A Dual-module Framework for Counterfactual Estimation over Time

Xin Wang [1] **Shengfei Lyu** [2] **Lishan Yang** [1] **Yibing Zhan** [3] **Huanhuan Chen** [1]

Abstract

Efficiently and effectively estimating counterfactuals over time is crucial for optimizing treatment strategies. We present the Adversarial Counterfactual Temporal Inference Network (ACTIN), a novel framework with dual modules to enhance counterfactual estimation. The balancing module employs a distribution-based adversarial method to learn balanced representations, extending beyond the limitations of current classification-based methods to mitigate confounding bias across various treatment types. The integrating module adopts a novel Temporal Integration Predicting (TIP) strategy, which has a wider receptive field of treatments and balanced representations from the beginning to the current time for a more profound level of analysis. TIP goes beyond the established Direct Predicting (DP) strategy, which only relies on current treatments and representations, by empowering the integrating module to effectively capture long-range dependencies and temporal treatment interactions. ACTIN exceeds the confines of specific base models, and when implemented with simple base models, consistently delivers state-of-the-art performance and efficiency across both synthetic and real-world datasets.

1. Introduction

Assessing the temporal impact of diverse treatments is critical in fields like personalized healthcare, which necessitates the efficient and effective estimation of counterfactuals over time, a crucial element in customizing treatment strategies (Hill and Su, 2013). While Randomized Controlled Trials are the gold standard for causal inference (Hariton and Locascio, 2018), the inherent high costs and ethical constraints have increased the emphasis on estimating counterfactuals from observational data.

In causal inference with observational data in static settings, addressing selection bias and emphasizing the unique roles of treatments is crucial, with recent studies (Shalit *et al.*, 2017; Yao *et al.*, 2018; Schwab *et al.*, 2020; Nie *et al.*, 2021; Johansson *et al.*, 2022) significantly advancing these aspects, thus enhancing the accuracy of treatment effect estimation. In longitudinal settings, similar challenges are encountered, often presenting increased complexity.

Observational longitudinal data presents complex confounding bias due to time-varying confounders, which impact subsequent treatment allocations and outcomes (Platt *et al.*, 2009). Current studies (Bica *et al.*, 2020b; Melnychuk *et al.*, 2022) address this issue by learning balanced representations from historical data, centering on the principle that these representations should not accurately classify current treatment assignments, which effectively removes the association between patient history and treatment assignments. While proven effective, these methods are primarily designed for scenarios with categorical treatments. Real-world applications, however, present diverse treatments like continuous or mixed types, posing a challenge to designing a wide-ranging method compatible with various treatment types for mitigating confounding bias.

In longitudinal studies, another challenge is managing long-range dependencies, like enduring effects in long-term medical treatments (Latner *et al.*, 2000; Jacobson *et al.*, 2013). Causal Transformer (CT) (Melnychuk *et al.*, 2022) leverages the advanced Transformer architecture, surpassing simpler Long Short-Term Memory (LSTM) networks in capturing these dependencies (Lim *et al.*, 2018; Bica *et al.*, 2020b; Li *et al.*, 2021). While CT adeptly handles general long-range dependencies, it may not thoroughly tackle the intricate temporal synergy between historical and current treatments, a critical aspect in medical settings. For instance, Roemhild *et al.* (2022) highlight the importance of grasping the temporal interactions between antibiotics for antibiotic optimization and resistance minimization. CT employs the Direct Predicting (DP) strategy (Bica *et al.*, 2020b), incorporating current treatments and balanced representations for counterfactual estimation, yet it may not fully harness the potential of complex temporal treatment interactions.

[1]School of Computer Science and Technology, University of Science and Technology of China, China [2]Nanyang Technological University, Singapore [3]JD Explore Academy. Correspondence to: Huanhuan Chen <hchen@ustc.edu.cn>, Shengfei Lyu <shengfei.lyu@ntu.edu.sg>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

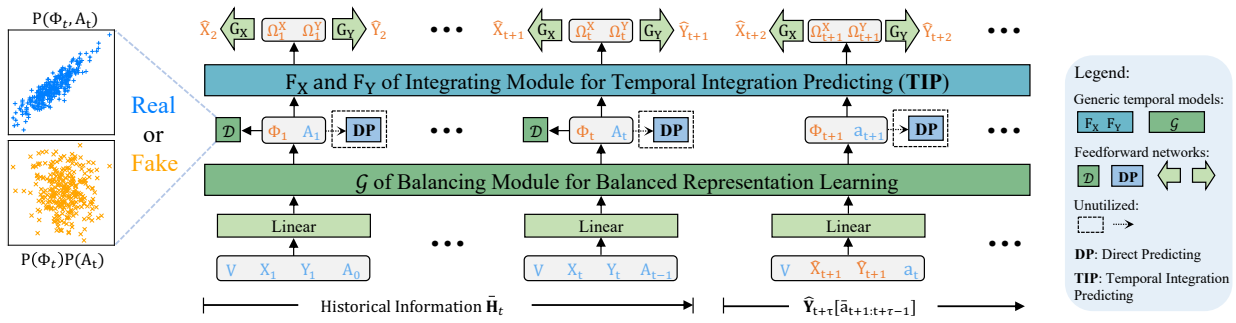


Figure 1. ACTIN features a dual-module architecture, namely a balancing module and an integrating module. The balancing module aims to align the joint and marginal distributions of representations and treatments to learn balanced representations. It incorporates a generator $\mathcal{G}$ that learns balanced representations from historical data, attempting to fool the discriminator $\mathcal{D}$, which distinguishes between real and generated samples. The integrating module's TIP strategy, diverging from established DP strategy that only relies on current treatments and balanced representations, merges these elements with their antecedents at each time point for a more profound level of analysis. Subsequently, counterfactual estimation is performed on the outputs of the integrating module using feedforward networks G_X and G_Y. ACTIN is trained on observational historical data spanning moments $(1, \cdots, t)$ and adopts an autoregressive approach for predicting counterfactuals at future moments $(t+1, \cdots, t+\tau)$.

To address these challenges, we propose the Adversarial Counterfactual Temporal Inference Network (ACTIN), as illustrated in Figure 1, a novel end-to-end framework developed for counterfactual estimation over time. The balancing module of ACTIN adopts an adversarial generative way of learning balanced representations. It features a discriminator that differentiates between real samples drawn from the joint distribution of representations and treatments, and fake samples generated from their marginal distributions. A generator within this module aims to learn representations for deceiving the discriminator. Theoretically, once equilibrium is achieved, the representation at each time point becomes independent of the corresponding treatment, thereby effectively reducing confounding bias. Be aware this learning mechanism is applicable to various types of treatments and enables ACTIN to be widely applied. The integrating module of ACTIN leverages the Temporal Integration Predicting (TIP) strategy, integrating balanced representations and treatments from current time and earlier, to refine counterfactual estimation. This approach employs in-depth processing of historical data embedded in balanced representations at each time point and its antecedents, thereby enhancing effectiveness in learning long-range dependencies. Moreover, it offers a more effective utilization of historical treatment information than DP methods, leading to enhanced capabilities in handling temporal treatment interactions.

ACTIN enables comparatively simpler base models, such as Temporal Convolutional Networks (TCN) and LSTM, to perform effectively. Note that while we instantiate ACTIN using TCN or LSTM in this paper, it is important to emphasize that it is not confined to any specific base model.

Overall, our main contributions are threefold[1]:

- ACTIN adopts a novel adversarial generative approach for learning balanced representations, which effectively reduces confounding bias and is applicable across various types of treatments.
- ACTIN, with its TIP strategy, boosts processing of long-range dependencies and temporal treatment interactions, enabling even simple base models to perform effectively and efficiently.
- ACTIN achieves state-of-the-art performance and efficiency on both synthetic and real-world datasets. Notably, on datasets constructed from real medical data, its running time is approximately only 10% of that of CT, highlighting its efficiency advantages.

2. Related Work

Initial methodologies for estimating time-varying outcomes encompassed the G-computation formula, Structural Nested Models, and Marginal Structural Models (MSM) (Robins, 1986; 1994; Robins *et al.*, 2000; Robins and Hernán, 2009). To address the shortcomings of conventional linear regression methods in managing intricate time dependencies (Mortimer *et al.*, 2005), the academic community has gravitated towards the adoption of Bayesian non-parametric approaches (Xu *et al.*, 2016; Soleimani *et al.*, 2017; Roy *et al.*, 2017). Nevertheless, these approaches are constrained in practical applications owing to their significant presuppositions about the structure of models.

