Generation of realistic synthetic data using multimodal neural ordinary differential equations

Philipp Wendland‡[1,2], Colin Birkenbihl‡[1,3], Marc Gomez-Freixa[3], Meemansa Sood[1,3], Maik Kschischo[2], Holger Fröhlich*[1,3]

‡ Authors contributed equally to this work.

1. Department of Bioinformatics, Fraunhofer Institute for Algorithms and Scientific Computing (SCAI), Sankt Augustin 53754, Germany
2. Department of Mathematics and Technology, University of Applied Sciences Koblenz, Remagen 53424, Germany
3. Bonn-Aachen International Center for IT, Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn 53115, Germany

*Corresponding Author:
Holger Fröhlich
holger.froehlich@scai.fraunhofer.de
Fraunhofer-Institute for Algorithms and Scientific Computing (SCAI)
Schloss Birlinghoven
D-53757 Sankt Augustin

Abstract

Individual organizations, such as hospitals, pharma companies and health insurance providers are currently limited in their ability to collect data that is fully representative of a disease population. This can in turn negatively impact the generalization ability of statistical models and scientific insights. However, sharing data across different organizations is highly restricted by legal regulations. While federated data access concepts exist, they are technically and organizationally difficult to realize. An alternative approach would be to exchange synthetic patient data instead. In this work, we introduce the Multimodal Neural Ordinary Differential Equation (MultiNODE), a hybrid, multimodal AI approach, which allows for generating highly realistic synthetic

patient trajectories on a continuous time scale, hence enabling smooth interpolation and extrapolation of clinical studies. Our proposed method can integrate both static and longitudinal data and implicitly handles missing values. We demonstrate the capabilities of our approach by applying it to real patient-level data from two independent clinical studies and simulated epidemiological data of an infectious disease.

Introduction

Patient-level data build the foundation for a plethora of healthcare research endeavors such as drug discovery, clinical trials, biomarker discovery, and precision medicine [1]. Collecting respective data is extremely time-consuming and cost-intensive, and additionally access-restricted by ethical and legal regulations in most countries. Individual organizations, such as hospitals, pharma companies and health insurance providers are currently limited in their ability to collect data that is fully representative of a disease population. This issue is especially pronounced in clinical studies, where patients are usually included based on defined inclusion and exclusion criteria. Differences in these selection criteria between studies, can introduce cohort-specific statistical biases [2] which, in turn, can negatively impact the generalization ability of machine learning models, since the usual i.i.d. assumption is violated [3]. A naïve idea to counteract the situation might be to build up large data repositories pooling diverse clinical studies from several organizations. However, a major obstacle is that sharing of patient-level data across different organizations is exceedingly difficult from a legal point of view, as formulated, for example, in the General Data Protection Rule (GDPR) in the European Union. Furthermore, naïve pooling of several biased datasets would bias a machine learning model to preferentially learn from the most abundant data source, hence still resulting in a biased model [2]. While this aspect might in theory be

addressable via transfer learning strategies, it must be understood that clinical studies within the same indication area can also differ in the set of collected variables as well as the intervals between consecutive follow-up assessments. In addition to these technical considerations, the setup of a federated machine learning framework across several institutions is a major organizational undertaking, which is costly and time consuming.

The idea we propagate in this paper is to learn a continuous-time generative machine learning model from clinical study data. Given the distribution of the real training data was appropriately learned by the model, the generated synthetic datasets maintain the real data signals, such as variable interdependencies and time-dependent trajectories. Furthermore, these synthetic datasets can overcome crucial limitations of their real counterparts like missing values or irregular assessment intervals, hence opening the opportunity to make at least subsets of variables from different studies statistically comparable. A further strong motivation for generating synthetic datasets is the aim to use the generated data as an anonymized version of its real-world counterpart and thereby mitigate the increased restrictions for sharing human data [4, 5, 6]. However, synthetic patient-level datasets open opportunities that reach far beyond data sharing. For example, trained generative models could be used for synthesizing control arms for clinical trials based on data from previously conducted trials, or from real-world clinical routine data [7]. This helps addressing major ethical concerns in disease areas, such as cancer, where it is impossible to leave patients untreated. Both, the American Food and Drug Administration (FDA) and the European Medicines Agency (EMA) have recognized this issue and taken initiatives to allow for synthetic control arms [7].

Over the last years, generative models (mostly Generative Adverserial Networks - GANs) have found notable success, mostly in the medical imaging domain [8,9,19-

22]. However, GANs are often found to show a collapse to the statistical mode of a distribution, which raises concerns regarding coverage of the real patient distribution by synthetic data. Moreover, these methods are not necessarily suited to cope with the complex nature of clinical data collected in observational, longitudinal cohort studies, which is the main focus of our work: In addition to the previously mentioned issue of irregular measurement frequencies and missing values not at random (for example due to participant drop-out), clinical studies often comprise several modalities combining time-dependent variables (e.g., measures of disease severity) and static information (e.g., biological sex). One approach specifically designed for the joint modeling and generation of multimodal, time-dependent and static patient-level data containing missing values are the recently introduced Variational Autoencoder Modular Bayesian networks (VAMBN) [4]. However, VAMBN only operate on a discrete time scale while relevant clinical indicators such as, for example, disease progression expressed through cognitive decline or rising inflammatory markers, are intrinsically time continuous. Recently, Neural Ordinary Differential Equations (NODEs) have been introduced as a hybrid approach fusing neural networks and ordinary differential equations (ODE) [10]. While NODEs are time continuous and thus enable smooth interpolation between observed data points and extrapolation beyond the observations in the data, they are not able to integrate static variables.

In this work, we present the Multimodal Neural Ordinary Differential Equations (MultiNODEs) as an extension of the NODEs. MultiNODEs allow learning a generative model from multimodal longitudinal and static data that may contain missing values not at random. To demonstrate MultiNODEs' generative capabilities, we applied the model to clinical, patient-level data from an observational Parkinson's disease (PD) cohort study (the Parkinson's Progression Markers Initiative, PPMI [11]) and,

additionally, a longitudinal Alzheimer's disease (AD) data collection (National Alzheimer's Coordination Center, NACC [12]). We compared the generated trajectories and correlation structure against the real counterpart. In this context, we additionally evaluated MultiNODEs' performance against the previously published VAMBN approach. Furthermore, we assessed MultiNODEs' interpolation and extrapolation performance. Finally, we investigated the influence of sample size, noisiness of the data, and longitudinal assessment density onto the training of MultiNODEs in a systematic benchmark on data simulated from a mathematical model well-known in the epidemiology field.

