HIGH-DIMENSIONAL COUNTERFACTUAL ANALYSIS WITH LATENT NEURAL DIFFERENTIAL EQUATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Counterfactual estimation is important in longitudinal data analysis, particularly in healthcare, where the goal is to infer how an observed patient trajectory would have evolved under an alternative intervention. This problem is especially challenging in real-world settings with longitudinal data, where measurements are typically high-dimensional and exhibit complex temporal structure. Although causal inference has been studied extensively, existing approaches are often tailored to univariate responses and do not naturally accommodate continuous-time dynamics. We address these limitations by proposing CLL-NODE, a Counterfactual analysis of high-dimensional Longitudinal data using Latent Neural ODE model. Our model parameterizes latent continuous-time dynamics as a function of treatment assignment and instance-level auxiliary covariates, enabling treatment and covariate dependent evolution while supporting flexible interpolation and extrapolation in time. To facilitate reliable counterfactual estimation, we introduce an adversarial regularization term that encourages treatment-invariant representations of the initial latent state. We evaluate CLL-NODE on multiple datasets and demonstrate competitive performance relative to strong deep-learning baselines for high-dimensional counterfactual modeling.

1 INTRODUCTION

Longitudinal data are pervasive in healthcare, sociology, seismology and many other domains (Liu, 2015). Such datasets consist of repeated measurements of individual instances (e.g., patients) collected over time. In many applications, causal analysis of longitudinal data, especially counterfactual estimation, is a primary focus. For example, in healthcare one may wish to estimate how patients' measurements would have evolved under an alternative treatment. However, counterfactual inference in longitudinal settings is challenging because the measurements are often high-dimensional, sparsely and irregularly sampled, and exhibit complex temporal structure. These difficulties are characteristic of many real-world biomedical datasets, such as electronic health records.

Although causal inference has been studied extensively over the past decades (Pearl, 2009), most existing methods are designed for low-dimensional, and often univariate, response data. Moreover, comparatively fewer approaches address causal inference in the presence of temporal dynamics. A common strategy for causal inference with temporal data is to represent each time point as a separate node in the causal graph (Loh and Ren, 2023). However, this discretization prevents generalization to time points that are not explicitly included in the graph. Another line of research focuses on continuous-time dynamics to mitigate the limitations of discretization. For example, Hess et al. (2024) propose a Bayesian neural controlled differential equation model; however, their approach is developed for a one-dimensional outcome. On the other hand, recent deep learning approaches based on variational autoencoders (VAEs) (Kingma and Welling, 2014) enable counterfactual modeling with high-dimensional observations (Louizos et al., 2017; Wu et al., 2025), but they do not naturally account for continuous-time dynamics.

In this work, we address these limitations by building on neural ordinary differential equations (ODEs) (Chen et al., 2018) and modelling latent temporal dynamics in continuous-time via an ODE parameterized by a neural network. This formulation supports flexible interpolation and extrapolation across time while remaining scalable to high-dimensional observations. Our goal is to perform retrospective counterfactual estimation by inferring the unobserved baseline latent state from post-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

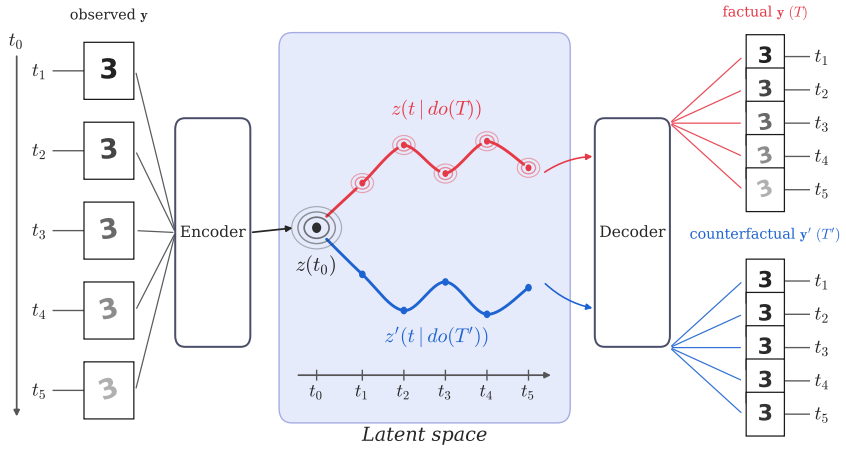


Figure 1: Conceptual overview of our model. Here, $z(t)$ and $z'(t)$ correspond to latent dynamics under factual treatment T and counterfactual treatment T', whereas $\boldsymbol{y}$ and $\boldsymbol{y}'$ refer to the observations decoded from the latent space.

treatment factual outcomes and subsequently predicting the outcomes under an alternative intervention.

Contributions. To summarize, our contributions are that

- *(i)* we propose a novel latent neural ODE model, titled CLL-NODE, that is specifically designed for counterfactual analysis in the context of high-dimensional longitudinal data;
- *(ii)* we develop an adversarial training procedure for latent neural ODE dynamics models that encourages the inferred initial latent state to be invariant to treatment assignment, thereby supporting more accurate identification of causal effects in counterfactual generation;
- *(iii)* we compare the performance of our model on counterfactual estimation task against baseline methods on several high-dimensional longitudinal datasets and report competitive performance.

Our proposed model is conceptualized in Figure 1.

2 Related Work

Causal inference. Causal inference aims to estimate the effect of interventions from observational or experimental data (Pearl, 2009) and has become an important area of modern machine learning (Yao et al., 2021). A wide range of methods have been developed for tasks such as treatment effect estimation, counterfactual prediction, and causal representation learning. These approaches typically rely on assumptions that make causal effects identifiable from observed data, while practical applications must additionally account for challenges such as confounding, selection bias, and limited overlap between treatment groups. Melnychuk et al. (2024) provides a comprehensive overview of the main application areas, methodological approaches, and challenges encountered in practice.

Causal inference with generative models. A popular approach that gained traction over the past decade is to model causal effects using deep latent variable models (DLVMs). Among the earliest is CEVAE (Louizos et al., 2017), which estimates the individual treatment effect (ITE) by leveraging a variational autoencoder (VAE) (Kingma and Welling, 2014) to infer the posterior distribution over unobserved confounders. Subsequent studies have examined limitations of CEVAE (Rissanen and Marttinen, 2021) and introduced alternative deep latent variable model formulations (Harada and Kashima, 2022; Xu et al., 2021). Wu et al. (2025) introduced a training objective tailored to counterfactual inference in high-dimensional settings, designed to separate exogenous variation

from treatment-related variation and thereby facilitate more accurate identification of causal effects in counterfactual generation. However, these aforementioned methods have been designed for time-fixed data, hence are not directly suitable for longitudinal data settings.

Causal inference with temporal data. Several works have also sought to extend causal inference to temporal cases (Loh and Ren, 2023). Many of these approaches represent each time point as a separate node in a causal graph and perform inference on the resulting discrete-time structure (Li et al., 2021; Nauta et al., 2019). Other approaches, such as those proposed by Bica et al. (2020) and Melnychuk et al. (2022), employ RNNs (Elman, 1990) and Transformers (Vaswani et al., 2017), respectively. However, these methods consider sequential treatment assignment at multiple time points and assume the availability of observations preceding each intervention. In contrast, our setting involves a single treatment assigned at baseline, with no pre-treatment observations available. The way to move beyond discrete-time formulations is to model the data using continuous-time dynamical systems, such as ODEs. Hess et al. (2024) propose a Bayesian neural controlled differential equation model for one-dimensional outcomes. The method most conceptually similar to ours is Counterfactual ODE (CF-ODE) (De Brouwer et al., 2022); however, it estimates interventional outcomes using only the pre-treatment sequence and does not scale well to high-dimensional settings. In contrast, our method, based on latent neural ODEs, takes the full factual sequence as input and is specifically designed to handle counterfactual estimation for high-dimensional data.

Neural ODEs. Neural ordinary differential equations (NODEs) were introduced as a continuous-depth alternative to discrete residual networks, where the evolution of a hidden state is defined by an ODE whose vector field is parameterized by a neural network (Chen et al., 2018). This formulation enables adaptive computation via black-box ODE solvers and provides a general framework for continuous-time modeling. Several works have focused on improving the classical neural ODE formulation to better handle long-horizon trajectories. A common strategy is to adopt a multiple-shooting approach (Hemker, 1974; Iakovlev et al., 2023; Giampiccolo et al., 2024; Abrevaya et al., 2023), which partitions the time grid into non-overlapping segments and introduces a separate shooting variable for each segment to stabilize optimization, enable parallelization and improve long-range generalization.

3 METHODS

3.1 PROBLEM SETUP

We are given $N = \sum_{p=1}^{P} n^{(p)}$ observations, where P denotes the number of unique instances, e.g., patients, and $n^{(p)}$ is the number of observations of instance p. The longitudinal, high-dimensional *response variables* of instance p collected at time points $\boldsymbol{t}^{(p)} = [t_1^{(p)}, \dots, t_{n^{(p)}}^{(p)}]$ are denoted as $\boldsymbol{y}^{(p)} = [y_1^{(p)}, \dots, y_{n^{(p)}}^{(p)}] \in \mathbb{R}^{D \times n^{(p)}}$, where each sample $y_i^{(p)} \in \mathcal{Y} = \mathbb{R}^D$. The *auxiliary covariate* information of instance p is a Q-dimensional vector $x^{(p)} \in \mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_Q$. Each patient p is assigned a treatment $T^{(p)}$, which we assume is binary, $T^{(p)} \in \{0, 1\}$. In our setting, treatment is received at the initial time point $t_0^{(p)}$, while the first measurement is recorded at $t_1^{(p)} > t_0^{(p)}$. We assume the observation schedule $\boldsymbol{t}^{(p)}$ is independent of assigned treatment $T^{(p)}$. In the following, we will omit p for notational convenience and formulate methods for a single instance.