[1]Code is available online: https://github.com/waxin/ACTIN

Currently, state-of-the-art methodologies predominantly rely on the development of deep neural networks. This includes RMSN (Lim *et al.*, 2018), CRN (Bica *et al.*, 2020b), G-Net (Li *et al.*, 2021), and CT (Melnychuk *et al.*, 2022). RMSN integrates two propensity networks and employs a training method based on Inverse Probability of Treatment Weighting (IPTW) scores for the predictive network. G-Net augments the traditional G-computation method through a deep learning framework. Both CT and CRN aim to generate a series of balanced representations designed to be predictive of outcomes while being non-predictive of the corresponding treatment assignments. The distinction lies in the two approaches: CRN uses a gradient reversal balancing strategy, while CT achieves its objective through minimizing reversed KL-divergence. However, the two approaches presuppose categorical treatments, limiting their applicability in broader contexts.

Moreover, RMSN, CRN, and G-Net are all built on simple LSTM networks. This presents certain limitations in capturing long-range complex dependencies among time-varying confounders in longitudinal data (Hochreiter *et al.*, 2001), which may impact their performance in real-world medical data processing. As an improvement, CT employs more powerful Transformer architecture (Vaswani *et al.*, 2017) for counterfactual estimation. However, the high computational complexity of Transformers may lead to inefficiency in processing large-scale real data. To balance effectiveness and efficiency, we propose a dual-module architecture, which enables comparatively simpler base models to more effectively capture long-range complex dependencies and temporal treatment interactions, providing a more effective method for analyzing real-world data.

Several studies, though varying in setup, explore causal inference from longitudinal data (Bica *et al.*, 2020a; Hatt and Feuerriegel, 2024; Frauen *et al.*, 2023; Seedat *et al.*, 2022; Meng *et al.*, 2023; Hess *et al.*, 2024). For instance, Time Series Deconfounder (Bica *et al.*, 2020a) leverages a novel recurrent neural network architecture to infer latent variables for adjusting multi-cause hidden confounders, enhancing causal inference from time-varying exposures. DeepACE (Frauen *et al.*, 2023) focuses on estimating time-varying average causal effects using a deep learning model incorporating iterative G-computation. Treatment Effect Neural Controlled Differential Equation (TE-CDE) (Seedat *et al.*, 2022) estimates counterfactuals in continuous time, while Bayesian Neural Controlled Differential Equation (BNCDE) (Hess *et al.*, 2024) offers uncertainty estimates using Bayesian methods. COSTAR (Meng *et al.*, 2023) distinguishes itself by addressing counterfactual estimation over time under distributional shifts and improves model performance by incorporating self-supervised learning.

The discussion above primarily focuses on methods used for estimating counterfactuals over time. A more in-depth review of pertinent literature is available in Appendix A.

3. Problem Formulation

Consider an i.i.d. observational dataset, denoted as $\mathbf{D}$, which comprehensively encapsulates the detailed information of N patients and is mathematically characterized as $\mathbf{D} = \left\{\{\mathbf{x}_t^{(i)}, \mathbf{a}_t^{(i)}, \mathbf{y}_t^{(i)}\}_{t=1}^{T^{(i)}} \cup \{\mathbf{v}^{(i)}\}\right\}_{i=1}^{N}$. For each individual patient, indexed by i, observations encompass time-varying covariates, $\mathbf{X}_t^{(i)} \in \mathcal{X}$, various types of treatments received like continuous or categorical types, $\mathbf{A}_t^{(i)} \in \mathcal{A}$, and resultant outcomes, $\mathbf{Y}_t^{(i)} \in \mathcal{Y}$, over $T^{(i)}$ discrete timesteps. Additionally, static covariates of patients, such as gender and age, are consistently recorded as $\mathbf{V}^{(i)} \in \mathcal{V}$. For enhanced notational clarity, the patient-specific superscript (i) will be omitted, unless contextually requisite.

Building upon the foundation of the potential outcomes framework (Rubin, 1978) and its extension to accommodate time-varying treatments (Robins and Hernan, 2008), we seek to estimate counterfactual outcomes over time as the previous studies (Lim *et al.*, 2018; Bica *et al.*, 2020b; Li *et al.*, 2021). Let historical information of a patient be denoted as $\bar{\mathbf{H}}_t = (\bar{\mathbf{X}}_t, \bar{\mathbf{A}}_{t-1}, \bar{\mathbf{Y}}_t, \mathbf{V})$, where $\bar{\mathbf{X}}_t = (\mathbf{X}_1, \cdots, \mathbf{X}_t)$, $\bar{\mathbf{Y}}_t = (\mathbf{Y}_1, \cdots, \mathbf{Y}_t)$, and $\bar{\mathbf{A}}_{t-1} = (\mathbf{A}_1, \cdots, \mathbf{A}_{t-1})$. We focus on estimating the potential outcome $\mathbf{Y}_{t+\tau}[\bar{\mathbf{a}}_{t:t+\tau-1}]$ following the administration of a treatment sequence $\bar{\mathbf{a}}_{t:t+\tau-1} = (\mathbf{a}_t, \cdots, \mathbf{a}_{t+\tau-1})$. The primary objective is to estimate:

$$\mathbb{E}[\mathbf{Y}_{t+\tau}[\bar{\mathbf{a}}_{t:t+\tau-1}]|\bar{\mathbf{H}}_t]. \tag{1}$$

To ensure the identifiability of treatment effects based on observational data, we make assumptions in reference to previous work (Lim *et al.*, 2018; Bica *et al.*, 2020b; Melnychuk *et al.*, 2022), including consistency, sequential ignorability, and sequential overlap (see Appendix B for details).

4. ACTIN and Theoretical Analysis

The proposed framework, ACTIN, is depicted in Figure 1. It is composed of two core modules. The balancing module synthesizes a balanced representation Φ_t by integrating historical time-dependent covariates $\bar{\mathbf{X}}_t$, static covariates $\mathbf{V}$, antecedent outcomes $\bar{\mathbf{Y}}_t$, and antecedent treatments $\bar{\mathbf{A}}_{t-1}$. The integrating module employs these balanced representations $\bar{\Phi}_t = \{\Phi_1, \cdots, \Phi_t\}$ alongside prior treatments $\bar{\mathbf{A}}_t$ to project forthcoming outcome $\hat{\mathbf{Y}}_{t+1}$ and covariate $\hat{\mathbf{X}}_{t+1}$. Additionally, the architecture adopts an autoregressive schema, iteratively inputting the projected outcomes $\hat{\mathbf{Y}}_{t+1:t+\tau-1}$ and covariates $\hat{\mathbf{X}}_{t+1:t+\tau-1}$ from the treatment juncture for multi-step-ahead prediction.

4.1. Balanced Representation

The historical data of a patient, represented as $\bar{\mathbf{H}}_t = (\bar{\mathbf{X}}_t, \bar{\mathbf{Y}}_t, \bar{\mathbf{A}}_{t-1}, \mathbf{V})$, includes time-dependent covariates $\bar{\mathbf{X}}_t$ which are pivotal in determining the treatment allocation $\mathbf{A}_t$. As elucidated by (Robins, 1999; Bica *et al.*, 2020b), severing the link between the historical data, which encompasses time-dependent covariates $\bar{\mathbf{X}}_t$, and the current treatment $\mathbf{A}_t$ enables the estimation of unbiased counterfactual treatment outcomes. Therefore, we introduce an innovative approach to formulating the representation of historical data $\bar{\mathbf{H}}_t$ and ensuring the independence of this representation from the treatment, which results in the severed link between the historical data $\bar{\mathbf{H}}_t$ and the corresponding treatment $\mathbf{A}_t$.

Let Φ serve as a function mapping the historical data $\bar{\mathbf{H}}_t$ onto a representation space $\mathcal{R}$, and let F represent any base model capable of processing temporal data. Figure 1 depicts our approach to fitting the representation function Φ through a neural network. This involves an initial step-by-step transformation of $\bar{\mathbf{H}}_t$ using a linear layer M, and the combination of a base model F_Φ with an activation function to generate the time-dependent representation. The parameters of this procedure, denoted as θ_Φ, allow us to formulate the representation function Φ as:

$$\Phi(\bar{\mathbf{H}}_t|\theta_\Phi) = \text{Activation}(F_\Phi(M(\bar{\mathbf{H}}_t))). \tag{2}$$

Let $\Phi_t = \Phi(\bar{\mathbf{H}}_t|\theta_\Phi)$ denote the learned representation at time t. For the purpose of achieving an unbiased treatment effect assessment, we aim to learn a treatment-independent representation Φ_t, i.e., $P(\Phi_t|\mathbf{A}_t = \mathbf{A}_0) = P(\Phi_t|\mathbf{A}_t = \mathbf{A}_1), \forall \mathbf{A}_0, \mathbf{A}_1 \in \mathcal{A}$. To realize this goal, we adopt the concept of adversarial training as put forth by (Ganin *et al.*, 2016). Figure 1 illustrates the working mechanism of this approach, where the base model of the balancing module is employed to learn the function Φ, serving as a generator $\mathcal{G}$, i.e., $\mathcal{G} = \Phi$. This generator forms the basis for two distinct sets of samples: one comprising real samples which align with the joint distribution $P(\Phi_t, \mathbf{A}_t)$, extracted from observational data; the other comprising fake samples generated from their marginal distributions $P(\Phi_t)P(\mathbf{A}_t)$.