Results

Conceptual Introduction of the MultiNODEs

MultiNODEs represent an extension of the original NODEs framework [10] that overcome the limitations of its predecessor such that an application to incomplete datasets consisting of both static and time-dependent variables becomes feasible. Conceptually, MultiNODEs build on three key components **(Figure 1)**: 1) latent NODEs, 2) a variational autoencoder (more specifically a Heterogenous Incomplete Variational Autoencoder - HI-VAE - designed to handle multimodal data with missing values [13]), and 3) an implicit imputation layer [14]. The latent NODEs enable the learning and subsequent generation of continuous longitudinal variable trajectories. The longitudinal properties of the initial condition (i.e., the starting point for the ODE system solver of the latent NODEs) are defined by the output of a recurrent variational encoder which embeds the longitudinal input data into a latent space **(Figure 1 orange box)**. In order to allow for an additional influence of static variables onto the estimation of the longitudinal variable trajectories, the second component, a HI-VAE, is introduced **(Figure 1 blue box)**. This component transforms the static information into

a distinct latent space and the resulting embedding is used to augment the latent starting condition of the NODEs by concatenating the static variable embedding and the latent representation of the longitudinal variables **(Figure 1 'augmentation')**. The HI-VAE component itself holds generative properties and conducts the synthesis of the static variables when MultiNODEs are applied in a generative setting. Conclusively, MultiNODEs integrate static variables (e.g., biological sex or genotype information) both to inform the learning of longitudinal trajectories, and in the generative process. Finally, to mitigate the original NODEs' incapability of dealing with missing values, we introduced the imputation layer which implicitly replaces missing values during model training with learned estimates **(Figure 1 green box)**. For further details on the model architecture, training, and hyperparameter optimization, we refer to the Method section and Supplements, respectively.

Synthetic Data Generation using MultiNODEs

Generating synthetic data using MultiNODEs starts by randomly sampling a latent representation for both the static and longitudinal variables, respectively. The longitudinal variables in data space are then generated by first constructing the initial conditions of the latent ODE system (i.e., concatenating the static latent representation to the longitudinal one), followed by solving the ODE system given these initial conditions, and finally by decoding the result into data space. The static variables are generated by directly transforming their sampled latent representation into data space using the HI-VAE decoder.

MultiNODEs support two different approaches for the initial sampling of the latent representations, namely sampling from the prior distribution employed during model training and sampling from the learned posterior distribution of the input data.

During the posterior sampling procedure, the reparameterization trick [15] is applied to draw a latent representation from the posterior distribution learned from the training data. The amount of noise added in this process can be tuned, whereas greater noise will lead to a wider spread of the generated marginal distributions of the synthetic data. Alternatively, latent representations can be drawn independently from the prior distributions imposed on the latent space during variational model training (multivariate standard Gaussian for longitudinal data, Gaussian Mixture Model for static data). However, independent sampling from two prior distributions and subsequent decoding may result into synthetic trajectories that have different statistical properties than the original real data, because we ignore statistical dependencies between static and longitudinal data. In conclusion, posterior sampling is the preferred method for data generation when aiming for a realistic synthetic counterpart of a real dataset, while the prior sampling might still be useful for increasing the sample size of data for machine learning purposes. More detailed descriptions of both procedures are provided in the Method section.

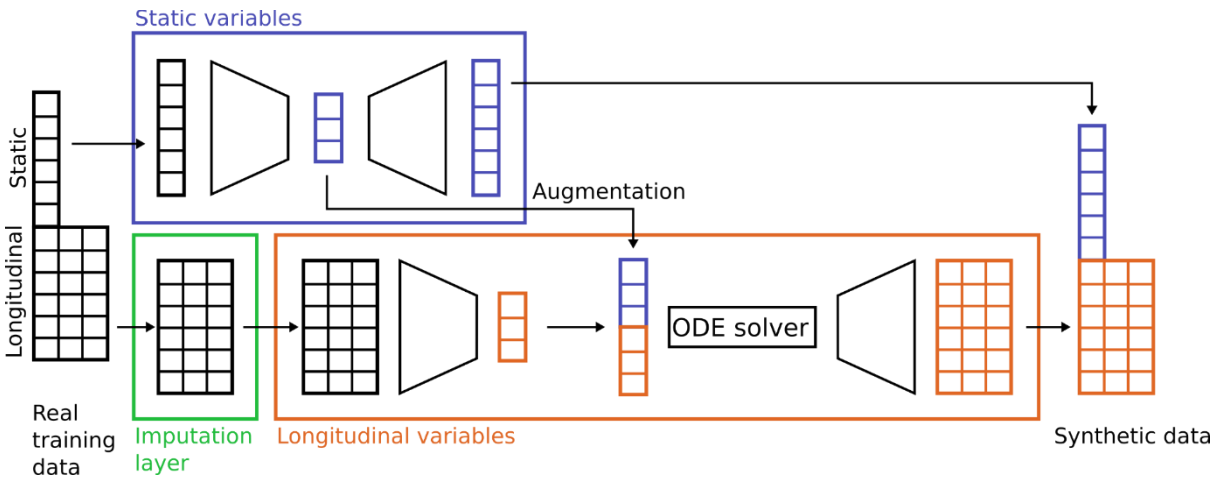

Figure 1: Conceptual framework of MultiNODEs. Blue box: HI-VAE for the encoding and generation of static variables. Orange box: NODEs that learn and generate longitudinal trajectories. Green box: The imputation layer which can handle missing

data implicitly during model training.

Application cases: Parkinson's disease and Alzheimer's disease

We applied MultiNODEs to longitudinal, multimodal data from two independent clinical datasets with the goal of generating realistic synthetic datasets that maintain the real data properties. Details about the data pre-processing steps are described in the Supplementary Material.

The first dataset was the Parkinson's Progression Markers Initiative (PPMI), an observational clinical study containing 354 de-novo PD patients who participated in a range of clinical, neurological, and demographic assessments which form the variables of the dataset. In total, a set of 25 longitudinal and 43 static variables was investigated.

Furthermore, as a second example, we applied MultiNODEs to longitudinal, multimodal data from the National Alzheimer's Coordinating Center (NACC). NACC is a database storing patient-level AD data collected across multiple memory clinics. After preprocessing, the dataset used in this study contained 2284 patients and a set of 4 longitudinal and 4 static variables was investigated.

In the following sections, we will focus on the results achieved on the PPMI data and refer to the equivalent experiments based on the NACC data which are presented in the Supplementary Material.

MultiNODEs generate realistic synthetic patient-level datasets

Although it was expected that data generated using the posterior sampling would resemble the real-world data more closely, we additionally applied the prior sampling for comparison purposes. With each method, we generated the same number of synthetic patients as encountered in the real dataset to allow for a fair comparison. To

assess whether the generated data followed the real data characteristics sufficiently closely, we conducted thorough comparisons of the marginal distributions and investigated the underlying correlation structure of the measured variables. Across all these aspects, we evaluated MultiNODEs' performance against the previously published VAMBN approach [4] as a benchmark.

The synthetic data generated using MultiNODEs' posterior sampling exhibited marginal distributions which were highly similar to their corresponding real counterparts. As expected, sampling and subsequent data generation from the posterior distribution resulted in synthetic data that resembled the real data more closely than those generated from the prior distribution **(Figure 2 A-C)**. This was also indicated by lower Kullback-Leibler divergences (KL-divergence) between the marginal distributions generated from the posterior and the real distribution relative to those generated from the prior distribution.

