Hizli, Caglar; John, ST; Juuti, Anne; Saarinen, Tuure; Pietiläinen, Kirsi; Marttinen, Pekka

Causal Modeling of Policy Interventions From Treatment-Outcome Sequences

Published in:
Proceedings of the 40th International Conference on Machine Learning

Published: 01/07/2023

Document Version
Publisher's PDF, also known as Version of record

Published under the following license:
CC BY

Please cite the original version:
Hizli, C., John, ST., Juuti, A., Saarinen, T., Pietiläinen, K., & Marttinen, P. (2023). Causal Modeling of Policy Interventions From Treatment-Outcome Sequences. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th International Conference on Machine Learning* (pp. 13050-13084). (Proceedings of Machine Learning Research; Vol. 202). JMLR. https://proceedings.mlr.press/v202/hizli23a.html

Causal Modeling of Policy Interventions From Treatment–Outcome Sequences

Çağlar Hızlı [1] ST John [1] Anne Juuti [2] Tuure Saarinen [2] Kirsi Pietiläinen [2] Pekka Marttinen [1]

Abstract

A *treatment policy* defines when and what treatments are applied to affect some outcome of interest. Data-driven decision-making requires the ability to predict *what happens if a policy is changed*. Existing methods that predict how the outcome evolves under different scenarios assume that the tentative sequences of future treatments are fixed in advance, while in practice the treatments are determined stochastically by a policy and may depend, for example, on the efficiency of previous treatments. Therefore, the current methods are not applicable if the treatment policy is unknown or a counterfactual analysis is needed. To handle these limitations, we model the treatments and outcomes jointly in continuous time, by combining Gaussian processes and point processes. Our model enables the estimation of a treatment policy from observational sequences of treatments and outcomes, and it can predict the interventional and counterfactual progression of the outcome *after an intervention on the treatment policy* (in contrast with the causal effect of a single treatment). We show with real-world and semi-synthetic data on blood glucose progression that our method can answer causal queries more accurately than existing alternatives.

1. Introduction

What policy should we adopt? In healthcare, for example, we observe patients' physiological markers (*outcomes*) that change over time. We want to (positively) affect these outcomes by actions (*treatments*) such as doses of a medicine. Sequences of outcomes and treatments are recorded as a time series. A *policy* dictates what actions to take and when. To improve our policies, we must be able to assess their consequences: What is the effect of a given policy? What will be the effect of a change to a different policy? What would have happened if a patient had followed a different treatment policy? These questions correspond to observational, interventional, and counterfactual queries, and answering them is crucial, especially in domains such as public health and healthcare, where quantifying risks and expectations accompanying a policy decision, as well as evaluating past policies, is essential (Schulam & Saria, 2017; Bica et al., 2020; Oberst & Sontag, 2019; Tsirtsis & Gomez Rodriguez, 2020; Tsirtsis et al., 2021). Answering the interventional and counterfactual queries requires modeling policy interventions using a *causal* model.

We want to infer a causal model of recurrent treatments from sequential treatment–outcome data. Such data are always created by some policy, which induces dependencies between past events and future treatments. However, the policy itself is generally not recorded and may be known only implicitly from the observed distribution of treatments and outcomes. In epidemiology, this problem has been studied using linear models for discrete-time sequences (Robins, 1986; Robins & Hernán, 2009; Taubman et al., 2009; Zhang et al., 2018), though these works are unable to capture nonlinear, long-term dependencies in real-world, continuous-time treatment–outcome sequences (Bica et al., 2021).

To handle such complex, long-term dependencies, sequential treatment–outcome models based on Bayesian nonparametrics and neural networks have been proposed (Xu et al., 2016; Schulam & Saria, 2017; Soleimani et al., 2017; Bica et al., 2020; Seedat et al., 2022). However, they neglect the policy-induced link between past events and future treatments, and their causal analyses are limited to estimating the impact of a fixed sequence of treatments set by hand or generated by a simplistic parametric model. Such models do not generalize beyond simulations to the analysis of realistic treatment policies in real-world applications. Recently, Hua et al. (2021) modeled treatments explicitly; however, they focused on optimizing treatment strategies instead of causal estimation and only considered a parametric model tailored for a particular application.

With an appropriate causal model, we can also evaluate treatment policies using counterfactual reasoning, which

[1] Department of Computer Science, Aalto University, Helsinki, Finland [2] Helsinki University Hospital and University of Helsinki, Helsinki, Finland. Correspondence to: Çağlar Hızlı <caglar.hizli@aalto.fi>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

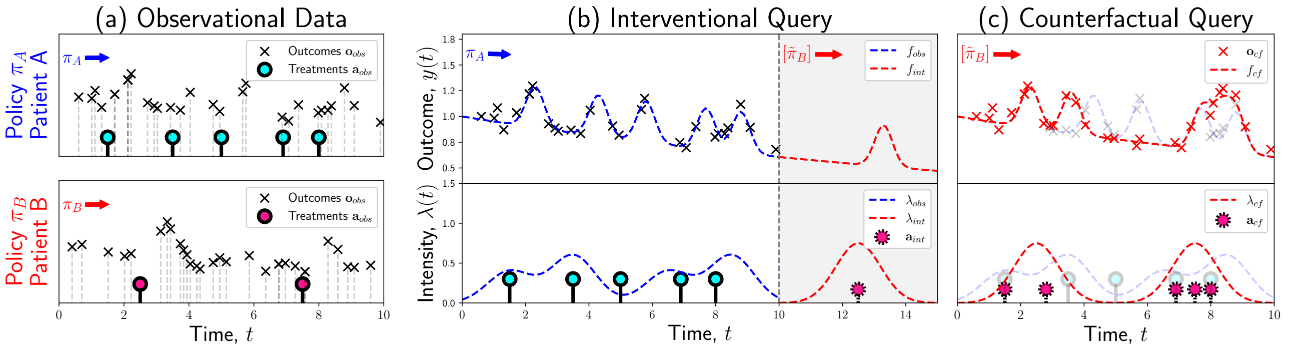


Figure 1: **(a)** Sequential treatment–outcome data for patients A (top) and B (bottom). Patients follow distinct policies π_A and π_B in the observation period $[0, 10]$, where patient A is treated more frequently. A treatment causes a (slightly delayed) increase or 'bump' in the outcome. **(b, c)** We now focus on patient A and show outcome trajectory f (top) and treatment intensity λ (bottom) in dashed lines. The estimates based on the observational data are in blue. **(b)** The interventional query corresponds to how the outcome trajectory of A will progress *after* the observation period (shaded area) under a treatment intensity induced by a *different* policy π_B, shown in red. **(c)** The counterfactual query corresponds to how the outcome trajectory of A *would have* progressed if the policy in the observation period $[0, 10]$ had been set to π_B instead. The observational data and estimations are shown in the background, while the counterfactual data and predictions are shown in red. Notice some of the observed treatments are kept as counterfactual treatments $\mathbf{a}_{cf}$, where the counterfactual intensity λ_{cf} is higher than the observational intensity λ_{obs}. Counterfactual treatments in turn affect the counterfactual trajectory f_{cf}.

allows for learning from mistakes by considering alternative scenarios to past events (Epstude & Roese, 2008; Oberst & Sontag, 2019). This is not considered by most of the literature, which focuses on future outcome progression. One recent work (Noorbakhsh & Rodriguez, 2022) applies counterfactual reasoning to event data using a counterfactual temporal point process, but does not consider a treatment–outcome setup (for further related work, see Appendix A).

To address these limitations, we propose a joint treatment–outcome model to estimate treatment policies and treatment responses in continuous time. Our contributions are:

Joint treatment–outcome model. We combine a marked point process and a conditional Gaussian process (GP). Our *non-parametric* model can be learned from observational sequential treatment–outcome data (Fig. 1(a)) and can estimate future and counterfactual progression.

Interventional and counterfactual queries of interventions on policies. We show that an intervention on a treatment policy is equivalent to a sequence of stochastic interventions on treatments, which we can model with our joint model, and use this to answer interventional (Fig. 1(b)) and counterfactual queries (Fig. 1(c)).

Counterfactual sampling of arbitrary point processes. We extend the algorithm of Noorbakhsh & Rodriguez (2022) from Poisson processes (independent events) to arbitrary point processes, allowing events to depend on past events.

We fit our model to a real-world data set on blood glucose progression, and from this generate realistic semi-synthetic use cases, where we show that our model accurately answers interventional and counterfactual policy queries.

2. Preliminaries

Our work builds on Gaussian processes (for more details see Williams & Rasmussen (2006)), marked point processes (Section 2.1) and causal inference (Section 2.2).

2.1. Marked Point Processes (MPP)

A temporal point process is a stochastic process that models a set of ordered random points $\{t_i\}_{i=1}^N$ on an interval $[0, T]$ (Daley & Vere-Jones, 2003; Rasmussen, 2011). This process is uniquely determined by its conditional intensity function $\lambda^*(\tau) = \lambda(\tau \mid \mathcal{H}_{<\tau})$, where the star superscript * denotes dependence on the past history $\mathcal{H}_{<\tau}$. The intensity function describes the instantaneous rate, i.e., the expected number of events in an infinitesimal interval $[\tau, \tau + \mathrm{d}\tau)$.

When each event time t_i is associated with additional information ('mark') m_i, we model $\mathcal{D} = \{(t_i, m_i)\}_{i=1}^N$ with a marked point process (MPP). For an MPP, the conditional intensity function additionally includes the conditional mark probability $p^*(m \mid t)$: $\lambda^*(t, m) = \lambda^*(t)p^*(m \mid t)$. The likelihood of $\mathcal{D}$ observed in the interval $[0, T]$ is given by

$$p(\mathcal{D} \mid \lambda^*(t, m)) = \left(\prod_{i=1}^{N} \lambda^*(t_i)p^*(m_i \mid t_i)\right) \exp(-\Lambda),$$

where Λ denotes the integral term $\Lambda = \int_{[0,T]} \lambda^*(\tau)\mathrm{d}\tau$.

2.2. Causal Inference

Causal effects are widely studied in the language of potential outcomes (PO, Neyman, 1923; Rubin, 1978) or structural causal models (SCM, Pearl, 2009). In this section, we briefly introduce both frameworks, as we use POs to define causal queries formally and show their identifiability, while using SCMs to emphasize the distinction between probability distributions induced by different types of target causal queries.

2.2.1. Potential Outcomes

Let A and Y denote a treatment and an outcome variable. In the PO framework, the primitive building block is the potential outcome $Y[\tilde{a}]$, which is a random variable that represents the value of the outcome Y under a treatment intervention $[A = \tilde{a}]$. The treatment intervention $[\tilde{a}]$ corresponds to the *do-operator* $\mathrm{do}(A = \tilde{a})$ in the SCM framework: $P(Y[\tilde{a}]) \equiv P(Y \mid \mathrm{do}(A = \tilde{a}))$ (Pearl, 2009).

2.2.2. Observational, Interventional and Counterfactual Distributions

For a set of variables $\mathbf{X} = \{X_k\}_{k=1}^{K}$, an SCM $\mathcal{M} = (\mathbf{S}, p(\mathbf{N}))$ consists of (i) a set of structural assignments $\mathbf{S} = \{X_k := f_k(\mathrm{pa}(X_k), N_k)\}_{k=1}^{K}$, where $\mathrm{pa}(\cdot)$ returns the parents of a given variable, and (ii) a distribution over noise variables $\mathbf{N} \sim p(\mathbf{N})$. A causal graph $\mathcal{G}$ is obtained by representing each variable X_k as a node and drawing edges to X_k from its parents $\mathrm{pa}(X_k)$. The SCM defines a generative process for $\mathbf{X}$ by sampling $\mathbf{N}$ from $p(\mathbf{N})$ and performing ancestral sampling over structural assignments $\mathbf{S}$, inducing an observational distribution $\mathbf{X} \sim p(\mathbf{X})$. Each data point $\mathbf{x}^{(n)} = \{x_k^{(n)}\}_{k=1}^{K}$ in the observed data $\mathcal{D} = \{\mathbf{x}^{(n)}\}_{n=1}^{N}$ is considered an i.i.d. sample from the observational distribution.

An *intervention* $\mathrm{do}(X_k = \tilde{x}_k)$ is performed by removing the assignment $f_k(\cdot, \cdot)$ and setting X_k to $\tilde{x}_k$. This yields an interventional SCM $\mathcal{M}_{\text{int}} = (\mathbf{S}_{\text{int}}, p(\mathbf{N}))$, inducing an interventional distribution: $p(\mathbf{X} \mid \mathrm{do}(X_k = \tilde{x}_k)) \equiv p(\mathbf{X}[\tilde{x}_k])$.

In the context of an observed data point $\mathbf{x}^{(n)}$, we define a *counterfactual* SCM by plugging in the individual's posterior noise distribution $p(\mathbf{N} \mid \mathbf{x}^{(n)})$ in place of the general (prior) noise distribution $p(\mathbf{N})$ and then updating the relevant structural assignment: $\mathcal{M}_{\text{cf}} = (\mathbf{S}_{\text{int}}, p(\mathbf{N} \mid \mathbf{x}^{(n)}))$. The resulting joint distribution $p(\mathbf{X} \mid \mathbf{x}^{(n)}, \mathrm{do}(X_k = \tilde{x}_k)) \equiv p(\mathbf{X}[\tilde{x}_k] \mid \mathbf{x}^{(n)})$ is called the counterfactual distribution.

We categorize causal queries by the required joint distribution to answer them: (i) an interventional query requires access to the interventional distribution and (ii) a counterfactual query requires access to the counterfactual distribution. The key difference between the two queries is that a counterfactual query requires the posterior distribution of noise variables $p(\mathbf{N} \mid \mathbf{x}^{(n)})$ before performing the intervention, while the interventional query uses the noise prior $p(\mathbf{N})$.

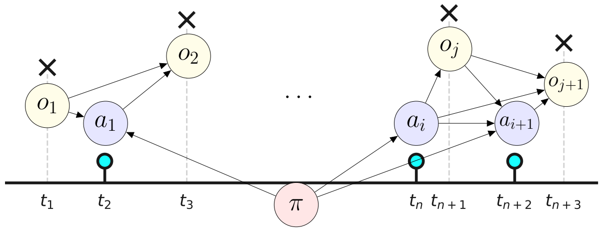


Figure 2: The causal graph $\mathcal{G}$ assumes a sequential treatment–outcome setup in continuous time, where past treatments and outcomes have a causal effect on future treatments and outcomes. The causal effect of the policy π on outcomes is fully mediated through treatments, i.e. there are no direct edges between the policy π and outcomes $\mathbf{o}$.

3. Problem Definition

Consider an observational data set $\mathcal{D}$:

$$\mathcal{D} = \{ \underbrace{\pi_{[0,T]}}_{\text{policy label}}, \underbrace{\{(t_i, m_i)\}_{i=1}^{N_a}}_{\text{treatments } \mathbf{a}}, \underbrace{\{(t_j, y_j)\}_{j=1}^{N_o}}_{\text{outcomes } \mathbf{o}} \}, \quad (1)$$

in a period $[0, T]$. For notational ease, the data set is defined for a single patient. Our model can be trivially generalized to multiple patients, under exchangeability.

A policy is a pair $(\pi_{[0,T]}, \lambda_\pi^*(t, m))$ that defines when and how treatments are applied, and is effectively determined by its treatment intensity function $\lambda_\pi^*(t, m)$. The policy label $\pi_{[0,T]}$ is assumed fixed for the period $[0, T]$ stated in the subscript, and it is assumed observed, while the corresponding intensity $\lambda_\pi^*(t, m)$ is unobserved. We assume that patients belong to different groups, and each group is characterized by a shared policy (treatment intensity). Background knowledge of having, e.g., different hospitals, countries, or environments in the observed data set implies distinct policies. In the absence of such information, each patient can be assumed to have their own policy.

A treatment tuple $a_i = (t_i, m_i)$ consists of an arrival time $t_i \in [0, T]$ and a treatment mark (dosage) $m_i \in \mathbb{R}$. An outcome tuple $o_j = (t_j, y_j)$ consists of a measurement time $t_j \in [0, T]$ and an outcome value $y_j \in \mathbb{R}$. Treatment times $\mathbf{t_a} = \{t_i\}_{i=1}^{N_a}$ and outcome times $\mathbf{t_o} = \{t_j\}_{j=1}^{N_o}$ are irregularly sampled points on the interval $[0, T]$. The history $\mathcal{H}_{\leq t} = \{\pi_{\leq t}, \mathbf{a}_{\leq t}, \mathbf{o}_{\leq t}\}$ contains the information about the past policy $\pi_{\leq t}$, past treatments $\mathbf{a}_{\leq t} = \{(t_i, m_i) : t_i \leq t\}$ and past outcomes $\mathbf{o}_{\leq t} = \{(t_j, y_j) : t_j \leq t\}$.

We observe a continuous-time process $\mathbf{Y}_{\leq T} = \{y(\tau) : \tau \leq T\}$ as outcome tuples $\mathbf{o}$ measured at times $\mathbf{t_o} = \{t_j\}_{j=1}^{N_o}$. To answer causal queries, we model the potential outcome trajectory $\mathbf{Y}_{>\tilde{\tau}}[\tilde{\pi}_{>\tilde{\tau}}]$, under an intervened policy specified by $\tilde{\pi}_{>\tilde{\tau}}$, where the subscript emphasizes the period when a policy intervention takes place. When the intervention time

$\tilde{\tau}$ is set to the end of the observation period $\tilde{\tau} = T$, we call the estimation task a *policy intervention*, as its computation requires access to the interventional distribution (Fig. 1(b)):

$$P(\mathbf{Y}_{>T}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}). \tag{2}$$

Also, we can set the intervention time $\tilde{\tau}$ to the start of the observation period $\tilde{\tau} = 0$ and consider a hypothetical scenario under an alternative treatment policy specified by $\tilde{\pi}_{[0,T]}$. We call this estimation task a *policy counterfactual*, as its computation requires access to the counterfactual distribution:

$$P(\mathbf{Y}_{[0,T]}[\tilde{\pi}_{[0,T]}] \mid \mathcal{H}_{\leq T}). \tag{3}$$

The difference between a policy intervention and a policy counterfactual is illustrated in Figs. 1 (b) and (c).

4. Causal Assumptions and Identifiability

In this section, we first show that a policy intervention query, formalized in Eq. (2), is equivalent to the potential outcome trajectory of a sequence of stochastic (conditional) interventions on treatments. Then, we show the latter is identifiable under causal assumptions, i.e., it can be estimated solely based on statistical terms (Pearl, 2009).

We assume a sequential treatment–outcome setup where past treatments and outcomes may affect future events (Robins, 1986; 1987). We add the policy label π to this setup, which has a direct causal effect on sequential treatments $\mathbf{a}$. The interplay between a policy label π, treatments $\mathbf{a}$, and outcomes $\mathbf{o}$ is illustrated in the causal graph $\mathcal{G}$ in Fig. 2. We make the standard causal assumptions of *consistency* (Assumption 1) and *no unobserved confounders* (NUC, Assumption 2), and furthermore the continuous-time NUC (Assumption 3, Schulam & Saria, 2017, see formal definitions in Appendix B). In words, the continuous-time NUC states that treatments are assigned stochastically in continuous time, conditioned on the past history. In addition, we assume that the causal effect of the policy label π on outcomes $\mathbf{o}$ is fully mediated through treatments $\mathbf{a}$, such that the causal graph $\mathcal{G}$ excludes direct edges from the policy label π to the outcomes $\mathbf{o}$, stated formally as:

Assumption 4: Fully-mediated policy effect. Conditioned on the past history $\mathcal{H}_{\leq \tau}$, the causal effect of the policy specified by $\tilde{\pi}_{>\tau}$ on the outcome trajectory $\mathbf{Y}_{>\tau}$ is fully mediated through sequential treatments $\tilde{\mathbf{a}}_{>\tau}$:

$$P(Y_{>\tau}[\tilde{\pi}_{>\tau}, \tilde{\mathbf{a}}_{>\tau}] \mid \mathcal{H}_{\leq \tau}) = P(Y_{>\tau}[\tilde{\mathbf{a}}_{>\tau}] \mid \mathcal{H}_{\leq \tau}).$$

Using Assumptions {1,2,3,4}, we inspect the potential outcome trajectory $\mathbf{Y}_{>T}[\tilde{\pi}_{>T}]$ at a discrete set of ordered query times $\mathbf{q} = \{q_1, \ldots, q_m\}$ (all $q_k > T$): $\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] = \{y(q_k)[\tilde{\pi}_{>T}] : q_k \in \mathbf{q}\}$ (Schulam & Saria, 2017). We factorize the outcome query $\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}]$ in temporal order, and develop Theorem 1 below to reformulate it as the sequential estimation of treatments and outcomes:

Theorem 1. Under Assumptions $\{1{,}2{,}3{,}4\}$, the potential outcome query $\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}]$ under a policy $\tilde{\pi}_{>T}$ is equivalent to the potential outcome query under a sequence of stochastic (conditional) interventions on treatments $\tilde{\mathbf{a}}_{>T}$:

$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\tilde{\mathbf{a}}_{>T}} \prod_{k=0}^{m-1} P(\tilde{\mathbf{a}}_{[q_k,q_{k+1})}[\tilde{\pi}_{>T}] | \mathcal{H}_{\leq q_k})\, P(Y_{q_{k+1}}[\tilde{\mathbf{a}}_{[q_k,q_{k+1})}] | \mathcal{H}_{\leq q_k}), \tag{4}$$

where $q_0 = T$, and $\tilde{\mathbf{a}}_{[q_k,q_{k+1})}$ denotes treatments in the interval $[q_k, q_{k+1})$ without outcome observations between consecutive query times. Here, the (whole) treatment sequence $\tilde{\mathbf{a}}_{>T}$ follows a distribution $p(\tilde{\mathbf{a}}_{>T}[\tilde{\pi}_{>T}])$, induced by the intervened policy label $\tilde{\pi}_{>T}$. The proof of Theorem 1 is given in Appendix C. Accordingly, the potential outcomes $\tilde{\mathbf{a}}_{[q_k,q_{k+1})}[\tilde{\pi}_{>T}]$ and $Y_{q_{k+1}}[\tilde{\mathbf{a}}_{[q_k,q_{k+1})}]$ can be identified under Assumptions $\{1{,}2{,}3\}$, using the following conditionals, both of which can be estimated with a statistical model (Schulam & Saria, 2017; Seedat et al., 2022):

$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\tilde{\mathbf{a}}_{>T}} \prod_{k=0}^{m-1} \underbrace{P(\tilde{\mathbf{a}}_{[q_k,q_{k+1})} | \tilde{\pi}_{>T}, \mathcal{H}_{\leq q_k})}_{\text{Treatment Term}} \underbrace{P(Y_{q_{k+1}} | \tilde{\mathbf{a}}_{[q_k,q_{k+1})}, \mathcal{H}_{\leq q_k})}_{\text{Outcome Term}}. \tag{5}$$

The identifiability result by Schulam & Saria (2017) was limited to a fixed sequence of treatments. Theorem 1 and the identifiability result in Eq. (5) generalize that result to a stochastic (conditional) intervention on a sequence of treatments, and one can recover the fixed sequence of treatments by assuming that the target density $p_{\tilde{\pi}}(\tilde{\mathbf{a}}_{>T})$ puts all probability on a single sequence of treatments. The identifiability of a policy counterfactual is more challenging and discussed in Section 6, since it requires functional assumptions related to the model definition (Section 5).

