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

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

1) UCB Pharma S.A, 208 Bath Road, Slough, Berkshire, SL1 3WE, UK
2) Fraunhofer Institute for Algorithms and Scientific Computing (SCAI), 53754 Sankt Augustin, Germany
3) University of Bonn, Bonn-Aachen International Center for IT, Endenicher Allee 19c, 53115 Bonn, Germany
4) UCB Pharma S.A., Alfred-Nobel-Str. 10, 40789 Monheim, Germany

*) correspondence: frohlich@bit.uni-bonn.de

Keywords: Bayesian Networks, autoencoders, simulation, longitudinal data

Abstract

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, there is still the need to investigate, mine and understand clinical data that cannot be accessed directly. Simulation of sufficiently realistic virtual patients could be a way to fill this gap.

In this work we propose a new machine learning approach (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.

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 over-representation 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 (Ghosh et al., 2019; McMahan et al., 2016), their implementation is technically challenging and does not permit researchers to unbiasedly investigate, mine and understand clinical study data located within different organizations.

Sufficiently realistic simulations of virtual patient cohorts could not 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 (Chase et al., 2018; Knab et al., 2016) or for better design of clinical trials (Galbusera et al., 2018; Lim et al., 2017). 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 Networks - 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.

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. It combines two classes of generative modeling techniques: Bayesian Networks (BNs) (Heckerman, 1997) and Variational Autoencoders (VAEs) (Kingma and Welling, 2013). Bayesian Networks (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

- 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 has often dozens of variables (measured over time). But, the number of patients is typically only in the order of a few hundred. Hence, the chance to identify the correct graph from these limited data is questionable.

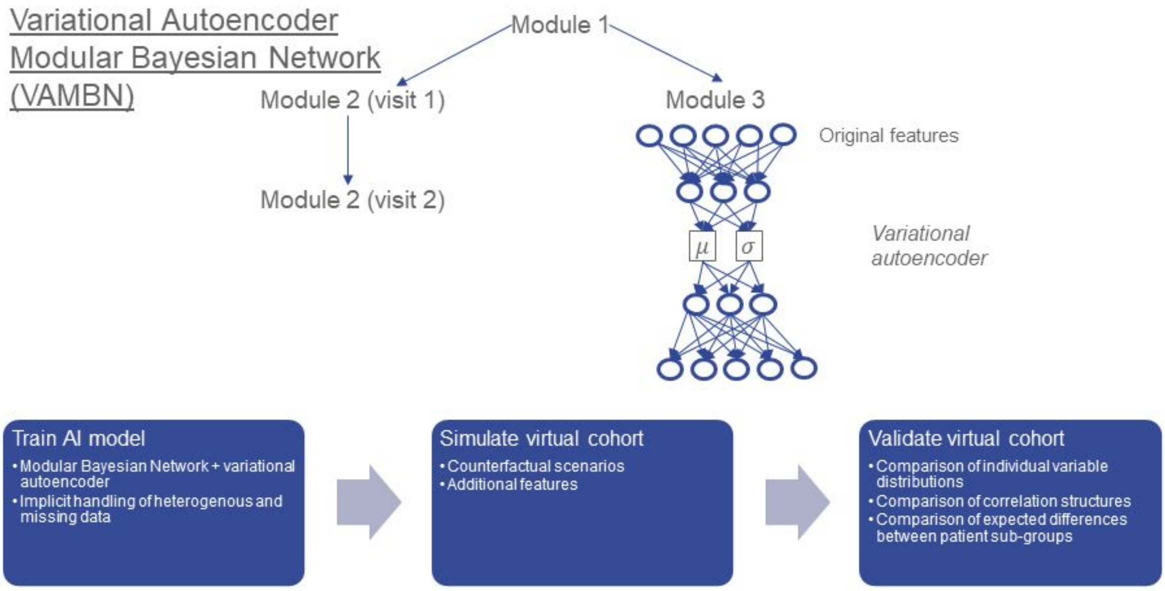
VAEs are a neural network based approach that maps input data to a low dimensional latent distribution (typically a Gaussian) through several sequential encoding steps. VAEs are typically trained via stochastic gradient descent to optimize an evidence / variational lower bound (ELBO) on the log-likelihood of the data. VAEs have recently been extended to deal with heterogeneous multi-modal 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 for combining the advantages of BNs and VAEs while mitigating their limitations (Figure 1): Following the idea of Module Networks (Segal et al., 2003, 2004) we first define modules of variables that group together according the design of the study. For example, demographic features, clinical assessment scores, medical history, treatment might each form such a module. Our aim is then to learn a BN between these modules. In contrast to Segal et al. we do not use regression trees to represent conditional joint distributions of variables within each module, but VAEs, because they are generative. Each VAE is thus only trained on a small subset of variables, hence significantly reducing the number of network weights compared to a full VAE model for the entire dataset and allowing for applying the well established “do” calculus for simulating interventions. 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.