We learn the function Φ by utilizing a discriminator $\mathcal{D}$ which is trained specifically to differentiate between real and fake samples within these two distinct sample sets. Concurrently, we also refine the generator $\mathcal{G} = \Phi$ to generate representations that can effectively mislead the discriminator $\mathcal{D}$. The training process ends with reaching an equilibrium point where the generator-derived representation Φ_t satisfies the condition $P(\Phi_t, \mathbf{A}_t) = P(\Phi_t)P(\mathbf{A}_t)$. Next, we will demonstrate the validity of this hypothesis from a theoretical perspective. Note that the novelty of our approach arises from its ability to handle various types of treatments employing this distribution-based balancing methodology, marking a notable progression from the previous studies (Bica *et al.*, 2020b; Melnychuk *et al.*, 2022) that were primarily limited to categorical treatments.

4.2. Theoretical Analysis

In this section, we provide theoretical guarantees for learning balanced representations with the adversarial training procedure described in Section 4.1.

Let $\mathbf{S} = (\Phi_t; \mathbf{A}_t)$ denote the generated sample, with $\zeta \in \{0, 1\}$ classifying its category. $\zeta = 1$ indicates that $\mathbf{s}$ is a real sample derived from observational data, as $\mathbf{s} \sim P(\Phi_t, \mathbf{A}_t)$. Conversely, $\zeta = 0$ suggests that $\mathbf{s}$ is a fabricated fake sample, conforming to $\mathbf{s} \sim P(\Phi_t)P(\mathbf{A}_t)$. We define $\mathcal{D}(\mathbf{s}|\theta_\mathcal{D})$ as the probability that the discriminator $\mathcal{D}$ predicts the sample $\mathbf{s}$ as $\zeta = 1$. Aligning with the adversarial training approach shown in Figure 1, our objective function is established as:

$$\max_{\Phi} \min_{\mathcal{D}} \mathcal{L}_\mathcal{D}(\theta_\mathcal{D}, \theta_\Phi, t), \tag{3}$$

where $\mathcal{L}_\mathcal{D}(\theta_\mathcal{D}, \theta_\Phi, t)$ is defined as $\mathbb{E}[L(\mathcal{D}(\mathbf{s}), \zeta)]$, with $L(\cdot, \cdot)$ denoting a loss function. The objective described in Eq. 3 is a minimax game. It is demonstrated that when equilibrium is achieved in this game, the representation function Φ will synchronize the distributions of the real and fake samples, i.e., e $P(\Phi_t, \mathbf{A}_t) = P(\Phi_t)P(\mathbf{A}_t)$. We begin our exposition using Lemma 4.1 to expound on the achievable condition of the optimal discriminator $\mathcal{D}^*$ with a fixed Φ.

Lemma 4.1. *For a fixed Φ, when employing the squared loss function L, the optimal discriminator $\mathcal{D}^*$ should satisfy the following condition:*

$$\mathcal{D}^*(\mathbf{s}) = \mathbb{E}_{\zeta \sim P(\zeta|\mathbf{s})}[\zeta]. \tag{4}$$

Proof. See Appendix C. □

When the optimal discriminator is realized, the process of optimizing the objective in Eq. 3 becomes synonymous with the maximization of the following objective $\mathcal{O}$:

$$\mathcal{O} \equiv \mathbb{E}[L(\mathcal{D}^*(\mathbf{s}), \zeta)]. \tag{5}$$

To prove the resultant condition of the representation function Φ in the context of maximizing $\mathcal{O}$, we employ Theorem 4.2.

Theorem 4.2. *The objective $\mathcal{O}$ achieves its maximum value if and only if the learned representation function Φ ensures that all $\mathbf{s}$ satisfy the condition $P(\zeta|\mathbf{s}) = P(\zeta)$. This signifies that Φ has established a balance in the distribution of samples, namely:*

$$P(\Phi_t, \mathbf{A}_t) = P(\Phi_t)P(\mathbf{A}_t). \tag{6}$$

Proof. See Appendix C. □

Theorem 4.2 posits that when the minimax game described in Eq. 3 reaches equilibrium, the condition $P(\Phi_t|\mathbf{A}_t = \mathbf{A}_0) = P(\Phi_t|\mathbf{A}_t = \mathbf{A}_1), \forall \mathbf{A}_0, \mathbf{A}_1 \in \mathcal{A}$ is satisfied, i.e., the learned representations are balanced. Subsequently, we will elaborate on the application of these representations in the estimation of counterfactuals.

4.3. Counterfactual Estimation

As shown in Figure 1, after obtaining the balanced representation Φ_t at each time point, the integrating module takes $\bar{\Phi}_t = \{\Phi_1, \cdots, \Phi_t\}$ and $\bar{\mathbf{A}}_t = \{\mathbf{A}_1, \cdots, \mathbf{A}_t\}$ as inputs. It revisits the historical treatments $\bar{\mathbf{A}}_t$ and balanced representations $\bar{\Phi}_t$ to effectively tackle long-range dependencies and temporal treatment interactions. This TIP strategy forms one of the innovations of our work and differs from the DP strategy (Bica *et al.*, 2020b) that only uses $\mathbf{A}_t$ and Φ_t for predictions.

The integrating module comprises two submodules: one for predicting outcomes and another for predicting covariates. Both submodules take $\bar{\Phi}_t$ and $\bar{\mathbf{A}}_t$ as inputs and have their own parameters.

The outcome prediction submodule, denoted as J_Y, comprises a base model F_Y and a feedforward neural network G_Y, i.e., $J_Y = G_Y \circ F_Y$. We refer to the parameters of J_Y as θ_Y. We train $J_Y = G_Y \circ F_Y$ using the Mean Squared Error (MSE) as the loss function. The loss function can be expressed as:

$$\mathcal{L}_Y(\theta_Y, \theta_\Phi, t) = \|\mathbf{Y}_{t+1} - J_Y(\bar{\Phi}_t\,; \bar{\mathbf{A}}_t|\theta_Y)\|^2. \quad (7)$$

Similarly, the covariate prediction submodule J_X consists of a base model F_X and two feedforward networks denoted as G_X and I_X, and its parameters are denoted as θ_X. Considering the slow rate of change for some covariates, e.g., cholesterol levels, we accommodate this gradual trend using a smoothing mechanism:

$$J_X(\bar{\Phi}_t\,; \bar{\mathbf{A}}_t|\theta_X) = \eta G_X(F_X(\bar{\Phi}_t\,; \bar{\mathbf{A}}_t)) + (1-\eta)\mathbf{X}_t, \quad (8)$$

where $\eta = \text{Sigmoid}(I_X(F_X(\bar{\Phi}_t\,; \bar{\mathbf{A}}_t)))$ modulates the time-varying feature of covariates, thereby achieving a balance between past observed values and predicted ones. It is worth noting that this strategy was inspired by the gating mechanism in Gated Recurrent Unit (GRU) (Chung *et al.*, 2014). Similarly, we define the loss function for J_x as follows:

$$\mathcal{L}_X(\theta_X, \theta_\Phi, t) = \|\mathbf{X}_{t+1} - J_X(\bar{\Phi}_t\,; \bar{\mathbf{A}}_t|\theta_X)\|^2. \quad (9)$$

Next, a detailed introduction to the training process of ACTIN will be provided, followed by an explanation of its specific inference steps.