5. Joint Treatment–Outcome Model

To estimate both statistical terms in Eq. (5), (i) Treatment Term and (ii) Outcome Term, from observational data, we propose a joint treatment–outcome model, combining a marked point process and a conditional Gaussian process.

We encode conditional independence statements in the graph $\mathcal{G}$ by defining two dependent MPPs with conditional intensity functions: (i) treatment intensity for a treatment $a = (t, m)$: $\lambda^*_\pi(t, m) = \lambda^*_\pi(t) p^*(m \mid t)$ and (ii) outcome intensity for an outcome $o = (t, y)$: $\lambda^*_o(t, y) = \lambda^*_o(t) p^*(y \mid t)$. The past history $\mathcal{H}_{<t} = \{\pi_{<t}, \mathbf{a}_{<t}, \mathbf{o}_{<t}\}$ containing the information about the policy, past actions, and past outcomes is a valid history for both conditional intensity functions

$\lambda^*_\pi(t, m)$ and $\lambda^*_o(t, y)$. The joint distribution for the observational data set $\mathcal{D}$ in Eq. (1) can be written in terms of treatment and outcome intensity functions:

$$p(\mathcal{D}) = \prod_{i=1}^{I} \lambda^*_\pi(t_i, m_i) \prod_{j=1}^{J} \lambda^*_o(t_j, y_j) \exp(-\Lambda),$$

with the integral term $\Lambda = \int_{[0,T]} \big(\lambda^*_\pi(\tau) + \lambda^*_o(\tau)\big) \mathrm{d}\tau$.

We further assume that the measurement times of the outcomes $\mathbf{t_o} = \{t_j\}_{j=1}^{N_o}$ are given, which is valid for example when the data are collected through automated patient monitoring in healthcare. This assumption is equivalent to setting the outcome time intensity to an indicator function $\mathbb{1}_{\mathbf{t_o}}(t)$:[1] $\lambda^*_o(t, y) = \mathbb{1}_{\mathbf{t_o}}(t) p^*(y \mid t) = p^*(y \mid t)|_{t \in \mathbf{t_o}}$. Then, the joint distribution becomes

$$p(\mathcal{D}) = \prod_{i=1}^{I} \underbrace{\lambda^*_\pi(t_i) p^*_\pi(m_i \mid t_i)}_{\text{Treatment Intensity}} \prod_{j=1}^{J} \underbrace{p^*(y_j \mid t_j)}_{\text{Outcome Model}} |_{t_j \in \mathbf{t_o}}$$
$$\times \exp(-\textstyle\int_{[0,T]} \lambda^*_\pi(\tau) \mathrm{d}\tau). \quad (6)$$

5.1. Treatment Intensity

We consider the treatment time intensity $\lambda^*_\pi(\tau)$ as the output of a square transformation of a latent function $g^*_\pi(\tau)$, which we model as the sum of a constant baseline β_0 and three time-dependent functions with GP priors, $g_b, g^*_a, g^*_o \sim \mathcal{GP}$. We assume the patients who follow the same treatment policy π share the same treatment model, i.e., the same hyperparameters, but the intensity curves may differ because past treatments and outcomes are not the same.

The latent-state function $g_b(\tau)$ models the history-independent intensity. The regressive component $g^*_a(\tau)$ models the dependence on past treatments by taking the last Q_a treatments as input and the regressive component $g^*_o(\tau)$ models the dependence on past outcomes by taking the last Q_o outcomes as input. The treatment intensity $\lambda^*_\pi(\tau)$ is

$$\lambda^*_\pi(\tau) = \Big(\underbrace{\beta_0}_{\substack{\text{Poisson}\\\text{Baseline}}} + \underbrace{g_b(\tau)}_{\substack{\text{NHPP}\\\text{Baseline}}} + \underbrace{g^*_a(\tau; \mathbf{a})}_{\substack{\text{Treatment}\\\text{Effect}}} + \underbrace{g^*_o(\tau; \mathbf{o})}_{\substack{\text{Outcome}\\\text{Effect}}} \Big)^2. \quad (7)$$

The resulting treatment intensity extends the model proposed by Liu & Hauskrecht (2019) with (i) additional baseline components β_0 and $g_b(\tau)$ and (ii) the functional dependence on marks. Without regressive components $g^*_a(\tau)$ and $g^*_o(\tau)$, the treatment model becomes a non-homogeneous Poisson process (NHPP), whose instantaneous intensity is independent of past events: $\lambda(\tau) = (\beta_0 + g_b(\tau))^2$. If we further exclude the time-varying baseline $g_b(\tau)$, the model becomes a simple Poisson process with constant intensity. We model the treatment dosages (marks) by a GP-prior: $m(\tau) \sim \mathcal{GP}$. The model definition is detailed in Appendix E.

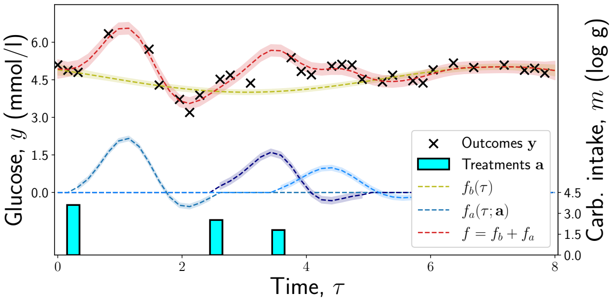


Figure 3: The outcome model f (red) is the sum of baseline f_b (yellow) and total response f_a, which adds up single treatment responses of different magnitude (3 blue lines).

5.2. Outcome Model

We model the outcome trajectory $\mathbf{Y} = \{y(\tau) : \tau \in \mathbb{R}_{\geq 0}\}$ over time τ by a conditional GP model, combining three independent components: (i) a baseline progression, (ii) a treatment response curve, and (iii) a noise variable (Schulam & Saria, 2017; Cheng et al., 2020; Zhang et al., 2020):

$$y(\tau) = \underbrace{f_b(\tau)}_{\text{Baseline}} + \underbrace{f_a(\tau; \mathbf{a})}_{\text{Treatment Response}} + \underbrace{\epsilon(\tau)}_{\text{Noise}}, \quad (8)$$

with independent Gaussian noise $\epsilon(\tau) \sim \mathcal{N}(0, \sigma^2_\epsilon)$.

Baseline Progression. This is modeled by a GP prior on $f_b(\tau) \sim \mathcal{GP}$. Cheng et al. (2020) proposed a sum of a squared exponential (SE) and a periodic kernel to model recurring patterns in heart rates. In our experiments, we model the baseline progression of blood glucose as a sum of a constant and a periodic kernel to capture daily blood glucose profiles of non-diabetic patients fluctuating around a constant baseline (Ashrafi et al., 2021). We assume that the outcome baseline f_b is patient-specific.

Treatment Response Model. We model the treatment responses as additive, similar to Cheng et al. (2020):

$$f_a(\tau; \mathbf{a}) = \sum_{a_i = (t_i, m_i) \in \mathbf{a}} f_m(m_i) f_t(\tau; t_i), \quad (9)$$

where the time-dependent response function $f_t \sim \mathcal{GP}$ represents the response 'shape' and is shared across patients, and $f_m(m_i)$ is a patient-specific linear function that scales the magnitude of the response for the given treatment dosage value $m_i \in \mathbb{R}$, as commonly done in the literature (Cheng et al., 2020; Zhang et al., 2020). This means the effects of nearby treatments simply sum up as shown in Fig. 3.

[1] The indicator function is: $\mathbb{1}_{\mathbf{t_o}}(t) = \begin{cases} 1, & \text{if } t \in \mathbf{t_o} \\ 0, & \text{otherwise} \end{cases}$

The scaling function f_m has patient-specific parameters that have hierarchical priors, to share information between patients. For the response shape function f_t we use an SE kernel, as it provides sufficient performance for our use-case (Cheng et al. (2020) also propose an extension with a latent force model (Alvarez et al., 2009)). The model definition is detailed in Appendix F.

6. Inference

Policy Intervention. To model the policy intervention, we first fit the joint model to the observational data using variational inference (see Appendix G), which gives us the (observational) SCM $\mathcal{M} = (\mathbf{S}, p(\mathbf{N}))$. An intervention $[\tilde{\pi}]$ on the policy results in an interventional SCM: $\mathcal{M}_{\text{int}} = (\mathbf{S}_{\text{int}}, p(\mathbf{N}))$. Following Theorem 1, this intervention is equivalent to a stochastic intervention on the sequence of treatments, where the treatments now follow the new policy $\tilde{\pi}$. In practice, this is modeled by replacing the observed treatment intensity $\lambda^*_\pi(\cdot)$ with the new interventional intensity $\lambda^*_{\tilde{\pi}}(\cdot)$, while keeping the outcome model the same.

Policy Counterfactual. The policy counterfactual is calculated in exactly the same way as the policy intervention, except that the generic noise term $p(\mathbf{N})$ is replaced by individual-specific noise $p(\mathbf{N} \mid \mathcal{D}^{(v)})$ that has been inferred from the observations for that individual (v): $\mathcal{M}_{\text{cf}} = (\mathbf{S}_{\text{int}}, p(\mathbf{N} \mid \mathcal{D}^{(v)}))$. In our model, the noise consists of two components, treatment and outcome noise: $\mathbf{N} = \{\mathcal{E}_a, \mathcal{E}_o\}$. The outcome noise $\mathcal{E}_o$ is defined in Eq. (8) and its posterior is available in closed form from the GP. The treatment noise $\mathcal{E}_a$ comprises noise associated with the sampling process of the treatments. In practice, we estimate the noise and condition the counterfactual sampling of the treatments with these estimates using a novel variant of the algorithm proposed by Noorbakhsh & Rodriguez (2022), which we have extended to history-dependent point processes (see Appendix H.2.2).

Identifiability of the policy counterfactual. The counterfactual trajectories involve two components: treatments and outcomes. The counterfactual *treatments* are identifiable, as our algorithm satisfies the monotonicity assumption (Pearl, 2009; Noorbakhsh & Rodriguez, 2022). On the other hand, the counterfactual *outcomes* are not guaranteed to be identifiable, because this would require identifiability of the baseline and treatment response separately, which is not guaranteed as they both are modeled non-parametrically. To address this, we use priors that encourage the baseline (slow-changing) and the response functions (fast-changing) to learn different aspects of the data, and we empirically show that in practice the response function $f_a(\tau)$ (Eq. (9)) and hence the counterfactual trajectory can be identified with good accuracy. The counterfactual algorithm and identifiability are further detailed in Appendix H.2.3.

7. Experiments

In this section, we validate that our model can estimate interventional and counterfactual outcome trajectories under policy interventions. The empirical validation is composed of two parts. First, on a *real-world observational dataset*, we show the proposed model can learn clinically meaningful treatment response curves and treatment intensities that can handle time-varying confounding. Second, we evaluate our model on two causal inference tasks: (i) the policy intervention and (ii) the policy counterfactual. To evaluate the performance on interventional and counterfactual predictions, we set up a *realistic semi-synthetic simulation* scenario, as the true causal effects are unknown for real-world observational data sets. To ensure a realistic simulation scenario, we use a subset of learned models from the real-world study as the ground-truth data generators. The study is reproducible and our implementation in GPflow (van der Wilk et al., 2020) can be found at `https://github.com/caglar-hizli/modeling-policy-interventions`.

7.1. Learning Treatment Intensities and Response Curves on Real-World Observational Data

We first demonstrate that our model can learn clinically meaningful treatment intensities and response curves from a real-world observational dataset on physiological dynamics of blood glucose (Zhang et al., 2020; Wyatt et al., 2021). The experimental setup is further detailed in Appendix I.1.

Real-world dataset. The dataset consists of treatment–outcome measurements of 14 non-diabetic patients over a 3-day period (Zhang et al., 2020), corresponding to meals (treatments) and blood glucose measurements (outcomes). Patients record their meals in a meal diary and their blood glucose is measured by a monitoring device at regular time intervals. Meal ingredients are transformed into five nutrients: sugar, starch, protein, fiber, and fat. As meal dosage (mark), we preprocess nutrient covariates and use a single carbohydrate intake covariate equal to the sum of sugar and starch values, following Ashrafi et al. (2021).

Causal Assumptions. In the real-world dataset, the causal assumptions do **not** (exactly) hold and are not statistically testable. This is why (i) in this section, we perform only a *statistical* analysis to show how our joint model works, and (ii) in Section 7.2, we perform a *causal* analysis on a semi-synthetic simulation study.

Assumption 2 states that an interventional meal has non-zero probability, which is likely to hold since we assume each patient follows their own diet (policy) and one patient's diet includes reasonable meal events that would have non-zero probability for the others. Assumption 3 does *not* hold, since there can be unobserved processes, e.g., physical

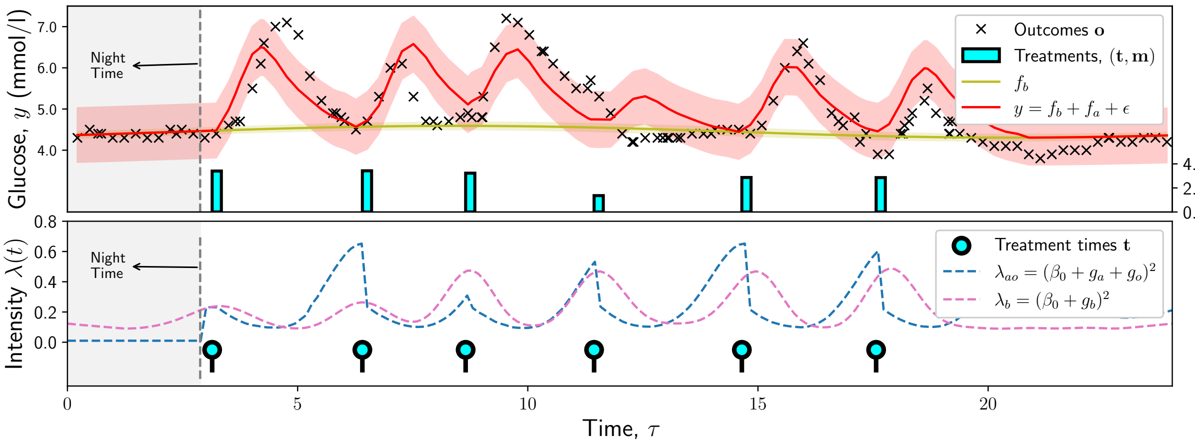


Figure 4: Estimated treatment–outcome model on the real-world meal–glucose data. **Top:** Glucose (outcome) measurements (black crosses), inferred baseline (yellow), and predicted glucose levels obtained by adding treatment responses, the baseline, and the noise (red). The confidence intervals around the means refer to ±1 standard deviation. We see that the increase in blood glucose is correctly predicted by the outcome model. **Bottom:** Treatment times and estimated intensities for the best model λ^*_{ao} and the simple baseline λ_b. The baseline treatment model has a large meal intensity at times of the day when the patient usually eats. In addition, in the best treatment model, λ^*_{ao}, we see that the intensity of a new meal immediately after a previous meal decreases either directly through g^*_a or indirectly through the increase in blood glucose represented by g^*_o. The intensities of the other models are shown in Appendix I.1.2.

activity, which influence both meal times and daily glucose profiles. However, the influence of confounding variables on blood glucose, apart from meals, is expected to be minor for non-diabetic individuals, as blood glucose is typically steady between meals (Ashrafi et al., 2021). Assumption 4 states that the policy effect is fully mediated through meals. For example, a policy that satisfies this is a dietary policy that dictates how the patient eats and in that way affects the blood glucose, but does not affect blood glucose in any other direct or indirect way. In Section 7.2, we perform a causal analysis under such dietary policy interventions. As an example that violates this assumption, we could consider a more generic lifestyle intervention that includes aspects other than the diet, such as physical activity.

Model. The sequential meal–blood glucose data of a patient is modelled jointly, as detailed in Section 5. To show how the treatment intensity $\lambda^*(\tau)$ handles the time-varying confounding due to previous treatments and outcomes, we define five treatment intensities $\{\lambda_b, \lambda^*_{ba}, \lambda^*_{bo}, \lambda^*_{ao}, \lambda^*_{bao}\}$ for comparison. In the most complex model λ^*_{bao}, the intensity of treatments depends on the baseline (b), previous treatments (a for 'action'), and previous outcomes (o); the simpler models have only some of these components as specified in their respective subscripts, see the functional forms in Table 1.

Results. We show an example fit of the joint model for one patient in Fig. 4. The treatment- and outcome-dependent intensity λ^*_{ao} (blue dashed line) correctly estimates that the probability of a meal event decreases (i) right after a new meal or (ii) with increasing blood glucose. This empirical finding suggests that functions g^*_a and g^*_o are useful in handling time-varying confounding due to past meal and glucose values.

Table 1: Comparison of models for real-world data. The last column shows the test log-likelihood (TLL, higher is better; mean ± standard deviation) for the different models. Models are trained on the first 2 days of meal–glucose data. Day 3 is used to compute the TLL.

INTENSITY	COMPONENTS	TLL ↑
λ_b	$\big(\beta_0 + g_b(\tau)\big)^2$	-13.3 ± 0.5
λ^*_{ba}	$\big(\beta_0 + g_b(\tau) + g^*_a(\tau; \mathbf{a})\big)^2$	-12.2 ± 0.3
λ^*_{bo}	$\big(\beta_0 + g_b(\tau) + g^*_o(\tau; \mathbf{o})\big)^2$	-11.0 ± 0.3
λ^*_{ao}	$\big(\beta_0 + g^*_a(\tau; \mathbf{a}) + g^*_o(\tau; \mathbf{o})\big)^2$	-10.7 ± 0.4
λ^*_{bao}	$\big(\beta_0 + g_b(\tau) + g^*_a(\tau; \mathbf{a}) + g^*_o(\tau; \mathbf{o})\big)^2$	-12.5 ± 0.4

We compare the different models in terms of the test log likelihood (TLL) for 14 patients in Table 1. We see that the simple history-independent intensity $\lambda_b(\tau)$ produces the lowest TLL. The treatment- and outcome-dependent intensity $\lambda^*_{ao}(\tau)$ produces the highest TLL mean, and slightly better results than the more complex intensity $\lambda^*_{bao}(\tau)$.

Table 2: Causal tasks on semi-synthetic data: The *observed* policy is $\pi_{[0,1d]} = \pi_A$, and we report mean squared error (MSE, lower is better) for observational $[\tilde{\pi}_{>1d} = \pi_A]$ and interventional $[\tilde{\pi}_{>1d} = \pi_B]$ queries as well as for the policy counterfactual $[\tilde{\pi}_{[0,1d]} = \pi_B]$. The best results are **bolded**. Methods developed by us are highlighted in **blue**.

		Query (MSE ↓)			Query (MSE ↓)	
Treatment Model	Response Model	$\mathcal{M}_{\text{SCM}}$	Observational $[\tilde{\pi}_{>1d} = \pi_A]$	Interventional $[\tilde{\pi}_{>1d} = \pi_B]$	$\mathcal{M}_{\text{SCM}}$	Counterfactual $[\tilde{\pi}_{[0,1d]} = \pi_B]$
Gamma (H21)	Param. (H21)	H21_{int}	0.50 ± 0.02	0.50 ± 0.02	H21_{cf}	0.49 ± 0.02
Gamma (H21)	**GP**	AB1_{int}	0.30 ± 0.02	0.30 ± 0.01	AB1_{cf}	0.21 ± 0.01
NHPP (L15)	Param. (S17)	AB2_{int}	0.97 ± 0.03	1.00 ± 0.03	AB2_{cf}	0.59 ± 0.02
NHPP (L15)	**GP**	AB3_{int}	0.34 ± 0.02	0.32 ± 0.01	AB3_{cf}	0.23 ± 0.01
GP-PP	Param. (H21)	AB4_{int}	0.46 ± 0.01	0.46 ± 0.01	AB4_{cf}	0.45 ± 0.02
GP-PP	Param. (S17)	AB5_{int}	0.96 ± 0.03	1.00 ± 0.03	AB5_{cf}	0.56 ± 0.02
GP-PP	**GP**	**OUR**$_{\text{obs}}$	$\mathbf{0.19 \pm 0.01}$	0.52 ± 0.02	**OUR**$_{\text{obs}}$	0.82 ± 0.03
GP-PP	**GP**	**OUR**$_{\text{int}}$	$\mathbf{0.19 \pm 0.01}$	$\mathbf{0.20 \pm 0.01}$	**OUR**$_{\text{int}}$	0.71 ± 0.03
GP-PP	**GP**	**OUR**$_{\text{cf}}$	$\mathbf{0.19 \pm 0.01}$	$\mathbf{0.20 \pm 0.01}$	**OUR**$_{\text{cf}}$	$\mathbf{0.14 \pm 0.01}$

7.2. Causal Tasks on Semi-Synthetic Data

In this section, we set up a semi-synthetic simulation study to demonstrate the ability of the proposed model to answer interventional and counterfactual queries resulting from an intervention on the treatment policy. Due to space constraints, the experimental setup is detailed in Appendix I.2.3.

Simulator. We obtain the ground-truth simulator by fitting our joint model to the meal–glucose data set, as discussed in Section 7.1. To be able to assess the sequential causal inference tasks, we have to include both sources of time-varying confounding. Therefore, we select as the ground-truth meal simulator the intensity that depends on both meal and glucose histories: $\lambda^*_\pi \sim \lambda^*_{ao}$.