Compared to VAMBN, MultiNODEs' posterior sampling produced marginal distributions that resembled the original data more closely for 51,7 % of the generated distributions, again indicated through lower respective KL-divergences between the generated and real distributions. Especially considering skewed, continuous variables, MultiNODEs showed closer approximation than VAMBN **(e.g., Figure 2 A)**. With respect to the static variables, both VAMBN and MultiNODEs' posterior sampling produced realistic distributions while prior sampling led to substantial deviations between synthetic and real data **(Figure 2 D)**. More examples of generated distributions on the PPMI data can be found in the Supplementary Material **(Figure S1)**. Equivalent results for the NACC data are presented in **Figure S4**.

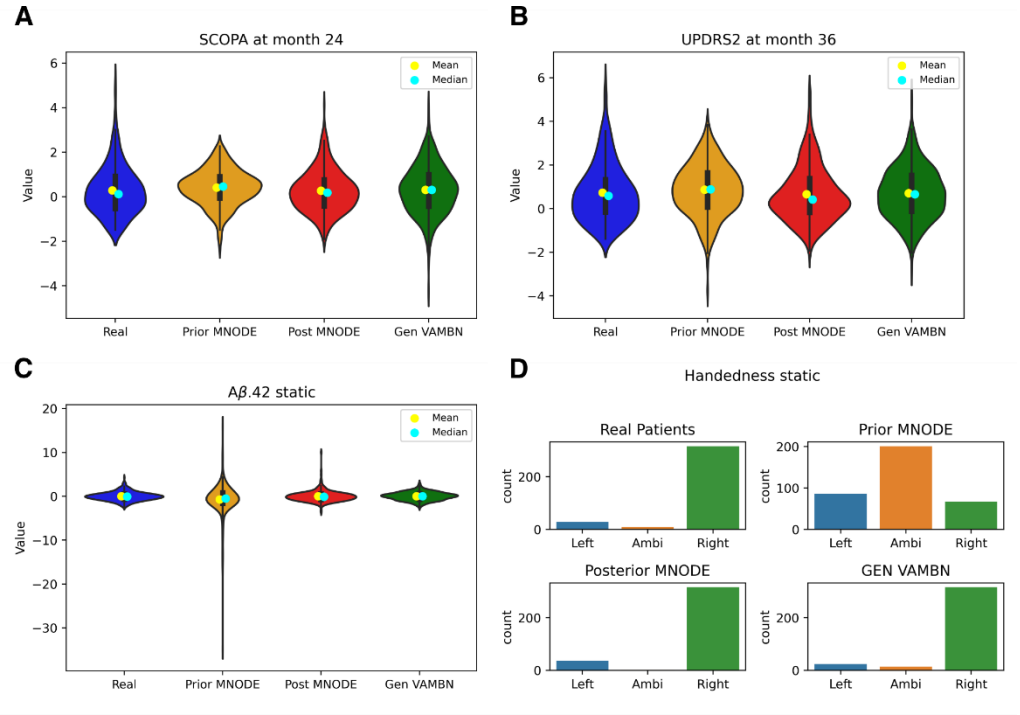

Figure 2: Marginal distributions of real and synthesized data for multiple variables. Mean, standard deviation and KL-Divergence for the displayed variables can be found in **Table S1**. Equivalent results for the NACC data are presented in **Figure S5**. **A,** time dependent variable 'SCOPA' at month 24. **B,** time dependent variable 'UPDRS2' at month 36. **C,** static variable 'Aβ.42'. **D,** categorical static variable 'Handedness'.

In order to evaluate whether MultiNODEs learned not only to reproduce univariate distributions but actually captured their interdependencies accurately, we compared the correlation structure of the generated data to the that of the real variables. Visualizations of the Spearman rank correlation coefficients showed that both the prior and posterior sampling generated synthetic data which successfully reproduced the real variables' interdependencies **(Figure 3)**. The only exception to this was that the synthetic data sampled from the prior failed to recover the correlations among the static variables (absence of the lower right block in **Figure 3 C**). Comparing the results against VAMBN generated data revealed that both generation procedures of

MultiNODEs were significantly better at reproducing the real data characteristics: the Frobenius norm of real data correlation matrix resulted in 45.3, and with a Frobenius norm of 25.66 the VAMBN generated data placed substantially further from the real data than the MultiNODEs approaches with 58.72 and 52.58 for the prior and posterior sampling, respectively. Here, it shows that MultiNODEs slightly overestimated the present correlations, while VAMBN substantially underestimated them. Concordantly, the relative error (i.e., the deviation of the respective synthetic dataset's correlation matrix from the real one normalized by the norm of the real correlation matrix), was 0.81, 0.56 and 0.40 respectively for VAMBN and MultiNODEs' prior and posterior sampling, leaving MultiNODEs with a significantly lower error than the VAMBN approach.

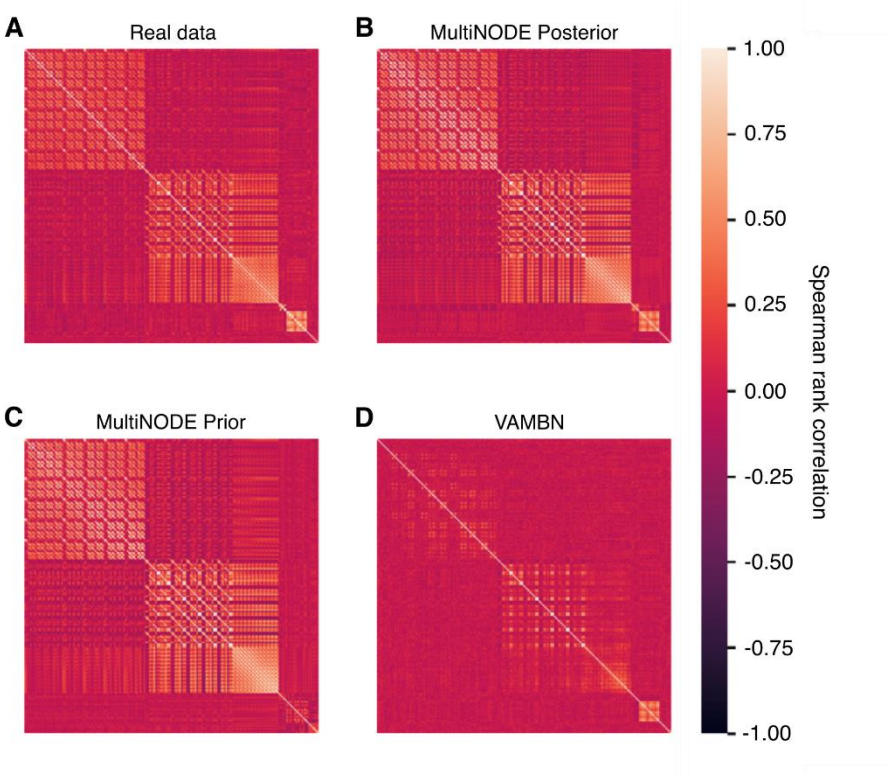

Figure 3: Correlation structure of real and synthetic data expressed as spearman rank correlation coefficients. Equivalent results for the NACC data are shown in **Figure S6**.