Let $y_*(t \mid \mathrm{do}(T = a))$ for $a \in \{0, 1\}$ and do-operator (do) denote the true (noise-free) continuous-time potential trajectory under treatment T, from which we observe noisy trajectory $\boldsymbol{y}$, where $y_i = y_*(t_i \mid \mathrm{do}(T = a)) + \epsilon_{y,i}$ for each time point t_i. Let the counterfactual trajectory be $\boldsymbol{y}'$, where observation for each time point is $y'_i = y_*(t_i \mid \mathrm{do}(T = 1 - a)) + \epsilon'_{y,i}$. The goal is to estimate counterfactual predictive distribution $p(\boldsymbol{y}' | \boldsymbol{y}, x, T, T', \boldsymbol{t})$ for $T' = 1 - T$.

3.2 LATENT NEURAL ODES

The core idea behind latent neural ODE (L-NODE) models (Chen et al., 2018) is to learn the system dynamics in a latent space $\mathcal{Z} = \mathbb{R}^L$ and then generatively decode latent states to the observation space, so that each observation $y_i := y(t_i)$ is associated with a corresponding latent state variable

$z_i := z(t_i) \in \mathcal{Z}$. Formally, an L-NODE model is defined as:

$$\begin{aligned} z_0 &\sim p(z(t_0)) \\ z_i &= \text{ODEsolve}(z_0, t_0, t_i, f_{\theta_{\text{dyn}}}), & i = 1, ..., n, \\ y_i | z_i &\sim p(y_i | g_{\theta_{\text{dec}}}(z_i)), & i = 1, ..., n. \end{aligned}$$

Variable z_0 is the initial state at time t_0. Dynamics function $f_{\theta_{\text{dyn}}}$ is the time derivative of $z(t)$, and $\text{ODEsolve}(z_0, t_0, t_i, f_{\theta_{\text{dyn}}})$ is defined as the solution of the following initial value problem at time t_i:

$$\frac{dz(t)}{dt} = f_{\theta_{\text{dyn}}}(z(t)), \quad z(t_0) = z_0, \quad t \in [t_0, t_i].$$

Likelihood model is application dependent. We assume additive Gaussian model where decoder $g_{\theta_{\text{dec}}}$ maps the latent state z_i to the noise-free trajectory $y_{*,i} = g_{\theta_{\text{dec}}}(z_i)$, which parametrizes observational likelihood model as $p(y_i | g_{\theta_{\text{dec}}}(z_i)) = \mathcal{N}(y_i | g_{\theta_{\text{dec}}}(z_i), \sigma_y^2 I)$. Dynamics and decoder functions are neural networks with corresponding parameters $\theta = (\theta_{\text{dyn}}, \theta_{\text{dec}})$.

3.3 Counterfactual analysis of high-dimensional longitudinal data

Our dynamical causal model builds on L-NODE by modifying the latent dynamics function $f_{\theta_{\text{dyn}}}$ to incorporate treatment T and covariate information x. Concretely, we define the latent dynamics as

$$\frac{dz(t)}{dt} = f_{\theta_{\text{dyn}}}(z(t), T, x).$$

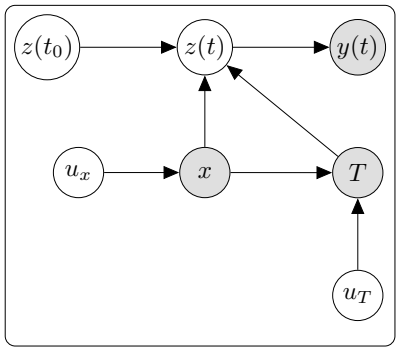


Figure 2: Structural causal model (SCM) of CLL-NODE. Shaded nodes are observed quantities.

Next, we describe the structural dependencies of our model. We assume u_x and u_T to be exogenous noise variables in our structural causal model (SCM), capturing uncertainty in the covariates x and the treatment assignment T, respectively, such that $x = f_x(u_x)$, $T = f_T(x, u_T)$. In addition, ϵ_y represents observation noise in the high-dimensional outcomes, i.e., we assume $y_i = y_{*,i} + \epsilon_{y,i}$. A graphical overview of the proposed SCM is provided in Figure 2. We adopt the standard causal assumptions of positivity, ignorability, and consistency. Positivity requires both treatment assignments to have non-zero probability over the relevant support of the covariates. Ignorability requires independency of potential outcome of noise-free trajectories under treatment intervention for given covariates. For an instance (or unit) receiving treatment $T = a$, consistency requires that the observed noise-free trajectory coincides with the corresponding potential trajectory, i.e., $y_*(t) = y_*(t | \text{do}(T = a))$. Further details on these assumptions are provided in Appendix A.1.

3.3.1 Sparse multiple shooting

Training dynamics models on long trajectories is challenging as complexity of the loss function grows with the length of the observed trajectory (Ribeiro et al., 2019; Voss et al., 2004). To mitigate this challenge and enable parallel training, we adopt sparse multiple shooting for CLL-NODE.

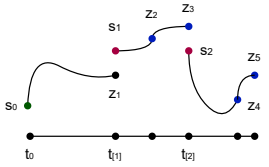


Figure 3: Sparse multiple shooting ($B = 2$): $t_{[1]} = t_1$, $t_{[2]} = t_3$, $I_1 = \{2, 3\}$, $I_2 = \{4, 5\}$; s_0 is the initial latent state.

The idea behind sparse multiple shooting, proposed by (Iakovlev et al., 2023), is to partition the time grid into B non-overlapping blocks and assign a single shooting variable s_b at time $t_{[b]}$ to each block, collectively denoted as $s_{1:B}$. For each block b, we define an index set I_b containing the time-point indices included in that block, such that $\bigcup_{b=1}^{B} I_b = \{2, \ldots, n\}$. In our case, the partition starts from t_1. In CLL-NODE model, we allocate an additional shooting variable, or initial state, s_0 to the initial time point t_0. The multiple shooting formulation enables reliable estimation of long trajectories and supports parallel optimization during training. It generalizes the classical multiple shooting method (Hemker, 1974), where a distinct shooting

variable is introduced at every time point. Figure 3 illustrates the multiple-shooting setup. The latent state is then computed as:

$$z_i = \text{ODEsolve}(s_b, t_{[b]}, t_i, T, x, \theta_{\text{dyn}}), \quad i \in I_b.$$

3.3.2 Generative CLL-NODE model

We define independent priors for neural network parameters $\theta_{\text{dyn}} \sim p(\theta_{\text{dyn}})$, $\theta_{\text{dec}} \sim p(\theta_{\text{dec}})$ and for the initial latent state $z(t_0) = s_0 \sim p(s_0)$. Following (Iakovlev et al., 2023), the joint prior distribution of initial state s_0, all shooting variables $s_{1:B}$ and neural network parameters is defined as:

$$p(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|T, x) = p(s_{0:B}|\theta_{\text{dyn}}, T, x)p(\theta_{\text{dyn}})p(\theta_{\text{dec}}),$$

where

$$\begin{aligned} p(s_{0:B}|\theta_{\text{dyn}}, T, x) &= p(s_0) \prod_{b=1}^{B} p(s_b|s_{b-1}, \theta_{\text{dyn}}, T, x) \\ &= p(s_0) \underbrace{\prod_{b=1}^{B} \mathcal{N}(s_b|\text{ODEsolve}(s_{b-1}, t_{[b-1]}, t_{[b]}, T, x, \theta_{\text{dyn}}), \sigma_c^2 I)}_{\text{continuity prior}}. \end{aligned}$$

The *continuity prior* (Iakovlev et al., 2023) defines a joint prior distribution for the shooting variables recursively from the initial state s_0 to the shooting variable s_B of the last shooting block. The continuity prior is conditional on $\theta_{\text{dyn}}, T, x$ and it effectively uses the continuous-time latent dynamics θ_{dyn} to map the shooting variable of the previous block, s_{b-1}, to the mean of the normal distribution that defines the distribution of the next shooting variable, s_b.

An arbitrary latent state z_i is obtained by first identifying the shooting block that contains t_i and then integrating the latent dynamics forward from the block's shooting variable to time t_i. The corresponding observation y_i is subsequently generated by decoding z_i through $g_{\theta_{\text{dec}}}$:

$$\begin{aligned} z_i &= \text{ODEsolve}(s_b, t_{[b]}, t_i, T, x, \theta_{\text{dyn}}) \\ y_i|z_i &\sim p(y_i|g_{\theta_{\text{dec}}}(z_i)). \end{aligned}$$

3.3.3 Variational inference

In order to approximate intractable posterior distribution $p(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, T, x)$, we introduce factorizable variational distribution parameterized by $\psi = (\psi_{\text{enc}}, \psi_{\text{dyn}}, \psi_{\text{dec}})$

$$q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) = \Big[\prod_{b=0}^{B} q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y}, \boldsymbol{t}) \Big] \cdot q_{\psi_{\text{dyn}}}(\theta_{\text{dyn}}) q_{\psi_{\text{dec}}}(\theta_{\text{dec}}),$$

where ψ_{dyn} and ψ_{dec} refer to the global variational parameters of θ_{dyn} and θ_{dec}, respectively, and ψ_{enc} refers to the parameters of the encoder neural network for amortized variational inference of the shooting variables. For $q_{\psi_{\text{enc}}}$ we employ a transformer-based encoder with time-aware attention and continuous relative positional encodings from (Iakovlev et al., 2023). These modifications allow the encoder to effectively operate on input sequences with a temporal component.