Semi-synthetic dataset. The ground-truth simulator is used to simulate samples from observational, interventional and counterfactual distributions of each patient (v). Simulated patients are divided into (i) two policy groups $\{\pi_A, \pi_B\}$ representing different diets and (ii) three patient groups with distinct baseline and response functions, enabling individualization among patients. The meal simulators of two diets correspond to learned intensity functions of two real-world patients. The glucose simulators of three groups are learned from three real-world patients.

For the observational data set, we simulate 1-day long meal–glucose time-series for 50 patients: $\mathcal{D} = \{\mathcal{D}^{(v)}\}_{v=1}^{50}$. For the interventional data set, we sample the next day of each patient under the specified policy intervention $[\tilde{\pi}_{>1d}]$. For the counterfactual data set, we condition on the observed data $\mathcal{D}^{(v)}$ for each patient and sample a hypothetical first day under a policy intervention $[\tilde{\pi}_{[0,1d]}]$, using the noise posteriors. We train all models on the observational dataset, and use interventional and counterfactual datasets for testing.

Benchmarks. First, we compare the proposed model with a parametric joint model (H21, Hua et al., 2021), which uses a renewal process with a parametric Gamma-shaped trigger function as the treatment model, and a linear mixed effects function as the outcome model. Next, we create ablations with different treatment and outcome model combinations, since other existing methods do not model treatments. With ablations having different treatment intensities, we measure the impact of having a flexible treatment intensity that can handle complex time-varying confounding scenarios. With ablations having different outcome models, we measure the impact of predicting the glucose trajectory well. The ablation AB1 combines the Gamma-based treatment intensity with our GP-response model. Ablations AB2 and AB3 combine a history-independent treatment model (NHPP, Lloyd et al., 2015, (L15)) with (i) a parametric response model proposed by Schulam & Saria (2017, (S17)) and (ii) our GP-response model. Ablations AB4 and AB5 combine our flexible treatment intensity (GP-PP) with two parametric response models proposed by Hua et al. (2021, (H21)) and Schulam & Saria (2017, (S17)).

The columns $\mathcal{M}_{\text{SCM}}$ in Table 2 specify the combination of (i) a joint model identifier $\mathcal{M}$, e.g., OUR, and (ii) the target distribution of the SCM stated as a subscript: $(\cdot)_{\text{obs}}$ for observational, $(\cdot)_{\text{int}}$ for interventional, and $(\cdot)_{\text{cf}}$ for counterfactual. Similar to Schulam & Saria (2017), we use our own observational model (OUR_{obs}) as another baseline in the interventional task. In the counterfactual task, we use both our observational (OUR_{obs}) and interventional model (OUR_{int}) as baselines. Our counterfactual model is denoted by OUR_{cf}. To give our baseline and ablation models the most strength, we use their interventional version, $(\cdot)_{\text{int}}$, for estimating the interventional query and their counterfactual version, $(\cdot)_{\text{cf}}$, for estimating the counterfactual query.

Metric. We report the mean squared error (MSE) between ground-truth and estimated glucose trajectories under the interventional and counterfactual distributions, over all patients and all measurement times. To obtain estimated trajectories comparable to the ground-truth trajectories, we fix the noise variables of the point process (meal) sampling algorithm between the ground-truth simulator and each estimated model. Each experiment is repeated 10 times. We report mean and standard deviation of the MSE metric.

Policy Intervention. We estimate the interventional query: What will happen if patient (v) with the diet $\pi^{(v)} = \pi_A$ will continue for the next day in the future, adopting another patient's diet π_B? Following Eq. (2), the causal query for this task is formalized as the interventional query:

$$P(\mathbf{Y}^{(v)}_{>1d}[\tilde{\pi}_{>1d} = \pi_B] \mid \pi_{[0,1d]} = \pi_A, \mathcal{H}^{(v)}_{\leq 1d}).$$

MSE results are shown in Table 2. We see that the interventional model OUR$_{\text{int}}$ is able to estimate observational and interventional trajectories with low error when the intervention policy is (i) the same as the observed policy $[\tilde{\pi}_{>1d} = \pi_A]$ and (ii) different from the observed policy $[\tilde{\pi}_{>1d} = \pi_B]$, while the observational model OUR$_{\text{obs}}$ fails in the latter case. This is because OUR$_{\text{obs}}$ can only answer a statistical query of the following form: Observing the history of patient (v), how will the patient state progress for the next day, $P(\mathbf{Y}^{(v)}_{>1d} \mid \pi_{[0,1d]} = \pi_A, \mathcal{H}^{(v)}_{\leq 1d})$? Furthermore, our empirical findings suggest that simplistic models with parametric glucose responses (Param.(S17,H21)) perform badly, even when they target the interventional distribution.

Policy Counterfactual. We estimate the counterfactual query: What *would have* happened if patient (v) with the diet $\pi^{(v)} = \pi_A$ had followed another patient's diet π_B in the observation period $[0, 1d]$? Following Eq. (3), the causal query for this task is formalized as the counterfactual

$$P(\mathbf{Y}^{(v)}_{\leq 1d}[\tilde{\pi}_{[0,1d]} = \pi_B] \mid \pi_{[0,1d]} = \pi_A, \mathcal{H}^{(v)}_{\leq 1d}).$$

MSE results are shown in the last column of Table 2. We see that the interventional model OUR$_{\text{int}}$ fails to estimate counterfactual trajectories, as it does not take into account the noise posterior of each patient. On the other hand, the counterfactual model OUR$_{\text{cf}}$ accurately samples counterfactual trajectories and has the smallest error of the methods considered.

8. Discussion

To study what happens if the (possibly implicit) treatment policy of one individual (hospital, unit, country, ...) is or had been adopted by another individual, we proposed a model that jointly considers sequences of treatments and outcomes of each individual. Theoretically, we showed that an intervention on a treatment policy is equivalent to a sequence of stochastic interventions on treatments, whose potential outcomes can be estimated from observational data with the joint model. In a real-world scenario, we demonstrated that our non-parametric model can be learned from observational sequences of treatments and outcomes, and it can handle time-varying confounding in continuous time. In a semi-synthetic experiment, we demonstrated that the joint model can answer causal queries about the interventional and counterfactual distributions of the outcome after an intervention on the treatment policy.

8.1. Limitations and Future Work

Modeling Limitations. The main limitation of the proposed outcome model is scalability, as it is not trivial to use inducing point approximations for the current 'time-marked' treatment response function. This could be improved in future work, e.g., by clever inducing point selections. Furthermore, we considered treatment responses to be additive, which does not allow for modeling interactions of nearby treatments. This could be improved by integrating more sophisticated kernels into the treatment response function.

The thinning algorithm for sampling the point process is sequential and known to be slow, which can raise a challenge in using our method for downstream tasks such as optimizing the policy by a model-based reinforcement learning algorithm. It could be possible to adapt an inverse sampling method with $\mathcal{O}(1)$ time complexity (Rasmussen, 2011), but this would require a completely different treatment model definition as in Shchur et al. (2020), whose applicability would need to be studied separately.

Causal Assumptions. For real-world observational datasets, our causal assumptions do not necessarily hold, in which case our method should not be used for *causal* analysis (the model may still be useful for statistical analysis). For example, the violation of the NUC assumption would result in an error in the causal effect estimates due to the unblocked back-door paths, whose magnitude depends on the size of the violation. An interesting future direction is to extend existing sensitivity analysis methods (Robins et al., 2000b) in order to estimate the size of the error in causal effects of continuous-time treatment sequences.

Acknowledgements

The authors would like to thank Çağatay Yıldız for fruitful discussions. This work was supported by the Academy of Finland (Flagship programme: Finnish Center for Artificial Intelligence FCAI, and grants 336033, 352986) and EU (H2020 grant 101016775 and NextGenerationEU).

References

Aalen, O. O., Stensrud, M. J., Didelez, V., Daniel, R., Røysland, K., and Strohmaier, S. Time-dependent mediators in survival analysis: Modeling direct and indirect effects with the additive hazards model. *Biometrical Jour-*

nal, 62(3):532–549, 2020.

Abid, A., Yuksekgonul, M., and Zou, J. Meaningfully debugging model mistakes using conceptual counterfactual explanations. In *International Conference on Machine Learning*, pp. 66–88, 2022.

Adams, R. P., Murray, I., and MacKay, D. J. Tractable nonparametric bayesian inference in poisson processes with gaussian process intensities. In *International Conference on Machine Learning*, pp. 9–16, 2009.

Alvarez, M., Luengo, D., and Lawrence, N. D. Latent force models. In *Artificial Intelligence and Statistics*, pp. 9–16, 2009.

Ashrafi, R. A., Ahola, A. J., Rosengård-Bärlund, M., Saarinen, T., Heinonen, S., Juuti, A., Marttinen, P., and Pietiläinen, K. H. Computational modelling of self-reported dietary carbohydrate intake on glucose concentrations in patients undergoing roux-en-y gastric bypass versus one-anastomosis gastric bypass. *Annals of Medicine*, 53(1):1885–1895, 2021.

Bica, I., Alaa, A. M., Jordon, J., and van der Schaar, M. Estimating counterfactual treatment outcomes over time through adversarially balanced representations. In *International Conference on Learning Representations*, 2020.

Bica, I., Alaa, A. M., Lambert, C., and Van Der Schaar, M. From real-world patient data to individualized treatment effects using machine learning: current and future methods to address underlying challenges. *Clinical Pharmacology & Therapeutics*, 109(1):87–100, 2021.

Cheng, L.-F., Dumitrascu, B., Zhang, M., Chivers, C., Draugelis, M., Li, K., and Engelhardt, B. Patient-specific effects of medication using latent force models with Gaussian processes. In *Artificial Intelligence and Statistics*, pp. 4045–4055, 2020.

Cunningham, J., Ghahramani, Z., and Rasmussen, C. Gaussian processes for time-marked time-series data. In *Artificial Intelligence and Statistics*, pp. 255–263, 2012.

Daley, D. J. and Vere-Jones, D. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

Epstude, K. and Roese, N. J. The functional theory of counterfactual thinking. *Personality and Social Psychology Review*, 12(2):168–192, 2008.

Gao, T., Subramanian, D., Bhattacharjya, D., Shou, X., Mattei, N., and Bennett, K. P. Causal inference for event pairs in multivariate point processes. *Advances in Neural Information Processing Systems*, 34:17311–17324, 2021.

Hawkes, A. G. Hawkes processes and their applications to finance: a review. *Quantitative Finance*, 18(2):193–198, 2018.

Hernán, M. A. and Robins, J. M. Causal inference, 2010.

Hua, W., Mei, H., Zohar, S., Giral, M., and Xu, Y. Personalized dynamic treatment regimes in continuous time: A Bayesian approach for optimizing clinical decisions with timing. *Bayesian Analysis*, 1(1):1–30, 2021.

John, S. and Hensman, J. Large-scale Cox process inference using variational Fourier features. In *International Conference on Machine Learning*, pp. 2362–2370, 2018.

Lewis, P. W. and Shedler, G. S. Simulation of nonhomogeneous poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979.

Lim, B., Alaa, A. M., and van der Schaar, M. Forecasting treatment responses over time using recurrent marginal structural networks. *Advances in Neural Information Processing Systems*, 31:7494–7504, 2018.

Liu, S. and Hauskrecht, M. Nonparametric regressive point processes based on conditional Gaussian processes. In *Advances in Neural Information Processing Systems*, volume 32, pp. 1064–1074, 2019.

Lloyd, C., Gunter, T., Osborne, M., and Roberts, S. Variational inference for Gaussian process modulated Poisson processes. In *International Conference on Machine Learning*, pp. 1814–1822, 2015.

Lok, J. J. Statistical modeling of causal effects in continuous time. *The Annals of Statistics*, 36(3):1464–1507, 2008.

Matthews, A. G. d. G., Hensman, J., Turner, R., and Ghahramani, Z. On sparse variational methods and the Kullback-Leibler divergence between stochastic processes. In *Artificial Intelligence and Statistics*, pp. 231–239, 2016.

Møller, J., Syversveen, A. R., and Waagepetersen, R. P. Log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 25(3):451–482, 1998.

Neyman, J. Sur les applications de la théorie des probabilités aux experiences agricoles: Essai des principes. *Roczniki Nauk Rolniczych*, 10:1–51, 1923.

Noorbakhsh, K. and Rodriguez, M. G. Counterfactual temporal point processes. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Oberst, M. and Sontag, D. Counterfactual off-policy evaluation with Gumbel-max structural causal models. In *International Conference on Machine Learning*, pp. 4881–4890, 2019.

Ogata, Y. On Lewis' simulation method for point processes. *IEEE Transactions on Information Theory*, 27(1):23–31, 1981.

Pearl, J. *Causality*. Cambridge University Press, 2009.

Pearl, J. and Robins, J. M. Probabilistic evaluation of sequential plans from causal models with hidden variables. In *Uncertainty in Artificial Intelligence*, volume 95, pp. 444–453, 1995.

Rasmussen, J. G. Temporal point processes: the conditional intensity function. *arXiv preprint arXiv:1806.00221*, 2011.

Robins, J. A new approach to causal inference in mortality studies with a sustained exposure period—application to control of the healthy worker survivor effect. *Mathematical Modelling*, 7(9-12):1393–1512, 1986.

Robins, J. Addendum to "a new approach to causal inference in mortality studies with sustained exposure periods—application to control of the healthy worker survivor effect". *Computers and Mathematics with Applications*, 14(9-12):923–945, 1987.

Robins, J. Estimation of the time-dependent accelerated failure time model in the presence of confounding factors. *Biometrika*, 79(2):321–334, 1992.

Robins, J., Hernan, M. A., and Brumback, B. Marginal structural models and causal inference in epidemiology, 2000a.

Robins, J. M. and Hernán, M. A. Estimation of the causal effects of time-varying exposures. *Longitudinal Data Analysis*, 553:599, 2009.

Robins, J. M., Rotnitzky, A., and Scharfstein, D. O. Sensitivity analysis for selection bias and unmeasured confounding in missing data and causal inference models. *IMA Volumes In Mathematics and Its Applications*, 116: 1–94, 2000b.

Rubin, D. B. Bayesian inference for causal effects: The role of randomization. *The Annals of Statistics*, pp. 34–58, 1978.

Ryalen, P. C., Stensrud, M. J., Fosså, S., and Røysland, K. Causal inference in continuous time: an example on prostate cancer therapy. *Biostatistics*, 21(1):172–185, 2020.

Schulam, P. and Saria, S. A framework for individualizing predictions of disease trajectories by exploiting multi-resolution structure. *Advances in Neural Information Processing Systems*, 28:748–756, 2015.

Schulam, P. and Saria, S. Reliable decision support using counterfactual models. *Advances in Neural Information Processing Systems*, 30:1696–1706, 2017.

Seedat, N., Imrie, F., Bellot, A., Qian, Z., and van der Schaar, M. Continuous-time modeling of counterfactual outcomes using neural controlled differential equations. In *International Conference on Machine Learning*, 2022.

Shchur, O., Gao, N., Bilos, M., and Günnemann, S. Fast and flexible temporal point processes with triangular maps. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Soleimani, H., Subbaswamy, A., and Saria, S. Treatment-response models for counterfactual reasoning with continuous-time, continuous-valued interventions. In *Uncertainty in Artificial Intelligence*, 2017.

Taubman, S. L., Robins, J. M., Mittleman, M. A., and Hernán, M. A. Intervening on risk factors for coronary heart disease: an application of the parametric g-formula. *International journal of epidemiology*, 38(6):1599–1611, 2009.

Tsirtsis, S. and Gomez Rodriguez, M. Decisions, counterfactual explanations and strategic behavior. *Advances in Neural Information Processing Systems*, 33:16749–16760, 2020.

Tsirtsis, S., De, A., and Rodriguez, M. Counterfactual explanations in sequential decision making under uncertainty. *Advances in Neural Information Processing Systems*, 34: 30127–30139, 2021.

van der Wilk, M., Dutordoir, V., John, S., Artemev, A., Adam, V., and Hensman, J. A framework for interdomain and multioutput Gaussian processes. *arXiv:2003.01115*, 2020. URL `https://arxiv.org/abs/2003.01115`.

Williams, C. K. and Rasmussen, C. E. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Wyatt, P., Berry, S. E., Finlayson, G., O'Driscoll, R., Hadjigeorgiou, G., Drew, D. A., Khatib, H. A., Nguyen, L. H., Linenberg, I., Chan, A. T., et al. Postprandial glycaemic dips predict appetite and energy intake in healthy individuals. *Nature Metabolism*, 3(4):523–529, 2021.

Xu, Y., Xu, Y., and Saria, S. A Bayesian nonparametric approach for estimating individualized treatment-response curves. In *Machine Learning for Healthcare Conference*, pp. 282–300, 2016.

Zhang, G., Ashrafi, R. A., Juuti, A., Pietiläinen, K., and Marttinen, P. Errors-in-variables modeling of personalized treatment-response trajectories. *IEEE Journal of Biomedical and Health Informatics*, 25(1):201–208, 2020.

Zhang, Y., Young, J. G., Thamer, M., and Hernán, M. A. Comparing the effectiveness of dynamic treatment strategies using electronic health records: an application of the parametric g-formula to anemia management strategies. *Health Services Research*, 53(3):1900–1918, 2018.

Table 3: Taxonomy of the related work, with respect to treatment model type, outcome model type, continuity of time-steps and their target distributions: observational (Obs.), interventional (Int.) and counterfactual (Cf.).

			Target Distribution			
Treatment Model	Response Model	Continuous-time	Obs.	Int.	Cf.	Reference
✗	Parametric	✓	✓	✓	✗	Schulam & Saria (2017)
✗	RNN	✗	✓	✓	✗	Lim et al. (2018)
✗	RNN	✗	✓	✓	✗	Bica et al. (2020)
✗	Neural CDE	✓	✓	✓	✗	Seedat et al. (2022)
✗	NHPP	✓	✓	✓	✓	Noorbakhsh & Rodriguez (2022)
Gamma	Parametric	✓	✓	✗	✗	Hua et al. (2021)
GP-PP	Non-parametric	✓	✓	✓	✓	Our Work

A. Further Related Work

Treatment Responses for Sequential Treatment–Outcome Data. Estimating sequential treatment effects has been studied by a large number of works (Schulam & Saria, 2017; Xu et al., 2016; Soleimani et al., 2017; Lim et al., 2018; Bica et al., 2020; Zhang et al., 2020; Cheng et al., 2020; Seedat et al., 2022). Recurrent marginal structural networks (RMSN, Lim et al., 2018) extend marginal structural models (MSM, Robins et al., 2000a) using a recurrent neural network (RNN) architecture to estimate propensity weights and treatment effects over time. The counterfactual recurrent network (CRN, Bica et al., 2020) also uses an RNN architecture, but extends the previous work by using domain adversarial training instead of estimating the propensity weights explicitly. Both works focus on a *discrete-time* sequential treatment–outcome setup.

For *continuous-time* treatments and outcomes, a number of works have proposed generative models over the outcome trajectory (Schulam & Saria, 2015; Xu et al., 2016; Soleimani et al., 2017; Zhang et al., 2020; Cheng et al., 2020), in the direction of the g-computation method (Robins, 1986; 1987). These works model the outcome trajectory as a sum of (i) a counterfactual baseline that captures the no-treatment case and (ii) an additive treatment response function. Most of these methods use a Bayesian non-parametric baseline and a parametric treatment response (Schulam & Saria, 2015; Xu et al., 2016; Soleimani et al., 2017; Zhang et al., 2020). In our work, we combine a non-parametric baseline with a more flexible, non-parametric treatment response function, similar to Cheng et al. (2020). In addition to these continuous-time methods, Seedat et al. (2022) estimate continuous-time treatment responses, by combining (i) neural controlled differential equations (Neural CDE) to model the latent outcome trajectory, and (ii) domain adversarial training to deal with the time-varying confounding. However, none of these methods model treatments, so they require a fixed sequence of treatment interventions as *input* at test time. Besides, they do not consider counterfactuals of observed time-series.

Treatment (Event) Models in Continuous Time. Continuous-time treatments can be modeled by temporal point processes (TPP), which have been investigated from a causal inference perspective by a large body of work (Lok, 2008; Schulam & Saria, 2017; Gao et al., 2021; Aalen et al., 2020; Ryalen et al., 2020; Hua et al., 2021). Lok (2008) estimates the causal effect of interventions on continuous-time treatments, by extending structural nested models (Robins, 1992) using a martingale approach. The methods proposed by Aalen et al. (2020) and Ryalen et al. (2020) consider interventions affecting continuous-time events on a survival outcome, where the TPP terminates after a single survival/death event. Gao et al. (2021) investigate the average treatment effect between pairs of event variables of a multivariate point process. Similar to our work, Schulam & Saria (2017) estimate interventional outcome (mark) trajectories affected by continuous-time treatments; however, they assume a fixed sequence of treatment interventions as input. Instead, we model continuous-time treatments jointly with the outcome trajectories. This enables our model to make predictions under more realistic, alternative scenarios.

From a modeling perspective, the model proposed in Hua et al. (2021) is the closest to our work, as they propose a Bayesian joint model for the sequences of treatments and outcomes. However, their goal is to find an optimal treatment policy for continuous-time treatments, while our goal is to estimate interventional and counterfactual outcome trajectories resulting from policy interventions. Besides, they use parametric models for both treatment and outcome models, heavily-tailored for their kidney transplantation application, which do not generalize trivially to other problem setups.

Counterfactual Treatments. Counterfactual reasoning has recently raised interest in the machine learning literature, where it has been used for evaluating and explaining model predictions (Oberst & Sontag, 2019; Tsirtsis & Gomez Rodriguez, 2020; Tsirtsis et al., 2021; Abid et al., 2022; Aalen et al., 2020; Ryalen et al., 2020; Noorbakhsh & Rodriguez, 2022). Most of these work focus on discrete-time treatment–outcome setups (Oberst & Sontag, 2019; Tsirtsis & Gomez Rodriguez, 2020; Tsirtsis et al., 2021; Abid et al., 2022). Aalen et al. (2020) and Ryalen et al. (2020) consider counterfactuals of a continuous-time survival outcome as discussed in the previous paragraph. From the perspective of counterfactual treatments, the closest to our work is counterfactual TPPs proposed in Noorbakhsh & Rodriguez (2022). However, Noorbakhsh & Rodriguez (2022) consider continuous-time event occurrences as target outcomes, while our main target is an outcome (mark) trajectory that is causally affected by a treatment policy intervention through continuous-time treatment events. Besides, the counterfactual sampling algorithm proposed in Noorbakhsh & Rodriguez (2022) is limited to non-homogeneous Poisson processes (NHPP) (independent events) or their Hawkes process variants since they choose Lewis' thinning algorithm (Lewis & Shedler, 1979) as the generative process. This limits the capacity of the algorithm to handle time-varying confounding in a sequential treatment–outcome setup. Therefore, we extend their counterfactual sampling algorithm to history-dependent point processes, by choosing Ogata's thinning algorithm (Ogata, 1981) as the generative process.