A, real data. **B,** posterior sampling from MultiNODEs. **C,** prior sampling from MultiNODEs. **D,** VAMBN generated data.

Generating data in continuous time through smooth interpolation and extrapolation

One particular strength of MultiNODEs, that sets it apart from alternative approaches such as VAMBN, is its ability to model variable trajectories in continuous time. The latent ODE system allows for estimation of variable trajectories at any arbitrary time point and thereby opens possibilities for 1) the generation of smooth trajectories, 2) overcoming panel-data limitations through interpolation, and finally 3) extrapolation beyond the time span covered in training data themselves. Again, we evaluated these capabilities based on the PPMI and NACC datasets.

Comparing the median trajectories of variables from the real data to those generated using MultiNODEs revealed that MultiNODEs accurately learned and reproduced the longitudinal dynamics exhibited in the real data **(Figure 4)**. Generation from both the prior and posterior distribution led to synthesized median trajectories that closely resembled the real median trajectories. Equivalently, also the 97.5% and 2.5% quantiles of the synthetic data approximated the corresponding real quantiles closely, indicating a realistic distribution of the synthetic data across the observed time points. This observation held true for most of the time-dependent variables (plots for all variables are linked in the Supplementary Material).

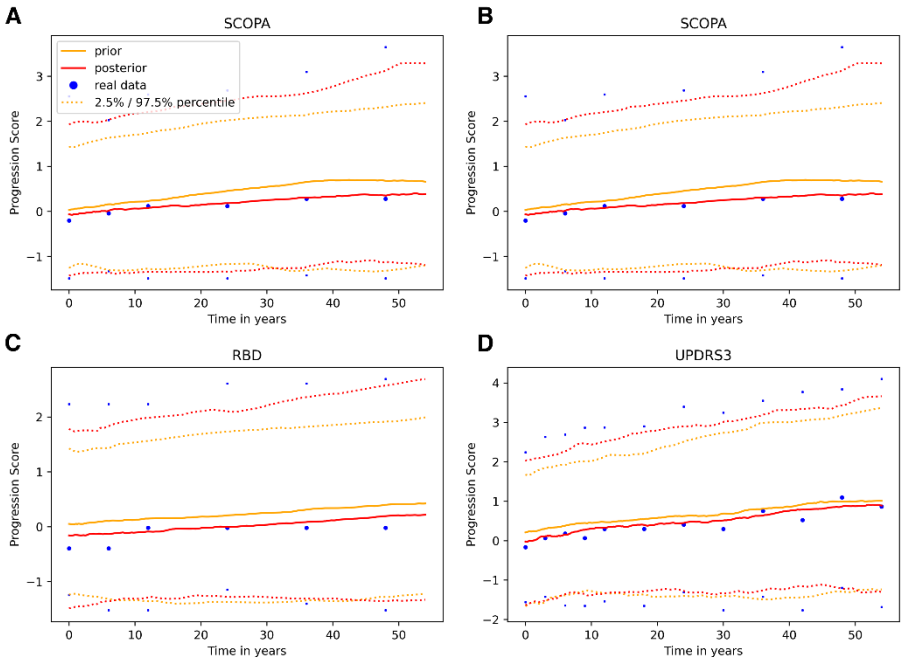

Figure 4: Comparison of median trajectories including the 2.5% / 97.5% quantiles of longitudinal variables from synthetic and real PPMI data. Additional examples are provided in **Figure S2**. A corresponding example for the NACC dataset is shown in **Figure S7. A, B, C, D,** depict different longitudinal variables from the PPMI dataset.

We further assessed the interpolation and extrapolation capabilities of MultiNODEs. For interpolation, one time point was excluded from model training and subsequently data was generated for all time points including the one left out. Contrasting the interpolated values against the corresponding real values showed that MultiNODEs accurately reproduced the longitudinal dynamics of a variable, even for unobserved time points **(Figure 5 B)**. In this context, we further compared the interpolated values against synthetic data that was generated based on the complete, real data trajectory. We observed that the median KL-divergence between the interpolated data and the real data was only slightly higher than between the real data and the synthetic data generated after training MultiNODEs on the complete trajectory (0.08 and 0.05, respectively). Similarly, the relative error between the interpolated correlation matrix

and the real data was again only marginally higher than between the complete data and the real data (0.56 and 0.53, respectively; **Figure S4**).

In order to test MultiNODEs' extrapolation capabilities, only the first 24 months of assessment follow-up and the static variables were used during model training. The trained model was then applied to generate data for the remaining, left out time points of the longitudinal variables. In this course, 77 values were extrapolated while not every variable had the same number of follow-up assessments after month 24. Comparing the extrapolated synthetic data to the left out real data demonstrated reliable extrapolation beyond the training data **(Figure 5 A, C)**. Similar to the interpolation setting, we also compared the median KL-divergence between the extrapolated data and the real data with that between the real data and synthetic data that were generated after training MultiNODEs on the complete trajectory. As expected, we could see a larger difference between the KL-divergences compared to the interpolation setting with 0.16 for the extrapolated data and 0.08 for the synthetic data based on the complete trajectory. The correlation structure was well preserved also in the extrapolation setting with a relative error of 0.56 compared to the .55 when using the complete trajectory for training MultiNODEs (**Figure S4**).

In addition, also the marginal distributions at both the interpolated and extrapolated time points followed those of the real data **(Figure 5 C, D)**.