Algorithm 1 Pseudocode of Training ACTIN

Input: $\mathbf{D} = \left\{\{\mathbf{x}_t^{(i)}, \mathbf{a}_t^{(i)}, \mathbf{y}_t^{(i)}\}_{t=1}^{T^{(i)}} \cup \{\mathbf{v}^{(i)}\}\right\}_{i=1}^{N}$
Parameters: $\theta := \{\theta_\Phi, \theta_X, \theta_Y\}$, learning rates l and $l_\mathcal{D}$, max epochs $p_{\max}$, iteration number $j = 0$, weight coefficients λ_D, λ_X and exponential smoothing factor β
Output: Optimized Parameters θ

1: Initialize EMA parameters θ^0_{EMA}
2: **for** $p = 1, \cdots, p_{\max}$ **do**
3: Compute $\lambda_{\mathcal{D}_p} = \lambda_\mathcal{D}\left(\frac{2}{1+\exp(-10\cdot(p/p_{\max}))} - 1\right)$
4: **for** batch $\mathcal{B}$ in epoch **do**
5: Construct real and fake samples using $\mathcal{B}$ and Φ
6: Compute $\mathcal{L}^\mathcal{B}_\mathcal{D} = \frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\sum_{t=1}^{T^{(i)}}\mathcal{L}^{(i)}_\mathcal{D}(\theta_\mathcal{D}, \theta_\Phi, t)$
7: $\theta_\mathcal{D} \leftarrow \theta_\mathcal{D} - l_\mathcal{D}\frac{\partial \mathcal{L}^\mathcal{B}_\mathcal{D}}{\partial \theta_\mathcal{D}}$
8: Compute $\mathcal{L}^\mathcal{B}_Y = \frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\sum_{t=1}^{T^{(i)}}\mathcal{L}^{(i)}_Y(\theta_Y, \theta_\Phi, t)$
9: Compute $\mathcal{L}^\mathcal{B}_X = \frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\sum_{t=1}^{T^{(i)}}\mathcal{L}^{(i)}_X(\theta_X, \theta_\Phi, t)$
10: Compute $\mathcal{L}^\mathcal{B}_\mathcal{D} = \frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\sum_{t=1}^{T^{(i)}}\mathcal{L}^{(i)}_\mathcal{D}(\theta_\mathcal{D}, \theta_\Phi, t)$
11: $\theta_\Phi \leftarrow \theta_\Phi - l(\frac{\partial \mathcal{L}^\mathcal{B}_Y}{\partial \theta_\Phi} + \lambda_X\frac{\partial \mathcal{L}^\mathcal{B}_X}{\partial \theta_\Phi} - \lambda_{\mathcal{D}_p}\frac{\partial \mathcal{L}^\mathcal{B}_\mathcal{D}}{\partial \theta_\Phi})$
12: $\theta_Y \leftarrow \theta_Y - l\frac{\partial \mathcal{L}^\mathcal{B}_Y}{\partial \theta_Y}$
13: $\theta_X \leftarrow \theta_X - l\lambda_X\frac{\partial \mathcal{L}^\mathcal{B}_X}{\partial \theta_X}$
14: $j \leftarrow j + 1$
15: Update EMA: $\theta^j_{\text{EMA}} = \beta\theta^{j-1}_{\text{EMA}} + (1-\beta)\theta$
16: **end for**
17: **end for**

4.4. Training Algorithm and Inference

We provide a detailed pseudo-code in Algorithm 1 to show the training process of ACTIN. It takes observational data $\mathbf{D}$ as input and outputs optimized parameters for subsequent inference tasks. During training, we employ an Exponential Moving Average (EMA) strategy (Tarvainen and Valpola, 2017), a technique that has been proven to yield more reliable results (Athiwaratkun *et al.*, 2019). Specifically, Algorithm 1 initializes the EMA parameters θ^0_{EMA} (see Line 1 of Algorithm 1) and iteratively updates them according to the following update formula (see Line 15 of Algorithm 1):

$$\theta^j_{\text{EMA}} = \beta\theta^{j-1}_{\text{EMA}} + (1-\beta)\theta^j, \quad (10)$$

where j denotes the iteration number in training and β denotes an exponential smoothing factor.

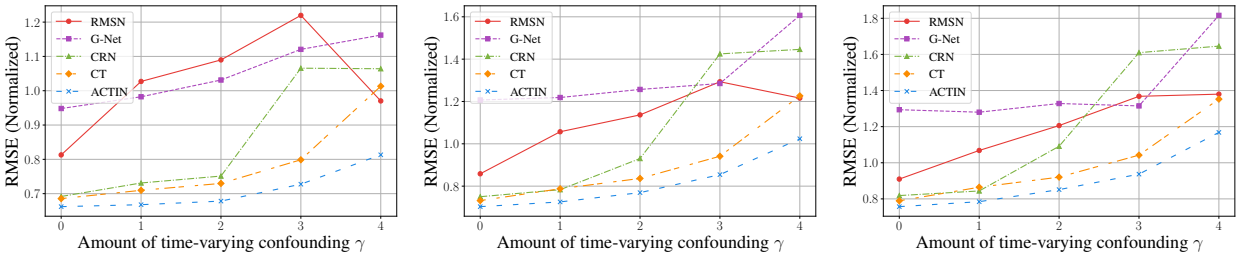
Within Algorithm 1, the discriminator parameters $\theta_\mathcal{D}$ are refined in Lines 5 to 7, with the intention of reducing the following objective function:

$$\mathcal{L}_\mathcal{D} = \frac{1}{N}\sum_{i\in\mathcal{D}}\sum_{t=1}^{T^{(i)}}\mathcal{L}^{(i)}_\mathcal{D}(\theta_\mathcal{D}, \theta_\Phi, t). \quad (11)$$

Note that fake samples are generated in Line 5 by employing a random shuffle of the treatments $\mathbf{A}_t$ during each time

Table 1. One-step-ahead prediction results ($\tau = 1$, lower values are better, with the best highlighted in bold) for the TG, RW, and SS datasets, with the latter two constructed based on MIMIC-III. Shown: RMSE as mean $\pm$ standard deviation over five runs.

	Tumor					MIMIC-III	
	$\gamma = 0$	$\gamma = 1$	$\gamma = 2$	$\gamma = 3$	$\gamma = 4$	RW	SS
RMSN	0.91±0.04	1.12±0.10	1.14±0.07	1.35±0.11	1.39±0.18	5.26±0.13	0.23±0.01
CRN	0.78±0.05	0.82±0.05	0.89±0.07	1.13±0.17	1.33±0.17	4.85±0.06	0.29±0.01
G-Net	0.83±0.05	0.87±0.08	1.00±0.06	1.30±0.23	1.37±0.25	5.05±0.04	0.36±0.01
CT	0.78±0.06	0.80±0.08	0.87±0.08	1.02±0.12	1.37±0.24	4.60±0.09	0.20±0.00
ACTIN (LSTM-based)	0.77±0.06	0.83±0.04	0.90±0.05	1.07±0.13	1.42±0.23	4.67±0.08	0.17±0.00
ACTIN	**0.75±0.06**	**0.78±0.04**	**0.85±0.07**	**1.01±0.13**	**1.26±0.21**	**4.56±0.10**	**0.15±0.00**



(a) 2-step-ahead prediction (b) 4-step-ahead prediction (c) 6-step-ahead prediction

Figure 2. Performance comparison of the ACTIN model with alternative models for 2-step, 4-step, and 6-step predictions is conducted under the *single sliding treatment* setting on TG datasets. The comparison is made across varying levels of the time-varying confounding factor γ, with the results presented as the average RMSE over five runs.

point. The parameters of the generator and the integrating module, i.e., θ_Φ, θ_X, and θ_Y, are updated in Line 8 to 13 by minimizing the objective as follows:

$$\mathcal{L} = \frac{1}{N}\sum_{i\in\mathcal{D}}\sum_{t=1}^{T^{(i)}}(\mathcal{L}_Y^{(i)}(\theta_Y,\theta_\Phi,t) + \lambda_X\mathcal{L}_X^{(i)}(\theta_X,\theta_\Phi,t) - \lambda_{\mathcal{D}_p}\mathcal{L}_{\mathcal{D}}^{(i)}(\theta_{\mathcal{D}},\theta_\Phi,t)). \quad (12)$$

Optimizing this objective is to enhance predictive accuracy while promoting the representation function Φ, which serves as the generator, to mislead the discriminator, by maximizing $\mathcal{L}_{\mathcal{D}}$. Here, λ_X denotes a preset weight coefficient and $\lambda_{\mathcal{D}_p}$ denotes a weight coefficient that gradually increases with training epochs to balance the training speed of the generator and the discriminator.

We implement ACTIN using the Pytorch Lightning framework and choose the Adam algorithm (Kingma and Ba, 2014) for gradient optimization. Once trained, the resultant model is ready to perform inference tasks.

During one-step-ahead inference, the model primarily relies on observational data. In the multi-step-ahead inference process, as shown in Figure 1, ACTIN utilizes the autoregressive recursive strategy for predictions in a more distant future, a well-established approach in multi-step time series forecasting (Chevillon, 2007; Taieb and Hyndman, 2014) that has also been successfully applied in G-Net.