A taxonomy of the related work is presented in Table 3, which classifies the closest of the aforementioned works in terms of the treatment model type, the outcome model type, the continuity of time-steps and their target distributions: observational (Obs.), interventional (Int.) and counterfactual (Cf.).

B. Causal Assumptions

Assume a structural causal model (SCM) with an intervention variable $A \in \mathcal{A}$, outcome variable $Y \in \mathbb{R}$ and all other variables $Z \in \mathcal{Z}$.

B.1. Intervention Types

In practice, an intervention on a variable A removes the structural equation $A := f_a(\mathrm{pa}(A), U_a)$. Intervention types differ in terms of the way they set the intervention value $do(A)$.

Atomic intervention. Sets A to a static value $\tilde{a}$. Under identifiability conditions, the effect of the intervention on the outcome variable is as follows:

$$p(Y \mid do(A = \tilde{a})) = \sum_z p(Y \mid A = \tilde{a}, Z = z) p(Z = z).$$

Conditional intervention. Sets A by a deterministic function $\tilde{h} : \mathcal{Z} \to \mathcal{A}$. Under identifiability conditions, the effect of the intervention on the outcome variable is as follows:

$$\begin{aligned} p(Y \mid do(A = \tilde{h}(z))) &= \sum_z p(Y \mid do(A = \tilde{h}(z)), Z = z) p(Z = z \mid do(A = \tilde{h}(z))) \\ &= \sum_z p(Y \mid a, z)|_{a=\tilde{h}(z)} p(z). \end{aligned}$$

Stochastic (conditional) intervention. Sets A by a stochastic relationship $\tilde{p}(A \mid Z)$. Given Z, the intervention $do(A = \tilde{a})$ occurs with probability $\tilde{p}(A \mid Z)$. Therefore, the effect of an atomic intervention $do(A = \tilde{a})$ is averaged over all possible $\tilde{a} \sim \tilde{p}(A|Z)$:

$$\begin{aligned} p(Y \mid & do(A \sim \tilde{p}(A \mid Z))) \\ &= \sum_{\tilde{a}} \sum_z p(Y \mid do(A = \tilde{a}), z = z) \tilde{p}(A = \tilde{a} \mid Z = z) p(Z = z) \\ &= \sum_{\tilde{a}} \sum_z p(Y \mid \tilde{a}, z) \tilde{p}(\tilde{a} \mid z) p(z). \end{aligned}$$

B.2. Formal Definitions of Causal Assumptions

Consistency, no-unmeasured confounding (NUC) and positivity are standard causal assumptions required for the identifiability of a static causal effect (Pearl, 2009; Hernán & Robins, 2010; Schulam & Saria, 2017; Bica et al., 2020).

Assumption 1: Consistency. The potential outcome $Y[a]$ under an action $a \in \mathcal{A}$ is consistent with its factual outcome, provided that the outcome Y is observed under the action $A = a$: $P(Y[a] \mid A = a) = P(Y \mid A = a)$.

Assumption S1: Positivity (Overlap). For a well-defined causal effect, the intervention must be have a non-zero probability: $P(A = a \mid Z) > 0$.

Assumption S2: No-Unmeasured Confounding (NUC). The potential outcome $Y[a]$ is independent of the action A conditioned on Z, if the action assignment is done at random given Z: $Y[a] \perp\!\!\!\perp A \mid Z$.

For continuous-time interventions, we require a continuous-time version of the positivity assumption (Assumption 2). In addition, Schulam & Saria (2017) propose the continuous-time NUC assumption (Assumption 3), for the continuous-time sequential treatment–outcome setup. Furthermore, we assume that a policy intervention affects the outcome trajectory only through the sequence of treatments, i.e., there is no direct effect from the policy variable to the outcomes (Assumption 4).

Assumption 2: Continuous-Time Positivity. The conditional treatment intensity $\lambda^*_\pi(t, m)$ is non-zero for all intervention times t and all intervention marks m, given any intervention history $\mathcal{H}_{<t}$: $\lambda^*_\pi(t, m) > 0, \forall t \in \mathbb{R}_{\geq 0}, \forall m \in \mathbb{R}$.

Assumption 3: Continuous-time NUC. Let sequential treatments $\mathbf{a}$ occur at a discrete set of time points on a given interval, characterized by the conditional intensity function $\lambda^*_\pi(t, m)$. The conditional treatment intensity $\lambda^*_\pi(t, m)$ of a treatment $a = (t, m)$, is independent of the potential outcome trajectory $\mathbf{Y}_{>t}[\tilde{\mathbf{a}}]$, conditioned on the past history $\mathcal{H}_{<t}$, $\forall t \in \mathbb{R}_{\geq 0}$[2].

Assumption 4: Fully-mediated policy effect. Conditioned on the past history $\mathcal{H}_{<\tau}$, the causal effect of the policy $\pi_{>\tau}$ on the outcome trajectory $\mathbf{Y}_{>\tau}$ is fully mediated through sequential treatments $\mathbf{a}_{>\tau}$:

$$P(Y_{>\tau}[\tilde{\pi}_{>\tau}, \tilde{\mathbf{a}}_{>\tau}] \mid \mathcal{H}_{\leq\tau}) = P(Y_{>\tau}[\tilde{\mathbf{a}}_{>\tau}] \mid \mathcal{H}_{\leq\tau}).$$

C. Causal Identifiability

We evaluate the potential outcome trajectory $\mathbf{Y}_{>T}[\tilde{\pi}_{>T}]$ at a discrete set of ordered query points $\mathbf{q} = \{q_1, ..., q_m : q_i > T, \forall i \in 1, ..., m\}$: $\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}] = \{y(q_k)[\tilde{\pi}_{>T}] : q_k \in \mathbf{q}\}$, similar to Schulam & Saria (2017). We denote the outcome value $y(q_k)$ at time q_k by $Y_{q_k} = y(q_k)$. Under the NUC Assumption (Assumption S2), the assignment of the target policy label $\tilde{\pi}$ is independent of the potential outcome query $\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}]$:

$$P(\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = P(\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}] \mid \tilde{\pi}_{>T}, \mathcal{H}_{\leq T}) \tag{10}$$

The causal graph $\mathcal{G}$ suggests that future sequential treatments $\mathbf{a}_{>T}$ act as mediators for the causal effect of the policy intervention $[\tilde{\pi}_{>T}]$ on the outcome query $\mathbf{Y}_\mathbf{q}$. To consider this mediation effect, we include future sequential treatments $\mathbf{a}_{>T}$ to Eq. (10):

$$P(\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}] \mid \tilde{\pi}_{>T}, \mathcal{H}_{\leq T}) = \sum_{\mathbf{a}_{>T}} P(\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}], \mathbf{a}_{>T} \mid \tilde{\pi}_{>T}, \mathcal{H}_{\leq T}). \tag{11}$$

Now, let us consider an intermediate outcome Y_{q_k} at query time q_k. The intermediate outcome Y_{q_k} acts as a common cause for some future actions $\mathbf{a}_{[q_k, q_{k+1})}$ and a future outcome $Y_{q_{k+1}}$. However, it is not possible to adjust for Y_{q_k} as it simultaneously acts as a mediator for a different set of causal paths from previous actions $\mathbf{a}_{<q_k}$ to a future outcome $Y_{q_{k+1}}$. This phenomenon is known as time-varying confounding (Robins, 1986; Pearl & Robins, 1995). To eliminate the time-varying confounding effect, we factorize the counterfactual query $\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}]$ and sequential actions $\mathbf{a}_{>T}$ in Eq. (11) in time-order:

$$\begin{aligned} P(\mathbf{Y}_\mathbf{q}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\mathbf{a}} \prod_{k=1}^{m} \Big(& P(Y_{q_k}[\tilde{\pi}_{>T}] \mid \mathbf{a}_{<q_k}, \mathbf{Y}_{\leq q_{k-1}}[\tilde{\pi}_{>T}], \tilde{\pi}_{>T}, \mathcal{H}_{\leq T}) \\ & P(\mathbf{a}_{[q_{k-1}, q_k)} \mid \mathbf{a}_{<q_{k-1}}, \mathbf{Y}_{\leq q_{k-1}}[\tilde{\pi}_{>T}], \tilde{\pi}_{>T}, \mathcal{H}_{\leq T}) \Big), \end{aligned} \tag{12}$$

[2] In fact, the history $\mathcal{H}_{<t}$ can be extended to include an outcome variable that occurs at time t, if it exists: $\mathcal{H}_{\leq t} \cup \{o_t\}$, assuming instantaneous effects from outcomes to treatments: $o \to a$. However, it would clutter the notation and in practice, the probability of an outcome $o = (t, y)$ to occur at time t is 0. Therefore, assuming no instantaneous effects between variables can be made without any practical effects.

similar to the g-estimation formula (Robins, 1986). By consistency assumption (Assumption 1), the potential outcome $\mathbf{Y}_{\leq q_{k-1}}[\tilde{\pi}_{>T}]$ is equal to the factual outcome when we observe (condition on) the policy specified by $\tilde{\pi}_{>T}$:

$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\mathbf{a}} \prod_{k=1}^{m} \Big(P(Y_{q_k}[\tilde{\pi}_{>T}] \mid \tilde{\pi}_{>T}, \underbrace{\mathbf{a}_{<q_k}, \mathbf{Y}_{\leq q_{k-1}}, \mathcal{H}_{\leq T}}_{\mathcal{H}_{q_{k-1}}}) P(\mathbf{a}_{[q_{k-1},q_k)} \mid \tilde{\pi}_{>T}, \underbrace{\mathbf{a}_{<q_{k-1}}, \mathbf{Y}_{\leq q_{k-1}}, \mathcal{H}_{\leq T}}_{\mathcal{H}_{q_{k-1}}}) \Big). \tag{13}$$

For both the outcome variable Y_{q_k} and actions $\mathbf{a}_{[q_{k-1},q_k)}$, the set of variables $\{\mathbf{a}_{<q_{k-1}}, \mathbf{Y}_{\leq q_{k-1}}, \mathcal{H}_{\leq T}\}$ is a valid history until the query point q_{k-1}: $\mathcal{H}_{q_{k-1}} = \mathcal{H}_{\leq q_{k-1}} \setminus \{a_{q_{k-1}}\} = \{\mathbf{a}_{<q_{k-1}}, \mathbf{Y}_{\leq q_{k-1}}, \mathcal{H}_{\leq T}\}$. Hence, Eq. (13) can be simplified as follows:

$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\mathbf{a}} \prod_{k=1}^{m} \Big(\underbrace{P(Y_{q_k}[\tilde{\pi}_{>T}] \mid \mathbf{a}_{[q_{k-1},q_k)}, \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}})}_{\text{Outcome Factor}} \underbrace{P(\mathbf{a}_{[q_{k-1},q_k)} \mid \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}})}_{\text{Treatment Factor}} \Big). \tag{14}$$

Under Assumptions $\{1, S2\}$, the treatment factor term in Eq. (14) is equivalent to the potential outcome $\mathbf{a}_{[q_{k-1},q_k)}[\tilde{\pi}_{>T}]$:

$$\begin{aligned} P(\mathbf{a}_{[q_{k-1},q_k)} \mid \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}}) &= P(\mathbf{a}_{[q_{k-1},q_k)}[\tilde{\pi}_{>T}] \mid \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}}) && \text{A.1} \\ &= P(\mathbf{a}_{[q_{k-1},q_k)}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{q_{k-1}}). && \text{A.S2} \end{aligned} \tag{15}$$

Conditioned on the past history $\mathcal{H}_{q_{k-1}}$, the causal effect of the policy intervention $[\tilde{\pi}_{>T}]$ on the outcome value Y_{q_k} is fully mediated through sequential treatments $\mathbf{a}_{[q_{k-1},q_k)}$. Using Assumptions $\{1, 2, 3, 4\}$, the outcome factor term in Eq. (14) is equivalent to the potential outcome $Y_{q_k}[\tilde{\mathbf{a}}_{[q_{k-1},q_k)}]$:

$$\begin{aligned} &P(Y_{q_k}[\tilde{\pi}_{>T}] \mid \mathbf{a}_{[q_{k-1},q_k)}, \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}}) \\ &\qquad = P(Y_{q_k} \mid \mathbf{a}_{[q_{k-1},q_k)}, \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}}) && \text{A.1} \\ &\qquad = P(Y_{q_k} \mid \mathbf{a}_{[q_{k-1},q_k)}, \mathcal{H}_{q_{k-1}}) && \text{A.4} \\ &\qquad = P(Y_{q_k}[\mathbf{a}_{[q_{k-1},q_k)}] \mid \mathbf{a}_{[q_{k-1},q_k)}, \mathcal{H}_{q_{k-1}}) && \text{A.1} \\ &\qquad = P(Y_{q_k}[\mathbf{a}_{[q_{k-1},q_k)}] \mid \mathcal{H}_{q_{k-1}}). && \text{A.3} \end{aligned} \tag{16}$$

Plugging the potential outcome representations of the treatment factor (Eq. (15)) and of the outcome factor (Eq. (16)) into Eq. (14), the potential query can be considered as a sequence of stochastic (conditional) interventions on treatment variables:

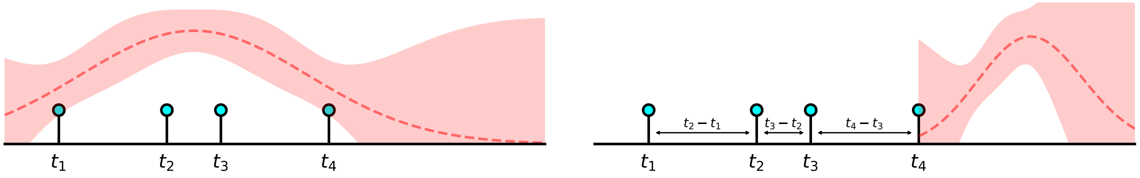
$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\tilde{\mathbf{a}}} \prod_{k=1}^{m} P(Y_{q_k}[\tilde{\mathbf{a}}_{[q_{k-1},q_k)}] \mid \mathcal{H}_{q_{k-1}}) P(\tilde{\mathbf{a}}_{[q_{k-1},q_k)}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{q_{k-1}}). \tag{17}$$

Under Assumptions $\{1, 2, 3\}$, a sequence of interventions on treatments is identified (Schulam & Saria, 2017; Seedat et al., 2022):

$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\tilde{\mathbf{a}}} \prod_{k=1}^{m} P(Y_{q_k} \mid \tilde{\mathbf{a}}_{[q_{k-1},q_k)}, \mathcal{H}_{q_{k-1}}) P(\tilde{\mathbf{a}}_{[q_{k-1},q_k)} \mid \tilde{\pi}_{>T}, \mathcal{H}_{q_{k-1}}). \tag{18}$$

D. Nonparametric Modeling approaches for TPPs

TPPs can be modeled by two nonparametric approaches: (i) latent-state point processes (Møller et al., 1998; Adams et al., 2009; Lloyd et al., 2015) and (ii) regressive point processes (Hawkes, 2018; Liu & Hauskrecht, 2019). The difference between the two approaches is shown in Figure 5.



(a) Latent-state point processes.

(b) Regressive point processes.

Figure 5: **(a)** An example of the conditional intensity function λ for a latent-state point process, which takes absolute times as input. **(b)** An example of the conditional intensity function λ for a regressive point process, which takes relative times as input.

D.1. Latent-state point processes

Latent-state point processes, e.g. a log/sigmoidal Gaussian Cox process (Møller et al., 1998; Adams et al., 2009) and a variational Bayes point process (VBPP, Lloyd et al., 2015), model the temporal dependence of the intensity indirectly, in the form of smoothness assumptions encoded by a latent (GP) prior (Williams & Rasmussen, 2006). For example, the VBPP model defines a latent GP prior $f \sim \mathcal{GP}$ with a squared-exponential (SE) kernel over a period $[0, T]$. The latent function f is passed through a square transformation to obtain the conditional intensity function $\lambda(\tau)$. The latent function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$ takes an absolute time $\tau \in [0, T]$ as input and outputs the latent rate for the same time point. The choice of the SE kernel assumes a smooth latent function, which implicitly enforces some temporal dependence of nearby intensity values. For example, the intensity values for points $\{t_i\}_{i=1}^{4}$ in Fig. 5a is given by:

$$\lambda^* \left(\begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{bmatrix} \right) = f \left(\begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{bmatrix} \right)^2 .$$

D.2. Regressive Point Processes

Unlike a latent-state formulation, the regressive point processes model the dependence on past events explicitly, by using relative times as input instead of absolute times (Hawkes, 2018; Liu & Hauskrecht, 2019). Liu & Hauskrecht (2019) propose the conditional GP regressive point process (CGPRPP), which defines a latent function $f \sim \mathcal{GP}$ that takes time differences of the last D events relative to a future time point τ as input: $f : \mathbb{R}_{\geq 0}^{D} \to \mathbb{R}$. They define τ to be a query time and $r : \mathbb{R}_{\geq 0} \to \{\mathbb{R}_{\geq 0} \cup \infty\}^D$ be a function that returns the relative times of the last D events: $r(\tau) = (\Delta t_1, \ldots, \Delta t_D)$, where $\Delta \tau_d$ denotes the relative time between τ and d^{th} last event: $\Delta \tau_d = \tau - t_d$ for $d \in 1, \ldots, D$:

$$r(\tau) = \{\Delta \tau_d\}_{d=1}^{D} = (\tau - t_1, \ldots, \tau - t_D).$$

If d^{th} last event does not exist, the function $r(\tau)$ returns the placeholder value ∞ for the d^{th} index.

CGPRPP regresses relative times $\{\Delta \tau_d\}_{d=1}^{D}$ on the latent intensity $f(\tau)$, which is passed through a square transformation to obtain the conditional intensity value $\lambda^*(\tau)$. As an example, the intensity values for points $\{t_i\}_{i=1}^{4}$ in Figure 5b, with $D = 2$, is given by:

$$\lambda^* \left(\begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{bmatrix} \right) = f \left(\begin{bmatrix} \infty & \infty \\ t_2 - t_1 & \infty \\ t_3 - t_2 & t_3 - t_1 \\ t_4 - t_3 & t_4 - t_2 \end{bmatrix} \right)^2 .$$

E. Treatment Model Details

E.1. Model Definition

Similar to the CGPRPP model proposed by Liu & Hauskrecht (2019), we model the treatment time intensity $\lambda^*(\tau)$ as the square transformation of a latent function, which is a sum of a constant scalar β_0 and three time-dependent functions with GP priors, $g_b, g_a^*, g_o^* \sim \mathcal{GP}$. The latent-state function $g_b : \mathbb{R}_{\geq 0} \to \mathbb{R}$ models the baseline intensity. The regressive component $g_a^* : \{\mathbb{R}_{\geq 0} \times \mathbb{R}\}^{Q_a} \to \mathbb{R}$ models the dependence on past treatments by taking the last Q_a treatment pairs (treatment time and dosage) as input. Similarly, the regressive component $g_o^* : \{\mathbb{R}_{\geq 0} \times \mathbb{R}\}^{Q_o} \to \mathbb{R}$ models the dependence on past outcomes by taking the last Q_o outcome pairs (measurement time and value) as input.

The treatment time intensity $\lambda^*(\tau)$ is defined as follows:

$$\lambda^*(\tau) = \Big(\underbrace{\beta_0}_{\substack{\text{PP}\\\text{baseline}}} + \underbrace{g_b(\tau)}_{\substack{\text{NHPP}\\\text{Baseline}}} + \underbrace{g_a^*(\tau)}_{\substack{\text{Treatment}\\\text{Effect}}} + \underbrace{g_o^*(\tau)}_{\substack{\text{Outcome}\\\text{Effect}}} \Big)^2.$$

E.2. Kernel Definition

Regressive components g_a^* and g_o^* explicitly model how the intensity depends on the past history. They take relative times to the last Q_a treatments and Q_o outcomes as input, while the baseline function g_b takes the absolute time. For example, let τ be a query time. Let $(t_{a_1}, \ldots, t_{a_{Q_a}})$ be the arrival times of last Q_a treatments before time τ. Similarly, let $(t_{o_1}, \ldots, t_{o_{Q_o}})$ be the list of arrival times of the last Q_o outcomes, before time τ. Then, three function components has the following $1 + 2 * Q_a + 2 * Q_o$ dimensional vector as input:

$$\tau, \underbrace{(\tau - t_{a_1}, m_1), \ldots}_{\text{last } Q_a \text{ Treatments}}, \underbrace{(\tau - t_{o_1}, y_1), \ldots}_{\text{last } Q_o \text{ Outcomes}}$$

To represent the information regarding the last $Q_a + Q_o$ events, we define retrieval functions:

- Let $r_{k_i,t} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ denote a retrieval function that takes time τ as input and outputs the time of the i^{th} last event of type $k \in \{a, o\}$. For example, the retrieval function $r_{a_1,t}$ outputs the time of the last action occurred before time τ: $r_{a_1,t}(\tau) = t_{a_1}$. Similar to $r_{k_i,t}$, let $\Delta_{k_i,t} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a retrieval function that returns the relative time: $\Delta_{a_1,t}(\tau) = \tau - t_{a_1}$.
- Let $r_{k_i,m} : \mathbb{R}_{\geq 0} \to \mathbb{R}$ denote a retrieval function that takes time τ as input and returns the mark $m \in \mathbb{R}$ of i^{th} last tuple of type $k \in \{a, o\}$.
- Let $r_{k_i} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} \times \mathbb{R}$ denote a retrieval function that outputs both the relative time and the mark of the i^{th} last event of type $k \in \{a, o\}$. For example, the retrieval function r_{a_1} outputs the relative time and the mark of the last action occurred before time τ: $r_{a_1}(\tau) = (\tau - t_{a_1}, m_1)$.
- For a query time τ, the retrieval function $r(\tau) : \mathbb{R}_{\geq 0} \to \mathcal{X}$ returns the overall input vector:

  $$r(\tau) = \{\tau, \underbrace{r_{a_1}(\tau), \ldots, r_{a_{Q_a}}(\tau)}_{\text{last } Q_a \text{ Treatments}}, \underbrace{r_{o_1}(\tau), \ldots, r_{o_{Q_o}}(\tau)}_{\text{last } Q_o \text{ Outcomes}}\},$$

  where $\mathcal{X}$ denotes the input domain $\mathcal{X} = \mathbb{R}_{\geq 0} \times \{\mathbb{R}_{\geq 0} \times \mathbb{R}\}^{Q_a} \times \{\mathbb{R}_{\geq 0} \times \mathbb{R}\}^{Q_o}$.