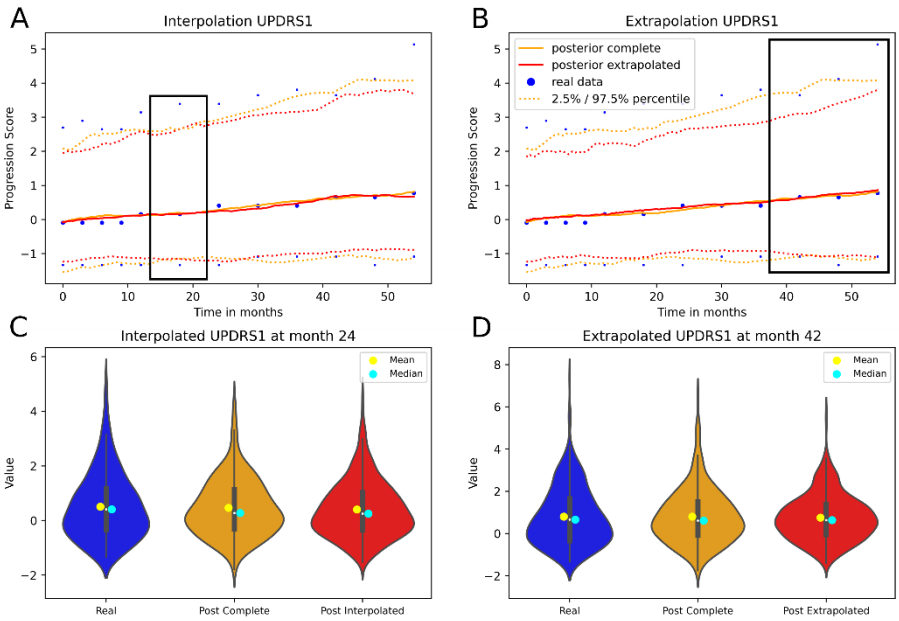

Figure 5: Time-continuous interpolation and extrapolation of exemplary PPMI variables. The black box indicates the interpolated and extrapolated sections. Plots for additional variables are presented in **Figure S3**. A corresponding example for the NACC dataset is shown in **Figure S8**. **A,** extrapolation of the last two assessments of the UPDRS1 variable. **B,** interpolation of the UPDRS1 variable at month 24. **C,** distribution of the interpolated values for UPDRS1 at visit 24. **D,** distribution of the extrapolated values for UPDRS1 at month 42. distribution of the extrapolated values for UPDRS1 at month 42.

Systematic model benchmarking on simulated data

To explore the learning properties of MultiNODEs more systematically, we investigated how alternating training conditions with respect to measurement frequency, sample size, and noisiness of the data influence MultiNODEs' generative performance.

The benchmarking data was simulated via the well-established Susceptible-Infected-Removed (SIR) model which is often used to describe the spread of infectious

diseases and follows a highly non-linear structure: Let $S(t)$ be the number of susceptible individuals at a timepoint t,$I(t)$ be the number of infectious individuals at a timepoint t and $R(t)$ be the number of removed or recovered individuals at a timepoint t. With β as transmission rate, γ as mean recovery / death rate and $N = S(t) + I(t) + R(t)$ as fixed population size the SIR model can be defined by the following ODE system:

$$\begin{aligned} \frac{dS}{dt} &= -\frac{\beta SI}{N} \\ \frac{dI}{dt} &= \frac{\beta SI}{N} - \gamma I. \\ \frac{dR}{dt} &= \gamma I \end{aligned}$$

Details about the SIR parameter settings are described in the supplementary material.

As baseline settings for each investigation, we simulated 1000 data points with 10 equidistant assessment time points each, distributed over a span of 40 time intervals and added 5% Gaussian noise to each measurement. That means we added a normally distributed variable with the standard deviation set to 5% of the theoretical range of each of the variables S(t), I(t) and R(t). During the benchmarking, we individually alternated the sample size, time points and the noise level. For the time point investigation, we compared MultiNODEs' trained on 5, 10 and 100 equidistant assessments; for the sample size we considered 100, 1000, and 5000 samples; and for the noise level we tested 50%, 75% and 100% of the maximum encountered value added as noise.

Alternating the amount of equidistant, longitudinal time points exposed a strong dependency of MultiNODEs on the longitudinal coverage of the time dependent process **(Figure 6 A)**. While the general trends in the data were appropriately learned for all explored assessment frequencies, the position of the observations in time

influenced how close the learned function approximated the true data underlying process. Especially the peak of the 'Infected'-function represented a challenge for MultiNODEs if no data point was located close to it **(Figure 6 A 'Infected')**. Similarly, the start of the decline in the 'Susceptible'-function and the incline in the 'Removed'-function were shifted, depending on the positioning of measurements. In conclusion, and as expected, a higher observation frequency of the data underlying time-dependent process significantly increased the fit of MultiNODEs to the process, although, general trends could already be approximated for lower assessment frequencies.

Investigating the effect of the sample size on training MultiNODEs, we observed that an increase of the sample size led to an expected improvement of the model fit to the SIR dynamics **(Figure 6 B)**. While the general trends could again be learned from limited data (n = 100), sample sizes of 1000 or 5000 substantially reduced the model's deviation from the true SIR model. With 1000 samples, the learned dynamic is less stable than when trained on 5000 samples, where a smooth dynamic was learned that closely resembled the true underlying process. In conclusion, MultiNODEs can already learn longitudinal dynamics based on only a few data points, however, they tend to underfit under these circumstances and benefit from larger sample sizes.

Adding an increasing noise level to the SIR training data revealed that MultiNODEs remain very robust. Only when introducing 100% of the maximal encountered value as additional noise, a clear deviation from the underlying true model could be observed.