5. Experiments

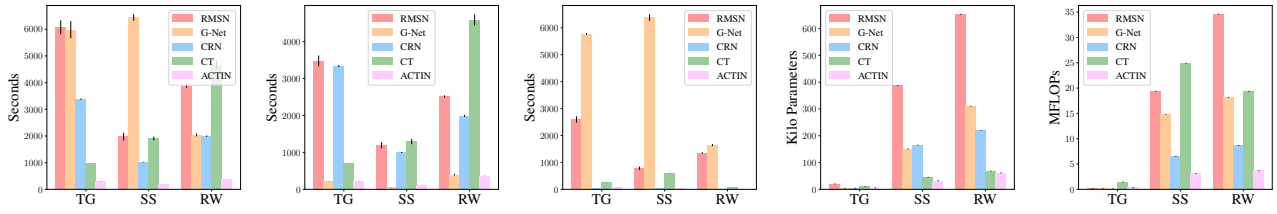
In this section, we validate the effectiveness of the proposed ACTIN through a series of experiments. Following the conventional workflow of counterfactual inference benchmarks (Melnychuk *et al.*, 2022), we conduct comparative analyses of ACTIN against existing models on both simulated and real datasets. Subsequently, we examine in detail the running time and complexity of the baseline methods and ACTIN on different datasets. Ultimately, we experimentally explore the roles of different components in ACTIN.

Baselines. In this study, models from the state-of-the-art literature on time-varying counterfactual outcome estimation are selected as baselines for comparison. These include: **RMSN** (Lim *et al.*, 2018), **CRN** (Bica *et al.*, 2020b), **G-Net** (Li *et al.*, 2021), and **CT** (Melnychuk *et al.*, 2022). To ensure the fairness in comparison, we employ hyperparameter tuning for these baselines (see Appendix F for details).

Datasets. The fully-synthetic tumor growth dataset, constructed using a pharmacokinetic-pharmacodynamic model (Geng *et al.*, 2017), simulates the combined effects of chemotherapy and radiotherapy on lung cancer patients. This dataset has also been used for evaluation in previous studies such as (Lim *et al.*, 2018; Bica *et al.*, 2020b; Melnychuk *et al.*, 2022). A parameter γ in the advanced biomathematical model controls time-dependent confounding in the dataset. An increase in γ signifies that historical

Table 2. Multi-step-ahead prediction results on the SS dataset. Shown: RMSE as mean ± standard deviation over five runs.

	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$	$\tau = 8$	$\tau = 9$	$\tau = 10$	$\tau = 11$
RMSN	0.47±0.01	0.59±0.01	0.70±0.03	0.79±0.06	0.88±0.08	0.95±0.08	1.01±0.07	1.04±0.05	1.07±0.04	1.10±0.03
CRN	0.46±0.01	0.57±0.01	0.62±0.01	0.66±0.01	0.69±0.01	0.70±0.02	0.73±0.02	0.75±0.03	0.77±0.03	0.80±0.03
G-Net	0.67±0.01	0.84±0.01	0.96±0.02	1.05±0.02	1.13±0.03	1.20±0.04	1.26±0.05	1.31±0.06	1.37±0.07	1.41±0.09
CT	0.37±0.00	0.45±0.01	0.49±0.01	0.51±0.01	0.53±0.01	0.55±0.01	0.56±0.02	0.58±0.02	0.59±0.02	0.60±0.02
ACTIN (LSTM-based)	0.35±0.00	0.41±0.00	0.45±0.01	0.48±0.01	0.50±0.01	0.51±0.01	0.52±0.01	0.54±0.02	0.55±0.02	0.56±0.02
ACTIN	**0.33±0.00**	**0.39±0.00**	**0.43±0.00**	**0.46±0.00**	**0.48±0.00**	**0.49±0.00**	**0.50±0.00**	**0.51±0.01**	**0.52±0.01**	**0.54±0.01**



(a) Running time (b) Training time (c) Inference time (d) Parameters (e) Computational Load

Figure 3. The comparison of running costs and model complexity between ACTIN and other models is presented for the TG dataset ($\gamma = 0$), as well as for the SS and RW datasets constructed based on MIMIC-III.

data play a more critical role in the allocation of treatments, leading to more pronounced confounding bias.

The Medical Information Mart for Intensive Care III (MIMIC-III) (Johnson *et al.*, 2016) is a comprehensive database of electronic health records for patients in intensive care units, often used to evaluate the effectiveness of models in real and complex medical scenarios. Following the studies in (Hatt and Feuerriegel, 2021; Kuzmanovic *et al.*, 2021; Melnychuk *et al.*, 2022), we extract 25 patients covariates and 3 static features from MIMIC-III, treat diastolic blood pressure as the predictive target, and choose two treatment interventions: the use of vasopressors and mechanical ventilation.

It is important to note that MIMIC-III, as a real data source, does not include information of counterfactual outcomes. To further explore the performance of ACTIN in the analysis of high-dimensional, long-range patient trajectories, we construct a semi-synthetic dataset based on the method described in (Melnychuk *et al.*, 2022) on top of MIMIC-III. This dataset, grounded in the research approach (Schulam and Saria, 2017), generates patient trajectories considering treatment effects, as well as endogenous and exogenous dependencies. This allows us to control the degree of confounding and access counterfactuals for evaluation.

Additionally, to explore ACTIN's adaptability to different treatments, we introduce a synthetic dataset for continuous interventions, the Continuous Interventions Synthetic Dataset (CISD). This dataset models continuous interventions through an autoregressive process that integrates historical covariate data to guide treatment decisions. Treatments are sampled from a Beta distribution after undergoing non-linear transformations and noise adjustments, reflecting the stochastic nature of treatment assignments. CISD is designed to assess the impact of sequential treatment strategies, highlighting the continuous nature of interventions and their complex temporal dependencies.

For ease of discussion, henceforth we denote the fully-synthetic tumor growth dataset as TG, and both the semi-synthetic and the real-world datasets, constructed from MIMIC-III, as SS and RW respectively. For details on the generation of all aforementioned datasets, please refer to Appendix E.

5.1. Performance Comparison

In this study, we primarily use TCN and LSTM networks as the base models for the ACTIN's balancing and integration modules, denoted as ACTIN (TCN-based) and ACTIN (LSTM-based), respectively. For a detailed discussion on TCN, please refer to Appendix D. In the descriptions that follow, unless otherwise specified, we refer to ACTIN (TCN-based) simply as ACTIN. Moreover, to demonstrate ACTIN's compatibility, we also applied the TIP strategy to the CT model, with detailed results available in Section 5.3.

One-step-ahead prediction. As shown in Table 1, ACTIN exhibits best performance on TG datasets across various levels of time-dependent confounding variable γ. On the RW and SS datasets constructed based on MIMIC-III, ACTIN also demonstrates higher prediction accuracy, with the lowest Root Mean Square Error (RMSE) and relatively small standard deviation. This experimental evidence indicates the effectiveness of ACTIN. Particularly, its outstanding

Table 3. Multi-step-ahead prediction results on the RW dataset. Shown: RMSE as mean ± standard deviation over five runs.

	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$
RMSN	10.02±0.53	11.18±0.71	11.93±1.07	12.56±1.47	13.12±1.82
CRN	9.13±0.16	9.77±0.16	10.10±0.17	10.36±0.20	10.58±0.22
G-Net	11.89±0.19	12.92±0.25	13.59±0.28	14.09±0.30	14.52±0.38
CT	8.99±0.21	9.59±0.22	9.91±0.25	10.14±0.29	10.34±0.32
ACTIN (LSTM-based)	9.05±0.16	9.64±0.16	9.96±0.17	10.21±0.19	10.42±0.21
ACTIN	**8.98±0.18**	**9.56±0.18**	**9.87±0.19**	**10.11±0.21**	**10.30±0.23**

performance on the semi-synthetic dataset further reveals ACTIN's applicability in scenarios closely mirroring real-world complexities.

Multi-step-ahead prediction. In the context of multi-step-ahead prediction on TG datasets, we adopt two settings following (Melnychuk *et al.*, 2022): (1) *single sliding treatment* to simulate trajectories with a single treatment where the treatments are iteratively moved over a window, and (2) *random trajectories* to simulate trajectories with random treatment assignments. Given that the former setting is also employed in (Bica *et al.*, 2020b), we present in Figure 2 a comparison of ACTIN with the baselines for multi-step-ahead prediction results under the relevant settings. In all testing scenarios, ACTIN consistently achieves the best prediction accuracy. With the gradual increase of the time-varying confounding factor γ, the RMSE of all models rises, yet ACTIN's performance remains comparatively stable, which confirms its effectiveness.

In the RW and SS datasets constructed based on MIMIC-III (as shown in Table 2 and Table 3), ACTIN also demonstrates superior multi-step-ahead prediction accuracy, showing its impressive ability to handle complex long-term dependencies. Additionally, we employ LSTM as an alternative base model for ACTIN, executing experiments on the SS dataset. The experimental results shown in Table 2 indicate that LSTM-based ACTIN slightly outperforms CT. Compared to other LSTM-based models, such as CRN, LSTM-based ACTIN achieved a significant performance improvement. These results demonstrate the compatibility of ACTIN and its capability of in enhancing the performance of simpler base models. For a more comprehensive exposition of the experimental results, please refer to Appendix G.