We represent the unavailable past information by an identifier value $r_{k_i} = (\infty, \infty)$, similar to (Liu & Hauskrecht, 2019). To set covariance values concerning the unavailable information to 0, an indicator function $\mathbb{1}_k : \mathbb{R}_{\geq 0} \times \mathbb{R} \to \{0, 1\}$ is defined:

$$\mathbb{1}_k[r_{k_i}(\tau)] = \begin{cases} 1, & \text{if } r_{k_i,t} < \infty, r_{k_i,m} < \infty \\ 0, & \text{otherwise} \end{cases}$$

Using the indicator function, the kernel function is defined as follows:

$$K(v, v') = K_b(\tau, \tau') + \sum_{k \in \{a,o\}} \sum_{i=1}^{Q_k} \mathbb{1}[r_{k_i}(\tau)] \mathbb{1}[r_{k_i}(\tau')] K_{k_i}(r_{k_i}(\tau), r_{k_i}(\tau')),$$

where K_b is an SE kernel, $v = r(\tau)$ and $v' = r(\tau')$ are input vectors, and the single-event kernel function $K_{k_i} : \mathbb{R}_{\geq 0} \times \mathbb{R} \rightarrow \mathbb{R}$ is a sum of SE kernels acting on relative time and mark dimensions independently:

$$\begin{aligned} K_{k_i}(r_{k_i}(\tau), r_{k_i}(\tau'))) &= \gamma_i \underbrace{\exp\left(-\frac{(\Delta_{k_i,t}(\tau) - \Delta_{k_i,t}(\tau'))^2}{\sigma_{i,t}}\right)}_{\text{Relative Time Kernel } K_t} \underbrace{\exp\left(-\frac{(r_{k_i,m}(\tau) - r_{k_i,m}(\tau'))^2}{\sigma_{i,m}}\right)}_{\text{Mark Kernel } K_m} \\ &= \gamma_i K_t(\Delta_{k_i,t}(\tau), \Delta_{k_i,t}(\tau')) K_m(r_{k_i,m}(\tau), r_{k_i,m}(\tau')) \end{aligned}$$

The overall kernel is a sum of the baseline SE kernel and $Q_a + Q_o$ two-dimensional single-event kernels K_k, each of which is an SE kernel on two dimensions: relative time and mark.

E.3. Likelihood

The likelihood for a set of treatments $\mathcal{D} = \{(t_n, m_n)\}_{n=1}^N$ can be written in terms of the conditional intensity function $\lambda^*(t, m) = \lambda^*(t)p^*(m \mid t)$:

$$p(\mathcal{D} \mid \lambda^*(\cdot)) = \prod_n \lambda^*(t_n)p^*(m_n \mid t_n) \times \exp\Big\{ - \int_{\mathcal{T}} \lambda^*(\tau)d\tau \Big\} \tag{19}$$

The mark intensity $p^*(m_n \mid t_n)$ factorizes and is modeled by a GP prior independent from the time intensity $\lambda^*(t)$. The inference for the mark intensity $p^*(m_n \mid t_n)$ is straightforward. Therefore, in the following, we derive the inference objective for the time intensity $\lambda^*(t)$.

E.3.1. Multiple Observations

We assume multiple observations $\mathcal{D} = \{\mathcal{D}_1, ..., \mathcal{D}_O\}_{o=1}^O$ are conditionally independent given the treatment intensity λ^*, e.g., daily meals for a patient for O days, where we consider each day as conditionally independent. Then, the likelihood factorizes over multiple observations:

$$p(\mathcal{D}_1, ..., \mathcal{D}_O \mid \lambda^*(\cdot)) = \prod_o p(\mathcal{D}_o \mid \lambda^*(\cdot)).$$

where the term $p(\mathcal{D}_o \mid \lambda^*(\cdot))$ follows Eq. (19).

E.4. Variational Inference

In variational inference, a lower bound (ELBO), denoted by $\mathcal{L}$, of the log likelihood $\log p(\mathcal{D})$ is maximized, which is equivalent to minimizing the KL divergence between the variational distribution $q(f)$ and the true posterior $p(f \mid \mathcal{D})$:

$$\log p(\mathcal{D}) = \mathcal{L} + \text{KL}[q(f) \mid\mid p(f \mid \mathcal{D})],$$

where the lower bound $\mathcal{L}$ is given by:

$$\begin{aligned} \mathcal{L} &= \mathbb{E}_q[\log p(\mathcal{D} \mid f)] - \text{KL}[q(f) \mid\mid p(f)] \\ &= \mathcal{L}_{\mathcal{D}} - \text{KL}[q(f) \mid\mid p(f)]. \end{aligned}$$

E.4.1. Inducing Points

Inducing point approximations are commonly used for scalable inference in GPs. Let $\mathbf{Z} = \{z_m\}_{m=1}^M$ be a set of inducing points. Their function evaluations are collected in a set of inducing variables $\mathbf{u} = f(Z) \in \mathbb{R}^M$, where each $u_m = f(z_m)$. Conditioned on inducing variables $\mathbf{u}$, we assume that the variational conditional distribution $q(\mathbf{f} \mid \mathbf{u})$ is equal to the true conditional $p(\mathbf{f} \mid \mathbf{u})$. Then, the variational distribution $q(\mathbf{f}, \mathbf{u})$ can be written as follows:

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f} \mid \mathbf{u})q(\mathbf{u}),$$

where $q(\mathbf{u}) \sim N(\mathbf{u}; \mathbf{m}, \mathbf{S})$. Generally, we integrate out $\mathbf{u}$:

$$\begin{aligned} q(\mathbf{f}) &= \int_{\mathcal{U}} p(\mathbf{f} \mid \mathbf{u}) q(\mathbf{u}) d\mathbf{u} = \mathcal{GP}(f; \tilde{\mu}, \tilde{\Sigma}), \\ \tilde{\mu}(\mathbf{x}) &= \mathbf{k_{xz}} \mathbf{K_{zz}^{-1}} \mathbf{m}, \\ \tilde{\Sigma}(\mathbf{x}, \mathbf{x'}) &= \mathbf{K_{xx'}} - \mathbf{k_{xz}} \mathbf{K_{zz}^{-1}} \mathbf{k_{zx}} + \mathbf{k_{xz}} \mathbf{K_{zz}^{-1}} \mathbf{S} \mathbf{K_{zz}^{-1}} \mathbf{k_{zx'}}. \end{aligned}$$

E.4.2. Evidence Lower Bound (ELBO)

The lower bound $\mathcal{L}$ is:

$$\mathcal{L} = \mathcal{L}_{\mathcal{D}} - \text{KL}[q(f) \mid\mid p(f)].$$

Similar to Lloyd et al. (2015); Matthews et al. (2016); John & Hensman (2018), we compute the KL term $\text{KL}[q(f) \mid\mid p(f)]$ by computing the KL divergence at inducing points $\text{KL}[q(\mathbf{u}) \mid\mid p(\mathbf{u})]$:

$$\mathcal{L} = \mathcal{L}_{\mathcal{D}} - \text{KL}[q(\mathbf{u}) \mid\mid p(\mathbf{u})]$$

We now consider the log-likelihood expectation term $\mathcal{L}_{\mathcal{D}} = \mathbb{E}_q[\log p(\mathcal{D} \mid f)]$:

$$\begin{aligned} \mathcal{L}_{\mathcal{D}} &= \sum_o \sum_n \mathbb{E}_q[\log \lambda^*(t_n^{(o)})] - \sum_o \mathbb{E}_q \Big[\int_{\mathcal{T}} \lambda^*(t) dt \Big] \\ &= \sum_o \mathcal{L}_n^{(o)} - \sum_o \mathcal{L}_t^{(o)} \end{aligned}$$

First, we start with the data term $\mathcal{L}_n^{(o)}$ for a single observed point process realization, indexed by (o):

$$\begin{aligned} \mathcal{L}_n^{(o)} &= \sum_n \mathbb{E}_q[\log \lambda^*(t_n)] \\ &= \sum_n \mathbb{E}_q[\log (f(r(t_n)) + \beta)^2], \end{aligned}$$

As John & Hensman (2018), we use a change of variables $f = f_n + \beta$ and exploit the computational trick used in Lloyd et al. (2015) to compute the one-dimensional integral $\mathbb{E}_q[\log f_n^2]$:

$$\begin{aligned} \mathbb{E}_q[\log f^2] &= \int_{-\infty}^{\infty} \log(f^2) \mathcal{N}(f; \tilde{\mu}, \tilde{\sigma}^2) df \\ &= -\tilde{G}\Big(-\frac{\tilde{\mu}^2}{2\tilde{\sigma}^2} \Big) + \log \Big(\frac{\tilde{\sigma}^2}{2} \Big) - C \end{aligned}$$

where C is the Euler-Mascheroni constant and $\tilde{G}$ is defined by the confluent hyper-geometric function, which is approximated by a look-up table for faster computation.

To compute the integral term $\mathcal{L}_t^{(o)}$, we divide the observation period into $N + 1$ intervals with end points $(0, t_1, \ldots, t_N, T)$, where $\{t_i\}_{i=1}^N$ includes all event points that have an effect on the conditional intensity value. In each interval $[t_n, t_{n+1}]$, the conditional intensity values $\{\lambda^*(\tau) : \tau \in [t_n, t_{n+1}]\}$ can be estimated using the past history $\mathcal{H}_{<t_n}$. Using a shorthand notation for $\mathbf{r} = r(t) \in \mathcal{X}$, we write the integral term as a sum of three integrals:

$$\begin{aligned} \mathcal{L}_t^{(o)} &= \mathbb{E}_q \Big[\int_{\mathcal{T}} \lambda^*(\tau) d\tau \Big], \\ &= \mathbb{E}_q \Big[\int_{\mathcal{T}} (f(r(\tau)) + \beta)^2 d\tau \Big], \\ &= \Big[\int_{\mathcal{T}} \Big(\mathbb{E}_q[f^2(\mathbf{r})] + 2\beta \, \mathbb{E}_q[f(\mathbf{r})] + \beta^2 \Big) d\tau \Big], \\ &= \Big[\int_{\mathcal{T}} \mathbb{E}_q[f(\mathbf{r})]^2 d\tau + \int_{\mathcal{T}} \text{Var}_q[f(\mathbf{r})] d\tau + 2\beta \int_{\mathcal{T}} \mathbb{E}_q[f(\mathbf{r})] d\tau + \beta^2 |\mathcal{T}| \Big], \\ &= \sum_{n=1}^{N+1} \Big[\int_{t_{n-1}}^{t_n} \mathbb{E}_q[f(\mathbf{r})]^2 d\tau + \int_{t_{n-1}}^{t_n} \text{Var}_q[f(\mathbf{r})] d\tau + 2\beta \int_{t_{n-1}}^{t_n} \mathbb{E}_q[f(\mathbf{r})] d\tau + \beta^2 |\mathcal{T}| \Big]. \end{aligned} \tag{20}$$

We can compute each integral term as follows:

$$\begin{aligned}
\int_{t_{n-1}}^{t_n} \mathbb{E}_q[f(r(\tau))]^2 d\tau &= \mathbf{m}^{\mathbf{T}} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \boldsymbol{\Psi}_{\mathbf{n}} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \mathbf{m}, \\
\int_{t_{n-1}}^{t_n} \mathrm{Var}_q[f(\tau)] d\tau &= \sum_{d=1}^{D} \gamma_d \int_{t_{n-1}}^{t_n} \mathbb{1}[h_d(t)] dt + \mathrm{Tr}(\mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \boldsymbol{\Psi}_{\mathbf{n}}) + \mathrm{Tr}(\mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \mathbf{S} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \boldsymbol{\Psi}_{\mathbf{n}}), \\
\int_{t_{n-1}}^{t_n} \mathbb{E}_q[f(\tau)] d\tau &= \boldsymbol{\Phi}_{\mathbf{n}}^{\mathbf{T}} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \mathbf{m}, \\
\boldsymbol{\Psi}_{\mathbf{n}}(\mathbf{z}, \mathbf{z}') &= \int_{t_{n-1}}^{t_n} K(\mathbf{z}, r(\tau)) K(r(\tau), \mathbf{z}') d\tau, \\
\boldsymbol{\Phi}_{\mathbf{n}}(\mathbf{z}) &= \int_{t_{n-1}}^{t_n} K(\mathbf{z}, r(\tau)) d\tau.
\end{aligned}$$

Then, we plug in all integral terms into Eq. (20):

$$\begin{aligned}
\mathcal{L}_t^{(o)} &= \mathbf{m}^{\mathbf{T}} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \boldsymbol{\Psi}_{\mathbf{n}} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \mathbf{m} \\
&+ \sum_{d=1}^{D} \gamma_d \int_{t_{n-1}}^{t_n} \mathbb{1}[h_d(t)] dt + \mathrm{Tr}(\mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \boldsymbol{\Psi}_{\mathbf{n}}) + \mathrm{Tr}(\mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \mathbf{S} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \boldsymbol{\Psi}_{\mathbf{n}}) \\
&+ 2\beta \boldsymbol{\Phi}_{\mathbf{n}}^{\mathbf{T}} \mathbf{K}_{\mathbf{zz}}^{-\mathbf{1}} \mathbf{m} + \beta^2 |\mathcal{T}_o|.
\end{aligned}$$

The phi vector $\boldsymbol{\Phi}_{\mathbf{n}}$ and the psi matrix $\boldsymbol{\Psi}_{\mathbf{n}}$ have closed form solutions, obtained by evaluating the integrals for the sum of SE kernels. The difference to Liu & Hauskrecht (2019) is the inclusion of the SE mark kernel terms $k_m(\cdot, \cdot)$.

$$\begin{aligned}
\Phi_n(z) = \sum_{q=1}^{Q_a+Q_o} & \mathbb{1}[z_q] \mathbb{1}[r_q(t_n)] \gamma_q \frac{\sqrt{\pi \alpha_{q,t}}}{\sqrt{2}} \exp\left(\frac{(m_{z_q} - r_{q,m}(t_n))^2}{\alpha_{q,m}} \right) \\
& \left[\mathrm{erf}\left(\frac{t_n - r_{q,t}(t_n) - t_{z_q}}{\sqrt{2\alpha_{q,t}}} \right) - \mathrm{erf}\left(\frac{t_{n-1} - r_{q,t}(t_n) - t_{z_q}}{\sqrt{2\alpha_{q,t}}} \right) \right],
\end{aligned}$$

$$\begin{aligned}
\Psi_n(z, z') = \sum_{i,j=1}^{Q_a+Q_o} & \mathbb{1}[z_i] \mathbb{1}[z_j] \mathbb{1}[r_i(t_n)] \mathbb{1}[r_j(t_n)] \gamma_i \gamma_j \frac{\sqrt{\pi \alpha_{i,t} \alpha_{j,t}}}{\sqrt{2(\alpha_{i,t} + \alpha_{j,t})}} \\
& \exp\left(\frac{(m_{z_i} - r_{i,m}(t_n))^2}{\alpha_{i,m}} \right) \exp\left(\frac{(m_{z_j} - r_{j,m}(t_n))^2}{\alpha_{j,m}} \right) \\
& \exp\left(- \frac{(t_{z_i} + r_{i,t}(t_n) - t'_{z'_j} - r_{j,t}(t_n))^2}{2(\alpha_{i,t} + \alpha_{j,t})} \right) \\
& \left[\mathrm{erf}\left(\frac{\alpha_i (t_n - r_{j,t}(t_n) - t'_{z'_j}) + \alpha_j (t_n - r_{i,t}(t_n) - t_{z_i})}{\sqrt{2\alpha_{i,t}\alpha_{j,t}(\alpha_{i,t} + \alpha_{j,t})}} \right) \right. \\
& \left. - \mathrm{erf}\left(\frac{\alpha_i (t_{n-1} - r_{j,t}(t_n) - t'_{z'_j}) + \alpha_j (t_{n-1} - r_{i,t}(t_n) - t_{z_i})}{\sqrt{2\alpha_{i,t}\alpha_{j,t}(\alpha_{i,t} + \alpha_{j,t})}} \right) \right].
\end{aligned}$$

F. Outcome Model Details

We model the outcome trajectory $\mathbf{Y} = \{y(\tau) : \tau \in \mathbb{R}_{\geq 0}\}$ by a conditional GP model, combining three independent function components: (i) a baseline progression, (ii) a treatment response function and (iii) a noise variable (Schulam & Saria, 2017; Zhang et al., 2020):

$$y(\tau) = \underbrace{f_b(\tau)}_{\text{Baseline}} + \underbrace{f_a(\tau; \mathbf{a})}_{\text{Treatment Response}} + \underbrace{\epsilon(\tau)}_{\text{Noise}}, \tag{21}$$

Baseline progression. The baseline progression $f_b(\tau)$ over time τ can be modeled by a GP prior. The kernel function is chosen depending on the application context. For example, a baseline function $f_b(\tau) \sim \mathcal{GP}(0, K_{se} + K_{per})$, with a zero mean and a kernel function equal to the sum of a squared exponential (SE) and a periodic (PER) kernel was proposed to model blood pressure and the heart rate measurements in Cheng et al. (2020). A combination of two non-stationary kernel functions was chosen as the baseline kernel function to model creatinine measurements in Schulam & Saria (2017). In Xu et al. (2016), the baseline kernel function was chosen as the sum of a linear and an exponential kernel. In our experiments, we model the baseline progression of blood glucose as a sum of a constant and a periodic kernel to capture daily blood glucose profiles of non-diabetic patients fluctuating around a constant baseline (Ashrafi et al., 2021).

Treatment Response Model. We define the treatment response function $f_a(\tau; \mathbf{a})$ as a Gaussian process: $f_a(\tau; \mathbf{a}) \sim \mathcal{GP}$, similar to Cheng et al. (2020). While Cheng et al. (2020) propose an extension using a latent force model (Alvarez et al., 2009), we use an SE kernel, as it produces sufficient performance for our use case. For each treatment, we consider the treatment response as additive:

$$f_a(\tau; \mathbf{a}) = \sum_{a_i = (t_i, m_i) \in \mathbf{a}} f_m(m_i) f_t(\tau; t_i),$$

where the scaling function $f_m : \mathbb{R} \to \mathbb{R}$ is a function of the treatment mark $m_i \in \mathbb{R}$. Each treatment type is assumed to have a distinct pair of a scaling function f_m and a time-dependent response function f_t. The scaling function $f_m^{(v)}(m_i)$ is a linear function that captures the individualized treatment response magnitude for each patient (v):

$$f_m^{(v)}(m_i) = \beta_0^{(v)} + \beta_1^{(v)} m_i,$$

where a hierarchical Gaussian prior is placed on parameters $\{\beta_0^{(v)}, \beta_1^{(v)}\}$: $\beta_0^{(v)} \sim \mathcal{N}(\beta_0, \sigma_0^2)$, $\beta_1^{(v)} \sim \mathcal{N}(\beta_1, \sigma_1^2)$.

The time-dependent function $f_t(\tau; \mathbf{a})$ is defined as a piecewise function that is equal to 0 outside treatment response intervals $\mathcal{T}_{t_i} = [t_i, t_i + T_i]$ and $\mathcal{T}_{t'_i} = [t'_i, t'_i + T'_i]$:

$$f_t(\tau; t_i) = \begin{cases} \mathcal{GP}(0, k_{f_t, f'_t}(\tau, \tau'; t_i, t'_i)), & \text{if } \tau \in \mathcal{T}_{t_i}, \tau' \in \mathcal{T}_{t'_i} \\ 0, & \text{otherwise} \end{cases},$$

where the treatment response interval T_i is set by domain knowledge. The kernel function works with non-negative relative times $\Delta\tau_i = \text{RELU}(\tau - t_i)$, rather than the absolute time τ:

$$k_{f_t, f'_t}(\tau, \tau'; t_i, t'_i) = \exp\left(-\frac{(\Delta\tau_i - \Delta\tau'_i))^2}{l_t^2} \right).$$

The adoption of the piecewise definition and relative times is enforcing a ‘causal’ kernel for the treatment response, as in ‘causal’ GP proposed by Cunningham et al. (2012). Here, the term ‘causal’ is used differently than in the causality discussion in previous sections and has more of a common sense meaning implying the time direction.

G. Learning

For the treatment model, we maximize the ELBO $\mathcal{L}$ with respect to variational parameters $\theta_{a,z} = \{\mathbf{m_z}, \mathbf{S_z}\}$ and kernel hyperparameters $\theta_{a,h} = \{\gamma_q, \alpha_q\}_{q=0}^{Q_a+Q_o}$ (see Appendix E.4 for details). For the outcome model, we maximize the marginal likelihood $p(\mathcal{D} \mid \theta_{o,h})$ with respect to model hyperparameters $\theta_{o,h} = \{b_b^{(v)}, \ell_b^{(v)}, \alpha_b^{(v)}\}_{v=1}^{N_{\text{patient}}} \cup \{\ell_t, \beta_0, \beta_1, \{\beta_0^{(r)}, \beta_1^{(v)}\}_{v=1}^{N_{\text{patient}}}\}$, where $\{b_b^{(v)}, \alpha_b^{(v)}, \ell_b^{(v)}\}$ denote the parameters of the individual-specific baseline function and $\{\beta_0, \beta_1, \beta_0^{(v)}, \beta_1^{(v)}\}$ denote the parameters of the linear, hierarchical model for the scale function $f_m^{(v)}(\cdot)$.