Variational Autoencoder Modular Bayesian Network (VAMBN)
Module 1
Module 2 (visit 1)
Module 2 (visit 2)
Module 3
Original features
μ
σ
Variational autoencoder
Train AI model
• Modular Bayesian Network + variational autoencoder
• Implicit handling of heterogenous and missing data
Simulate virtual cohort
• Counterfactual scenarios
• Additional features
Validate virtual cohort
• Comparison of individual variable distributions
• Comparison of correlation structures
• Comparison of expected differences between patient sub-groups

Modular Bayesian Networks (MBNs)

The starting point of our proposed approach is a Modular Bayesian Network (MBN) describing in a longitudinal manner statistical dependencies of VAE encoded modules (i.e. sets of variables in the original data): Let $G = (V, E)$ be a directed acyclic graph (DAG) and $X = (X_v)_{v \in V}$ a set of random variables indexed by nodes in V. X is called a Bayesian Network (BN) with respect to G, if the joint distribution $p(X_1, X_2, ..., X_n)$ factorizes according to:

$$p(X_1 = x_1, X_2 = x_2, ..., X_n = x_n) = \prod_{v \in V} p(X_v = x_v \mid X_{pa(v)} = \boldsymbol{x}_{pa(v)})$$

where $pa(v)$ denotes the parents of node v and $\boldsymbol{x}_{pa(v)}$ their joint configuration (Koller and Friedman, 2009). For a given node v we summarize the set of associated conditional probabilities into a parameter vector θ_v, and these parameter vectors are assumed to be statistically independent for different nodes v, v'.

In our situation there exists a subset $\tilde{X} \subset X$ that is time dependent, i.e. $\tilde{\mathbf{x}} = (\tilde{x}(1)), ..., \tilde{x}(T))$ 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 $\tilde{X}(t)$ of $\tilde{X}$ 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. $p(\tilde{X}(t) \mid \tilde{X}(t-1)) \neq p(\tilde{X}(t+1) \mid \tilde{X}(t))$. Hence, we here use an unfolding strategy, in which we explicitly use different copies $\tilde{X}(t)$ 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 $D = \{x_{vi} \mid i = 1, ..., N,\ v \in V\}$ given G can be calculated analytically (Andrews et al., 2018):

$$\log p(D \mid G) = \sum_{v \in V} \log p(X_v \mid X_{pa(v)})$$

$$\log p(X_v \mid X_{pa(v)}) = \sum_{c \in \mathcal{C}} \ell_c(X_c)$$

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

where $\mathcal{C}$ is the set of possible partitionings of Gaussian variable X_v according to the configuration of its discrete parents, and n_c is the number of patients in partition c. X_c is the associated design matrix, and k the number of columns of that matrix. In a similar way the local log-likelihood $\ell_c(X_c)$ for a discrete node X_v with only discrete parents can be computed. 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 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; 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 (Kang, 2013; Rubin, 1976):

- missing completely at random (MCAR): The probability of missing information is not related to either the specific value which 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 drop out 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 modelling 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 super-exponentially 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 don't 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 tabu search (Hong et al., 2016), which is essentially a modified hill climber that is designed to better escape local optima. 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 was 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 v.

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 $p(x \mid z) = N(\mu(z), \sigma(z))$. One of the key ideas behind VAEs is to variationally approximate

$$\log q(z \mid x) = \log N(z \mid \mu(x), \sigma(x))$$

That means $\mu(x)$ 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

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

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

Variational Autoencoders for Heterogeneous and Incomplete Data (HI-VAE)

VAEs were originally developed for homogenous data without missing values. But, 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 is 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

$$p(x,z)=p(z)\prod_j p(x_j \mid z)$$

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

$$p(x \mid z)=\prod_{j\in\mathcal{O}} p(x_j \mid z)\prod_{j\in\mathcal{M}} p(x_j \mid z)$$

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

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

$$p(x_j \mid z)=p(x_j \mid \gamma_j=h_j(z))$$

where $h_j(\cdot)$ is a function learned by the neural network, and γ_j accordingly models data modality specific parameters (e.g. for real-valued data $\gamma_j=(\mu_j(z),\sigma_j^2(z)))$. 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.:

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

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, as training suggested that a single normal distribution was not a good fit for the data.

VAMBN: Bringing MBNs and HI-VAEs Together

Let Y_v denote the set of observed variables associated to module Y_v denote the set of observed variables associated to module $X_v, v \in V$. Note that variables X_v are low dimensional embeddings / encodings of the Y_v. The total likelihood $p(X, Y \mid G, \Theta)$ given graph G and model parameters Θ can be written as:

$$p(X, Y \mid G, \Theta) = \prod_{v \in V} p(X_v \mid pa(X_v), \Theta_v) p(Y_v \mid X_v, \Theta_v)$$

where $p(Y_v \mid X_v, \Theta_v)$ is the generative model of the data represented by HI-VAE (it is the decoder distribution). Moreover, $pa(X_v)$ denotes all module nodes plus (in our case 1 dimensional) categorical s variables, see last Section. Hence, $p(X_v \mid pa(X_v), \Theta_v)$ is a normal distribution with mean

$$m_v = \Theta_v^{(0)} + \sum_{\rho \in pa(X_v)} \Theta_v^{(\rho)} \rho$$

(i.e. modeled via a linear regression with intercept $\Theta_v^{(0)}$ and slope coefficients $\Theta_v^{(\rho)}$), and residual variance $\nu_v = Var(X_v - m_v)$.