Overall, the experimental results sufficiently prove the potential of ACTIN in time-series analysis for estimating counterfactual outcomes. ACTIN maintains reliable and precise forecasts even in complex scenarios characterized by substantial confounding bias and long-term dependencies.

5.2. Model Efficiency Evaluation

In practical applications, not only the accuracy of the model matters, but its operational efficiency is also crucial. Therefore, we evaluated the running cost and complexity of ACTIN and the baselines on various datasets, with the specific results presented in Figure 3. Figure 3(a) displays the running time, which is the sum of both training and inference times. This combined measure is a common practice for evaluating model efficiency, as demonstrated in recent studies like CoST (Woo *et al.*, 2022)

Table 4. Ablation study results with average RMSE for the ACTIN model and relative performance changes for model variants across TG ($\tau = 6$), SS ($\tau = 11$), and CISD ($\tau = 6$) Datasets, where "+" indicates a decrease and "-" an increase in performance.

	TG				
	$\gamma = 0$	$\gamma = 2$	$\gamma = 4$	SS	CISD
ACTIN	0.757	0.851	1.168	0.536	3.378
w/o balancing	−0.1%	+1.6%	+4.3%	+1.9%	+9.4%
w/o integrating	+6.8%	+4.0%	+12.6%	+41.2%	+17.1%
with DP	+5.4%	+6.6%	+6.2%	+5.8%	+18.9%

The computational load shown in Figure 3(e) is measured by the million floating-point operations (MFLOPs) of a model processing a single sample, as measured using fvcore[2].

ACTIN demonstrates superior efficiency across all datasets, particularly in contexts that closely mirror real-world scenarios, operating at roughly 10% the running time of CT (based on RW and SS datasets from MIMIC-III). This efficiency, alongside its enhanced performance, significantly bolsters ACTIN's suitability for real-world application scenarios. This exceptional efficiency is partially due to the rapidity of ACTIN in both training and inference stages. In contrast, CRN requires a longer training time due to the separate training of encoders and decoders, whereas G-Net's inference time is prolonged due to Monte Carlo sampling.

Regarding model complexity, ACTIN's requirements for the number of parameters and computational load are generally lower, especially notable in the more structurally complex MIMIC-III derived datasets.

Overall, ACTIN shows excellent efficiency and good scala-

[2]Fvcore is a lightweight core library that provides common functionality and useful utilities for deep learning. For more information, please visit https://github.com/facebookresearch/fvcore

	$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$
CT	1.35±0.24	1.01±0.28	1.13±0.35	1.23±0.40	1.29±0.43	1.33±0.45
ACTIN (CT-based)	1.38±0.22	0.99±0.14	1.06±0.18	1.15±0.20	1.22±0.22	1.28±0.24

Table 5. τ-step-ahead prediction results on the TG dataset ($\gamma = 4$ with single sliding treatment setting) for the CT and ACTIN (CT-based) models across different τ values. Results are presented as mean ± standard deviation over five runs.

bility, indicating its substantial potential in handling complex real-world application scenarios.

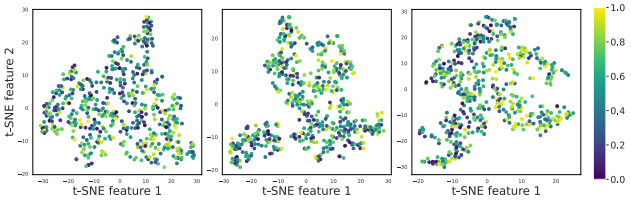
5.3. Ablation Study

In our ablation study, we evaluate the importance of different components of ACTIN by comparing the full model against variants with certain components removed. Specifically, "w/o balancing" refers to omitting the adversarial aspect within the balancing module, i.e., setting $\lambda_{\mathcal{D}} = 0$; "w/o integrating" denotes excluding the entire integrating module; "with DP" signifies the integrating module using only balanced representations for input and employing the DP strategy for counterfactual estimation at the output stage. The experiment is conducted on TG datasets, the SS dataset, and the dataset focusing on continuous interventions (CISA), which was generated through an autoregressive process (see Appendix E.3 for details).

The results of the ablation study are shown in Table 4. In TG datasets, the effectiveness of balanced representations learned through adversarial training becomes more pronounced with an increase in time-varying confounding bias. Additionally, experimental outcomes underscore that the integrating module improves model performance across all datasets, with particularly pronounced enhancements when contending with the complexities of the MIMIC-III data, thereby affirming its capability to address long-range dependencies. Notably, substituting the TIP strategy with the DP approach often results in a similar performance detriment to the removal of the entire integrating module, highlighting the critical importance of accounting for temporal treatment interactions in counterfactual estimation over time.

In the experiments with the CISD dataset, ACTIN also effectively mitigates the confounding bias in scenarios of the continuous treatment, thereby enhancing model performance. Furthermore, in Figure 4, we present the t-SNE embeddings of the balanced representations constructed by ACTIN for validation patients in the CISD dataset. By coloring based on the value of treatments, we observe that the treatments corresponding to different representations are essentially balanced.

To further explore the effectiveness of ACTIN in relation to other models, we implemented the TIP strategy using LSTM on CT, referred to as ACTIN (CT-based). We compared the performance of the original CT and ACTIN (CT-based)



(a) t=5 (b) t=15 (c) t=20

Figure 4. t-SNE embeddings of the balancing representations learned by ACTIN at different time.

on the TG dataset ($\gamma = 4$ with a single sliding treatment setting). Specifically, we maintained the other model hyperparameters of CT unchanged and added an integrating module composed of an LSTM to conduct experiments. The experimental results in Table 5 demonstrate that combining ACTIN with CT can indeed lead to performance improvements. However, the magnitude of improvement is smaller compared to that observed with simpler base models such as TCN. This can be attributed to the fact that the proposed dual-module framework primarily serves two purposes: (1) enhancing the ability of simple temporal models to handle long-term dependencies, and (2) improving the model's capacity to capture temporal treatment interactions. CT, through its elaborate design, already effectively enhances the handling of long-term dependencies. Consequently, while the improvement brought about by our method in the first point is comparatively limited, its contribution to capturing temporal treatment interactions remains notable.

6. Conclusion

In this paper, we introduce a dual-module framework, the Adversarial Counterfactual Temporal Inference Network (ACTIN), designed to enhance counterfactual estimation over time. The balancing module of ACTIN effectively mitigates confounding bias across different treatment scenarios, while the integrating module enables the construction of efficient and effective models based on simple base models. ACTIN has been rigorously validated through extensive experimental analysis, thereby underscoring ACTIN's considerable potential for real-world applications. Notably, a key limitation of all counterfactual estimation methods, including ours, is that they can not be directly validated with real-world data since counterfactuals are unobservable.

Acknowledgements

This research is supported by National Key R&D Program of China (No. 2021ZD0111700), National Nature Science Foundation of China (No. 62137002, 62176245), Key Science and Technology Special Project of Anhui Province (No. 202103a07020002). We thank the anonymous reviewers for their constructive comments that help improve the manuscript.

Impact Statement

This paper presents significant advancements in counterfactual estimation over time, primarily aimed at deepening the analysis and comprehension of longitudinal data, with a special focus on applications within the medical domain. The core objective of this research is to fortify the theoretical and methodological bedrock of causal inference. The insights and methodologies developed in this study are anticipated to wield notable influence, particularly in the healthcare sector. Potential applications of these advancements are wide-ranging, including but not limited to, enhancements in patient care, refinement of medical decision-making processes, and optimization of treatment protocols.

References

Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. There are many consistent explanations of unlabeled data: Why you should average. In *International Conference on Learning Representations*, 2019.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR, 2018.

Ioana Bica, Ahmed Alaa, and Mihaela Van Der Schaar. Time series deconfounder: Estimating treatment effects over time in the presence of hidden confounders. In *International Conference on Machine Learning*, pages 884–895. PMLR, 2020.

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

Guillaume Chevillon. Direct multi-step estimation and forecasting. *Journal of Economic Surveys*, 21(4):746–785, 2007.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Alicia Curth and Mihaela van der Schaar. Nonparametric estimation of heterogeneous treatment effects: From theory to learning algorithms. In *International Conference on Artificial Intelligence and Statistics*, pages 1810–1818. PMLR, 2021.

Dennis Frauen, Tobias Hatt, Valentyn Melnychuk, and Stefan Feuerriegel. Estimating average causal effects from patient trajectories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7586–7594, 2023.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Changran Geng, Harald Paganetti, and Clemens Grassberger. Prediction of treatment response for combined chemo- and radiation therapy for non-small cell lung cancer patients using a bio-mathematical model. *Scientific reports*, 7(1):13542, 2017.