H. Inference

H.1. Test Log-Likelihood (TLL)

The outcome model is a GP prior with an independent Gaussian noise. The posterior of the latent function $\mathbf{f_q}$ at query times $\mathbf{q}$ follows a multivariate normal distribution and its mean and covariance functions have closed form solutions:

$\mathbf{f_q} \mid \mathbf{y} \sim \mathcal{N}(\tilde{\mu}_q, \tilde{\Sigma}_q)$, with $\hat{\mu}_q = \mathbf{k}_*^{\mathbf{T}}(\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1}\mathbf{y}$ and $\hat{\Sigma}_q = \mathbf{K}_{**} - \mathbf{k}_*^{\mathbf{T}}(\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1}\mathbf{k}_*$. The test log-likelihood has a closed form solution and can be computed by integrating out the posterior $\mathbf{f_q} \mid \mathbf{y}$:

$$\log p(\mathcal{D}_{\text{test}} \mid \mathcal{D}) = \log \int_{\mathbf{f_q}} \log p(\mathcal{D}_{\text{test}} \mid \mathbf{f_q}) p(\mathbf{f_q} \mid \mathcal{D}) d\mathbf{f_q}.$$

For the treatment model, the marginal log-likelihood is intractable. A lower-bound on the test log-likelihood $\log p(\mathcal{D}_{\text{test}} \mid \mathcal{D})$ can be computed as follows:

$$\begin{aligned}
\log p(\mathcal{D}_{\text{test}} \mid \mathcal{D}) &= \log \int_{g^*} \log p(\mathcal{D}_{\text{test}} \mid g^*) p(g^* \mid \mathcal{D}) dg^*, \\
&= \log \mathbb{E}_{p(g^* \mid \mathcal{D})} \left[p(\mathcal{D}_{\text{test}} \mid g^*) \right], \\
&\approx \log \mathbb{E}_{q(g^*)} \left[p(\mathcal{D}_{\text{test}} \mid g^*) \right], \\
&\geq \mathbb{E}_{q(g^*)} \left[\log p(\mathcal{D}_{\text{test}} \mid g^*) \right],
\end{aligned}$$

where we approximate the true posterior $p(g^* \mid \mathcal{D})$ by the variational distribution $q(g^*) = p(g^* \mid u)q(u)$. The term $\mathbb{E}_{q(g^*)} \left[\log p(\mathcal{D}_{\text{test}} \mid g^*) \right]$ can be computed as in the steps detailed in Appendix E.4.2, as it has the same form as the expected log likelihood component of the ELBO.

Algorithm 1 Ogata's Thinning algorithm

Input: Start T_1, End T_2, Interval function $l(\cdot)$, Conditional intensity $\lambda^*(\cdot)$.
Output: Point process sample $\mathcal{T} = \{t_1, \ldots, t_n\}$in the interval $[T_1, T_2]$.

```
function SAMPLE-OGATA($T_1, T_2, l, \lambda^*$)
    $\tau = T_1, \mathcal{T} = \emptyset$.                                      ▷ Initialize
    while $\tau < T_2$ do
        $\lambda_{ub} = \sup_{s \in [\tau, \tau + l(\tau)]} \lambda^*(s)$.                  ▷ Upper-bound.
        $u_{ub}, u_a \sim \mathcal{U}(0, 1)$.                            ▷ Sample noise variables.
        $t_i = -1/\lambda_{ub} * \log(u_{ub})$       ▷ Draw the inter-arrival time, $t_i \sim \text{Exp}(\lambda_{ub})$.
        if $t_i \leq l(\tau)$ then                              ▷ Candidate in the interval.
            if $u_a \leq \lambda^*(\tau + t_i)/\lambda_{ub}$ then       ▷ Keep with probability $\lambda^*/\lambda_{ub}$
                $\mathcal{T} = \mathcal{T} \cup \{\tau + t_i\}$.                         ▷ Point accepted.
            end if
            $\tau = \tau + t_i$.                         ▷ Continue from the candidate point.
        else                                        ▷ No candidate point in the interval.
            $\tau = \tau + l(\tau)$.                  ▷ Continue from the end of the interval.
        end if
    end while
    return $\mathcal{T} = \{t_1, \ldots, t_n\}$.
end function
```

H.2. Counterfactual Query

For a policy counterfactual, we estimate the potential outcome of the past trajectory $P(\mathbf{Y}_{\leq T}[\tilde{\pi}_{\leq T}] \mid \mathcal{H}_{\leq T}, \pi_{\leq T})$, under a new treatment policy specified by $\tilde{\pi}_{\leq T}$ different than the observed policy specified by $\pi_{\leq T}$. To answer a counterfactual query, we require access to the posterior distribution of noise variables $\mathbf{N}$ in addition to the counterfactual intensity λ^*_{cf}.

In our model, the noise consists of two components: treatment and outcome noise, $\mathbf{N} = \{\mathcal{E}_a, \mathcal{E}_o\}$. The outcome noise $\mathcal{E}_o$ is defined in Eq. (8) and it is equal to the difference between the outcome measurement $y(\tau)$ and the latent function $f(\tau)$. Hence, its posterior is available in closed form from the GP posterior:

$$\epsilon(\tau) \mid \mathbf{y} = y(\tau) - f(\tau) \mid \mathbf{y}.$$

In our experiments, we use the mean of the noise posterior as a point estimate:

$$\mathbb{E}\left[\epsilon(\tau) \mid \mathbf{y}\right] = y(\tau) - \mathbb{E}\left[f(\tau) \mid \mathbf{y}\right].$$

The treatment noise $\mathcal{E}_a$ comprises noise associated with the sampling process of the treatments. In practice, we extend the counterfactual sampling algorithm for non-homogeneous Poisson processes provided in Noorbakhsh & Rodriguez (2022) to history-dependent point processes where future events depend on past events. The key difference in algorithms is that the counterfactual sampling algorithm provided in Noorbakhsh & Rodriguez (2022) is based on Lewis' thinning algorithm (Lewis & Shedler, 1979), while our algorithm is based on Ogata's thinning algorithm (Ogata, 1981). Both thinning algorithms follow a general recipe: they first sample candidate points from an upper-bounding intensity and then perform rejection sampling with a probability proportional to the target intensity, so that the intensity of accepted points is equivalent to the target intensity. Different than Lewis' algorithm, Ogata's algorithm considers smaller intervals where the target intensity can be upper-bounded. In the following, we first discuss Ogata's thinning algorithm and then present our counterfactual sampling algorithm. We conclude by discussing the identifiability of the counterfactual query.

H.2.1. Ogata's Thinning Algorithm

Consider an initial time point s, where we start sampling. Ogata's algorithm first selects an interval such as $[s, s + l(s)]$, where the function $l(\cdot)$ specifies the length of the interval $l(s)$ until which it is possible to upper bound the conditional intensity $\lambda^*(\tau)$ (Rasmussen, 2011). The upper bound intensity should be greater than or equal to the supremum of the conditional intensity $\lambda^*(\tau)$ in the interval $[s, s + l(s)]$:

$$\lambda_{ub} \geq \sup_{\tau \in [s, s+l(s)]} \lambda^*(\tau).$$

Using the constant upper bound intensity, one can sample a candidate point from an upper-bounding Poisson process $t_i \sim \mathcal{PP}(\lambda_{ub})$. After sampling t_i, the algorithm considers two options:

- If $t_i > l(s)$, then there exists no candidate points in the interval $[s, s + l(s)]$ and the sampling procedure continues from $s + l(s)$.
- If $t_i < l(s)$, t_i is kept with the probability $\lambda^*(t_i)/\lambda_{ub}$. In practice, a noise variable U_i is sampled from a uniform distribution: $u_i \sim \mathcal{U}(0, 1)$ and the point is kept if $u_i \leq \lambda^*(t_i)/\lambda_{ub}$. Regardless of the thinning decision, the procedure continues from the time point t_i.

The sampling process is illustrated in Algorithm 1.

H.2.2. Counterfactual Sampling Algorithm for Tpps

To estimate the policy counterfactual, we use the counterfactual treatment intensity λ^*_{cf} and individual-specific noise distribution $p(\mathbf{N} \mid \mathcal{D}^{(r)})$ (see Section 6). We define treatment noise variables $\mathcal{E}_a$ by assuming that the generative process for observed treatments follows Ogata's thinning algorithm (Ogata, 1981; Noorbakhsh & Rodriguez, 2022). Since the sampling algorithm is not actually used to generate the observed data, the noise variables are latent variables of the model.

In Ogata's algorithm, there are two sources for the stochasticity: (i) the candidate proposal noise and (ii) the acceptance noise. The candidate proposal noise tells us where the candidate points are. Since we already observe the accepted points, the missing part is the rejected points. Similar to Noorbakhsh & Rodriguez (2022), we sample the rejected points from the 'rejected' intensity equal to $\lambda^*_{rej} = \lambda_{ub} - \lambda^*_{obs}$, at each interval of the Ogata's algorithm. In addition to this, a noise variable is defined as the acceptance noise at every candidate point, i.e., at all rejected and accepted points.

For accepted points, rejected points and the noise posteriors to be well defined, we choose an upper-bound intensity λ_{ub} that is an upper bound for both the observational intensity λ^*_{obs} and the counterfactual intensity λ^*_{cf}, for each interval $[\tau, \tau + l(\tau)]$:

$$\lambda_{ub} \geq \sup_{s \in [\tau, \tau+l(\tau)]} \{\lambda^*(s) : \lambda^* \in \{\lambda^*_{obs}, \lambda^*_{cf}\}\}.$$

In practice, we select the function $l(\tau)$ so that it returns the next event after time τ, which can be a future treatment or an outcome. The observation period $[0, T]$ is split into intervals with end points $(0, t_1, \ldots, t_N, T)$, so that we can compute both the observational intensity λ^*_{obs} and the counterfactual intensity λ^*_{cf} for all points in each interval, conditioned on their respective histories $\mathcal{H}_{obs}$ and $\mathcal{H}_{cf}$. Here, the ordered set of events $t_1, \ldots, t_N$ are equal to the sorted union of all treatments and outcomes: $t_1, \ldots, t_N = \textsc{Sort}(\mathbf{t_a} \cup \mathbf{t_o})$.

Algorithm 2 Counterfactual Sampling Algorithm for TPPs (Based on Ogata's)

Input: Start T_1, End T_2, Interval function $l(\cdot)$, Observational intensity $\lambda^*_{obs}(\cdot)$, Counterfactual intensity $\lambda^*_{cf}(\cdot)$, Observed points $\mathcal{T}_{obs}$

Output: Counterfactual point process sample $\mathcal{T}_{cf} = \{t_1, \ldots, t_n\}$

```
 1: function SAMPLE($T, l, \lambda^*$)
 2:     $\tau = T_1, \mathcal{T} = \emptyset$.                                              ▷ Initialize
 3:     while $\tau < T_2$ do
 4:         $\lambda_{ub} = \sup_{s \in [\tau, \tau + l(\tau)]} \{\lambda^*(s) : \lambda^* \in \{\lambda^*_{obs}, \lambda^*_{cf}\}\}$.        ▷ Shared upper-bound.
 5:         $t_{rej} = \text{SAMPLE-OGATA}(\tau, \tau + l(t), \lambda_{ub}, \lambda_{ub} - \lambda^*_{obs})$.   ▷ Draw a rejected point (Algorithm 1).
 6:         if $t_{rej} \leq l(\tau)$ and $t_{rej} + \tau \leq T$ then
 7:             $u_{rej} \sim \mathcal{U}(\lambda^*_{obs}(\tau + t_r, \mathcal{H}_{obs}), \lambda_{ub})$.                 ▷ Noise posterior.
 8:             if $u_{rej} \leq \lambda^*_{cf}(\tau + t_r)$ then
 9:                 $\mathcal{T}_{cf} = \mathcal{T}_{cf} \cup \{\tau + t_r\}$.
10:             end if
11:             $\tau = \tau + t_r$.
12:         else                                                              ▷ No rejections in interval.
13:             if $\tau + l(\tau) \in \mathcal{T}_{obs}$ then                          ▷ Check if the end point is observed.
14:                 $t_{obs} = \tau + l(\tau)$.
15:                 $u_{obs} \sim \mathcal{U}(0, \lambda^*_{obs}(t_{obs}, \mathcal{H}_{obs}))$.                        ▷ Noise posterior.
16:                 if $u_{obs} \leq \lambda^*_{cf}(t_{obs}, \mathcal{H}_{cf})$ then
17:                     $\mathcal{T}_{cf} = \mathcal{T}_{cf} \cup \{t_{obs}\}$.
18:                 end if
19:             end if
20:             $\tau = \tau + l(\tau)$.
21:         end if
22:     end while
23:     return $\mathcal{T}_{cf} = \{t_1, \ldots, t_n\}$.
24: end function
```

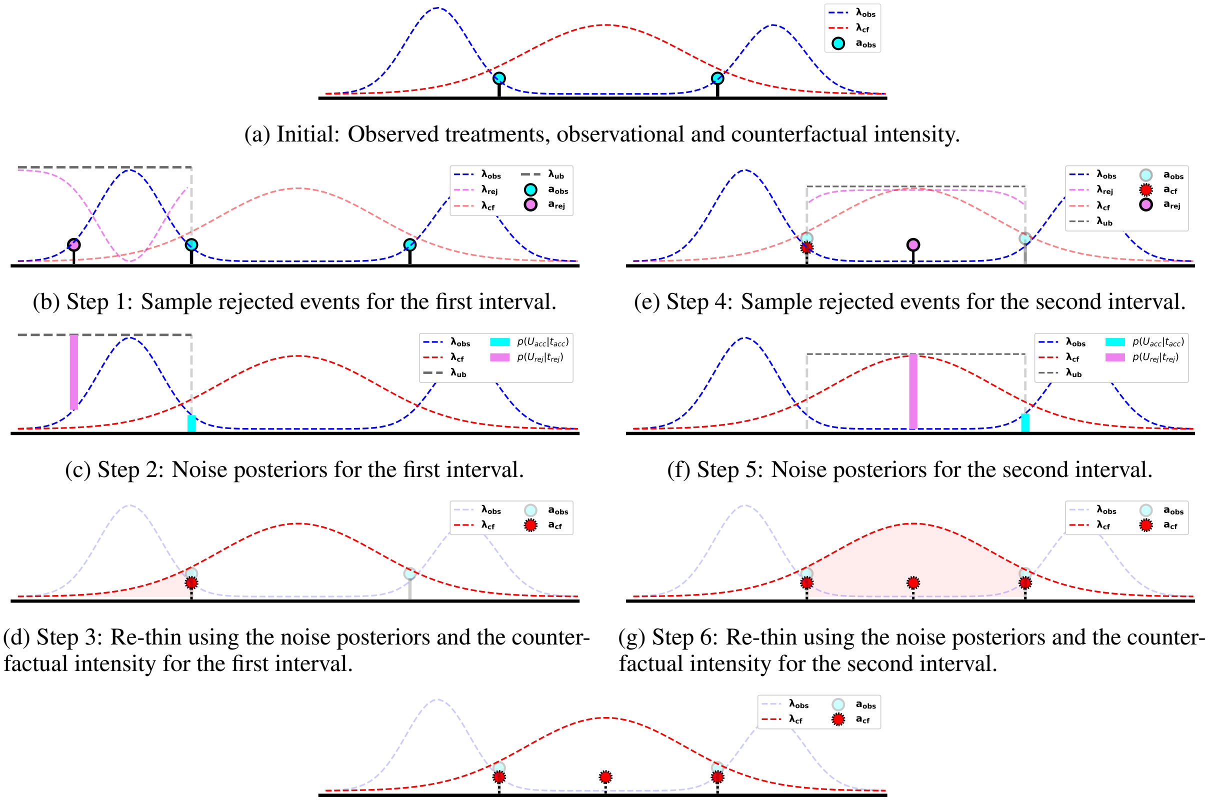
The algorithm performs three main tasks, for each interval $[\tau, \tau + l(\tau)]$. This procedure is described below in words and illustrated in Fig. 6.

1. (Figs. 6b and 6e) The algorithm finds the upper-bound intensity λ_{ub} for the interval $[\tau, \tau + l(\tau)]$ and it samples rejected events by using the difference of the upper bound intensity and the observational intensity: $\lambda^*_{rej} = \lambda_{ub} - \lambda^*_{obs}$.
2. (Figs. 6c and 6f) It computes the posterior distribution of each acceptance noise variable U_i in the interval $[\tau, \tau + l(\tau)]$, e.g. a rejected point t_{rej} or an observed point t_{obs}. By inspecting the sampling process in Algorithm 1, we can compute the noise posterior for a noise variable U_i as follows:

$$p(U_i \mid t_i, \lambda_{ub}, \lambda^*_{obs}) = \begin{cases} \mathcal{U}(0, \lambda^*_{obs}(t_i)), & \text{if } t_i \text{ is observed,} \\ \mathcal{U}(\lambda^*_{obs}(t_i), \lambda_{ub}), & \text{if } t_i \text{ is rejected.} \end{cases}$$

3. (Figs. 6d and 6g) Then, it uses the counterfactual intensity and the noise posteriors to re-thin the candidate accepted and rejected points.

As a result, once we have access to (i) observed (accepted) events, (ii) rejected events, (iii) posterior distributions of noise variables, (iv) the observational intensity and (v) the counterfactual intensity, we can sample point process realizations from the counterfactual distribution, in a similar fashion to the forward sampling process of Ogata's algorithm. The counterfactual sampling algorithm is presented in Algorithm 2. There are three key differences to the observational sampling process in Algorithm 1: (i) in each interval, the algorithm samples hypothetical rejected events using the rejection intensity $\lambda_{ub} - \lambda^*_{obs}$ (Line 5), (ii) the counterfactual algorithm uses the noise posteriors ($p(U_i \mid t_i)$) instead of noise priors $\mathcal{U}(0, \lambda_{ub})$ (Lines 7,15) and (3) the accept/reject decisions are based on the counterfactual intensity λ^*_{cf} instead of the observational intensity λ^*_{obs} (Line 8,16).



(a) Initial: Observed treatments, observational and counterfactual intensity.

(b) Step 1: Sample rejected events for the first interval.

(e) Step 4: Sample rejected events for the second interval.

(c) Step 2: Noise posteriors for the first interval.

(f) Step 5: Noise posteriors for the second interval.

(d) Step 3: Re-thin using the noise posteriors and the counterfactual intensity for the first interval.

(g) Step 6: Re-thin using the noise posteriors and the counterfactual intensity for the second interval.

(h) Final: Counterfactual treatments.

Figure 6: The algorithm is illustrated on a toy example. **(a-h)** The observed (cyan), rejected (pink) and counterfactual (red) points are shown by filled circles. The observational (blue), rejected (pink) and counterfactual (red) intensities are shown in dashed lines. The noise posteriors for the accepted points (cyan) and the rejected points (pink) are shown by filled rectangles. **(a)** The initial setup with observed treatments, the observational and the counterfactual intensities. **(b-d)** We demonstrate the three tasks for the first interval, which is up until the first observed point. **(b-d)** The same procedure for the second interval, which is between two observed points. **(h)** The final counterfactual treatments. Notice how the algorithm chooses to keep the observed treatments where the counterfactual intensity is larger than the observed intensity.

H.2.3. Identifiability of the Counterfactual Query

In this section, we first describe the monotonicity condition for binary SCMs. Then, we show that our counterfactual TPP algorithm satisfies the monotonicity condition and hence the counterfactual identifiability. We conclude by discussing the identifiability of the counterfactual outcome trajectory.

Consider an SCM with a binary intervention variable π and a binary target variable A. For the binary SCM, the monotonicity condition (Pearl, 2009; Oberst & Sontag, 2019; Noorbakhsh & Rodriguez, 2022) is a sufficient condition to render the counterfactuals of the target variable A identifiable. It is stated formally as:

Monotonicity (Noorbakhsh & Rodriguez, 2022). The binary target variable A is monotonic with respect to the binary intervention variable π, if the following condition holds for $a \neq a'$:

$$\mathbb{E}[A = a \mid do(\pi = 1)] \geq \mathbb{E}[A = a \mid do(\pi = 0)] \implies P(A = a' \mid A = a, \pi = 0, do(\pi = 1)) = 0. \tag{22}$$

The monotonicity condition states that an intervention which increases the probability of an already observed event $A = a$ can not produce a counterintuitive counterfactual value $A = a'$.

Let us denote the observational regime by $\pi = 0$ and the interventional regime by $\pi = 1$, such that $\pi \in \{0, 1\}$. Similarly, we denote their corresponding intensities by λ_0^* and λ_1^*. For a given point at time t, the probability of acceptance A under a regime $\pi \in \{0, 1\}$ is proportional to its intensity and the upper bound, e.g., for $\pi = 0$:

$$p(A \mid t, \lambda_{ub}, do(\pi = 0)) = p(A \mid t, \lambda_{ub}, \pi = 0) = \frac{\lambda_0^*(t)}{\lambda_{ub}}.$$

Without loss of generality, we consider a single time point t for our algorithm. An input point to the algorithm at time t is either an accepted ($A = 1$) or a rejected ($A = 0$) point:

- For a rejected point t with $A = 0$, the probability of the point to be kept as a counterfactual point is zero, after an intervention that decreases the intensity at time t:

  $$\mathbb{E}[A = 0 \mid do(\pi = 1)] \geq \mathbb{E}[A = 0 \mid do(\pi = 0)] \implies P(A = 1 \mid A = 0, \pi = 0, do(\pi = 1)) = 0. \tag{23}$$

  This follows from the posterior distribution of the rejected point $U_{rej} \sim \mathcal{U}(\lambda_0^*(t), \lambda_{ub})$ being larger than the interventional intensity $\lambda_1^*(t) \leq \lambda_0^*(t)$.

- For an accepted point t with $A = 1$, the probability of the point to be removed from the counterfactual set is zero after an intervention that increases the intensity at time t:

  $$\mathbb{E}[A = 1 \mid do(\pi = 1)] \geq \mathbb{E}[A = 1 \mid do(\pi = 0)] \implies P(A = 0 \mid A = 1, \pi = 0, do(\pi = 1)) = 0. \tag{24}$$

  This follows from the posterior distribution of the accepted point $U_{acc} \sim \mathcal{U}(0, \lambda_0^*(t))$ being smaller than the interventional intensity $\lambda_1^*(t) \geq \lambda_0^*(t)$.

Eqs. (23) and (24) imply that the counterfactual sampling algorithm satisfies the monotonicity condition in Eq. (22), and hence the counterfactual treatments are identifiable.

The counterfactual algorithm either adds or removes a treatment to the observed treatment sequence, which produces a new or removed bump in the outcome trajectory, formalized by the treatment response function f_a, while the baseline function f_b remains unchanged. Therefore, the counterfactual outcome trajectory is identifiable, conditioned on the identifiability of the treatment response function $f_a(\tau; \mathbf{a})$, which is not possible without further functional assumptions. In our semi-synthetic experiments, we empirically show that it is possible to identify the treatment response function using distinctive functional priors for the baseline and the treatment response, when the outcome trajectory follows our model definition. When this is the case, the counterfactual trajectory can be empirically estimated with good accuracy.