Our aim is to find parameters Θ maximizing $\log p(X, Y \mid G, \Theta)$. 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 $v \in V$: $\hat{\Theta}_v^* = \arg\max \log p(Y_v \mid X_v, \hat{\Theta}_v)$. This is achieved via training a HI-VAE model for each module X_v, i.e. optimizing associated network weights $\hat{\Theta}_v$.
2. For all $v \in V$: $\tilde{\Theta}_v^* = \arg\max \log p(X_v \mid pa(X_v), \tilde{\Theta}_v)$. This is achieved by learning the MBN structure G and associated parameters $\tilde{\Theta}_v$ 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 hyper-parameter optimization over
    a. learning rate ∈{0.01, 0.001}
    b. mini-batch size ∈ {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_v 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 $N(m_v, \nu_v)$ as a prior for X_v 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 of the total model likelihood $p(X, Y \mid G, \Theta)$ 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. That means 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 (Pearl, 2000). That means rather than sampling from a joint distribution $p(X_1, X_2, \ldots, X_n)$ we draw from $p(X_1, X_2, \ldots, X_{p-1}, X_{p+1}, \ldots, X_n \mid do(X_p = x))$ where X_p denotes a variable in the MBN, in which we simulate the counterfactual scenario $X_p = x$. 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 $B_{\mathcal{A}}$ and $B_{\mathcal{B}}$ for both datasets. Moreover, we suppose $pa(Y) \subset B_{\mathcal{A}}$, i.e. parents of Y are also in A. Hence, we can draw from the interventional distribution

$$p(Y \mid do(pa(Y) = \mathbf{a}))$$

where $\mathbf{a}$ 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 re-identify 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 $\mathcal{A}$ be a randomized algorithm and $0 < \epsilon, 0 < \delta < 1$. According to (Dwork et al., 2006a) $\mathcal{A} : \mathcal{D} \rightarrow \mathcal{R}$ is said to respect (ϵ, δ) differential privacy, if for any two datasets $D_1, D_2 \in \mathcal{D}$ that differ only in one single patient and for any output of the randomized algorithm $\mathcal{S} \subseteq \mathcal{R}$ we have

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

(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. That means the property for 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 up-titrated within a 13 weeks 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 & 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 multi-modal, 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 two 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, CSF (cerebrospinal fluid) biomarkers (A-β, α-synuclein, dopamine, phospho-tau, total tau) and UPDRS scores.

Results

VAMBN Reflects Expected Causal Relationships in Data

As outlined in the Methods part of this paper 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). That means that for each study, we resampled the existing N patients 1000 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.

Figures 2a), b) highlight that in both SP513 as well as PPMI inferred edges agree well with expected causal dependencies: For example, in SP513 (Figure 2a) 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 & Yahr stages and the illness severity score defined in SP513. Safety biomarkers depend on gender, but otherwise have no impact.

In PPMI (Figure 2b) 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.