3.3.4 Evidence lower bound objective

Below we give an overview of the training objective, while the full details are provided in Appendix B. The resulting ELBO of our model consists of four components.

$$\mathcal{L}(\psi|\boldsymbol{y}, x, T, \boldsymbol{t}) = \mathcal{L}_1 - \mathcal{L}_2 - \mathcal{L}_3 - \mathcal{L}_4. \tag{1}$$

The first term corresponds to the reconstruction term that measures how well our model fits the observed factual trajectory $\boldsymbol{y}$ in terms of the expected log likelihood

$$\mathcal{L}_1 = \mathbb{E}_{q_\psi} \left[\ln p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x) \right] + \sum_{b=1}^{B} \sum_{i \in \mathcal{I}_b} \mathbb{E}_{q_\psi} \left[\ln p(y_i|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x) \right],$$

where the expectations are computed w.r.t. $q_\psi(s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}|\boldsymbol{y}, \boldsymbol{t})$ and $q_\psi(s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t})$, respectively. The remaining three components correspond to the regularization terms that measure the Kullback-Leibler divergence of the variational approximations from their corresponding priors

$$\mathcal{L}_2 = \text{KL}\left(q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y}, \boldsymbol{t})\|p(s_0)\right) + \sum_{b=1}^{B} \mathbb{E}_{q_\psi(s_{b-1}, \theta_{\text{dyn}}|\boldsymbol{y}, \boldsymbol{t})}\left[\text{KL}\left(q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y}, \boldsymbol{t})\|p(s_b|s_{b-1}, \theta_{\text{dyn}}, T, x)\right)\right],$$

$$\mathcal{L}_3 = \text{KL}(q_{\psi_{\text{dyn}}}(\theta_{\text{dyn}})\|p(\theta_{\text{dyn}})), \quad \mathcal{L}_4 = \text{KL}(q_{\psi_{\text{dec}}}(\theta_{\text{dec}})\|p(\theta_{\text{dec}})).$$

The expectations are approximated using Monte Carlo samples drawn from the corresponding variational distributions, while the KL divergence terms admit closed-form solutions.

3.4 Counterfactual computation

Having specified the temporal dynamics of our model via latent neural ODEs, we can now state again the main objective of this work: high-dimensional, longitudinal counterfactual outcome modeling. Specifically, we aim to characterize the counterfactual predictive distribution

$$p(\boldsymbol{y}'|\boldsymbol{y}, x, T, T', \boldsymbol{t}),$$

where T and $\boldsymbol{y}$ denote the administered treatment and its corresponding observed outcome at time points $\boldsymbol{t}$, respectively, and $\boldsymbol{y}'$ represents the hypothetical outcome at the same time points that would have been observed under an alternative treatment T'. Under our model, this distribution is obtained by marginalizing over latent states and model parameters:

$$p(\boldsymbol{y}'|\boldsymbol{y}, x, T, T', \boldsymbol{t}) = \int p(\boldsymbol{y}'|s_0, x, T', \boldsymbol{t}, \theta_{\text{dyn}}, \theta_{\text{dec}}) q_\psi(s_0, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) ds_0 d\theta_{\text{dyn}} d\theta_{\text{dec}},$$

where the integral is approximated using the Monte Carlo estimate. Note that the counterfactual estimate involves only the initial state s_0, not the multiple shooting states $s_{1:B}$ that are used only as nuisance variables to improve training a continuous-time dynamics model from a long sequence of observations.

3.5 Adversarial discriminator

Since counterfactual outcomes are not observed during training, naively optimizing the ELBO can result in variational approximations of the latent initial states that are completely separated between different treatment groups, thereby effectively ignoring the role of treatment T from the generative model component. In other words, the encoder may already encode treatment-related information into the latent initial state, so that

$$\|\mathbb{E}[s_0 \mid T = 0] - \mathbb{E}[s_0 \mid T = 1]\| \gg 0.$$

As a result, the conditional latent distributions $\mathbb{P}[s_0 \mid T = 0]$ and $\mathbb{P}[s_0 \mid T = 1]$ form well-separated clusters, allowing accurate reconstruction even when the neural ODE dynamics effectively ignore T. Our goal is to enforce that the latent representation is invariant to treatment and instead captures individual-specific characteristics. Related concerns have been discussed in (Shen et al., 2022; Wu et al., 2025; Bica et al., 2020); in particular, Wu et al. (2025) address this issue by introducing an additional divergence-based regularizer and the approach described below is similar to that proposed by Bica et al. (2020).

To this end, we introduce a discriminator

$$D_{\phi_{\text{disc}}} : \mathcal{Z} \to (0, 1), \tag{2}$$

which predicts whether the initial state s_0 was inferred from an instance with $T = 1$ or $T = 0$. At each iteration, we update the discriminator parameters ϕ_{disc} by minimizing expectation of the standard cross-entropy loss, keeping the model parameters fixed:

$$\mathcal{L}_{\text{disc}}(D_{\phi_{\text{disc}}}|\boldsymbol{y}, T, \boldsymbol{t}, \psi_{\text{enc}}) = -\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y}, \boldsymbol{t})}\big[\log D_{\phi_{\text{disc}}}(s_0) \cdot T + \log(1 - D_{\phi_{\text{disc}}}(s_0)) \cdot (1 - T)\big]. \tag{3}$$

Next, when updating the variational model parameters ψ, we employ an adversarial objective (Goodfellow et al., 2014) to encourage treatment-invariant latent representations by making the encoder fool the discriminator:

$$\mathcal{L}_{\text{adv}}(\psi_{\text{enc}}|\boldsymbol{y}, T, \boldsymbol{t}, D_{\phi_{\text{disc}}}) = -\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y}, \boldsymbol{t})}\big[\log D_{\phi_{\text{disc}}}(s_0) \cdot T + \log(1 - D_{\phi_{\text{disc}}}(s_0)) \cdot (1 - T)\big]. \tag{4}$$

This adversarial mechanism encourages s_0 to encode general instance-level structure while suppressing treatment-specific information. The adversarial and discriminator losses have the same cross-entropy expression but are optimized in opposite directions: the discriminator minimizes this loss with respect to ϕ_{disc}, whereas the encoder maximizes it with respect to ψ_{enc}.

We then augment the ELBO $\mathcal{L}$ in Eq. (1) with $\mathcal{L}_{\text{adv}}$ to form the final objective to be maximized:

$$\mathcal{L}^*(\psi|\boldsymbol{y}, x, T, \boldsymbol{t}, \phi_{\text{disc}}) = \mathcal{L}_1 - \mathcal{L}_2 - \mathcal{L}_3 - \mathcal{L}_4 + \lambda_{\text{adv}} \cdot \mathcal{L}_{\text{adv}} \tag{5}$$

Training proceeds by alternating updates: (i) minimize $\mathcal{L}_{\text{disc}}$ w.r.t. discriminator parameters ϕ_{disc}, and (ii) maximize $\mathcal{L}^*$ w.r.t. variational parameters ψ. Hyperparameter λ_{adv} is application dependent and is necessary to balance the loss function. To ensure that the discriminator promotes a treatment-invariant initial latent state s_0, we establish a connection between the proposed adversarial loss and the mutual information between s_0 and the treatment T.

Lemma 1 (Optimal discriminator and treatment information)**.** *Assume a universal discriminator D that implements all measurable functions $D : S_0 \longrightarrow (0, 1)$ where $S_0 \subseteq \mathcal{Z}$. Define population discriminator value function, which corresponds to the negative expected value of instance-wise discriminator loss $\mathcal{L}_{\text{disc}}$ in Eq. (3):*

$$V(D|\psi_{\text{enc}}) = -\mathbb{E}_{p(\boldsymbol{y},T,\boldsymbol{t})}[\mathcal{L}_{\text{disc}}(D|\boldsymbol{y}, T, \boldsymbol{t}, \psi_{\text{enc}})]. \tag{6}$$

Let $\pi_a = \mathbb{P}(T = a)$ for $a \in \{0, 1\}$, and assume that $\pi_0, \pi_1 > 0$. Then, for any fixed encoder with parameters ψ_{enc}

$$\sup_D V(D|\psi_{\text{enc}}) = -H(T) + I_{\psi_{\text{enc}}}(T; s_0), \tag{7}$$

where $H(T) = -\pi_1 \log \pi_1 - \pi_0 \log \pi_0$ and $I_{\psi_{\text{enc}}}(T; s_0)$ is the mutual information between initial state and treatment. Please refer to Appendix C for the proof.

Assuming an expressive discriminator D optimized to Bayes optimality, maximizing the adversarial objective in Eq. (4) is equivalent to minimizing the mutual information $I_{\psi_{\text{enc}}}(T; s_0)$. Thus, although neural network discriminator $D_{\phi_{\text{disc}}}$ with finite capacity and optimization of the final objective in Eq. (5) do not guarantee zero mutual information in practice, the adversarial objective discourages treatment-specific information in s_0 and promotes a treatment-invariant initial representation.

4 Experiments

Evaluating counterfactual inference methods on real-world data is inherently challenging because the relevant counterfactual outcomes are never observed. As a result, empirical validation typically relies either on completely or semi synthetic benchmarks—where ground truth effects are available by construction—or on proxy metrics when working with observational datasets. We compare our CLL-NODE method against high-dimensional deep learning baselines: CEVAE (Louizos et al., 2017), a general-purpose framework for causal inference with latent variables, VCI (Wu et al., 2025), which is specifically tailored to high-dimensional counterfactual modeling, and CF-NODE (De Brouwer et al., 2022), a Neural ODE model for counterfactual predictions. More information regarding the implementation of the corresponding baselines can be found in Appendix E. In addition, we include a variant of our method without the adversarial discriminator, referred to as Causal-LNODE, to demonstrate the importance of this component. For each method, we report mean-squared error (MSE), presenting the mean and standard deviation over five independent runs. In each experiment, the best (lowest) mean MSE is highlighted in bold in the corresponding table. During training, we used the Adam optimiser (Kingma and Ba, 2015) as implemented in Pytorch (Paszke et al., 2019), with hyperparameters selected based on cross-validation.

