b). "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography,* eds S. Halevi and T. Rabin (Berlin; Heidelberg: Springer), 265–284.

Fröhlich, H., Balling, R., Beerenwinkel, N., Kohlbacher, O., Kumar, S., Lengauer, T., et al. (2018). From hype to reality: data science enabling personalized medicine. *BMC Med.* 16:150. doi: 10.1186/s12916-018-1122-7

Galbusera, F., Niemeyer, F., Seyfried, M., Bassani, T., Casaroli, G., Kienle, A., et al. (2018). Exploring the potential of generative adversarial networks for synthesizing radiological images of the spine to be used in *in silico* trials. *Front. Bioeng. Biotechnol.* 6:53. doi: 10.3389/fbioe.2018.00053

Ghahramani, Z. (1998). "Learning dynamic Bayesian networks," in *Adaptive Processing of Sequences and Data Structures. NN 1997. Lecture Notes in*

Computer Science, Vol. 1387, eds C. L. Giles and M. Gori (Berlin; Heidelberg: Springer), 168–197. doi: 10.1007/BFb0053999

Ghosh, A., Hong, J., Yin, D., and Ramchandran, K. (2019). Robust federated learning in a heterogeneous environment. *arXiv [Preprint]*. arXiv:1906.06629.

Giladi, N., Boroojerdi, B., Korczyn, A. D., Burn, D. J., Clarke, C. E., Schapira, A. H. V., and SP513 investigators (2007). Rotigotine transdermal patch in early Parkinson's disease: a randomized, double-blind, controlled study versus placebo and ropinirole. *Mov. Disord*. 22, 2398–2404. doi: 10.1002/mds.21741

Heckerman, D. (1997). A tutorial on learning with bayesian networks. *Data Min. Knowl. Disc.* 1, 79–119. doi: 10.1023/A:1009730122752

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput*. 9, 1735–1780. doi: 10.1162/neco.1997.9.8.1735

Holford, N., Ma, S. C., and Ploeger, B. A. (2010). Clinical trial simulation: a review. *Clin. Pharmacol. Ther*. 88, 166–182. doi: 10.1038/clpt.2010.114

Hong, Y., Xia, X., Le, J., and Zhou, X. (2016). "Learning bayesian network structure from large-scale datasets," in *2016 International Conference on Advanced Cloud and Big Data (CBD)* (Chengdu), 258–264.

Kang, H. (2013). The prevention and handling of the missing data. *Kor. J. Anesthesiol.* 64, 402–406. doi: 10.4097/kjae.2013.64.5.402

Kingma, D. P., and Welling, M. (2013). Auto-encoding variational bayes. *arXiv [Preprint]*. arXiv:1312.6114.

Knab, T. D., Clermont, G., and Parker, R. S. (2016). A "virtual patient" cohort and mathematical model of glucose dynamics in critical care. *IFAC-Papers Online* 49, 1–7. doi: 10.1016/j.ifacol.2016.12.094

Koller, D., and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Technique.* Cambridge, MA; London: The MIT Press.

Lim, S. S., Kivitz, A. J., McKinnell, D., Pierson, M. E., and O'Brien, F. S. (2017). Simulating clinical trial visits yields patient insights into study design and recruitment. *Patient Prefer. Adher.* 11, 1295–1307. doi: 10.2147/PPA.S137416

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. (2016). Communication-efficient learning of deep networks from decentralized data. *arXiv [Preprint]*. arXiv:1602.05629.

Mustillo, S., and Kwon, S. (2015). Auxiliary variables in multiple imputation when data are missing not at random. *J. Math. Sociol.* 39, 73–91. doi: 10.1080/0022250X.2013.877898

Nazabal, A., Olmos, P. M., Ghahramani, Z., and Valera, I. (2018). Handling incomplete heterogeneous data using VAEs. *arXiv [Preprint]*. arXiv: 1807.03653.

Pappalardo, F., Russo, G., Tshinanu, F. M., and Viceconti, M. (2018). *In silico* clinical trials: concepts and early adoptions. *Brief. Bioinformatics* 20, 1699–1708. doi: 10.1093/bib/bby043

Parkinson Progression Marker Initiative (2011). The Parkinson Progression Marker Initiative (PPMI). *Prog. Neurobiol*. 95, 629–635. doi: 10.1016/j.pneurobio.2011.09.005

Pearl, J. (2000). *Causality: Models, Reasoning and Inference*. Cambridge: Cambridge University Press.

Raghu, V. K., Poon, A., and Benos, P. V. (2018). Evaluation of causal structure learning methods on mixed data types. *Proc. Mach. Learn. Res.* 92, 48–65.

Rubin, D. B. (1976). Inference and missing data. *Biometrika* 63, 581–592. doi: 10.1093/biomet/63.3.581

Scutari, M. (2010). Learning Bayesian Networks with the bnlearn R package. *J. Stat. Softw.* 35, 1–22. doi: 10.18637/jss.v035.i03

Segal, E., Pe'er, D., Regev, A., Koller, D., and Friedman, N. (2005). Learning module networks. *J. Mach. Learn. Res.* 6, 557–588.

Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., et al. (2003). Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.* 34, 166–176. doi: 10.1038/ng1165

Zhuo, C., Zhu, X., Jiang, R., Ji, F., Su, Z., Xue, R., et al. (2017). Comparison for efficacy and tolerability among ten drugs for treatment of parkinson's disease: a network meta-analysis. *Sci. Rep.* 7:45865. doi: 10.1038/srep45865

Conflict of Interest: HF and LG-D received salaries form UCB Pharma. UCB Pharma had no influence on the content of this study.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.