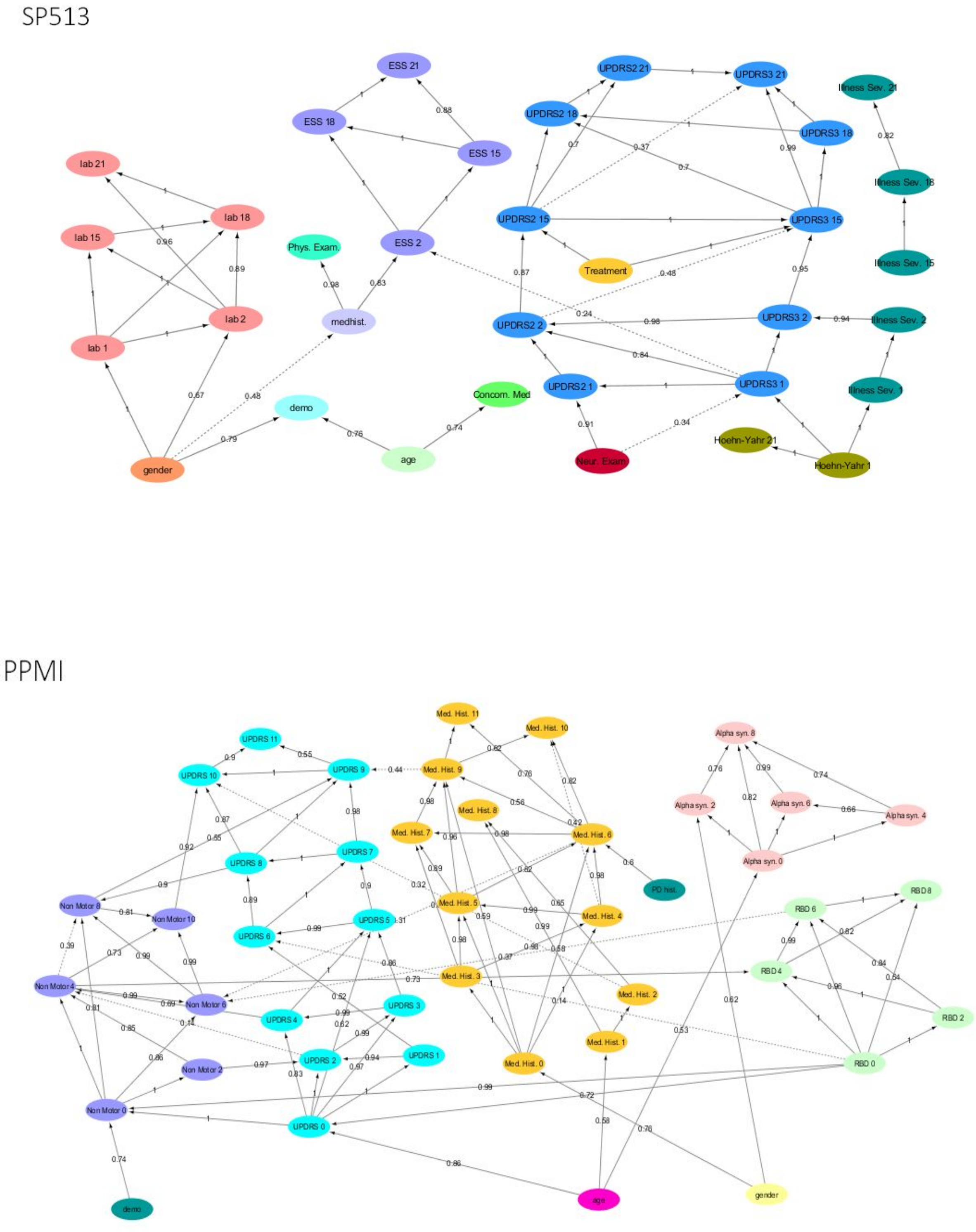


Figure 2. Final MBNs learned by VAMBN based on SP513 and PPMI data. The edges are labelled with the bootstrap frequencies of each connection. For readability, auxiliary variables and missing visit nodes were removed for the visualisation. Figures are also available as Cytoscape files in the Supplements for better convenience.

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 re-generate 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.

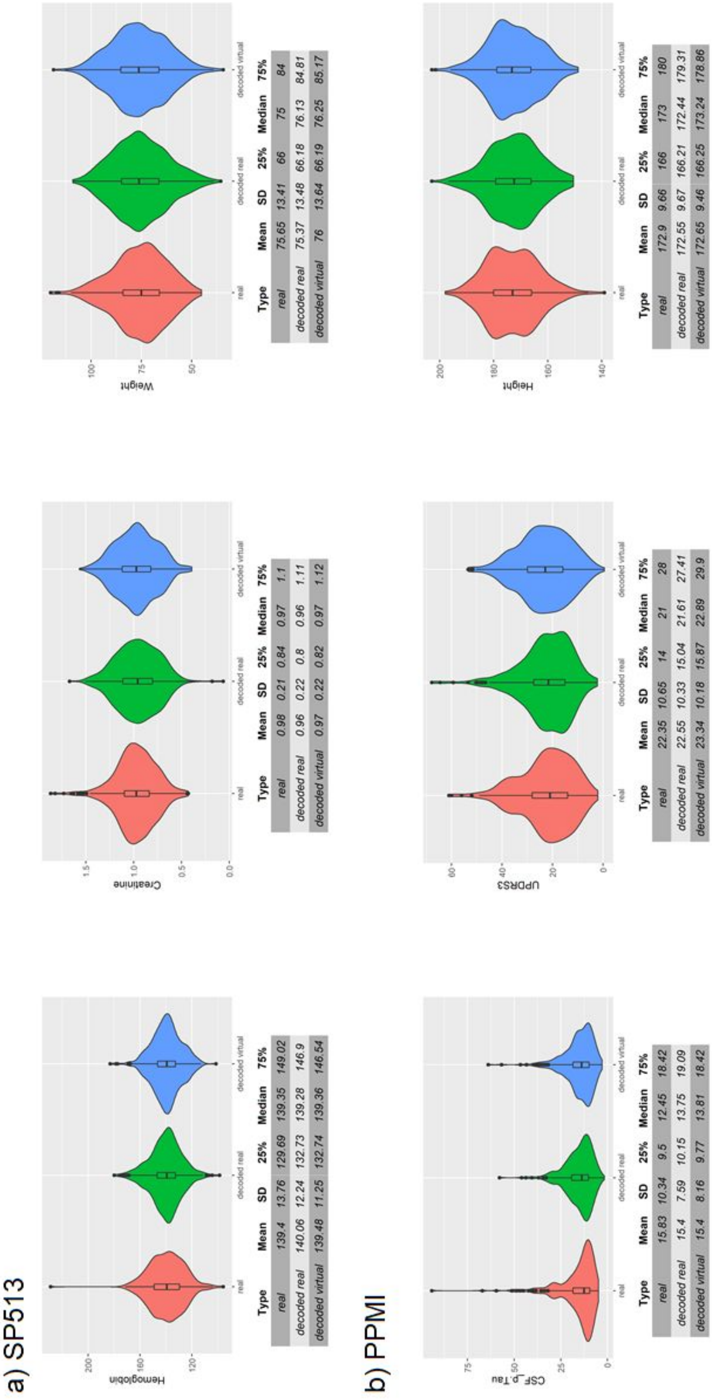
From our point of view simulated patients should mainly fulfill three criteria:

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) 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.