4.1 Causal MNIST

To simulate a high-dimensional longitudinal dataset for evaluating counterfactual modeling, we use a modified version of the MNIST dataset (LeCun and Cortes, 2010), commonly referred to as Rotated MNIST. In our setup, we select 5000 images that represent the digit '3' and impose a shared temporal effect by rotating the images across 20 angles spaced over $[0, 2\pi]$. In the regularly sampled version of the dataset, the angles are uniformly spaced, while in the irregularly sampled one they are unevenly spaced. Each image has size 32-by-32, i.e., our observations are 1024-dimensional. In addition,

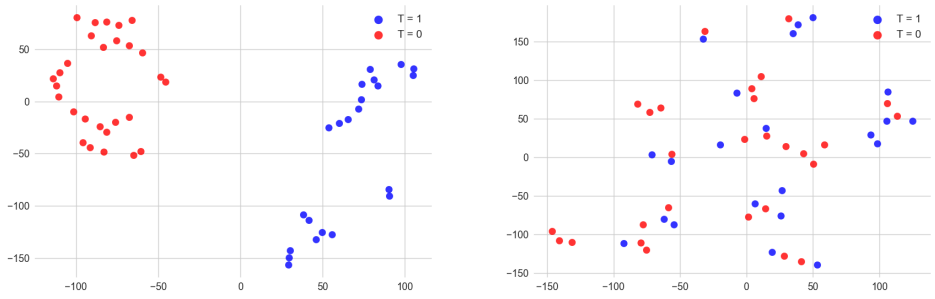


Figure 4: t-SNE projections of the inferred initial latent states s_0 for 50 instances produced by CAUSAL-LNODE without discriminative regularization (left) and by CLL-NODE with discriminative regularization (right).

each patient is randomly assigned to a treatment group, $T \in \{0, 1\}$. For treated patients ($T = 1$), the image intensity gradually decreases over time, i.e., the digit fades as t increases. This setup allows us to directly assess both temporal dynamics modeling and counterfactual prediction in a high-dimensional setting.

Table 1 shows that CLL-NODE achieves the best performance on both regularly and irregularly sampled data, demonstrating its effectiveness across different sampling settings. Although CF-NODE (De Brouwer et al., 2022) is specifically designed for longitudinal data, it performs poorly, likely due to its limited scalability to high-dimensional outcomes. Our ablation method, CAUSAL-LNODE, also performs poorly, as it can reconstruct observed trajectories but cannot reliably infer counterfactual outcomes. As noted in the previous section, this is due to treatment-related information being encoded in the latent initial state. This behavior is demonstrated by the t-SNE projection (van der Maaten and Hinton, 2008) of the initial latent state s_0 in Figure 4, where the projection produced by CAUSAL-LNODE reveals a clear separation between the latent distributions associated with different treatment groups, indicating that s_0 captures treatment-related information. By contrast, CLL-NODE alleviates this problem by encouraging the latent representation to retain individual-specific information while suppressing treatment-specific structure. Figure 5 in Appendix G provides a qualitative illustration of our model's behavior: it captures the underlying temporal evolution and produces plausible counterfactual trajectories as digits start to gradually fade. Also, in Appendix F, we provide an ablation study assessing the importance of the shooting variables.

4.2 CAUSAL MNIST GROUPS

The previous setup in Section 4.1 primarily serves as a feasibility check that the model can capture temporal dynamics and perform basic counterfactual prediction. To evaluate generalization to more complex causal mechanisms, we introduce a modified version of Causal MNIST. In this setup, we impose the following causal structure:

- Each subject is randomly assigned to one of five latent groups $g \in \{1, \ldots, 5\}$. Although g is unobserved, we provide an observed proxy variable $x \in \mathbb{R}^2$, sampled from a two-dimensional Gaussian distribution $\mathcal{N}(\mu_g, \sigma^2 I)$ whose mean depends on the group.
- Treatment assignment depends on group membership: the probability of receiving treatment decreases across groups as follows: $p(T = 1 \mid g = 1) = 0.9$, $p(T = 1 \mid g = 2) = 0.7$, and so on.
- We further introduce a binary *recovery* variable R, which may occur even in the absence of treatment. Its probability is also group-dependent as follows: $p(R = 1 \mid g = 1) = 0.9$, $p(R = 1 \mid g = 2) = 0.7$, and so on.
- For subjects with either $T = 1$ or $R = 1$, the image intensity decreases over time at a reduced rate compared with the previous setup in Section 4.1. If both $T = 1$ and $R = 1$, the decay rate is the same as in the previous setup.

This setting follows the general causal graphical model depicted in Figure 2: the auxiliary covariates x influence both treatment assignment T and the latent state z, and the observed trajectory is then

Table 1: Counterfactual prediction MSEs. Lower values are better.

Method	Causal MNIST		Causal MNIST Groups	
	Regular	Irregular	Regular	Irregular
CLL-NODE	$\mathbf{0.009}_{\pm 0.002}$	$\mathbf{0.013}_{\pm 0.002}$	$\mathbf{0.010}_{\pm 0.001}$	$\mathbf{0.011}_{\pm 0.001}$
CAUSAL-LNODE	$0.066_{\pm 0.001}$	$0.065_{\pm 0.001}$	$0.020_{\pm 0.001}$	$0.022_{\pm 0.001}$
VCI	$0.014_{\pm 0.001}$	$0.026_{\pm 0.001}$	$0.020_{\pm 0.001}$	$0.021_{\pm 0.001}$
CEVAE	$0.045_{\pm 0.001}$	$0.045_{\pm 0.001}$	$0.024_{\pm 0.001}$	$0.030_{\pm 0.001}$
CF-NODE	$0.065_{\pm 0.003}$	$0.068_{\pm 0.002}$	$0.067_{\pm 0.002}$	$0.071_{\pm 0.005}$

jointly determined by x and T through their effect on z, decoded to y. Table 1 shows that our method substantially outperforms the baselines, which demonstrates its robustness to more challenging settings.

4.3 PHYSIONET

We evaluated our model on healthcare data from the 2012 Physionet Challenge (Silva et al., 2012). The dataset contains 12,000 patients monitored on the intensive care unit (ICU) for 48 hours, with irregularly sampled timestamps, from which we selected only those with ICU type "Medical ICU", or "Surgical ICU". We modelled measurements of 37 different attributes, such as glucose level, heart rate, and body temperature. As with any real-world dataset, counterfactual outcomes are not observed in Physionet, and therefore the counterfactual prediction performance of our model cannot be evaluated directly. To address this limitation, we consider two variants of the Physionet.

In the first variant, we construct a semi-synthetic dataset based on the original observations. Here, we associate $T = 0$ with original trajectories, whereas trajectories for $T = 1$ are generated synthetically from a dynamical system. This formulation provides both factual and counterfactual outcomes, enabling direct evaluation of the model's counterfactual prediction performance. As auxiliary covariates, we use *gender*, *age*, *height*, and *ICU type*. More information about semi-synthetic generation can be found in Appendix H.

Table 2: Performance on the Physionet datasets. We report counterfactual prediction MSE on the semi-synthetic variant and $|\Delta_{\text{accuracy}}|$ on the original Physionet dataset.

Method	MSE	$\lvert\Delta_{\text{accuracy}}\rvert$
CLL-NODE	$\mathbf{1.143}_{\pm 0.02}$	$\mathbf{0.154}_{\pm 0.05}$
CAUSAL-LNODE	$1.411_{\pm 0.02}$	$0.29_{\pm 0.01}$
VCI	$2.852_{\pm 0.05}$	$0.3_{\pm 0.1}$
CEVAE	$1.210_{\pm 0.03}$	$0.224_{\pm 0.06}$

In the second variant, ICU type is used as the treatment variable, while *gender*, *age*, and *height* are retained as auxiliary covariates. Since counterfactual outcomes are unobserved, we cannot compute prediction error directly and therefore rely on a proxy evaluation task. Concretely, we train an *oracle* classifier based on the Transformer architecture (Vaswani et al., 2017) to predict the assigned treatment from laboratory measurements. We then apply this oracle to both the factual trajectories and the model-generated counterfactual trajectories, and report the difference in classification error between the two. We interpret smaller difference as indicating better counterfactual quality. Table 2 demonstrates that in both cases our model achieves the best results.

5 CONCLUSION

We introduced CLL-NODE, a continuous-time causal model designed for counterfactual analysis in high-dimensional longitudinal data. By parameterizing latent Neural ODE dynamics as a function of treatment assignment and auxiliary covariates, our approach captures treatment-dependent temporal evolution while naturally supporting interpolation and extrapolation across irregular time grids. To improve identifiability of causal effects, we further incorporated an adversarial regularization term that encourages treatment-invariant representations of the initial latent state. Across multiple experimental benchmarks, CLL-NODE achieves the best performance when comparing with strong deep-learning baselines for counterfactual modeling. Future work could extend the proposed framework to more general settings involving multi-valued or continuous treatments.

AI USE STATEMENT

In this work, we used generative AI tools to generate semi-synthetic data, provide feedback on research methodology or experiments, implement baseline methods, assist with translation, clean and reformat dataset, support qualitative and thematic data analysis, interpret results, improve on text.

We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses.

We have reviewed all AI-assisted work. For example, LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide a link to the Anonymous Github repo for the sake of reproducibility: https://anonymous.4open.science/r/Causal-NODE-7CF2

REFERENCES

Germán Abrevaya, Mahta Ramezanian-Panahi, Jean-Christophe Gagnon-Audet, Pablo Polosecki, Irina Rish, Silvina Ponce Dawson, Guillermo Cecchi, and Guillaume Dumas. Effective latent differential equation models via attention and multiple shooting, 2023.

Ioana Bica, Ahmed M Alaa, James Jordon, and Mihaela van der Schaar. Estimating counterfactual treatment outcomes over time through adversarially balanced representations. In *Proceedings of the Eighth International Conference on Learning Representations (ICLR)*, 2020.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, 2018.

Thomas M Cover, Joy A Thomas, and John Kieffer. *Elements of information theory*. wiley New York, 1991.