Eduardo Hariton and Joseph J Locascio. Randomised controlled trials—the gold standard for effectiveness research. *BJOG: an international journal of obstetrics and gynaecology*, 125(13):1716, 2018.

Negar Hassanpour and Russell Greiner. Counterfactual regression with importance sampling weights. In *IJCAI*, pages 5880–5887, 2019.

Tobias Hatt and Stefan Feuerriegel. Sequential deconfounding for causal inference with unobserved confounders. *arXiv preprint arXiv:2104.09323*, 2021.

Tobias Hatt and Stefan Feuerriegel. Sequential deconfounding for causal inference with unobserved confounders. In *Causal Learning and Reasoning*, pages 934–956. PMLR, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Konstantin Hess, Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Bayesian neural controlled differential equations for treatment effect estimation. In *The*

Twelfth International Conference on Learning Representations, 2024.

Jennifer Hill and Yu-Sung Su. Assessing lack of common support in causal inference using bayesian nonparametrics: Implications for evaluating the effect of breastfeeding on children's cognitive outcomes. *The Annals of Applied Statistics*, pages 1386–1420, 2013.

Sepp Hochreiter, Yoshua Bengio, Paolo Frasconi, Jürgen Schmidhuber, et al. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, 2001.

Alan Jacobson, Barbara Braffett, Patricia Cleary, Rose Gubitosi-Klug, and Mary Larkin. The long-term effects of type 1 diabetes treatment and complications on health-related quality of life. *Diabetes care*, 36, 07 2013.

Fredrik Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In *International conference on machine learning*, pages 3020–3029. PMLR, 2016.

Fredrik D Johansson, Uri Shalit, Nathan Kallus, and David Sontag. Generalization bounds and representation learning for estimation of potential outcomes and causal effects. *The Journal of Machine Learning Research*, 23(1):7489–7538, 2022.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Insung Kong, Yuha Park, Joonhyuk Jung, Kwonsang Lee, and Yongdai Kim. Covariate balancing using the integral probability metric for causal inference. *arXiv preprint arXiv:2305.13715*, 2023.

Milan Kuzmanovic, Tobias Hatt, and Stefan Feuerriegel. Deconfounding temporal autoencoder: estimating treatment effects over time using noisy proxies. In *Machine Learning for Health*, pages 143–155. PMLR, 2021.

JD Latner, AJ Stunkard, GT Wilson, ML Jackson, DS Zelitch, and E Labouvie. Effective long-term treatment of obesity: a continuing care model. *International Journal of Obesity*, 24(7):893–898, 2000.

Rui Li, Stephanie Hu, Mingyu Lu, Yuria Utsumi, Prithwish Chakraborty, Daby M. Sow, Piyush Madan, Jun Li, Mohamed Ghalwash, Zach Shahn, and Li-wei Lehman. G-net: a recurrent network approach to g-computation for counterfactual prediction under a dynamic treatment regime. In *Proceedings of Machine Learning for Health*, volume 158 of *Proceedings of Machine Learning Research*, pages 282–299. PMLR, 04 Dec 2021.

Bryan Lim, Ahmed Alaa, and Mihaela van der Schaar. Forecasting treatment responses over time using recurrent marginal structural networks. *Advances in neural information processing systems*, 31, 2018.

Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. *Advances in neural information processing systems*, 30, 2017.

Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Causal transformer for estimating counterfactual outcomes. In *International Conference on Machine Learning*, pages 15293–15329. PMLR, 2022.

Chuizheng Meng, Yihe Dong, Sercan Ö Arık, Yan Liu, and Tomas Pfister. Costar: Improved temporal counterfactual estimation with self-supervised learning. *arXiv preprint arXiv:2311.00886*, 2023.

Kathleen M Mortimer, Romain Neugebauer, Mark Van Der Laan, and Ira B Tager. An application of model-fitting procedures for marginal structural models. *American Journal of Epidemiology*, 162(4):382–388, 2005.

Lizhen Nie, Mao Ye, Dan Nicolae, et al. VCNet and functional targeted regularization for learning causal effects of continuous treatments. In *Proceedings of the 9th International Conference on Learning Representations*, 2020.

Lizhen Nie, Mao Ye, qiang liu, and Dan Nicolae. {VCN}et and functional targeted regularization for learning causal effects of continuous treatments. In *International Conference on Learning Representations*, 2021.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

Robert W Platt, Enrique F Schisterman, and Stephen R Cole. Time-modified confounding. *American journal of epidemiology*, 170(6):687–694, 2009.

James Robins and Miguel Hernan. Estimation of the causal effects of time-varying exposures. *Chapman & Hall/CRC Handbooks of Modern Statistical Methods*, pages 553–599, 2008.

James M Robins and Miguel A Hernán. *Estimation of the causal effects of time-varying exposures*. CRC Press, Boca Raton, FL, 2009.

James M Robins, Miguel Angel Hernan, and Babette Brumback. Marginal structural models and causal inference in epidemiology. *Epidemiology*, pages 550–560, 2000.

James Robins. 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.

James M Robins. Correcting for non-compliance in randomized trials using structural nested mean models. *Communications in Statistics-Theory and methods*, 23(8):2379–2412, 1994.

James M Robins. Association, causation, and marginal structural models. *Synthese*, 121(1/2):151–179, 1999.

Roderich Roemhild, Tobias Bollenbach, and Dan I Andersson. The physiology and genetics of bacterial responses to antibiotic combinations. *Nature Reviews Microbiology*, 20(8):478–490, 2022.

Jason Roy, Kirsten J Lum, and Michael J Daniels. A bayesian nonparametric approach to marginal structural models for point treatments and a continuous or survival outcome. *Biostatistics*, 18(1):32–47, 2017.

Donald B Rubin. Bayesian inference for causal effects: The role of randomization. *The Annals of statistics*, pages 34–58, 1978.

Peter Schulam and Suchi Saria. Reliable decision support using counterfactual models. *Advances in neural information processing systems*, 30, 2017.

Patrick Schwab, Lorenz Linhardt, Stefan Bauer, Joachim M Buhmann, and Walter Karlen. Learning counterfactual representations for estimating individual dose-response curves. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5612–5619, 2020.

Nabeel Seedat, Fergus Imrie, Alexis Bellot, Zhaozhi Qian, and Mihaela van der Schaar. Continuous-time modeling of counterfactual outcomes using neural controlled differential equations. In *International Conference on Machine Learning*, pages 19497–19521. PMLR, 2022.

Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pages 3076–3085. PMLR, 2017.

Claudia Shi, David Blei, and Victor Veitch. Adapting neural networks for the estimation of treatment effects. *Advances in neural information processing systems*, 32, 2019.

Hossein Soleimani, Adarsh Subbaswamy, and Suchi Saria. Treatment-response models for counterfactual reasoning with continuous-time, continuous-valued interventions. In *33rd Conference on Uncertainty in Artificial Intelligence, UAI 2017*. AUAI Press Corvallis, 2017.

Souhaib Ben Taieb and Rob Hyndman. Boosting multi-step autoregressive forecasts. In *International conference on machine learning*, pages 109–117. PMLR, 2014.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Shirly Wang, Matthew BA McDermott, Geeticka Chauhan, Marzyeh Ghassemi, Michael C Hughes, and Tristan Naumann. Mimic-extract: A data extraction, preprocessing, and representation pipeline for mimic-iii. In *Proceedings of the ACM conference on health, inference, and learning*, pages 222–235, 2020.

Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Cost: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. In *International Conference on Learning Representations*, 2022.

Yanbo Xu, Yanxun Xu, and Suchi Saria. A bayesian nonparametric approach for estimating individualized treatment-response curves. In *Machine learning for healthcare conference*, pages 282–300. PMLR, 2016.

Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. Representation learning for treatment effect estimation from observational data. In *Proceedings of the 32nd Advances in Neural Information Processing Systems*, volume 31, 2018.

Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. Ganite: Estimation of individualized treatment effects using generative adversarial nets. In *International conference on learning representations*, 2018.

A. Extended Related Work

Causal inference in static settings

Mitigating selection bias. A substantial body of literature focuses on estimating counterfactual outcomes in static settings to assess individual treatment effects, as explored in studies like (Johansson *et al.*, 2016; Shalit *et al.*, 2017; Yoon *et al.*, 2018; Curth and van der Schaar, 2021; Kong *et al.*, 2023). In observational data within these static environments, treatments are typically allocated based on covariates related to each unit, leading to imbalances in covariate distributions between subgroups receiving different treatments, a phenomenon known as selection bias. Mitigating this imbalance, which can compromise inference reliability, is a critical issue. Numerous studies have dedicated efforts to this challenge, with the closest approach to our concept being the learning of representations to balance distributions between treatment and control groups (Johansson *et al.*, 2016; Shalit *et al.*, 2017; Yoon *et al.*, 2018). For instance, Shalit *et al.* (2017) learn balanced representations by minimizing the Integral Probability Metric (IPM) measure of distance between treated and control distributions. However, methods developed for static settings do not directly extend to time-varying treatments. Our strategy for learning balanced representations draws inspiration from (Belghazi *et al.*, 2018), fundamentally aiming to render the mutual information between balanced representations and corresponding intervention distributions as zero.