UPDRS3 scores in simulated PD patients showed similar differences to healthy controls than real PD patients im PPMI (Figure 5 right). Moreover, the ropinirole treatment effect in simulated and real SP513 patients demonstrated a comparable effect size and p-value (Figure 5 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 was a simple re-generation of one of the patients in the training data.



Type	Mean	SD	25%	Median	75%
real	*139.4*	*13.76*	*129.69*	*139.35*	*149.02*
decoded real	*140.06*	*12.24*	*132.73*	*139.28*	*146.9*
decoded virtual	*139.48*	*11.25*	*132.74*	*139.36*	*146.54*

Type	Mean	SD	25%	Median	75%
real	*0.98*	*0.21*	*0.84*	*0.97*	*1.1*
decoded real	*0.96*	*0.22*	*0.8*	*0.96*	*1.11*
decoded virtual	*0.97*	*0.22*	*0.82*	*0.97*	*1.12*

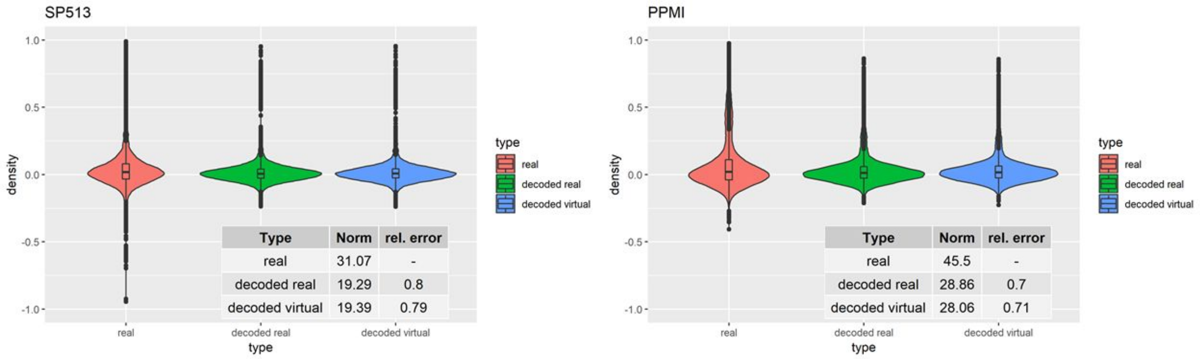
Type	Mean	SD	25%	Median	75%
real	*75.65*	*13.41*	*66*	*75*	*84*
decoded real	*75.37*	*13.48*	*66.18*	*76.13*	*84.81*
decoded virtual	*76*	*13.64*	*66.19*	*76.25*	*85.17*

Type	Mean	SD	25%	Median	75%
real	*15.83*	*10.34*	*9.5*	*12.45*	*18.42*
decoded real	*15.4*	*7.59*	*10.15*	*13.75*	*19.09*
decoded virtual	*15.4*	*8.16*	*9.77*	*13.81*	*18.42*

Type	Mean	SD	25%	Median	75%
real	*22.35*	*10.65*	*14*	*21*	*28*
decoded real	*22.55*	*10.33*	*15.04*	*21.61*	*27.41*
decoded virtual	*23.34*	*10.18*	*15.87*	*22.89*	*29.9*

Type	Mean	SD	25%	Median	75%
real	*172.9*	*9.66*	*166*	*173*	*180*
decoded real	*172.55*	*9.67*	*166.21*	*172.44*	*179.31*
decoded virtual	*172.65*	*9.46*	*166.25*	*173.24*	*178.86*

Figure 3. Examples of real and simulated / virtual patients for SP513 (a) and PPMI (b) datasets. The Figure compares the marginal distributions of selected variables for real patients (red), virtual patients (blue) and real patients decoded via the 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.



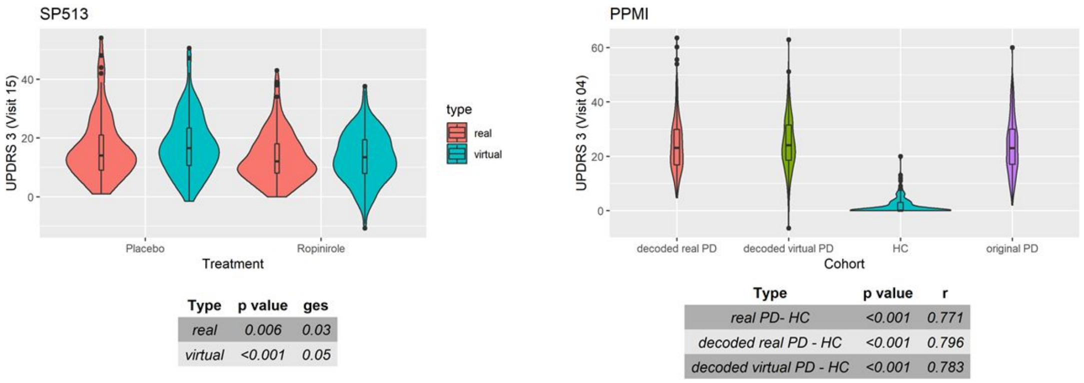
SP513

Type	Norm	rel. error
real	31.07	-
decoded real	19.29	0.8
decoded virtual	19.39	0.79

PPMI

Type	Norm	rel. error
real	45.5	-
decoded real	28.86	0.7
decoded virtual	28.06	0.71

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.