I. Experiment Details

I.1. Real-World Study Details

I.1.1. Data Preprocessing

The real-world data set consists of meal-blood glucose measurements for 14 non-diabetic individuals, where individuals are monitored over a 3-day period (Zhang et al., 2020). The blood glucose is measured by a continuous monitoring device, which takes a sample approximately every 15 minutes. The meal data is collected through a meal diary, recorded daily by individuals. Later, meal records in the diary are translated into nutrient values by a look-up table.

As the meal data set is collected by individuals, it inherently contains measurement errors (Zhang et al., 2020). To deal with the measurement errors, we take a data preprocessing step before fitting our treatment model to the meal data. For our application, there are two main measurement errors that may lead to incorrect results: (i) individuals may record their meals in parts, where a single meal is recorded as 2-3 meal occurrences, and (ii) individuals record their meal times with error.

To deal with measurement errors of type (i), we remove redundant meals by keeping only a single meal event if two meals co-occur in a 2-hour interval. To deal with measurement errors of type (ii), we update the meal time if a meal seems to occur at a time where the blood glucose is changing rapidly. For example, if an individual reports a meal with a delay, the meal seems to occur at a time point where the blood glucose measurements have already increased rapidly and is about to decline. The recorded meal time (with error) implies an incorrect result that the meal leads to a decrease in the blood glucose, or similarly, that the meal likelihood increases with increasing blood glucose. On the contrary, the bump in the glucose level is most likely caused by the same meal in the first place. In practice, we move a meal time to the next possible time point where the blood glucose derivative is less than a threshold, 0.5. At each time point t_i, we approximate the derivative of the blood glucose by averaging derivative values for two consecutive regions, (t_{i-1}, t_i) and (t_i, t_{i+1}):

$$dy_i \approx \frac{1}{2}\left(\frac{(y_i - y_{i-1})}{t_i - t_{i-1}} + \frac{(y_{i+1} - y_i)}{t_{i+1} - t_i}\right).$$

I.1.2. Joint Model Definition

Treatment Intensity. We assume each patient's meal habits are different, corresponding to a distinct treatment policy. Furthermore, we define five model variants $\{\lambda_b, \lambda^*_{ba}, \lambda^*_{bo}, \lambda^*_{ao}, \lambda^*_{bao}\}$ for each patient (see Table 1 for details). Each treatment model is defined and trained independently on each patient's meal–glucose measurements.

We choose the constant baseline intensity β_0 to be equal to 0.1. For the treatment-dependent function $g^*_a(\tau; \mathbf{a})$, we use a relative-time kernel k_t only, i.e. we assume meal dosages (marks) do not have an affect on the next meal time. For the outcome-dependent function $g^*_o(\tau; \mathbf{o})$, we use a combination of a mark kernel k_m and a relative-time kernel k_t. We choose the dimensionality parameters $\{Q_a, Q_o\}$ of the kernels as $Q_a = 1, Q_o = 1$, i.e., the meal intensity depends only on the last treatment and the last outcome observations. This is a simplification of complex real-world dynamics between meals and blood glucose. Yet, our goal is not to model these dynamics in the best possible way in this work, but rather to understand if our joint model can learn clinically meaningful treatment intensities, that can handle time-varying confounding.

At a query time τ, the retrieval function $r(\tau)$ returns the input vector $r(\tau) = \mathbf{r} \in \mathbb{R}^4$, as defined in Appendix E.2:

$$r(\tau) = \{ \underbrace{\tau}_{\substack{\text{Input for}\\ \text{Baseline}}}, \underbrace{\tau - t_i}_{\substack{\text{Input for}\\ \text{last } Q_a\\ \text{treatment}}}, \underbrace{\tau - t_j, y_j}_{\substack{\text{Input for}\\ \text{last } Q_o\\ \text{outcome}}} \}.$$

The intensities $\{\lambda_b, \lambda^*_{ba}, \lambda^*_{bo}, \lambda^*_{ao}, \lambda^*_{bao}\}$ use the relevant dimensions of the input. For example, the intensity λ_b uses only the absolute time information (first dimension) τ, which is for the baseline input.

We choose $M = 20$ inducing points: $\mathbf{Z} \in \mathbb{R}^{M \times 4}$. We place the inducing points in regular intervals inside the target domain of the treatment intensity component g_b, g^*_a or g^*_o. For example, the baseline component's input domain is the observation period $[0, T]$ and inducing points for the baseline dimension are placed onto $[0, T]$ regularly. Similarly, input domains of treatment-effect and outcome-effect kernels include relative times to the last Q_a actions and Q_o outcomes. For these two dimensions, we place inducing points between 0 and the maximum (or some quantile) time between the last Q_a actions or Q_o outcomes. Inducing points are assumed to be independent along each dimension similar to Liu & Hauskrecht (2019).

We choose kernel hyperparameters $\theta_{a,h} = \{\gamma_q, \ell_q\}_{q=0}^{Q_a+Q_o}$ by inspection, since our data set is small and there exists identifiability issues when one optimizes the effect of each component on the treatment effect simultaneously. As the number of meal events are small in the data set, we choose relatively small values for the variance parameters $(\gamma_b, \gamma_a, \gamma_o) = (0.1, 0.05, 0.15)$. We choose lengthscale parameters as $(\ell_b, \ell_a, \ell_{o,t}, \ell_{o,m}) = (7.0, 1.0, 100.0, 2.5)$. Each lengthscale value represents an underlying assumption based on the domain knowledge: (i) the baseline lengthscale $\ell_b = 7.0$ assumes a slow-changing smooth baseline function, (ii) the treatment-effect time lengthscale ℓ_a is 1.0 as the last meal generally has an effect on the next meal in the next 2-3 hours time span, (iii) a very large outcome-effect time lengthscale $\ell_{o,t} = 100$ is chosen so that the outcome-effect is constant in time for a given glucose value y, mimicking a simultaneous outcome-effect on the meal intensity with a large number of regular glucose measurements, (iv) the outcome-effect mark lengthscale $\ell_{o,m} = 2.5$ is chosen so that the mark effect is also a slow-changing, smooth function, as the glucose level should gradually affect meal intensity.

Outcome Model. We define a single hierarchical outcome model for all patients, as detailed in Appendix F.

The baseline progression f_b for each patient (v) is the sum of a constant kernel and a periodic kernel whose period is set to 24 hours. The periodic kernel captures daily glucose profiles of patients. The constant kernel function has an intercept parameter $\{b_b^{(v)}\}$, initialized to $\{1.0\}$. Each periodic kernel has parameters $\{\alpha_b^{(v)}, \ell_b^{(v)}\}$, initialized to $\{1.0, 1.0\}$.

The treatment response function is equal to the sum of two components: $f_a(\tau; \mathbf{a} = (\mathbf{m}, \mathbf{t})) = f_m(\mathbf{m}) f_t(\tau; \mathbf{t})$. The response shape function $f_t(\tau; \mathbf{t})$ has an SE kernel with hyperparameters $\{\ell_t^{(m)}\}$ for each treatment type (m). We assume all meals are of the same type: $m = 1$. We initialize the lengthscale to 0.5: $\ell_t^{(m)} = \ell_t = 0.5$. The effective interval $\mathcal{T}_{t_i}$ for each meal t_i is set to 3 hours: $\mathcal{T}_{t_i} = 3$. The treatment response scaling function $f_m^{(v)}$ specifies the amplitude of the treatment response for each patient (v), using patient-specific intercept and slope parameters: $\{\beta_0^{(v)}, \beta_1^{(v)}\}$. A hierarchical model is imposed on the intercept and the slope parameters $\{\beta_0^{(v)}, \beta_1^{(v)}\}_{v=1}^{14}$ by defining a hierarchical Gaussian prior on them: $\beta_0^{(v)} \sim \mathcal{N}(\beta_0, \sigma_0^2), \beta_0^{(v)} \sim \mathcal{N}(\beta_1, \sigma_1^2), \forall v \in \{1, \dots, 14\}$. All scaling function parameters $\{\{\beta_0^{(v)}, \beta_1^{(v)}\}_{v=1}^{14}, \beta_0, \beta_1, \sigma_0, \sigma_1\}$ are initialized to 0.1.

Training. We assume each day of meal observations as conditionally independent given the conditional intensity function. We train each model on the first two days of observations and use the third day as the test set. For the outcome model, the measurements follow a Gaussian likelihood, i.e., an exact GP model. We maximize the log marginal likelihood with respect to the hyperparameters $\theta_o = \{\beta_0, \beta_1, \{b^{(v)}, \gamma_b^{(v)}, \ell_b^{(v)}, \beta_0^{(v)}, \beta_1^{(v)}\}_{r=1}^{14}\}$. We fix hyperparameters $\{\sigma_0, \sigma_1\}$ at their initialized values. For the treatment model, we maximize the training objective ELBO with respect to the variational parameters $\theta_{a,z} = \{\mathbf{m_z}, \mathbf{S_z}\}$, while the treatment hyperparameters $\theta_{a,h}$ are fixed at their initial values.

Table 4: Results for the test log likelihood (TLL) values for treatment models with distinct meal intensities, for all patients. Models are trained on first 2-days of meal-glucose data. The last day is used for computing the TLL. The intensity $\lambda_{ba}^*(\tau)$ provide the highest TLL value for all patients.

		Meal Intensities				
Patient-id	Metric	λ_b^*	λ_{ba}^*	λ_{bo}^*	λ_{ao}^*	λ_{bao}^*
Patient 1	TLL	-10.38	-9.89	-9.84	$\mathbf{-7.68}$	-10.68
Patient 2		-14.37	-11.39	$\mathbf{-10.57}$	$\mathbf{-10.58}$	-11.85
Patient 3		-12.07	-12.85	$\mathbf{-12.05}$	-12.78	-12.79
Patient 4		-12.90	-12.75	$\mathbf{-9.38}$	-9.61	-11.52
Patient 5		-14.93	-11.93	$\mathbf{-10.99}$	-11.12	-12.35
Patient 6		-13.46	-9.52	$\mathbf{-8.51}$	-8.59	-9.73
Patient 7		-13.87	-12.19	-12.47	$\mathbf{-10.86}$	-13.98
Patient 8		-12.79	-12.62	-10.85	$\mathbf{-9.70}$	-12.20
Patient 9		-13.60	-14.07	-13.01	$\mathbf{-12.91}$	-13.95
Patient 10		-16.29	-13.10	$\mathbf{-11.37}$	$\mathbf{-11.39}$	-12.61
Patient 11		-13.49	-12.95	-12.33	$\mathbf{-12.10}$	-14.00
Patient 12		-15.75	-13.82	$\mathbf{-12.07}$	-13.61	-14.79
Patient 13		$\mathbf{-9.71}$	-10.81	-10.07	-9.99	-11.67
Patient 14		-12.08	-12.50	-10.95	$\mathbf{-9.13}$	-12.88
All	TLL	-13.26 ± 0.49	-12.17 ± 0.36	-11.03 ± 0.34	-10.72 ± 0.46	-12.50 ± 0.37

Results. For the treatment model, we report the test log-likelihood (TLL) for the third day to compare different treatment intensity variants. The computation of TLL is detailed in Appendix H.1. TLL results for all patients are shown in Table 4. In agreement with the mean TLL value, the intensity λ_{ao}^* produce largest TLL values for most patients. In addition, the simplest (baseline) intensity λ_b with no past treatment-outcome effect and the most complex intensity λ_{bao}^* with all components provide low TLL values in general. This suggests that a treatment model independent of the past history is not able to explain the meal intensity well. Similarly, the intensity with all components seem to be overly complex for this small data set. In addition, we demonstrate the model fits for all intensities in Fig. 7.

For the outcome model, we examine train and test fits for patients qualitatively. We show train/test fits for a subset of patients $\{3, 8, 12\}$ in Fig. 8. The empirical findings suggest that the outcome model is able to capture clinically meaningful treatment responses that are close to bell-shaped curves reported in Wyatt et al. (2021). It is important to note that the treatment response curves are the main learning objective in continuous-time treatment–outcome setups (Schulam & Saria, 2015; Xu et al., 2016; Schulam & Saria, 2017; Soleimani et al., 2017; Cheng et al., 2020; Seedat et al., 2022).

I.2. Semi-Synthetic Simulation Study Details

We perform a semi-synthetic simulation study to evaluate our model's performance on the causal tasks, as the true causal effects are unknown in a real-world observational data set.

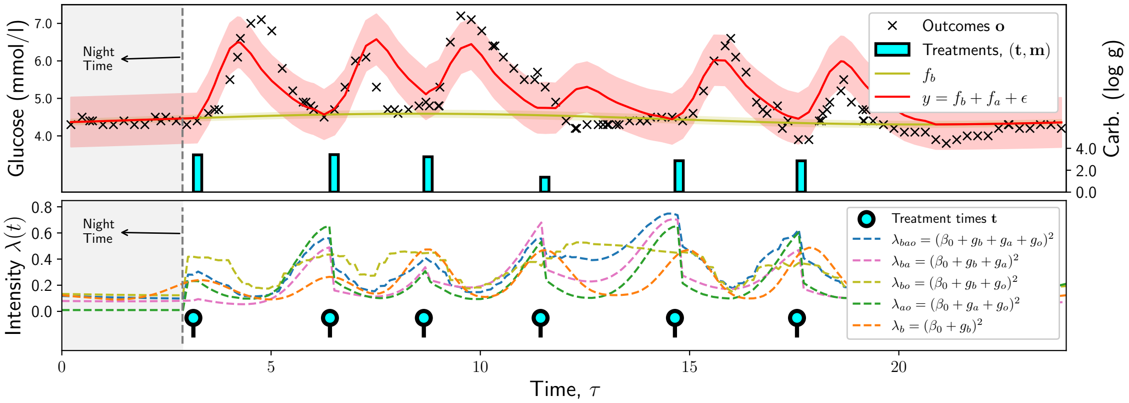


Figure 7: Estimated treatment–outcome model on the real-world meal–glucose data, similar to Fig. 4. **(Top)** Estimated outcome model on the glucose measurements (see Fig. 4 for details). **(Bottom)** Estimated intensities for all five treatment models $\{\lambda_b, \lambda^*_{ba}, \lambda^*_{bo}, \lambda^*_{ao}, \lambda^*_{bao}\}$ are shown. Inspecting intensities λ^*_{ba} (pink), λ^*_{ao} (green) and λ^*_{bao} (blue), we see that the intensity of a new meal immediately after a previous meal decreases through g^*_a. Similarly, inspecting intensities λ^*_{bo} (yellow), λ^*_{ao} (green) and λ^*_{bao} (blue), we see that the intensity of a new meal immediately after a previous meal decreases indirectly through the increase in blood glucose represented by g^*_o.

I.2.1. SIMULATOR

Simulator functions. Our ground-truth joint simulator is composed of a treatment and an outcome simulator. The ground-truth treatment simulator, has the form of a treatment- and outcome-dependent intensity λ^*_{ao} (defined in Table 1) to include both sources of time-varying confounding. The ground-truth outcome simulator has the form of the hierarchical outcome model (defined in Eq. (8)). The model definitions, initialization and the training procedures are detailed in Appendix I.1.2.

We fit our joint model to a subset of the real-world meal–glucose data set to obtain ground-truth simulator functions $\mathcal{M}^{(v)}_{gt} = \{\lambda^*_{\pi_v}(\tau), f^{(v)}_b(\tau), f^{(v)}_a(\tau; \mathbf{a})\}$. In the rest of this section, we show the characteristics of these functions and what kind of observational trajectories they produce.

Treatment simulator. We divide simulated patients into two policy groups $\{\pi_A, \pi_B\}$, representing distinct treatment policies of different hospitals, countries, etc. The simulator intensities of two policies correspond to learned intensity functions of two real-world patients: Patients $\{4{,}13\}$. We choose these two intensities, so that the treatment distributions induced by two policies show different characteristics, i.e. observational and interventional distributions are not similar. Otherwise, an intervention on the treatment policy might not have a visible effect on the treatment distribution. For example, consider the limiting case of having the same treatment intensity for both policies $\{\pi_A, \pi_B\}$, then the observational and the interventional distributions are identical.

Comparing simple statistics of two patients, Patient 4 has 5 meals on average per day, while Patient 13 has 4 meals on average per day, which leads to higher intensity values for Patient 4, and hence lower expected meal arrival times. Besides, the learned mark intensity functions $p(m \mid \tau)$ for both policies are distinct. The ground-truth functions g^*_a g^*_o and $p(m \mid \tau)$, that are learned from Patients $\{4{,}13\}$ are shown in Fig. 9.

Outcome simulator. To enable individualization among the blood glucose dynamics of patients, we assume there are three patient groups $\{gr_0, gr_1, gr_2\}$. Patients in the same group share the baseline and treatment response functions, e.g., patients in the group gr_1 have the following functions: $\{f_{b,gr_1}, f_{a,gr_1}\}$.

We choose three outcome simulator functions corresponding to the learned outcome models of three real-world patients (Patients $\{3{,}8{,}12\}$), by qualitatively assessing the learned functions with respect to the domain knowledge and findings about the effect of a meal on the blood glucose in Wyatt et al. (2021): (i) they have distinct daily baseline characteristics, fluctuating around a constant value (Fig. 10a) (Ashrafi et al., 2021) (ii) their treatment response curves are smooth functions with similar shapes to the bell-shaped response curves argued in Wyatt et al. (2021) (Fig. 10b) . The train and test fits of the

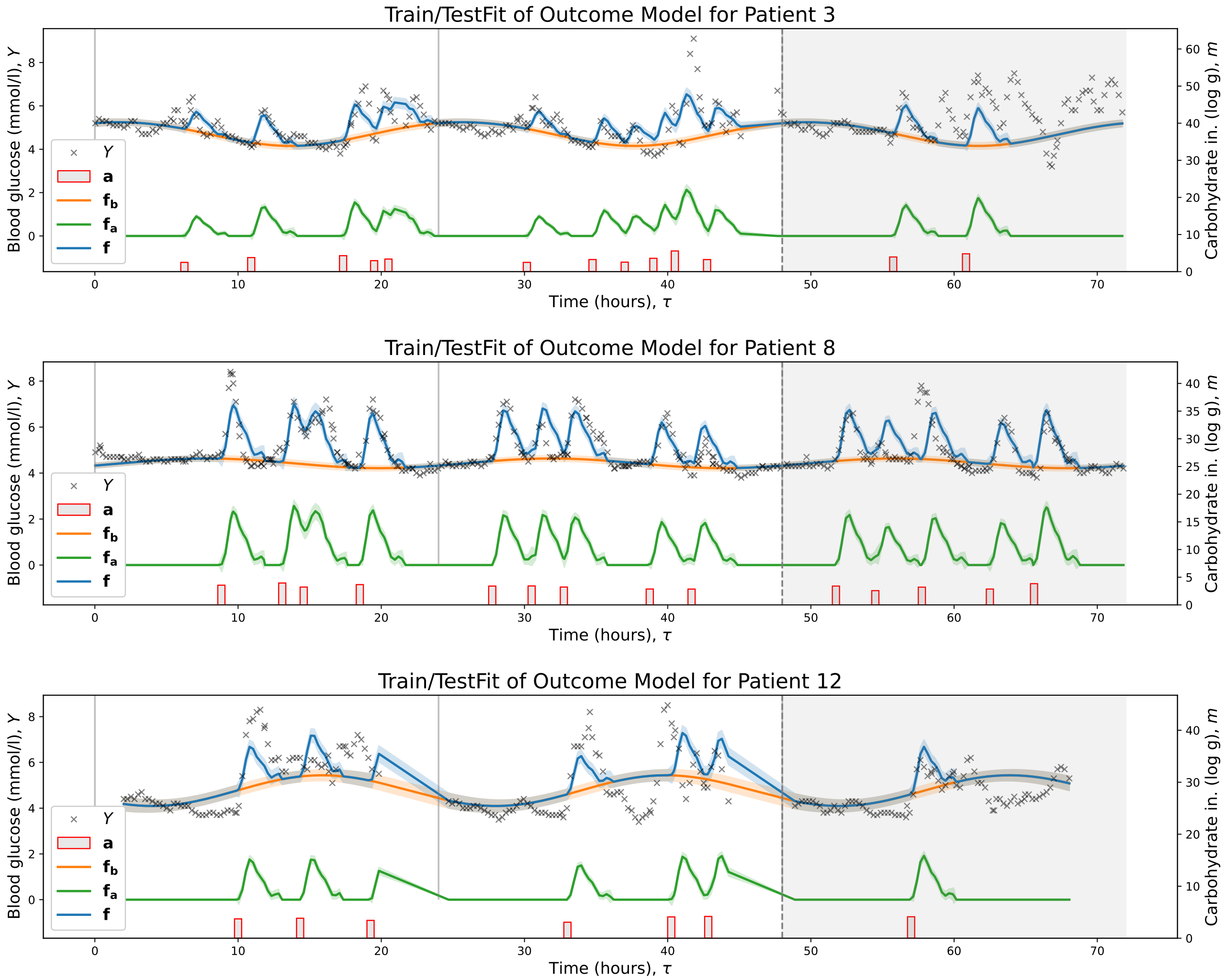


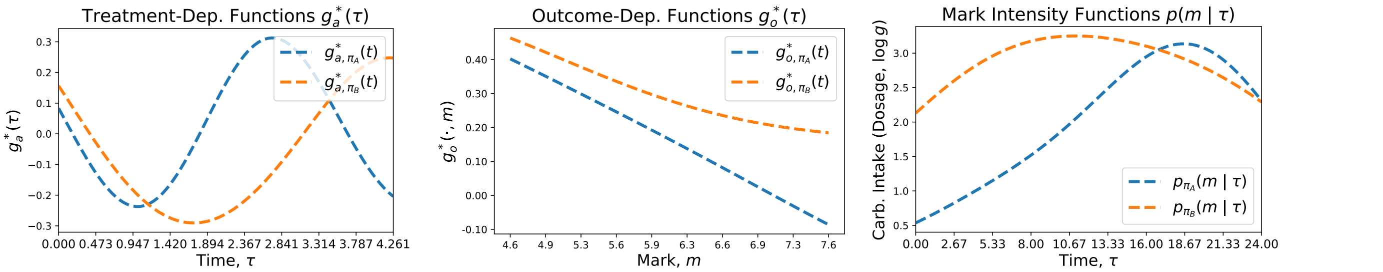
Figure 8: The learned ground-truth functions $f_b(\tau)$, $f_a(\tau)$ and $f(\tau) = f_b(\tau) + f_a(\tau)$ for Patients $\{3,8,12\}$. The grey-shaded area is the test period. The baseline kernels are sums of (i) a constant kernel (Ashrafi et al., 2021) and (ii) a periodic kernel (combined with a long-lengthscale SE kernel) whose period is fixed to 1-day in order to capture the daily blood glucose profiles. The treatment shape function is close to a Gaussian bump (Wyatt et al., 2021).

outcome simulators for these patients are shown in Figure 8.