Edward De Brouwer, Javier Gonzalez, and Stephanie Hyland. Predicting the impact of treatments over time with uncertainty aware neural differential equations. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, 2022.

Jeffrey L. Elman. Finding structure in time. *Cognitive science*, 1990.

Gerald B Folland. *Real analysis: modern techniques and their applications*. John Wiley & Sons, 1999.

Stefano Giampiccolo, Federico Reali, Anna Fochesato, Giovanni Iacca, and Luca Marchetti. Robust parameter estimation and identifiability analysis with hybrid neural ordinary differential equations in computational biology. *npj Systems Biology and Applications*, 2024.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, 2014.

Shonosuke Harada and Hisashi Kashima. Infocevae: treatment effect estimation with hidden confounding variables matching. *Mach. Learn.*, 2022.

Piet Hemker. Nonlinear parameter estimation in initial value problems. 1974.

Kevin Hess, Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Bayesian neural controlled differential equations for treatment effect estimation. In *12th International Conference on Learning Representations*, 2024.

V. Iakovlev, Çağatay Yıldız, Markus Heinonen, and H. Lähdesmäki. Latent neural odes with sparse bayesian multiple shooting. In *The Eleventh International Conference on Learning Representations*, 2023.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, 2015.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations*, 2014.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

Rui Li, Yebin Xia, Yeonjae Chang, and Mihaela van der Schaar. G-net: a recurrent network approach to g-computation for counterfactual outcome prediction under dynamic treatment regimes. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2021.

X Liu. *Methods and applications of longitudinal data analysis*. Elsevier, New York, 2015.

W. W. Loh and D. Ren. A tutorial on causal inference in longitudinal data with time-varying confounding using g-estimation. *Advances in Methods and Practices in Psychological Science*, 2023.

Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, 2017.

Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Causal transformer for estimating counterfactual outcomes. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

Valentyn Melnychuk, Jonas Schweisthal, Konstantin Hess, Alicia Curth, Stefan Bauer, Niki Kilbertus, Isaac S Kohane, Mihaela van der Schaar, Stefan Feuerriegel, and Dennis Frauen. Causal machine learning for predicting treatment outcomes. *Nature Medicine*, 2024.

Meike Nauta, Doina Bucur, and Christin Seifert. Causal discovery with attention-based convolutional neural networks. *Machine Learning and Knowledge Extraction*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 2019.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2009.

Antonio H. Ribeiro, Koen Tiels, Jack Umenberger, Thomas Schön, and Luis Aguirre. On the smoothness of nonlinear system identification, 2019.

Severi Rissanen and Pekka Marttinen. A critical look at the consistency of causal estimation with deep latent variable models. In *Advances in Neural Information Processing Systems*, 2021.

Xinwei Shen, Furui Liu, Hanze Dong, Qing Lian, Zhitang Chen, and Tong Zhang. Weakly supervised disentangled generative causal representation learning. *J. Mach. Learn. Res.*, 2022.

I. Silva, G. Moody, D J Scott, L A Celi, and R G Mark. Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012. In *2012 Computing in Cardiology.IEEE*, 2012.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 2008.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, 2017.

Henning U. Voss, Jens Timmer, and Jürgen Kurths. Nonlinear dynamical system identification from uncertain and indirect measurements. *International Journal of Bifurcation and Chaos*, 2004.

Yulun Wu, Louis McConnell, and Claudia Iriondo. Counterfactual generative modeling with variational causal inference. In *The Thirteenth International Conference on Learning Representations*, 2025.

Liyuan Xu, Heishiro Kanagawa, and Arthur Gretton. Deep proxy causal learning and its application to confounded bandit policy evaluation. In *Advances in Neural Information Processing Systems*, 2021.

Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. A survey on causal inference. *ACM Trans. Knowl. Discov. Data*, 2021.

Cağatay Yıldız, Markus Heinonen, and Harri Lähdesmäki. Ode2vae: Deep generative second order odes with bayesian neural networks. In *Advances in Neural Information Processing Systems 32*, 2019.

A CAUSAL ASSUMPTIONS AND REMARK ON CONSISTENCY

A.1 ASSUMPTIONS

To state the causal and structural conditions underlying our counterfactual model, we make following assumptions:

1. **Structural causal generation and shared initial state:**
   The potential trajectories are generated according to the SCM in Figure 2 from treatment, covariates, and common initial latent state s_0. A treatment intervention changes treatment-specific dynamics but leaves initial state s_0 unchanged. Formally, for each instance,
   $$z(t_0 \mid \mathrm{do}(T = 0)) = z(t_0 \mid \mathrm{do}(T = 1)) = s_0 \tag{8}$$
   where $z(t_0 \mid \mathrm{do}(T = a))$ is initial state under intervention $\mathrm{do}(T = a)$.
2. **Latent consistency:** Let $y_*(t \mid \mathrm{do}(T = a))$ denote the noise-free potential trajectory under treatment intervention $\mathrm{do}(T = a)$ for $a \in \{0, 1\}$. If the factual treatment is $T = a$, then
   $$y_*(t) = y_*(t \mid \mathrm{do}(T = a)) \quad t > t_0. \tag{9}$$
3. **Conditional Ignorability:** There are no unobserved confounders affecting treatment assignment and outcome trajectories $\{y_*(t \mid \mathrm{do}(T = 0)), y_*(t \mid \mathrm{do}(T = 1))\}_{t>t_0} \perp T|X$.
4. **Positivity**: Every individual has a non-zero probability of receiving any treatment assignment. So, for every $a \in \{0, 1\}$ and every x in the support of X: $0 < P(T = a|X = x) < 1$.

A.1.1 CONSISTENCY REMARK

Remark 1. *We formulate consistency for the noise-free structural trajectory y_*. Under treatment intervention* $\mathrm{do}(T = a)$*,* CLL-NODE *generates*

$$z_i = \mathrm{ODEsolve}(s_0, t_0, t_i, a, x, \theta_{dyn}), \quad y_{i*}(t_i \mid \mathrm{do}(T = a)) = g_{\theta_{dec}}(z_i), \tag{10}$$

followed by the observation model

$$y_i = y_{i*}(t_i \mid \mathrm{do}(T = a)) + \epsilon_{y,i}, \quad \epsilon_{y,i} \sim \mathcal{N}(0, \sigma_y^2 I) \tag{11}$$

Thus treatment acts through latent dynamics and changes the noise-free structural trajectory y_*, whereas ϵ_y is the decoder measurement error with variance $\sigma_y^2 I$. Non-degenerate measurement noise means that this error has positive variance $\sigma_y^2 > 0$, and therefore it is not identically zero across all observations. Persistent individual features are presented by s_0, while the encoder does not infer and reuse the factual realization of ϵ_y. We therefore impose consistency on y_*. When $T' = T$, the same structural trajectory and observation distribution are recovered, although independent draws of ϵ_y may produce different observations. Accordingly, y' is posterior predictive measurement rather than individual uniquely realized potential outcome.

B APPROXIMATE POSTERIOR AND ELBO

Joint distribution. For notational simplicity here we provide results for an individual instance only, where n denotes the number of observations in the sequence. Generalization to longitudinal setting with P instances is straightforward. The following derivations are based on results from Iakovlev et al. (2023), with addition of T and x to the dynamics function and certain changes to shooting variables s. Here we denote the n high-dimensional observations that are collected at time points $\boldsymbol{t} = [t_1, \ldots, t_n]$ by $\boldsymbol{y} = [y_1, \ldots, y_n]$.

The joint distribution factorizes as:

$$p(\boldsymbol{y}, s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|T, x, \boldsymbol{t}) = p(\boldsymbol{y}|s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, \boldsymbol{t})p(s_{0:B}|\theta_{\text{dyn}}, T, x)p(\theta_{\text{dyn}})p(\theta_{\text{dec}}),$$

where

$$p(s_{0:B}|\theta_{\text{dyn}}, T, x) = p(s_0)\prod_{b=1}^{B} p(s_b|s_{b-1}, \theta_{\text{dyn}}, T, x)$$
$$= p(s_0)\prod_{b=1}^{B} \mathcal{N}(s_b|\text{ODEsolve}(s_{b-1}, t_{[b-1]}, t_{[b]}, T, x, \theta_{\text{dyn}}), \sigma_c^2 I),$$

and

$$p(\theta_{\text{dyn}}) = \mathcal{N}(\theta_{\text{dyn}}|\mu_{\theta_{\text{dyn}}}, \sigma^2_{\theta_{\text{dyn}}} I), \quad p(\theta_{\text{dec}}) = \mathcal{N}(\theta_{\text{dec}}|\mu_{\theta_{\text{dec}}}, \sigma^2_{\theta_{\text{dec}}} I),$$

where $\mathcal{N}$ is the Gaussian distribution, and $I \in \mathbb{R}^{d\times d}$ is the identity matrix. Finally, the likelihood term, $p(\boldsymbol{y}|s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, \boldsymbol{t})$, factorizes across the B shooting blocks as well as the first measurement time point y_1 as follows

$$p(\boldsymbol{y}|s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, \boldsymbol{t}) = p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]})\prod_{b=1}^{B} p(\{y_i\}_{i\in\mathcal{I}_b}|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i)$$
$$= p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]})\prod_{b=1}^{B}\prod_{i\in\mathcal{I}_b} p(y_i|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i)$$
$$= \mathcal{N}(y_1|g_{\theta_{\text{dec}}}(\text{ODEsolve}(s_0, t_0, t_{[1]}, T, x, \theta_{\text{dyn}})), \sigma_y^2 I)$$
$$\times \prod_{b=1}^{B}\prod_{i\in\mathcal{I}_b} \mathcal{N}(y_i|g_{\theta_{\text{dec}}}(\text{ODEsolve}(s_b, t_{[b]}, t_i, T, x, \theta_{\text{dyn}})), \sigma_y^2 I),$$

where σ_y^2 is the observation noise variance that is shared across data dimensions.