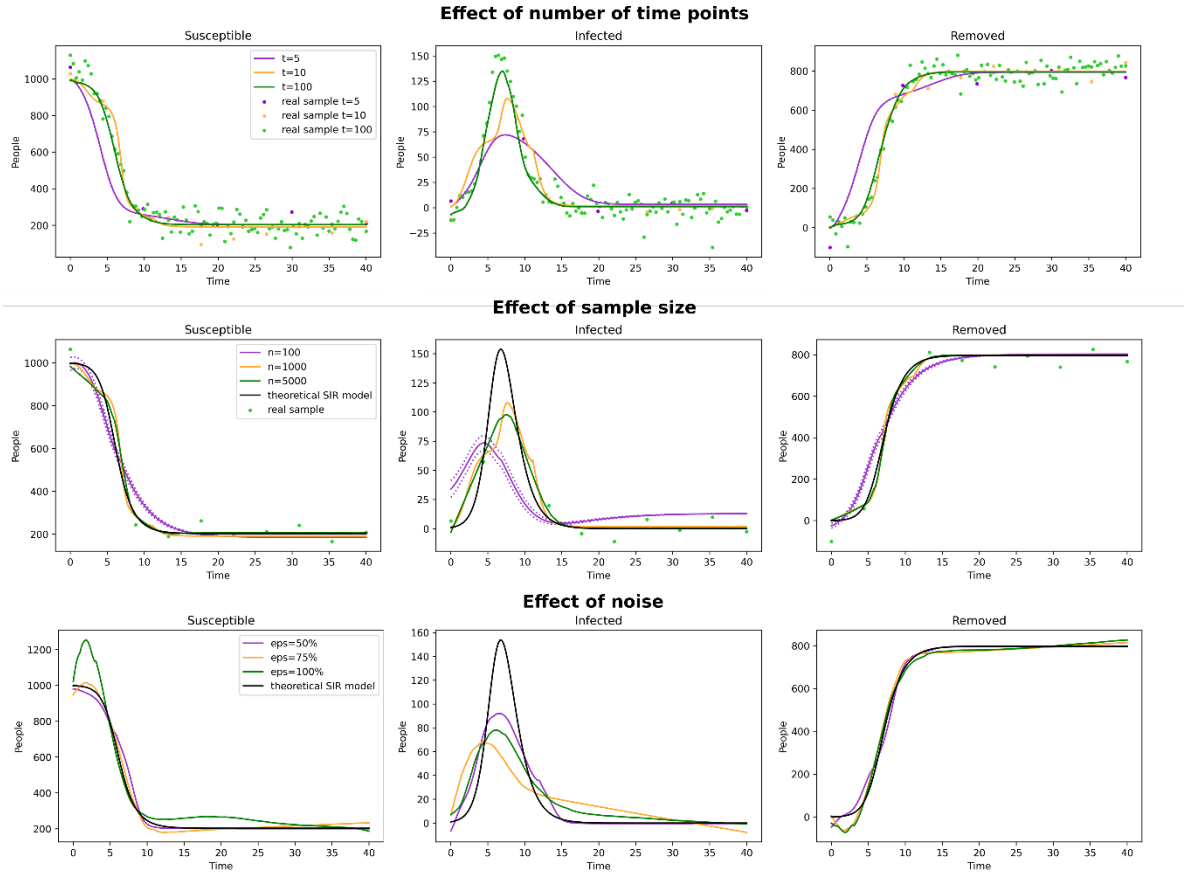

Figure 6: Model benchmarking on simulated data from the SIR-model. Each row represents the evaluation of another parameter (assessment frequency, sample size, noise level).

Discussion

In this work, we presented MultiNODEs, a hybrid AI approach to generate realistic synthetic patient-level datasets. MultiNODEs are designed to consider the characteristics of clinical studies. It extends its predecessor, the Neural ODEs, and enables the application of the latent ODE system framework to multimodal datasets comprising both time-dependent and static variables with missing values not at random. MultiNODEs learn a latent, continuous time trajectory from observed data. This concept fits well to disease progression, where relevant observations (e.g., biomarkers and disease symptoms) only indirectly mimic the true, underlying disease

mechanism. Consequently, MultiNODEs are well suited for an application to heterogeneous datasets holding complex signals as encountered, for example, in biomedical research.

Our evaluations showed that MultiNODEs successfully generated complex, synthetic medical datasets that accurately reproduced the characteristics of their real-world counterparts. By sampling from MultiNODEs' posterior distribution, the model outperformed the state-of-the-art VAMBN approach, most notably with respect to the integrity of the correlation structure. This finding implies that the single data instances generated using MultiNODEs exhibit more realistic properties and that the real data characteristics are not only reproduced on a population-level.

Advancing beyond the training data through time-continuous modeling

Besides the reproduction of marginal distributions and synthesis of realistic data instances, MultiNODEs most prominent strength lies in the generation of smooth longitudinal data. The latent ODE system allows MultiNODEs to learn dynamics which are continuous in time and cover the unobserved time intervals of real-world data. Here, both the prior and posterior sampling approach resulted in realistic trajectories that obey real variables dynamics.

Furthermore, the time-continuous generative capabilities of MultiNODEs create opportunities to fill gaps in the real data through interpolation and go beyond the observation time by extrapolating the longitudinal dynamics. Hence, MultiNODEs could be used to support the design of longitudinal clinical studies, in which the maximum observation period as well as visit frequency are always crucial decisions to make. Here, the question of how patients might develop between two visits or after the last one determines the optimal follow-up time, demonstrating, for example, the most

significant treatment effect. In this context, MultiNODEs provide the opportunity to learn time-continuous disease trajectories from pre-existing studies. Accordingly, its generated synthetic disease trajectories could then be compared to those generated based on other studies, even if the visit intervals employed in the real studies were not identical.

Requirements on training data scale with complexity of data generating process

Our benchmark experiments on the simulated SIR model data demonstrated that MultiNODEs are applicable in a variety of different data settings. While the general trends of a data underlying process could already be learned from a relatively limited dataset, similar to any machine learning task, the accuracy and trustworthiness of the model critically depends on the available data. Especially for complex, nonlinear processes, a sufficiently high observation frequency should be considered. Here, the position of the observation time-points relative to the true underlying process is crucial for MultiNODEs to accurately learn nonlinear dynamics. The sample size of the training data mainly impacts how well MultiNODEs fits the data dynamics and we observed that lower sample sizes can lead to underfitting and rather rigid ODE systems. On the other hand, only severe noise-levels led to a model deviation from the true data-underlying process and, with respect to noise, MultiNODEs proved to be highly robust. In conclusion, MultiNODEs' requirements towards the training data ultimately depend on the complexity of the data underlying process, whereas the learning of more complex processes requires more frequent observations and larger sample size, and more linear systems can already be learned from rather limited datasets.

Limitations

One limitation of MultiNODEs in their current form is that they do not work on time-

dependent categorical variables. Additionally, MultiNODEs are sensitive to several hyperparameters that should be optimized for best performance. The training process and all relevant hyperparameters are explained in the Method section.

Methods

Application case datasets

Both datasets, namely PPMI and NACC, are well known staples in their respective fields and can be accessed after successful data access applications. For PPMI see https://www.ppmi-info.org/. For NACC we refer to https://naccdata.org/.

Neural ODEs (NODEs)

NODEs are a hybrid of neural networks and ODEs [10]. They can be seen as an extension of a ResNet [16], which does not rely on a discrete sequence of hidden layers, but on a continuous hidden dynamical system defined by an ordinary differential equation.

For $0 < t < M$ and $z_0 \in \mathbb{R}^D$ the dynamics of the hidden layer of a NODE are given as

$$\begin{aligned} \frac{dz(t)}{dt} &= f(z(t), t, \theta) \\ z(0) &= z_0 \end{aligned} \qquad (1)$$

where $z(0)$ may be interpreted as the first hidden layer and $z(T)$ as the solution of the initial value problem at timepoint T. Importantly, f is a feed-forward neural network parameterized by θ.

NODEs as generative latent time series models

As demonstrated by the authors in their publication, NODEs can be trained as a continuous time Variational Autoencoder. The basic idea is to learn the initial

conditions z_0 of the dynamical system in Eq. (1) from observed time series data using a variational long-short term memory (LSTM) recurrent encoder [17]. Hence, Eq. (1) now describes the dynamics of a latent system, resulting into a classical state-observation model. Accordingly, a feed-forward neural network decoder is required to project the solution of Eq. (1) back to observed data at defined time points (**Figure S10**).