I.2.2. Benchmarks

For our estimation model, we use a joint model that has the same form with the simulator. In addition to ablations, we use two benchmark models used for comparison in Experiments:

H21 (Hua et al., 2021). They propose a joint treatment–outcome model, with a publicly available implementation. However, they assume treatments and outcomes occur jointly at the same continuous-time points, which is a strong assumption that does not hold for our problem definition. Therefore, we adapt their implementation by separating the joint likelihood into treatment and outcome parts, so that the model can work with treatments and outcomes that occur at different time points. Besides, we remove the irrelevant survival outcome part from the joint likelihood. The implementation uses a Markov chain Monte Carlo (MCMC) algorithm. We run the algorithm for 20000 iterations, with a burn-in period of 5000 iterations. To deal with sample correlation, we use every 50th sample after the burn-in period for inference. During inference, we use the mean values for the treatment and outcome model parameters.

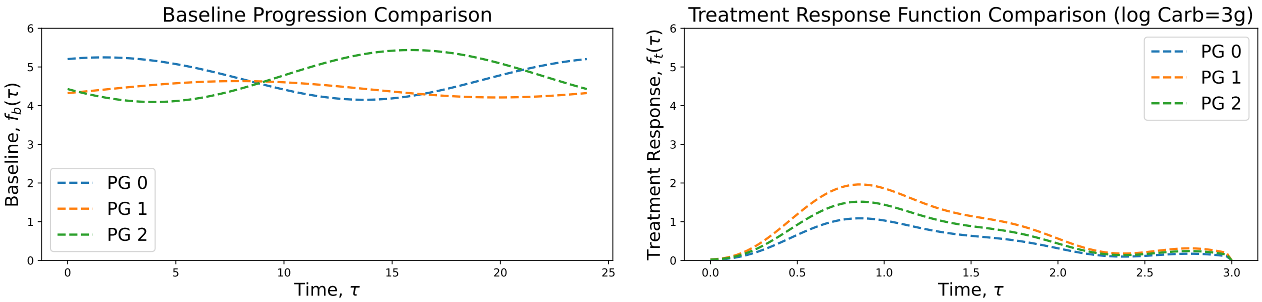


(a) Dependence on last meal g_a^*.

(b) Dependence on last glucose value g_o^*.

(c) Carb. intake intensity $p(m \mid \tau)$.

Figure 9: The learned simulator functions functions g_a^*, g_o^* and $p(m \mid \tau)$ for Patient 4 (blue dashed lines) and Patient 13 (orange dashed lines). **(a)** After a meal occurs, both intensities first decline and then increase to account for the next meal. The bump for the next meal occurs earlier for Patient 4 than Patient 13. Additionally, the magnitude of the bump is higher for Patient 4 than Patient 13. **(b)** The meal intensity of Patient 13 increases slightly more than Patient 4 for all blood glucose values. **(c)** Patient 13 is more likely to have meals with higher carbohydrate intake during early hours of the day, while Patient 4 is more likely to have higher carbohydrate intake after 5pm.



(a) Baseline progressions f_b in the interval $[0, 24]$ (hours).

(b) Treatment response curves f_a in the interval $[0, 3]$ (hours).

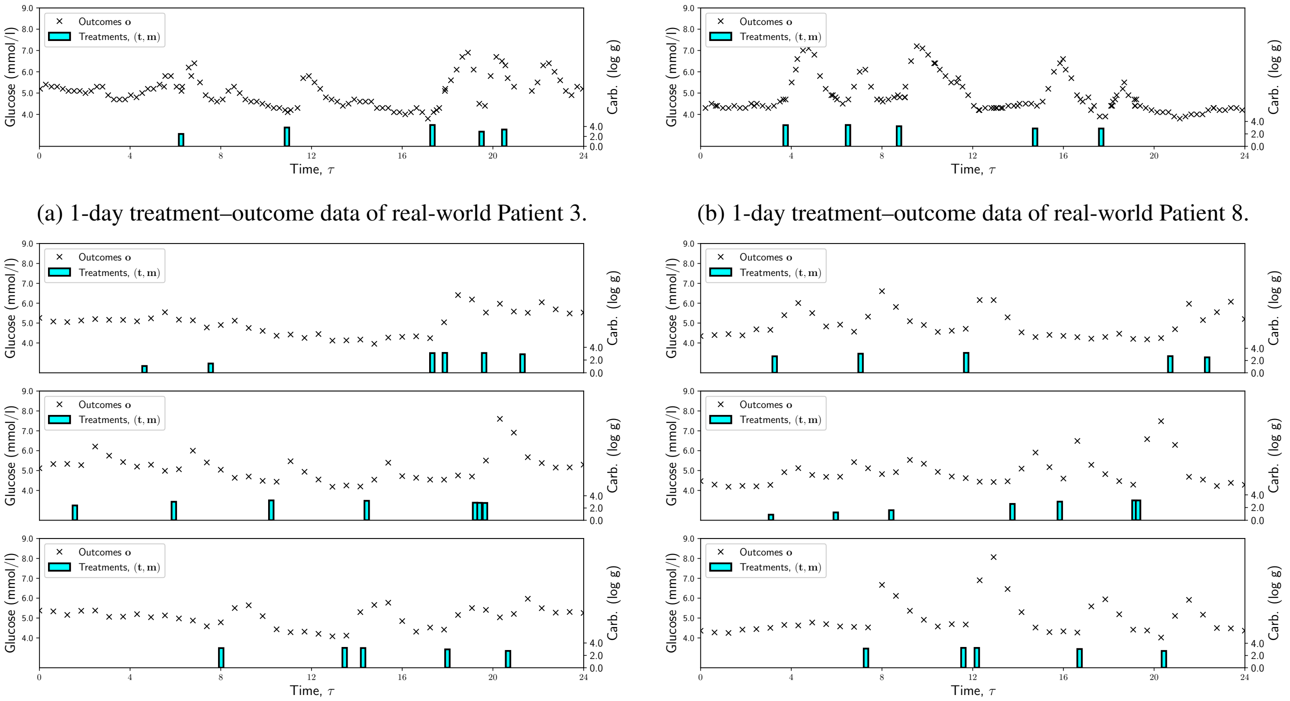
Figure 10: Baseline progression and treatment response functions of Patient 3 (blue), Patient 8 (orange) and Patient 12 (green), i.e., of the ground-truth outcome simulator.

S17 (Schulam & Saria, 2017). The outcome model proposed in Schulam & Saria (2017) is a conditional GP, with additive baseline and treatment response functions. We use their publicly available implementation that uses (i) the sum of a mixture of B-splines and a GP prior with a Matern kernel for the baseline function and (ii) a constant treatment response. It uses marginal likelihood for learning and GP posteriors for inference.

I.2.3. Semi-synthetic Dataset

We use the selected treatment and outcome simulators described in Appendix I.2.1 to simulate semi-synthetic samples from observational, interventional and counterfactual distributions. For the observational data set, we simulate 1-day long treatment–outcome sequences for 50 patients: $\mathcal{D} = \{\mathcal{D}^{(v)}\}_{v=1}^{50}$. We assume the outcome trajectory is measured in regular intervals, similar to a continuous glucose monitoring device. We divide the 1-day long observation period into 40 intervals and sample joint trajectories using the treatment and outcome simulators. We show some of the sampled outcome trajectories in Fig. 11, together with 1-day long examples of treatment–outcome sequences of two real-world patients {3,8}. We train all benchmark models on the observational data set, while using interventional and counterfactual data sets for test.

We train our hierarchical outcome model on the observational data set to obtain estimated outcome model components for all patients: $\{\hat{f}_b^{(v)}(\tau), \hat{f}_a^{(v)}(\tau; \mathbf{a}), \hat{\epsilon}^{(v)}(\tau)\}_{v=1}^{50}$. For each policy $\pi \in \{\pi_A, \pi_B\}$, we train a treatment model on the population data which follows the target policy: $\{\hat{\lambda}_A^*(\tau), \hat{\lambda}_B^*(\tau)\}$. Overall, we have the following estimated model components: $\mathcal{M}_{est} = \{\hat{\lambda}_A^*(\tau), \hat{\lambda}_B^*(\tau)\} \cup \{\hat{f}_b^{(v)}(\tau), \hat{f}_a^{(v)}(\tau; \mathbf{a}), \hat{\epsilon}^{(v)}(\tau)\}_{v=1}^{50}$.



(a) 1-day treatment–outcome data of real-world Patient 3.

(b) 1-day treatment–outcome data of real-world Patient 8.

(c) Outcome trajectory samples for 3 synthetic patients that use the baseline and response functions of Patient 3.

(d) Outcome trajectory samples for 3 synthetic patients that use the baseline and response functions of Patient 8.

Figure 11: Observational outcome trajectory samples of patients belonging to two different patient groups: **(left)** patient group 1 and (**right**) patient group 2. **(a-b)** 1-day long examples of treatment–outcome sequences for Patient 3 and Patient 8, on which we learn the simulator functions for patient groups $\{1{,}2\}$. **(c-d)** Three joint trajectory samples from both patient groups.

Test datasets and metric. After learning model components on the observational data, we sample test data sets from the interventional and counterfactual distributions resulting from the policy interventions $[\tilde{\pi}_{>1d}]$ and $[\tilde{\pi}_{\leq 1d}]$ respectively. For the interventional data set, we sample the second day of each patient under the policy intervention $[\tilde{\pi}_{>1d}]$, where the treatment policy of the patient is switched. For the counterfactual data set, we condition on the observed data $\mathcal{D}^{(r)}$ for each patient and sample a hypothetical first day under a policy intervention $[\tilde{\pi}_{\leq 1d}]$, using the posterior of the noise variables. This is performed using the simulator functions and the counterfactual sampling algorithm detailed in Appendix H.2.2.

More specifically, our policy intervention target has the following form:

$$P(\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}] \mid \mathcal{H}_{\leq T}) = \sum_{\tilde{\mathbf{a}}_{>T}} \prod_{k=0}^{m-1} \underbrace{P(\tilde{\mathbf{a}}_{[q_k, q_{k+1})} \mid \tilde{\pi}_{>T}, \mathcal{H}_{\leq q_k})}_{\text{Treatment Model}} \underbrace{P(Y_{q_k} \mid \tilde{\mathbf{a}}_{[q_k, q_{k+1})}, \mathcal{H}_{\leq q_k})}_{\text{Outcome Model}}. \tag{25}$$

Notice that the marginal distribution of the outcome trajectory $\mathbf{Y}_{\mathbf{q}}[\tilde{\pi}_{>T}]$ is not available in closed form, as we cannot integrate out treatments $\mathbf{a}$ in Eq. (25). Nevertheless, we have a generative treatment–outcome model, from which we can sample joint treatment–outcome trajectories to compare how close estimated outcome trajectories are to the ground-truth outcome trajectories in terms of the mean squared error (MSE).

Naively sampling joint trajectories leads to misaligned treatment times, and hence misaligned treatment response curves, no matter how close estimated intensities $\{\hat{\lambda}_A^*, \hat{\lambda}_B^*\}$ are to the ground-truth simulator intensities $\{\lambda_A^*, \lambda_B^*\}$. As an example, see three i.i.d. treatment samples $\mathbf{a}^{(v)} \sim p(\mathbf{a}^{(v)} \mid \lambda_A^*), v = 1, ..., V$ from the same simulator intensity λ_A^* in Fig. 12, where the misaligned treatment responses are annotated by red circles. This misalignment is due to the inherent stochasticity of the point process sampling.

The source of the stochasticity in the Ogata's sampling algorithm is two-folds: (i) the candidate point sampled from the upper-bound Poisson process and (ii) the uniform noise variable sampled for an accept/reject decision. To achieve aligned

Algorithm 3 Ogata's Thinning algorithm for P processes with Fixed Noise

Input: Start T_1, End T_2, Interval function $l(\cdot)$, P conditional intensities $\{\lambda_p^*\}_{p=1}^P$.
Output: Samples $\mathcal{T} = \{t_1, \ldots, t_n\}$ for P point processes in the interval $[T_1, T_2]$.

```
function SAMPLE-FIXED-NOISE($T_1, T_2, l(\cdot), \{\lambda_p^*\}_{p=1}^P$, P)
    $\mathcal{T}_p = \emptyset, \forall p \in \{1, \cdots, P\}$.                ▷ Initialize point sets.
    $\tau = T_1$.                                                          ▷ Initialize time.
    while $\tau < T_2$ do
        $\lambda_{ub} = \sup_{s \in [\tau, \tau + l(\tau)]} \{\lambda_p^*(s) : p \in \{1, \cdots, P\}\}$    ▷ Shared upper-bound.
        $u_{ub}, u_a \sim \mathcal{U}(0, 1)$.                                ▷ Sample and fix noise var. for current interval.
        $t_i = -1/\lambda_{ub} * \log(u_{ub})$                                ▷ Draw the inter-arrival time, $t_i \sim Exp(\lambda_{ub})$.
        if $t_i \leq l(\tau)$ then                                          ▷ Candidate in the interval.
            for $p \in \{1, \cdots, P\}$ do
                if $u_a \leq \lambda_p^*(\tau + t_i)/\lambda_{ub}$ then          ▷ Keep with probability $\lambda_p^*/\lambda_{ub}$ for each process $p$.
                    $\mathcal{T}_p = \mathcal{T}_p \cup \{\tau + t_i\}$.          ▷ Point accepted.
                end if
            end for
            $\tau = \tau + t_i$.                                            ▷ Continue from the candidate point.
        else                                                               ▷ No candidate point in the interval.
            $\tau = \tau + l(\tau)$.                                        ▷ Continue from the end of the interval.
        end if
    end while
    return $\{\mathcal{T}_p\}_{p=1}^P$.
end function
```

point process realisations and hence comparable treatment responses, we fix the noise variables responsible for both sources of randomness and sample a ground-truth and an estimated trajectory simultaneously. If an estimated intensity $\hat{\lambda}_{\text{est}}$ and a simulator (oracle) intensity λ_{oracle} are close, the fixed-noise sampling algorithm will produce a similar set of treatments. In the case where two intensities $\hat{\lambda}_{\text{est}}$ and λ_{oracle} are equal, we will obtain an equal set of treatments $\mathbf{a}_{\text{est}}^{(v)} \equiv \mathbf{a}_{\text{oracle}}^{(v)}, v = 1, ..., V$. The algorithm is shown in Algorithm 3.

For the policy intervention task, we illustrate the sampling process that compares the ground-truth (oracle) simulator with a set of estimated models (OUR$_{\text{int}}$, OUR$_{\text{obs}}$, AB1$_{\text{int}}$, AB3$_{\text{int}}$) in Fig. 13, where the subscripts 'obs' (observational) and 'int' (interventional) denote the target distribution. Similarly, the superscripts 'oracle', 'ab1', 'ab3' and 'our' denote the target model. The ground-truth intensity $\lambda_{\text{int}}^{\text{oracle}}$, the ground-truth treatments $\mathbf{a}_{\text{int}}^{\text{oracle}}$ and the ground-truth trajectory $\mathbf{Y}_{\text{int}}^{\text{oracle}}$ are shown in blue, while the predictions of our model OUR$_{\text{int}}$ are shown in red. For the model OUR$_{\text{int}}$, the estimated treatment intensity $\lambda_{\text{int}}^{\text{our}}$ and the outcome trajectory $\mathbf{Y}_{\text{int}}^{\text{our}}$ follow the ground-truth intensity and trajectory well. For the model OUR$_{\text{obs}}$, the estimated treatment distribution is different than the ground-truth treatment distribution and the estimated trajectory $\mathbf{Y}_{\text{int}}^{\text{our}}$ is not close to the ground-truth trajectory. For the model AB3$_{\text{int}}$, we see that the history-independent (NHPP) intensity λ_{int}^{ab3} does not take past treatments and outcome into account and only predicts a conservative approximation on the treatment distribution. The Gamma-based treatment intensity $\lambda_{\text{int}}^{ab1}$ of the model AB1$_{\text{int}}$ is a better approximation than the history-independent intensity $\lambda_{\text{int}}^{ab3}$, however, it lacks the flexibility of our non-parametric intensity $\lambda_{\text{int}}^{\text{our}}$.

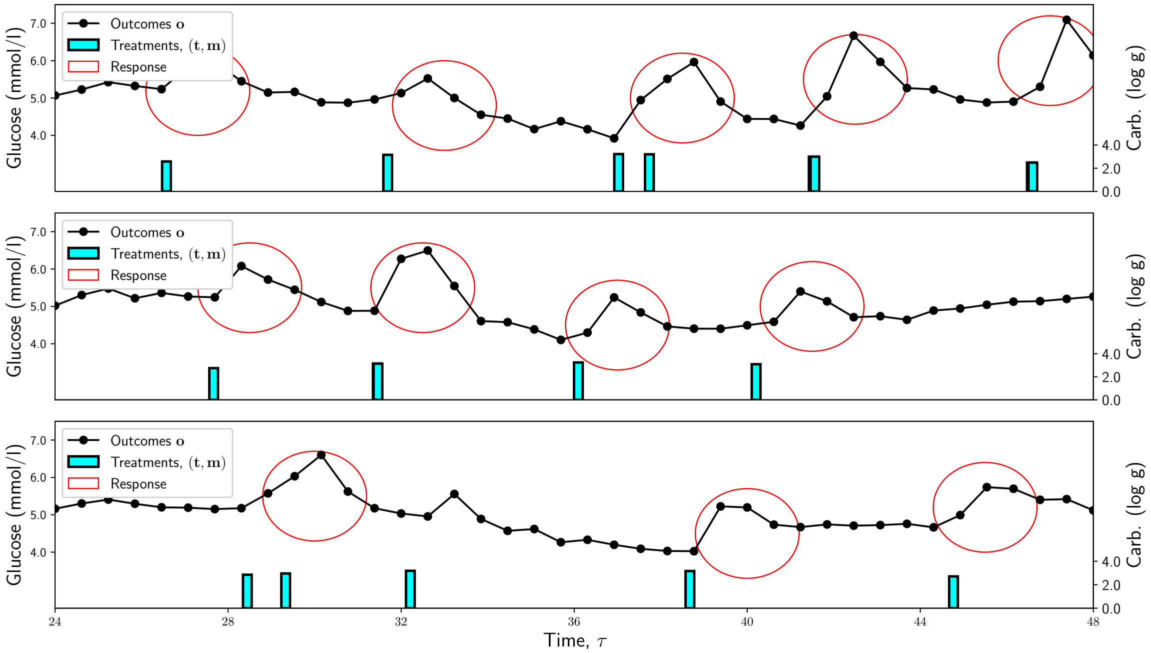


Figure 12: Three ground-truth i.i.d. treatment samples from policy A and patient group 0. Notice that sampled treatments (cyan) are not aligned, i.e., they do not occur at similar times each day. Hence, their treatment responses (annotated by red circles) are also misaligned and estimating the MSE between them naively would lead to high values.

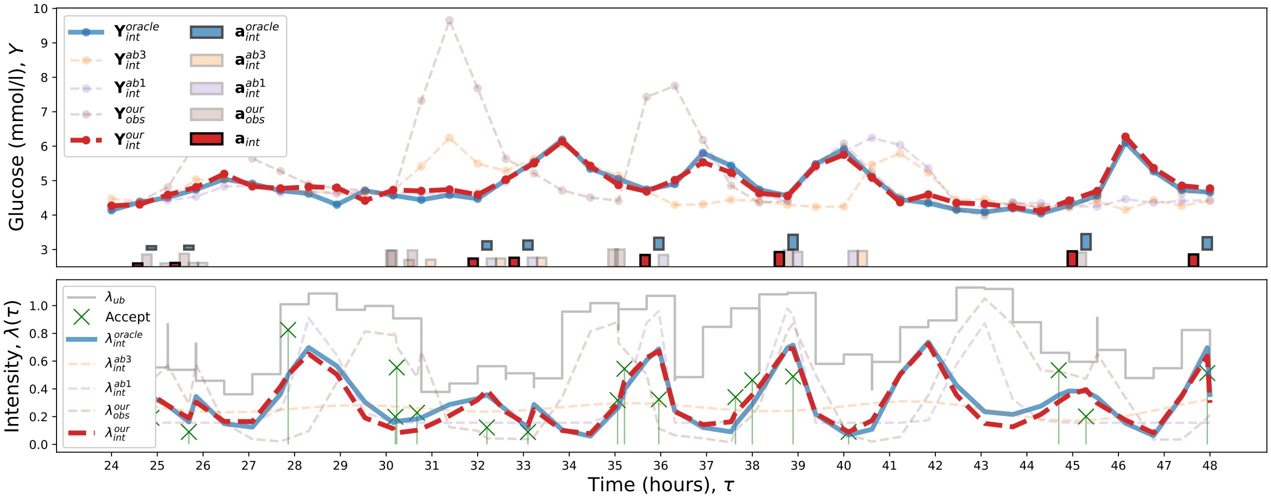


Figure 13: Interventional trajectory examples from the fixed-noise sampling algorithm (Algorithm 3), where treatment noise variables $\mathcal{E}_a$ are fixed for the ground-truth and the estimated models. The ground-truth intensity $\lambda^{\text{oracle}}_{\text{int}}$, the ground-truth treatments $\mathbf{a}^{\text{oracle}}_{\text{int}}$ and the ground-truth trajectory $\mathbf{Y}^{\text{oracle}}_{\text{int}}$ are shown in blue, while the predictions of our model OUR$_{\text{int}}$ are shown in red. The estimated treatment intensity $\lambda^{\text{our}}_{\text{int}}$ and the estimated outcome trajectory $\mathbf{Y}^{\text{our}}_{\text{int}}$ follow the ground-truth intensity and trajectory well, while benchmark models fail to do so.