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 5. Distribution of UPDRS3 scores in SP513 (left) and 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 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.

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 set and test set, indicating a sufficiently good agreement (Figure 6). We thus concluded that VAMBN models are generally not overfitted. That means the previously reported agreement of virtual and real patients cannot just be the result of overfitting the data with an overly complex model.

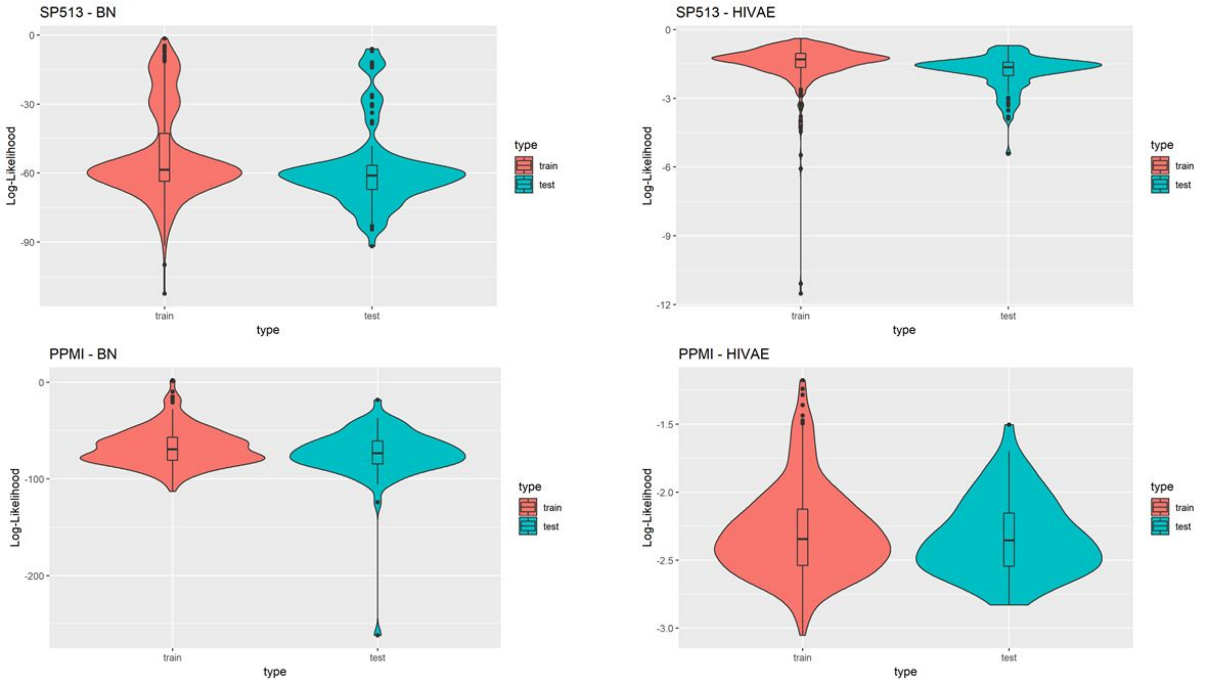


Figure 6. This figure compares the log-likelihoods of real patients in a training set (red) and a test set (blue) of the SP513 (top row) and PPMI datasets (bottom row) for the MBN and the HI-VAE models. The HI-VAE log-likelihoods are based on the participants included in the respective sets after averaging across all separate HI-VAE models (9 for SP513, 34 for PPMI).

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 the Methods part. Figure 7a) demonstrates the effect of counterfactually altering UPDRS2 and UPDRS3 baseline scores of all patients in SP513 to the mean observed in PPMI, i.e. towards 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 7b). 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 the Methods part: 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. That means there was only a simulated intervention into these features. By subtracting the simulated shift from the observed UPDRS3 off scores in PPMI we obtained a counterfactual treatment simulation with ropinirole. Figure 7c 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.