Overall NODEs are trained at once by maximizing the Evidence Lower Variational Bound (ELBO): Let the training data be $D = \{(x_{t_i}^n, t_i) | n = 1, \dots, N, i = 1, \dots, M\}$, where N is the number of patients and $t_{i_1}, \dots, t_{i_M}$ the observed time points / patient visits. That means $x_{t_i}^n \in \mathbb{R}^p$ is the p-dimensional vector of measurements taken for the n-th patient at visit t_i. The ELBO for NODEs is then given as:

$$ELBO^{NODE} = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M} -D_{KL}\left(q\left(z_{t_0}^n | \{x_{t_i}^n, t_i\}_i\right) \| \, p\left(z_{t_0}^n\right)\right) + E_{q\left(z_{t_0}^n | \{x_{t_i}^n, t_i\}_i\right)}\left(log\left(p\left(x_{t_i}^n | z_{t_i}^n\right)\right)\right)$$

where $p\left(z_{t_0}^n\right) = N(0, I)$, as usual. For details we refer to Chen *et al.* [10]

Multi Modal Neural NODEs (MultiNODEs)

Handling missing values

To handle missing values (potentially not at random) in longitudinal clinical data we build on our previously published work, in which we introduced an imputation layer to implicitly estimate missing values during neural network training [14]: Let $A := \left\{x_{t_i,j}^n \middle| x_{t_i,j}^n is\ not\ missing\right\}$, 1_A be the indicator function on set A with cardinality $|A|$. The imputation layer can be defined as a data transformation $\tilde{x}_{t_i,j}^n = x_{t_i,j}^n \times 1_A\left(x_{t_i,j}^n\right) +$

$b_{t_i,j} \times \left(1 - 1_A\left(x^n_{t_i,j}\right)\right)$, where parameters $b_{t_i,j}$ are trainable weights. That means missing values in a patient's data vector $x^n_{t_i,j}$ are replaced by $b_{t_i,j}$. The accordingly completed data is subsequently mapped through a recurrent neural network encoder to a static, lower dimensional vector, which is interpreted as the initial condition of the latent ODE system (**Figure S11**).

To learn parameters $b_{t_i,j}$ the NODEs' loss function needs to be adapted. More specifically, we use the modified ELBO criterion:

$$ELBO^{NODE}_{IMP} = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M} -D_{KL}\left(q\left(z^n_{t_0}\middle|\left\{x^n_{t_i}, t_i\right\}_i\right) \| \, p\left(z^n_{t_0}\right)\right)$$
$$+\frac{DM}{A}\sum_{n=1}^{N}\sum_{i=1}^{M}\sum_{j=1}^{D} 1_A\left(x^n_{t_i,j}\right)\left(x^n_{t_i,j} - \hat{x}^n_{t_i,j}\right)^2$$

where $\hat{x}^n_{t_i,j}$ denotes the reconstructed data. Note that we only aim for reconstructing the observed data, but not the imputed one. Due to the layer-wise architecture of a neural network $\hat{x}^n_{t_i,j}$ implicitly depends on $b_{t_i,j}$.

In practice we initialize $b_{t_i,j}$ for neural network training as $\frac{1}{N}\sum_{n=1}^{N} x^n_{t_i,j}$.

Dealing with multimodal data

In addition to implicit missing value imputation, the second main idea of MultiNODEs is to complement NODEs with a HI-VAE encoder [13] for static variables (**Figure S11**). A HI-VAE is an extension of a Variational Autoencoder which can implicitly impute missing values via an input drop-out model and handle heterogeneous multimodal data, including categorical data and count data, via an accordingly factorized generative model. In addition, a HI-VAE uses a Gaussian Mixture Model (GMM) as a

prior distribution rather than a single Gaussian. We refer to Nazabal *et al.* [13] for details.

The HI-VAE results into a lower dimensional latent representation z_{stat} of static variables, which can be used to augment the initial conditions z_{t_0} learned from time series data. Consequently, we arrive at the following formulation of the latent ODE system:

$$\begin{aligned} \frac{d}{dt} z^{aug}(t) &= \frac{d}{dt}\begin{bmatrix} z(t) \\ \tilde{z}(t) \end{bmatrix} = f\left(\begin{bmatrix} z(t) \\ \tilde{z}(t) \end{bmatrix}, t, \theta_f^{aug}\right) \\ z_{t_0}^{aug} &= \begin{bmatrix} z_{t_0} \\ z_{stat} \end{bmatrix} \end{aligned} \quad (2)$$

This approach resembles the Augmented Neural ODEs by Dupont *et al.* [18]. The main difference to our work is that in their work no additional features are added by the augmentation step, i.e. $z_{stat} = 0$. According to Dupont et al. the purpose of Augmented Neural ODEs is to smoothen f, whereas we here focus on multimodal data integration.

For training MultiNODEs, we have to jointly consider $ELBO_{IMP}^{NODE}$ as well as $ELBO^{HI-VAE}$. After bringing both quantities on a comparable numerical scale, we use a weighted sum as our final training objective:

$$\begin{aligned} \widetilde{ELBO}_{IMP}^{NODE} &= \frac{ELBO^{HI-VAE}}{ELBO^{HI-VAE} + ELBO_{IMP}^{NODE}} ELBO_{IMP}^{NODE} \\ \widetilde{ELBO}^{HI-VAE} &= \frac{ELBO_{IMP}^{NODE}}{ELBO^{HI-VAE} + ELBO_{IMP}^{NODE}} ELBO^{HI-VAE} \end{aligned}$$

$$ELBO^{MultiNODE} = \widetilde{ELBO}_{IMP}^{NODE} + \beta \widetilde{ELBO}^{HI-VAE}$$

where β is a tunable hyperparameter. Details about hyperparameter optimization are described in the Supplements.

Generating synthetic subjects

We tested two methods to generate synthetic subjects with MultiNODEs:

a) We could independently draw $z_{t_0} \sim N(0, I)$ from the prior for the longitudinal data and $z_{stat} \sim GMM(\pi)$ from the Gaussian Mixture prior (with mixture coefficients π) for the static data used by the HI-VAE. Subsequently, we concatenate $z_0 = [z_{t_0}, z_{stat}]$ into a vector forming the initial conditions for the latent ODE system, solve the ODE system and decode the solution. Unfortunately, the independent drawing from two priors destroys the correlation between static and longitudinal features in the real data. We call this approach “prior sampling”.

b) A second option is to draw $z_{t_0} \sim q\left(z_{t_0}^n | \{x_{t_i}^n, t_i\}_i\right) = N(\lambda(x_{t_i}^n, t_i), \sigma(x_{t_i}^n, t_i))$ for the longitudinal data and $z_{stat} \sim q(z_{stat}^n | x_{stat}^n, \pi) = N\left(\lambda(\hat{x}_{stat}^n, s^n), \sigma(\hat{x}_{stat}^n, s^n)\right)$, $s^n \sim Categorical(\pi(\hat{x}_{stat}^n))$ for the static data. That means we generate a blurred / noisy version of the original *n*-th patient. We call this approach “posterior sampling”. In practice, we doubled the posterior variance during sampling because we found the synthetic data otherwise to lie too close to the real data.

It should be mentioned that synthetic data can not only be generated for observed visits, but also for definable time points in between (interpolation) and after the end of study (extrapolation). This is possible because the latent ODE system is continuous in time.

Funding

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 826421, “TheVirtualBrain-Cloud” and from the Deutsche Forschungsgemeinschaft (DFG) funded project “NFDI4Health” (project number 442326535).