Approximate inference. We define the following approximate variational distribution:

$$q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) = \Big[\prod_{b=0}^{B} q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y}, \boldsymbol{t})\Big] \cdot q_{\psi_{\text{dyn}}}(\theta_{\text{dyn}}) q_{\psi_{\text{dec}}}(\theta_{\text{dec}})$$
$$= \Big[\prod_{b=0}^{B} \mathcal{N}(s_b|\boldsymbol{\gamma}_b, \text{diag}(\boldsymbol{\tau}_b^2))\Big]$$
$$\cdot \mathcal{N}(\theta_{\text{dyn}}|\boldsymbol{\gamma}_{\theta_{\text{dyn}}}, \text{diag}(\boldsymbol{\tau}^2_{\theta_{\text{dyn}}}))\mathcal{N}(\theta_{\text{dec}}|\boldsymbol{\gamma}_{\theta_{\text{dec}}}, \text{diag}(\boldsymbol{\tau}^2_{\theta_{\text{dec}}})),$$

where $[\boldsymbol{\gamma}_0, \boldsymbol{\tau}_0, \ldots, \boldsymbol{\gamma}_B, \boldsymbol{\tau}_B] = \text{Transformer}(\boldsymbol{y}, \boldsymbol{t}; \psi_{\text{enc}})$ is the transformer-based encoder, parameterized by ψ_{enc}, proposed in (Iakovlev et al., 2023).

ELBO. The ELBO can be written as

$$\mathcal{L} = \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \frac{p(\boldsymbol{y}, s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|T, x, \boldsymbol{t})}{q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t})} d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}$$
$$= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \frac{p(\boldsymbol{y}|s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, \boldsymbol{t}) p(s_{0:B}|\theta_{\text{dyn}}, T, x) p(\theta_{\text{dyn}}) p(\theta_{\text{dec}})}{q_{\psi_{\text{enc}}}(s_{0:B}|\boldsymbol{y}, \boldsymbol{t}) q_{\psi_{\text{dyn}}}(\theta_{\text{dyn}}) q_{\psi_{\text{dec}}}(\theta_{\text{dec}})} d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}$$
$$= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(\boldsymbol{y}|s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, \boldsymbol{t}) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}$$
$$- \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \frac{q_{\psi_{\text{enc}}}(s_{0:B}|\boldsymbol{y}, \boldsymbol{t})}{p(s_{0:B}|\theta_{\text{dyn}}, T, x)} d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}$$
$$- \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \frac{q_{\psi_{\text{dyn}}}(\theta_{\text{dyn}})}{p(\theta_{\text{dyn}})} d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}$$
$$- \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \frac{q_{\psi_{\text{dec}}}(\theta_{\text{dec}})}{p(\theta_{\text{dec}})} d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}$$
$$= \mathcal{L}_1 - \mathcal{L}_2 - \mathcal{L}_3 - \mathcal{L}_4.$$

Let's look at each term $\mathcal{L}_i$ separately. The first term can be written as

$$
\begin{aligned}
\mathcal{L}_1 &= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(\boldsymbol{y}|s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, \boldsymbol{t}) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \Bigg[p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]}) \\
&\quad \cdot \prod_{b=1}^{B} p(\{y_i\}_{i\in\mathcal{I}_b}|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i) \Bigg] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]}) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&\quad + \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \left[\prod_{b=1}^{B} p(\{y_i\}_{i\in\mathcal{I}_b}|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i) \right] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]}) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&\quad + \sum_{b=1}^{B} \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(\{y_i\}_{i\in\mathcal{I}_b}|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]}) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_0 \\
&\quad + \sum_{b=1}^{B} \int q_\psi(s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln p(\{y_i\}_{i\in\mathcal{I}_b}|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i) d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_b \\
&= \mathbb{E}_{q_\psi(s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}|\boldsymbol{y}, \boldsymbol{t})} \left[\ln p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]}) \right] \\
&\quad + \sum_{b=1}^{B} \mathbb{E}_{q_\psi(s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t})} \left[\ln p(\{y_i\}_{i\in\mathcal{I}_b}|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i) \right] \\
&= \mathbb{E}_{q_\psi(s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}|\boldsymbol{y}, \boldsymbol{t})} \left[\ln p(y_1|s_0, \theta_{\text{dec}}, \theta_{\text{dyn}}, T, x, t_{[1]}) \right] \\
&\quad + \sum_{b=1}^{B} \sum_{i\in\mathcal{I}_b} \mathbb{E}_{q_\psi(s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t})} \left[\ln p(y_i|s_b, \theta_{\text{dyn}}, \theta_{\text{dec}}, T, x, t_i) \right].
\end{aligned}
$$

The second term is

$$
\begin{aligned}
\mathcal{L}_2 &= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \frac{q_{\psi_{\text{enc}}}(s_{0:B}|\boldsymbol{y}, \boldsymbol{t})}{p(s_{0:B}|\theta_{\text{dyn}}, T, x)} d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \left[\frac{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y}, \boldsymbol{t})}{p(s_0)} \prod_{b=1}^{B} \frac{q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y}, \boldsymbol{t})}{p(s_b|s_{b-1}, \theta_{\text{dyn}}, T, x)} \right] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \left[\frac{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y}, \boldsymbol{t})}{p(s_0)} \right] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&\quad + \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \left[\prod_{b=1}^{B} \frac{q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y}, \boldsymbol{t})}{p(s_b|s_{b-1}, \theta_{\text{dyn}}, T, x)} \right] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&= \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \left[\frac{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y}, \boldsymbol{t})}{p(s_0)} \right] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B} \\
&\quad + \sum_{b=1}^{B} \int q_\psi(s_{0:B}, \theta_{\text{dyn}}, \theta_{\text{dec}}|\boldsymbol{y}, \boldsymbol{t}) \ln \left[\frac{q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y}, \boldsymbol{t})}{p(s_b|s_{b-1}, \theta_{\text{dyn}}, T, x)} \right] d\theta_{\text{dyn}} d\theta_{\text{dec}} ds_{0:B}
\end{aligned}
$$

$$
\begin{aligned}
&= \int q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t}) \ln\left[\frac{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}{p(s_0)}\right] ds_0 \\
&\quad + \sum_{b=1}^{B} \int q_{\psi}(s_b, s_{b-1}, \theta_{\text{dyn}}|\boldsymbol{y},\boldsymbol{t}) \ln\left[\frac{q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y},\boldsymbol{t})}{p(s_b|s_{b-1},\theta_{\text{dyn}},T,x)}\right] d\theta_{\text{dyn}} ds_{b-1} ds_b \\
&= \int q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t}) \ln\left[\frac{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}{p(s_0)}\right] ds_0 \\
&\quad + \sum_{b=1}^{B} \int q_{\psi}(s_{b-1}, \theta_{\text{dyn}}|\boldsymbol{y},\boldsymbol{t}) \left(\int q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y},\boldsymbol{t}) \ln\left[\frac{q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y},\boldsymbol{t})}{p(s_b|s_{b-1},\theta_{\text{dyn}},T,x)}\right] ds_b \right) d\theta_{\text{dyn}} ds_{b-1} \\
&= \int q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t}) \ln\left[\frac{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}{p(s_0)}\right] ds_0 \\
&\quad + \sum_{b=1}^{B} \int q_{\psi}(s_{b-1}, \theta_{\text{dyn}}|\boldsymbol{y},\boldsymbol{t}) \mathrm{KL}\left(q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y},\boldsymbol{t}) \| p(s_b|s_{b-1},\theta_{\text{dyn}},T,x)\right) d\theta_{\text{dyn}} ds_{b-1} \\
&= \mathrm{KL}\left(q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t}) \| p(s_0)\right) + \sum_{b=1}^{B} \mathbb{E}_{q_{\psi}(s_{b-1},\theta_{\text{dyn}}|\boldsymbol{y},\boldsymbol{t})}\left[\mathrm{KL}\left(q_{\psi_{\text{enc}}}(s_b|\boldsymbol{y},\boldsymbol{t}) \| p(s_b|s_{b-1},\theta_{\text{dyn}},T,x)\right)\right].
\end{aligned}
$$

It is easy to see that the last two terms correspond to the KL divergency

$$\mathcal{L}_3 = \mathrm{KL}(q_{\psi_{\text{dyn}}}(\theta_{\text{dyn}}) \| p(\theta_{\text{dyn}})), \quad \mathcal{L}_4 = \mathrm{KL}(q_{\psi_{\text{dec}}}(\theta_{\text{dec}}) \| p(\theta_{\text{dec}})).$$

C Proof of Lemma 1

Proof. The proof follows the optimal-discriminator analysis developed in Proposition 1 and Theorem 1 of Goodfellow et al. (2014). In their setting, the discriminator distinguishes between the data and generated distributions, while the generator is trained to make this distinction difficult. In our setting, the aggregated latent distributions of the treated and untreated groups play analogous roles, and the encoder is trained to make treatment assignment difficult to predict from the initial latent state. We therefore adopt the same general proof strategy: fixing the latent distributions, deriving the optimal discriminator, and substituting it back into the objective. However, we adapt each step to account for treatment-group probabilities and the latent-variable formulation.