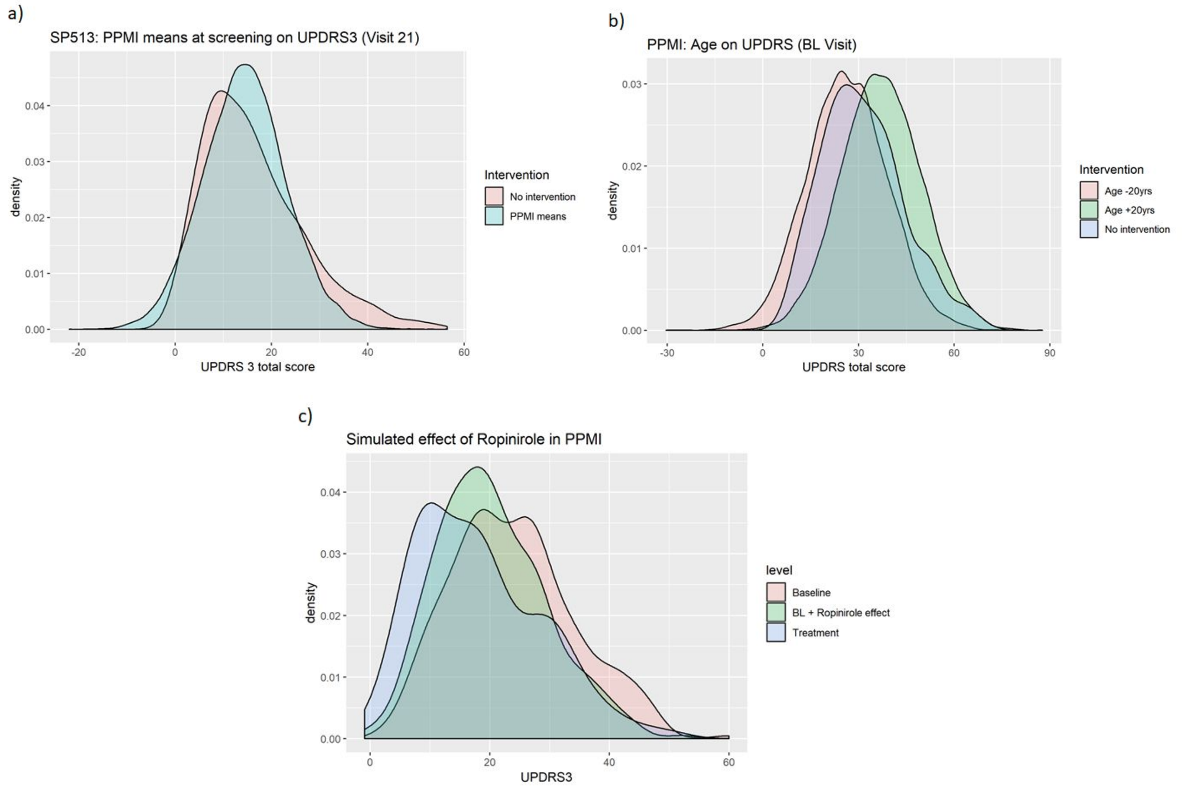


Figure 7. Counterfactual simulation of (a) lower disease severity in SP513 (shift of UPDRS3 scores at baseline to mean observed in PPMI at baseline); (b) shift of age 20 years up and down; (c) treatment effect of ropinirole in PPMI.

Differential Privacy Respecting Modeling Training

As a last point we investigated differential privacy respecting model training of VAMBN. As indicated in the Methods part 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, Figures 7 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 dependency 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 dependency 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.

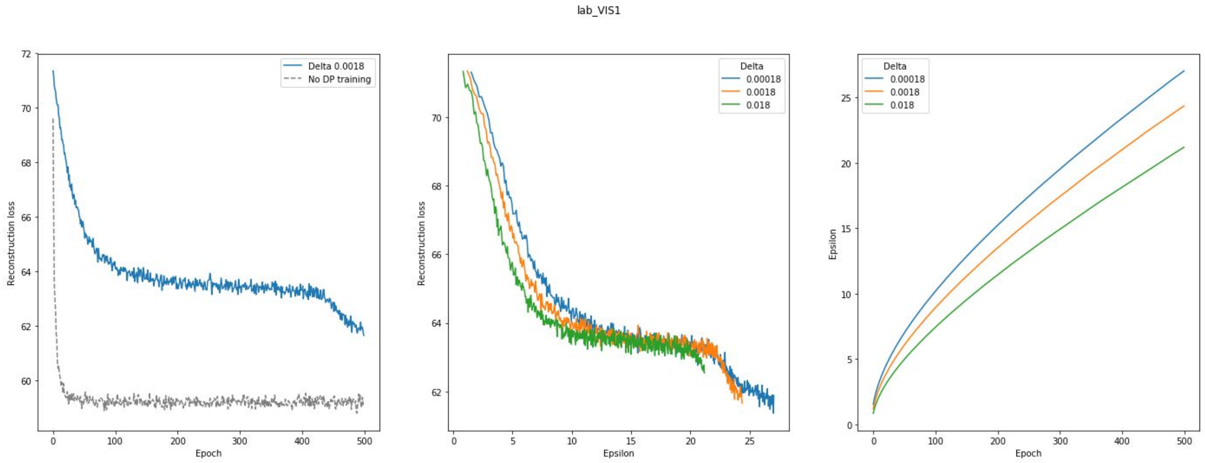


Figure 8. This figure shows the effects of differentially privacy respecting (DP) HI-VAE training on the HI-VAE step of the model. Left panel: Reconstruction loss change between DP and conventional model training for laboratory data at visit 1 for the SP513 study; middle panel: Epsilon plotted against reconstruction loss for different delta values; right panel: 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.

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 6000 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. Further distinction points of our VAMBN method include the 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 within a well-established theoretical framework, which could e.g. help in the design of clinical trials.

Our results demonstrate that VAMBN models generally do not overfit and allow for a sufficiently realistic simulation of virtual patients. Hence, we see virtual patients as valuable for data mining purposes and for counterfactually merging different datasets. In addition, we demonstrated that data privacy respecting model training is in principle possible with VAMBN.

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. Moreover, 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.

Acknowledgements

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 – is 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/.

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.

Conflict of Interest

HF and LGD received salaries from UCB Biosciences GmbH and UCB Celltech Ltd, respectively (both branches of UCB Pharma S.A.). UCB Pharma had no influence on the content of this work.