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, including [list the full names of all of the PPMI funding partners found at www.ppmi-info.org/fundingpartners].

The NACC database is funded by NIA/NIH Grant U01 AG016976. NACC data are contributed by the NIA-funded ADCs: P30 AG019610 (PI Eric Reiman, MD), P30 AG013846 (PI Neil Kowall, MD), P30 AG062428-01 (PI James Leverenz, MD) P50 AG008702 (PI Scott Small, MD), P50 AG025688 (PI Allan Levey, MD, PhD), P50 AG047266 (PI Todd Golde, MD, PhD), P30 AG010133 (PI Andrew Saykin, PsyD), P50 AG005146 (PI Marilyn Albert, PhD), P30 AG062421-01 (PI Bradley Hyman, MD, PhD), P30 AG062422-01 (PI Ronald Petersen, MD, PhD), P50 AG005138 (PI Mary Sano, PhD), P30 AG008051 (PI Thomas Wisniewski, MD), P30 AG013854 (PI Robert Vassar, PhD), P30 AG008017 (PI Jeffrey Kaye, MD), P30 AG010161 (PI David Bennett, MD), P50 AG047366 (PI Victor Henderson, MD, MS), P30 AG010129 (PI Charles DeCarli, MD), P50 AG016573 (PI Frank LaFerla, PhD), P30 AG062429-01(PI James Brewer, MD, PhD), P50 AG023501 (PI Bruce Miller, MD), P30 AG035982 (PI Russell Swerdlow, MD), P30 AG028383 (PI Linda Van Eldik, PhD), P30 AG053760 (PI Henry Paulson, MD, PhD), P30 AG010124 (PI John Trojanowski, MD, PhD), P50 AG005133 (PI Oscar Lopez, MD), P50 AG005142 (PI Helena Chui, MD), P30 AG012300 (PI Roger Rosenberg, MD), P30 AG049638 (PI Suzanne Craft, PhD), P50 AG005136 (PI Thomas Grabowski, MD), P30 AG062715-01 (PI Sanjay Asthana, MD, FRCP), P50 AG005681 (PI John Morris, MD), P50 AG047270 (PI Stephen Strittmatter, MD, PhD).

References

1. Fröhlich, H., Balling, R., Beerenwinkel, N., Kohlbacher, O., Kumar, S., Lengauer, T., ... & Zupan, B. (2018). From hype to reality: data science enabling personalized medicine. BMC medicine, 16(1), 1-15.
2. Birkenbihl, C., Salimi, Y., Fröhlich, H., Japanese Alzheimer's Disease Neuroimaging Initiative, & Alzheimer's Disease Neuroimaging Initiative. (2021). Unraveling the heterogeneity in Alzheimer's disease progression across multiple cohorts and the implications for data-driven disease modeling. Alzheimer's & Dementia.
3. Birkenbihl, C., Emon, M. A., Vrooman, H., Westwood, S., Lovestone, S., Hofmann-Apitius, M., & Fröhlich, H. (2020). Differences in cohort study data affect external validation of artificial intelligence models for predictive diagnostics of dementia-lessons for translation into clinical practice. EPMA Journal, 11(3), 367-376.
4. Gootjes-Dreesbach, L., Sood, M., Sahay, A., Hofmann-Apitius, M., & Fröhlich, H. (2020). Variational Autoencoder Modular Bayesian Networks for Simulation of Heterogeneous Clinical Study Data. Frontiers in Big Data, 3, 16.
5. Sood, M., Sahay, A., Karki, R., Emon, M. A., Vrooman, H., Hofmann-Apitius,

M., & Fröhlich, H. (2020). Realistic simulation of virtual multi-scale, multi-modal patient trajectories using Bayesian networks and sparse auto-encoders. Scientific reports, 10(1), 1-14.

6. Chen, Richard J., et al. "Synthetic data in machine learning for medicine and healthcare." Nature Biomedical Engineering (2021): 1-5.
7. Thorlund, K., Dron, L., Park, J. J., & Mills, E. J. (2020). Synthetic and External Controls in Clinical Trials–A Primer for Researchers. Clinical Epidemiology, 12, 457.
8. Lei, Y., Harms, J., Wang, T., Liu, Y., Shu, H. K., Jani, A. B., ... & Yang, X. (2019). MRI-only based synthetic CT generation using dense cycle consistent generative adversarial networks. Medical physics, 46(8), 3565-3581.
9. Yang, G., Yu, S., Dong, H., Slabaugh, G., Dragotti, P. L., Ye, X., ... & Firmin, D. (2017). DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction. IEEE transactions on medical imaging, 37(6), 1310-1321.
10. Chen, R. T., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2018) “Neural Ordinary Differential Equations”. Advances in Neural Information Processing Systems 31, pp. 6571–6583, Curran Associates, Inc.
11. Marek, K., Jennings, D., Lasch, S., Siderowf, A., Tanner, C., Simuni, T., ... & Parkinson Progression Marker Initiative. (2011). The parkinson progression marker initiative (PPMI). Progress in neurobiology, 95(4), 629-635.
12. Besser, L., Kukull, W., Knopman, D. S., Chui, H., Galasko, D., Weintraub, S., ... & Morris, J. C. (2018). Version 3 of the National Alzheimer’s Coordinating Center’s Uniform Data Set. Alzheimer disease and associated disorders.
13. Nazabal, A., Olmos, P. M., Ghahramani, Z., & Valera, I. (2020). Handling incomplete heterogeneous data using vaes. Pattern Recognition, 107, 107501.
14. de Jong, J., Emon, M. A., Wu, P., Karki, R., Sood, M., Godard, P., ... & Fröhlich, H. (2019). Deep learning for clustering of multivariate clinical patient trajectories with missing values. GigaScience, 8(11), giz134.
15. Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.
16. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
17. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
18. Dupont, E., Doucet, A., & Teh, Y. W. (2019). Augmented neural odes. *arXiv preprint arXiv:1904.01681*.
19. Lin, Z., Jain, A., Wang, C., Fanti, G., & Sekar, V. (2020, October). Using GANs for sharing networked time series data: Challenges, initial promise, and open questions. In *Proceedings of the ACM Internet Measurement Conference* (pp. 464-483).
20. Bae, H., Jung, D., Choi, H. S., & Yoon, S. (2019). AnomiGAN: Generative adversarial networks for anonymizing private medical data. In *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2020* (pp. 563-574).
21. Jordon, J., Yoon, J., & Van Der Schaar, M. (2018, September). PATE-GAN: Generating synthetic data with differential privacy guarantees. In *International conference on learning representations*.
22. Beaulieu-Jones, B. K., Wu, Z. S., Williams, C., Lee, R., Bhavnani, S. P., Byrd, J. B., & Greene, C. S. (2019). Privacy-preserving generative deep neural

networks support clinical data sharing. *Circulation: Cardiovascular Quality and Outcomes*, *12*(7), e005122.