Let $q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})$ denote the variational posterior over initial latent state s_0 for an observed trajectory $\boldsymbol{y}$ at observation times $\boldsymbol{t}$. Assume a universal discriminator D that implements all measurable functions $D : S_0 \longrightarrow (0,1)$ where $S_0 \subseteq \mathcal{Z}$. Recall the discriminator loss

$$\mathcal{L}_{\text{disc}}(D|\boldsymbol{y},T,\boldsymbol{t},\psi_{\text{enc}}) = -\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}\big[\log D(s_0)\cdot T + \log(1-D(s_0))\cdot(1-T)\big]. \tag{12}$$

and define discriminator value function, which corresponds to the negative expected value of instance-wise loss as follows:

$$V(D \mid \psi_{\text{enc}}) = -\mathbb{E}_{p(\boldsymbol{y},T,\boldsymbol{t})}[\mathcal{L}_{\text{disc}}(D|\boldsymbol{y},T,\boldsymbol{t},\psi_{\text{enc}})] \tag{13}$$

Let the marginal probabilities of treatment assignments be defined as follows:

$$\pi_0 := \mathbb{P}(T=0) \quad \pi_1 := \mathbb{P}(T=1) \quad \pi_0,\pi_1 > 0, \pi_0+\pi_1 = 1 \tag{14}$$

By the law of total expectation, $p(\boldsymbol{y},T,\boldsymbol{t}) = p(\boldsymbol{y},\boldsymbol{t} \mid T)P(T)$. We can explicitly expand the outer expectation over discrete states of $T \in \{0,1\}$, substituting the respective values for T into the cross-entropy expression:

$$V(D \mid \psi_{\text{enc}}) = \pi_1 \mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=1)}[\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}[\log D(s_0)]] \tag{15}$$

$$+ \pi_0 \mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=0)}[\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}[\log(1-D(s_0))]]. \tag{16}$$

Define the aggregated posterior densities of s_0 conditioned over $T=a$ as:

$$p_{a,\psi_{\text{enc}}}(s_0) = \mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=a)}[q_{\psi_{\text{enc}}}(s_0 \mid \boldsymbol{y},\boldsymbol{t})], \quad a \in \{0,1\}. \tag{17}$$

Since $q_{\psi_{\text{enc}}}(s_0 \mid \boldsymbol{y}, \boldsymbol{t})$ is non-degenerate Gaussian density (based on Appendix B, page 14), the aggregate density above exists (refer to Theorem 2.37 Folland (1999)).

Expanding expectation terms for treated and untreated groups and using Equation 17 leads to:

$$\begin{aligned}\mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=1)}\left[\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}[\log D(s_0)]\right] &= \int_{s_0} \log(D(s_0))\mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=1)}[q_{\psi_{\text{enc}}}(s_0 \mid \boldsymbol{y},\boldsymbol{t})]ds_0 \\ &= \int_{s_0} \log(D(s_0))p_{1,\psi_{\text{enc}}}(s_0)ds_0\end{aligned} \tag{18}$$

and

$$\begin{aligned}\mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=0)}\left[\mathbb{E}_{q_{\psi_{\text{enc}}}(s_0|\boldsymbol{y},\boldsymbol{t})}[\log(1-D(s_0))]\right] &= \int_{s_0} \log(1-D(s_0))\mathbb{E}_{p(\boldsymbol{y},\boldsymbol{t}|T=0)}[q_{\psi_{\text{enc}}}(s_0 \mid \boldsymbol{y},\boldsymbol{t})]ds_0 \\ &= \int_{s_0} \log(1-D(s_0))p_{0,\psi_{\text{enc}}}(s_0)ds_0.\end{aligned} \tag{19}$$

Substituting these terms in Equation 16 leads to:

$$V(D \mid \psi_{\text{enc}}) = \pi_1 \int_{S_0} \log D(s_0)p_{1,\psi_{\text{enc}}}(s_0)ds_0 + \pi_0 \int_{S_0} \log(1-D(s_0))p_{0,\psi_{\text{enc}}}(s_0)ds_0. \tag{20}$$

For any fixed encoder, the Bayes-optimal discriminator $D^*(s_0)$ is obtained by maximizing the integrand in Equation 20 point-wise. Taking derivative of integrand in Equation 20 with respect to $D(s_0)$ and setting it to zero yields:

$$\frac{\pi_1 p_{1,\psi_{\text{enc}}}(s_0)}{D(s_0)} - \frac{\pi_0 p_{0,\psi_{\text{enc}}}(s_0)}{1-D(s_0)} = 0. \tag{21}$$

Solving for $D(s_0)$, we obtain the optimal discriminator:

$$D^*(s_0) = \frac{\pi_1 p_{1,\psi_{\text{enc}}}(s_0)}{\pi_1 p_{1,\psi_{\text{enc}}}(s_0) + \pi_0 p_{0,\psi_{\text{enc}}}(s_0)} = \frac{\pi_1 p_{1,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}. \tag{22}$$

Note that $m_{\psi_{\text{enc}}}(s_0) = \pi_1 p_{1,\psi_{\text{enc}}}(s_0) + \pi_0 p_{0,\psi_{\text{enc}}}(s_0)$ is marginal density of s_0 and that $1 - D^*(s_0) = \frac{\pi_0 p_{0,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}$.

Substituting optimal discriminator and its complement into Equation 20 gives:

$$\begin{aligned}V(D^* \mid \psi_{\text{enc}}) &= \pi_1 \int_{S_0} p_{1,\psi_{\text{enc}}}(s_0) \log\left(\frac{\pi_1 p_{1,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &\quad + \pi_0 \int_{s_0} p_{0,\psi_{\text{enc}}}(s_0) \log\left(\frac{\pi_0 p_{0,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &= \pi_1 \log \pi_1 \int_{s_0} p_{1,\psi_{\text{enc}}}(s_0)ds_0 + \pi_0 \log \pi_0 \int_{s_0} p_{0,\psi_{\text{enc}}}(s_0)ds_0 \\ &\quad + \pi_1 \int_{s_0} p_{1,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{1,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &\quad + \pi_0 \int_{s_0} p_{0,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{0,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0.\end{aligned} \tag{23}$$

The last equation is obtained by separating the treatment probabilities inside logarithms. Using the fact that each conditional density integrates to one $\int_{S_0} p_{a,\psi_{\text{enc}}}(s_0)ds_0 = 1$ gives:

$$\begin{aligned}V(D^* \mid \psi_{\text{enc}}) = \pi_1 \log \pi_1 + \pi_0 \log \pi_0 + \pi_1 \int_{s_0} p_{1,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{1,\psi_{enc}}(s_0)}{m_{\psi_{enc}}(s_0)}\right) ds_0 + \\ + \pi_0 \int_{s_0} p_{0,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{0,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0.\end{aligned} \tag{24}$$

Rewriting second term based on KL-divergence (D_{KL}) gives:

$$V(D^* \mid \psi_{\text{enc}}) = \pi_1 \log \pi_1 + \pi_0 \log \pi_0 + \pi_1 D_{KL}(p_{1,\psi_{\text{enc}}} || m_{\psi_{\text{enc}}}) + \pi_0 D_{KL}(p_{0,\psi_{\text{enc}}} || m_{\psi_{\text{enc}}}). \tag{25}$$

Note that mutual information between treatment T and initial state s_0 can be written as Cover et al. (1991):

$$I_{\psi_{\text{enc}}}(T; s_0) = D_{KL}\left(p_{\psi_{\text{enc}}}(T, s_0) || p(T) m_{\psi_{\text{enc}}}(s_0)\right). \tag{26}$$

Because T takes values 0 and 1, we can expand KL divergence into two integrals:

$$\begin{aligned} I_{\psi_{\text{enc}}}(T; s_0) = &\int_{s_0} p_{\psi_{\text{enc}}}(T=1, s_0) \log\Big(\frac{p_{\psi_{\text{enc}}}(T=1, s_0)}{\mathbb{P}(T=1) m_{\psi_{enc}}(s_0)}\Big) ds_0 \\ &+ \int_{s_0} p_{\psi_{\text{enc}}}(T=0, s_0) \log\Big(\frac{p_{\psi_{\text{enc}}}(T=0, s_0)}{\mathbb{P}(T=0) m_{\psi_{\text{enc}}}(s_0)}\Big) ds_0. \end{aligned} \tag{27}$$

Based on probability chain rule: $p_{\psi_{\text{enc}}}(T=1, s_0) = \pi_1 p_{1,\psi_{\text{enc}}}(s_0)$ and $p_{\psi_{\text{enc}}}(T=0, s_0) = \pi_0 p_{0,\psi_{\text{enc}}}(s_0)$:

$$\begin{aligned} I_{\psi_{\text{enc}}}(T; s_0) &= \int_{s_0} \pi_1 p_{1,\psi_{\text{enc}}}(s_0) \log\left(\frac{\pi_1 p_{1,\psi_{\text{enc}}}(s_0)}{\pi_1 m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &\quad + \int_{s_0} \pi_0 p_{0,\psi_{\text{enc}}}(s_0) \log\left(\frac{\pi_0 p_{0,\psi_{\text{enc}}}(s_0)}{\pi_0 m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &= \int_{s_0} \pi_1 p_{1,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{1,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &\quad + \int_{s_0} \pi_0 p_{0,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{0,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &= \pi_1 \int_{s_0} p_{1,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{1,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &\quad + \pi_0 \int_{s_0} p_{0,\psi_{\text{enc}}}(s_0) \log\left(\frac{p_{0,\psi_{\text{enc}}}(s_0)}{m_{\psi_{\text{enc}}}(s_0)}\right) ds_0 \\ &= \pi_1 D_{KL}(p_{1,\psi_{\text{enc}}} || m_{\psi_{\text{enc}}}) + \pi_0 D_{KL}(p_{0,\psi_{enc}} || m_{\psi_{\text{enc}}}). \end{aligned} \tag{28}$$

Also, recall entropy $H(T) = -\pi_1 \log \pi_1 - \pi_0 \log \pi_0$ (Cover et al., 1991). Therefore Equation 24 becomes:

$$V(D^* \mid \psi_{\text{enc}}) = -H(T) + I_{\psi_{\text{enc}}}(T; s_0). \tag{29}$$

The optimized discriminator loss is therefore:

$$\mathcal{L}^*_{\text{disc}}(\psi_{\text{enc}}) := -\sup_D V(D|\psi_{\text{enc}}) = -V(D^*|\psi_{\text{enc}}) = H(T) - I_{\psi_{\text{enc}}}(T; s_0). \tag{30}$$

Since $H(T)$ does not depend on encoder parameters, maximizing $\mathcal{L}^*_{\text{disc}}(\psi_{\text{enc}})$ with respect to ψ_{enc} is equivalent to minimizing $I_{\psi_{\text{enc}}}(T; s_0)$.