References

1. Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., and

Zhang, L. (2016). Deep Learning with Differential Privacy. ArXiv:1607.00133 [Cs, Stat].

2. Andrews, B., Ramsey, J., and Cooper, G.F. (2018). Scoring Bayesian networks of mixed variables. Int J Data Sci Anal *6*, 3–18.
3. Beaulieu-Jones, B.K., Wu, Z.S., Williams, C., Lee, R., Bhavnani, S.P., Byrd, J.B., and Greene, C.S. (2018). Privacy-preserving generative deep neural networks support clinical data sharing. BioRxiv 159756.
4. Chase, J.G., Preiser, J.-C., Dickson, J.L., Pironet, A., Chiew, Y.S., Pretty, C.G., Shaw, G.M., Benyo, B., Moeller, K., Safaei, S., 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.
5. 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.
6. Davison, A.C., and Hinkley, D.V. (1997). Bootstrap Methods and Their Application (Cambdrige, UK: Cambridge University Press).
7. 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, S. Vaudenay, ed. (Springer Berlin Heidelberg), pp. 486–503.
8. Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006b). Calibrating Noise to Sensitivity in Private Data Analysis. In Theory of Cryptography, S. Halevi, and T. Rabin, eds. (Springer Berlin Heidelberg), pp. 265–284.
9. Fröhlich, H., Balling, R., Beerenwinkel, N., Kohlbacher, O., Kumar, S., Lengauer, T., Maathuis, M.H., Moreau, Y., Murphy, S.A., Przytycka, T.M., et al. (2018). From hype to reality: data science enabling personalized medicine. BMC Med *16*, 150.
10. Galbusera, F., Niemeyer, F., Seyfried, M., Bassani, T., Casaroli, G., Kienle, A., and Wilke, H.-J. (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*.
11. Ghahramani, Z. (1998). Learning dynamic Bayesian networks. In Adaptive Processing of Sequences and Data Structures, (Springer-Verlag), pp. 168–197.
12. Ghosh, A., Hong, J., Yin, D., and Ramchandran, K. (2019). Robust Federated Learning in a Heterogeneous Environment. ArXiv:1906.06629 [Cs, Stat].
13. 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.
14. Heckerman, D. (1997). A Tutorial on Learning With Bayesian Networks. Data Mining and Knowledge Discovery *1*, 79–119.
15. Hochreiter, S., and Schmidhuber, J. (1997). Long Short-Term Memory. Neural Comput. *9*, 1735–1780.
16. Holford, N., Ma, S.C., and Ploeger, B.A. (2010). Clinical Trial Simulation: A Review. Clinical Pharmacology & Therapeutics *88*, 166–182.
17. 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), pp. 258–264.
18. Kang, H. (2013). The prevention and handling of the missing data. Korean J Anesthesiol *64*, 402–406.
19. Kingma, D.P., and Welling, M. (2013). Auto-Encoding Variational Bayes. ArXiv:1312.6114 [Cs, Stat].

20. Knab, T.D., Clermont, G., and Parker, R.S. (2016). A "Virtual Patient" Cohort and Mathematical Model of Glucose Dynamics in Critical Care. IFAC-PapersOnLine *49*, 1–7.
21. Koller, D., and Friedman, N. (2009). Probabilistic Graphical Models: Principles and Technique (MIT Press).
22. 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 Adherence *11*, 1295–1307.
23. McMahan, H.B., Moore, E., Ramage, D., Hampson, S., and Arcas, B.A. y (2016). Communication-Efficient Learning of Deep Networks from Decentralized Data. ArXiv:1602.05629 [Cs].
24. Mustillo, S., and Kwon, S. (2015). Auxiliary Variables in Multiple Imputation When Data Are Missing Not at Random. The Journal of Mathematical Sociology *39*, 73–91.
25. Nazabal, A., Olmos, P.M., Ghahramani, Z., and Valera, I. (2018). Handling Incomplete Heterogeneous Data using VAEs. ArXiv:1807.03653 [Cs, Stat].
26. Pappalardo, F., Russo, G., Tshinanu, F.M., and Viceconti, M. (2018). In silico clinical trials: concepts and early adoptions. Briefings in Bioinformatics.
27. Parkinson Progression Marker Initiative (2011). The Parkinson Progression Marker Initiative (PPMI). Prog. Neurobiol. *95*, 629–635.
28. Pearl, J. (2000). Causality: Models, Reasoning and Inference (Cambridge: Cambridge University Press).
29. 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.
30. Rubin, D.B. (1976). Inference and Missing Data. Biometrika *63*, 581–592.
31. Scutari, M. (2010). Learning Bayesian Networks with the bnlearn R Package. Journal of Statistical Software *35*, 1–22.
32. Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., and Friedman, N. (2003). Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nat Genet *34*, 166–176.
33. Segal, E., Pe'er, D., Regev, A., Koller, D., and Friedman, N. (2004). Learning module networks. In Advances in Neural Information Processing Systems, pp. 297–304.
34. Zhuo, C., Zhu, X., Jiang, R., Ji, F., Su, Z., Xue, R., and Zhou, Y. (2017). Comparison for Efficacy and Tolerability among Ten Drugs for Treatment of Parkinson's Disease: A Network Meta-Analysis. Sci Rep *7*.

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