□

D Dynamics Function

Dynamics functions for neural ODEs are typically defined as the first or second-order models. Throughout this work we assume the second-order model, where the latent state consists of the state and velocity (or momentum) components $\boldsymbol{z}(t) := (z_s(t)^T, z_v(t)^T)^T$, $\frac{dz_s(t)}{dt} = z_v(t)$ and $\frac{dz_v(t)}{dt} = f_{\theta_{\text{dyn}}}(z_s(t), z_v(t))$. Further discussion of the advantages of second-order models can be found in Yıldız et al. (2019).

As mentioned in the main text, we assume the following parameterization:

$$\frac{dz(t)}{dt} = f_{\theta_{\text{dyn}}}(z(t), T, x).$$

Since our method assumes a binary treatment variable T, we implement $f_{\theta_{\text{dyn}}}(z(t), T, x)$ using two separate neural networks, one for each treatment value.

The general architecture of these networks is shown in Table 3. The output dimension $L/2$ follows from our use of a second-order latent dynamics model, while the remaining hyperparameters depend on the dataset.

Table 3: Dynamics function architecture

Layer	Output dimension	Operation
Input	$L + Q$	–
Hidden 1	$m_F \times L$	Linear + Activation
Hidden 2	$m_F \times L$	Linear + Activation
Output	$L/2$	Linear

In all experiments, we set $L = 32$ and $m_F = 16$. The value of Q, corresponding to the number of covariates, is dataset-specific: $Q = 0$ for Causal MNIST, $Q = 2$ for Causal MNIST Groups, $Q = 3$ for Physionet and $Q = 4$ for semi-synthetic Physionet.

E BASELINES IMPLEMENTATION

In this section, we describe how the baseline methods were adapted to the datasets considered in this work.

VCI and CEVAE. As VCI and CEVAE were not originally designed to handle temporal data, we adopt the following strategy: each element of a trajectory is treated as a separate data point and provided to the model, together with its corresponding timestamp, for counterfactual prediction.

CF-NODE. This method is conceptually similar to ours, as it also incorporates an ODE component to model temporal dynamics. The key difference is that it relies exclusively on pre-treatment observations to predict counterfactual outcomes, whereas our setting assumes that observations are available after treatment assignment. Therefore, to apply this method on the simulated Causal MNIST and Causal MNIST Groups datasets, we additionally generate pre-treatment trajectories using the same data-generating mechanism as for the original trajectories. Due to this difference in the required observational setting, the method cannot be applied to Physionet.

F SHOOTING VARIABLES ABLATION

Table 4: CLL-NODE performance on Causal MNIST regular with various number of datapoints in each shooting block. Lower is better.

block size	MSE
block size 1 (separate shooting variable for each datapoint)	$\mathbf{0.009}_{\pm 0.002}$
block size 5	$0.039_{\pm 0.001}$
block size 10	$0.027_{\pm 0.001}$
block size 20 (full trajectory)	$- - - - -$

As shown in Table 4, the best performance on Causal MNIST regular was achieved when assigning a separate shooting variable to each datapoint. This setting is also the most computationally efficient, as it enables parallel optimization across shooting intervals during training. In contrast, training the model on the full trajectory was unsuccessful due to numerical instabilities in the ODE solver, a limitation previously discussed in (Ribeiro et al., 2019; Voss et al., 2004; Iakovlev et al., 2023).

It is important to note that the shooting variables are used only to stabilize training. During counterfactual estimation, the entire trajectory is generated starting from s_0. Using the intermediate shooting variables $s_{1:B}$ for counterfactual prediction would be inappropriate, as they may retain treatment-specific information that is explicitly removed from s_0 and because the treatment is administered at time t_0.

G SUPPLEMENTARY FIGURES

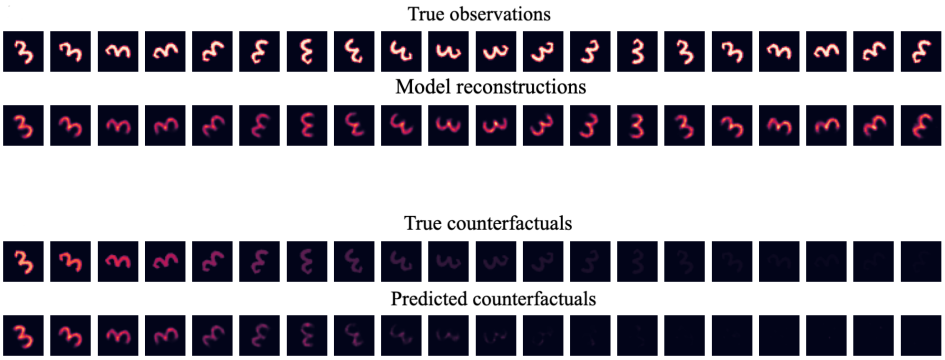


Figure 5: Predictions on Causal MNIST regular dataset. The top two figures show the original trajectory and the trajectory reconstructed by CLL-NODE. The bottom two figures show the true counterfactual trajectory and the counterfactual trajectory predicted by CLL-NODE.

H SEMI-SYNTHETIC COUNTERFACTUAL GENERATION

To enable quantitative evaluation of counterfactual predictions, we construct a semi-synthetic version of the Physionet dataset. We retain the observed physiological trajectories from the original dataset and introduce a synthetic treatment whose assignment mechanism and effect are fully known. This allows us to generate potential outcomes for every individual.

Let

$$\mathbf{y}_i^{(0)}(t) \in \mathbb{R}^D$$

denote the outcome of patient i with $T = 0$. We define $\mathbf{y}_i^{(0)}(t)$ using the original Physionet trajectory. The potential outcome of $T = 1$ is constructed as

$$\mathbf{y}_i^{(1)}(t) = \mathbf{y}_i^{(0)}(t) + \mathbf{\Delta}_i(t), \tag{31}$$

where $\mathbf{\Delta}_i(t) \in \mathbb{R}^D$ represents the treatment-induced modification of the physiological trajectory.

We assume that the treatment effect evolves according to the following dynamical system:

$$\frac{d\mathbf{\Delta}_i(t)}{dt} = \alpha h_i \mathbf{v} + \beta \mathbf{v} \odot \tanh\left(\mathbf{y}_i^{(1)}(t)\right) - \lambda \mathbf{\Delta}_i(t), \tag{32}$$

with initial condition

$$\mathbf{\Delta}_i(t_0) = \mathbf{0}. \tag{33}$$

Here, $\alpha > 0$ controls the overall treatment-effect magnitude, h_i determines patient-specific treatment-effect heterogeneity, $\mathbf{v} \in \mathbb{R}^D$ specifies the direction and relative magnitude of the treatment effect across physiological variables, $\beta \geq 0$ controls the dependence of the treatment response on the current physiological state, and $\lambda > 0$ determines the decay rate of the treatment effect. The operator $\odot$ denotes element-wise multiplication.

The first term in Equation 32,

$$\alpha h_i \mathbf{v},$$

corresponds to a patient-specific treatment forcing. The second term,

$$\beta \mathbf{v} \odot \tanh\left(\mathbf{y}_i^{(1)}(t)\right),$$

introduces state-dependent treatment effects. The hyperbolic tangent bounds this contribution and prevents unstable growth of the simulated trajectories. Finally, the term

$$-\lambda \mathbf{\Delta}_i(t)$$

prevents the treatment effect from growing indefinitely and causes it to approach an equilibrium value.

The individual treatment-response coefficient h_i is generated from the patient's baseline physiological state. In particular, we define

$$h_i = \frac{1}{2} + \sigma\left(\mathbf{w}_h^\top \mathbf{y}_i^{(0)}(t_0)\right), \tag{34}$$

where $\sigma(\cdot)$ denotes the logistic sigmoid function and $\mathbf{w}_h$ is a randomly generated and fixed coefficient vector. Consequently,

$$h_i \in (0.5, 1.5),$$

such that patients exhibit different treatment-effect magnitudes while the overall scale of the intervention remains bounded.

For numerical simulation, we normalize the observation times of each trajectory to the interval $[0, 1]$. Let

$$\tau_{ik} = \frac{t_{ik} - t_{i0}}{t_{iK} - t_{i0}}, \tag{35}$$

and define

$$\Delta\tau_{ik} = \tau_{ik} - \tau_{i,k-1}.$$

Equation 32 is then discretized using the forward Euler method:

$$\mathbf{\Delta}_{i,k} = \mathbf{\Delta}_{i,k-1} + \Delta\tau_{ik}\Big[\alpha h_i \mathbf{v} + \beta \mathbf{v} \odot \tanh\left(\mathbf{y}_{i,k-1}^{(1)}\right) - \lambda \mathbf{\Delta}_{i,k-1}\Big], \tag{36}$$

$$\mathbf{y}_{i,k}^{(1)} = \mathbf{y}_{i,k}^{(0)} + \mathbf{\Delta}_{i,k}. \tag{37}$$

Thus, potential outcomes are available for every individual.

Treatment assignment. We define the treatment indicator deterministically with respect to the trajectory type: $T = 0$ corresponds to the original trajectory, whereas $T = 1$ corresponds to the generated one. For each instance, the factual trajectory is then selected at random: with probability 0.5, the original trajectory is treated as factual, and otherwise the generated trajectory is used. During training, the model receives only the factual trajectory $\mathbf{y}_i^{\mathrm{F}}(t)$ together with its assigned treatment T_i. The alternative potential outcome $\mathbf{y}_i^{\mathrm{CF}}(t)$ is removed from the training and used exclusively to evaluate the accuracy of the counterfactual prediction.