Emphasizing the unique roles of treatments. In static settings, a critical problem for the successful application of neural networks in causal inference is the necessity to design a network structure that distinctly incorporates treatment variables from other covariates (Shalit *et al.*, 2017; Schwab *et al.*, 2020; Nie *et al.*, 2020). Under binary interventions, Shalit *et al.* (2017) proposed a general framework known as Counterfactual Regression (CFR). CFR initially learns a shared representation, on top of which two separate "heads" are utilized to predict outcomes post-intervention and control, effectively addressing the potential loss of treatment information in high-dimensional latent representations. This concept has been widely adopted in subsequent research (Louizos *et al.*, 2017; Shi *et al.*, 2019; Hassanpour and Greiner, 2019). Furthermore, Schwab *et al.* (2020) extended this idea to continuous interventions with Dose Response Network (DRNet). DRNet segments continuous treatment variables into discrete sections, assigning a distinct head for each segment and integrating the treatment into every layer of the hidden network. However, this structure fails to maintain the continuity of the average dose-response curve (ADRF). To address this, Nie *et al.* (2020) introduce the Varying Coefficient Network (VCNet). VCNet permits the weights of the prediction head to be continuous functions of the treatment, not only enhancing the impact of the treatment but also preserving the continuity of the ADRF.

In continuous settings, similar issues arise, such as the potential neglect of historical treatments over time, yet this problem has received little attention. CT (Melnychuk *et al.*, 2022) introduces a subnetwork for treatments, partially addressing the issue. However, as we have identified, CT's DP strategy utilizes historical treatments in a more implicit manner, whereas our TIP strategy engages historical treatments more explicitly, thus enhancing the model's ability to manage temporal treatment interactions more effectively.

B. Assumptions

To ensure the identifiability of treatment effects based on observational data, we make the following assumptions in reference to previous work (Lim *et al.*, 2018; Bica *et al.*, 2020b; Li *et al.*, 2021; Melnychuk *et al.*, 2022).

Assumption B.1 (**Consistency**). At time $t+1$, the observed outcome $\mathbf{Y}_{t+1}$ equals the potential outcome $\mathbf{Y}_{t+1}[\mathbf{a}_t]$ given the treatment $\mathbf{a}_t$ at t, i.e., $\mathbf{Y}_{t+1} = \mathbf{Y}_{t+1}[\mathbf{a}_t]$.

Assumption B.2 (**Sequential Overlap**). The probability of receiving a treatment at time t is always nonzero, i.e., $0 < P(\mathbf{A}_t = \mathbf{a}_t | \bar{\mathbf{H}}_t = \bar{\mathbf{h}}_t) < 1, \forall \mathbf{a}_t \in \mathcal{A}$ if $P(\bar{\mathbf{H}}_t = \bar{\mathbf{h}}_t) > 0$, where $\bar{\mathbf{h}}_t$ is a realization of $\bar{\mathbf{H}}_t$.

Assumption B.3 (**Sequential Ignorability**). The treatment applied at any time t is independent of the potential outcome, given the observed history, i.e., $\mathbf{A}_t \perp \mathbf{Y}_{t+1}[\mathbf{a}_t] \,|\, \bar{\mathbf{H}}_t, \forall \mathbf{a}_t \in \mathcal{A}$. This suggests that there are no unobserved confounders that affect both treatment and outcome.

C. Proofs

Lemma C.1. *For a fixed Φ, when employing the squared loss function L, the optimal discriminator $\mathcal{D}^*$ should satisfy the following condition:*

$$\mathcal{D}^*(\mathbf{s}) = \mathbb{E}_{\zeta \sim P(\zeta|\mathbf{s})}[\zeta]. \tag{13}$$

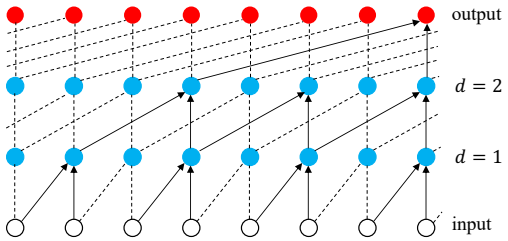


Figure 5. Illustration of a Temporal Convolutional Network (TCN) architecture with layers exhibiting progressively increased dilation factors d, thereby broadening the model's receptive field and enhancing its capacity to grasp extensive temporal dependencies.

Proof. When Φ is fixed and L represents a squared loss function, the objective function can be rewritten as:

$$\begin{aligned}\mathbb{E}_{(\mathbf{s},\zeta)\sim p(\mathbf{s},\zeta)}[L(\mathcal{D}(\mathbf{s}),\zeta)] &= \mathbb{E}_{(\mathbf{s},\zeta)\sim p(\mathbf{s},\zeta)}[(\mathcal{D}(\mathbf{s})-\zeta)^2] &(14)\\ &= \mathbb{E}_{\mathbf{s}\sim p(\mathbf{s})}\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[(\mathcal{D}(\mathbf{s})-\zeta)^2|\mathbf{s}] &(15)\\ &= \mathbb{E}_{\mathbf{s}\sim p(\mathbf{s})}\left[\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[\zeta^2|\mathbf{s}] - 2\mathcal{D}(\mathbf{s})\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[\zeta|\mathbf{s}] + \mathcal{D}(\mathbf{s})^2\right]. &(16)\end{aligned}$$

For a fixed $\mathbf{s}$, $\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[\zeta^2|\mathbf{s}]$ can be considered as a constant. Therefore, we only need to minimize:

$$-2\mathcal{D}(\mathbf{s})\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[\zeta|\mathbf{s}] + \mathcal{D}(\mathbf{s})^2. \tag{17}$$

It is clear that the optimal discriminator, denoted by $\mathcal{D}^*$, is characterized by the condition $\mathcal{D}^*(\mathbf{s}) = \mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[\zeta]$, which thus completes the proof. □

Theorem C.2. *The objective $\mathcal{O}$ achieves its maximum value if and only if the learned representation function Φ ensures that all* $\mathbf{s}$ *satisfy the condition $P(\zeta|\mathbf{s}) = P(\zeta)$. This signifies that Φ has established a balance in the distribution of samples, namely:*

$$P(\Phi_t, \mathbf{A}_t) = P(\Phi_t)P(\mathbf{A}_t). \tag{18}$$

Proof. With the optimal discriminator $\mathcal{D}^*$, Eq. 3 aims to maximize the objective $\mathcal{O}$ of the objective $\mathcal{O}$:

$$\begin{aligned}\mathcal{O} &\equiv \mathbb{E}[L(\mathcal{D}^*(\mathbf{s}),\zeta)] &(19)\\ &= \mathbb{E}_{\mathbf{s}\sim p(\mathbf{s})}\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[(\mathbb{E}_{\zeta\sim p(\zeta|\mathbf{s})}[\zeta]-\zeta)^2] &(20)\\ &= \mathbb{E}_{\mathbf{s}\sim p(\mathbf{s})}\mathbb{V}[\zeta|\mathbf{s}]. &(21)\end{aligned}$$

According to the Law of Total Variance Theorem, we have:

$$\mathbb{V}(\zeta) = \mathbb{E}_{\mathbf{s}\sim p(\mathbf{s})}[\mathbb{V}(\zeta|\mathbf{s})] + \mathbb{V}_{\mathbf{s}\sim p(\mathbf{s})}(\mathbb{E}[\zeta|\mathbf{s}]). \tag{22}$$

Due to the non-negativity of the variance:

$$\mathbb{E}_{\mathbf{s}\sim p(\mathbf{s})}\mathbb{V}[\zeta|\mathbf{s}] \le \mathbb{V}[\zeta]. \tag{23}$$

The equality holds if and only if $\mathbb{V}_{\mathbf{s}\sim p(\mathbf{s})}(\mathbb{E}[\zeta|\mathbf{s}]) = 0$, i.e., there exists a constant μ_c such that $\mathbb{E}[\zeta|\mathbf{s}] = \mu_c$.

Using the law of total expectation, $\forall\zeta$, we have $\mathbb{E}[\zeta|\mathbf{s}] = \mathbb{E}[\zeta]$.

Given that $\zeta \in \{0,1\}$, it follows that $p(\zeta|\mathbf{s}) = p(\zeta)$, i.e. $\zeta \perp \mathbf{s}$, implying that $P(\Phi_t, \mathbf{A}_t) = P(\Phi_t)P(\mathbf{A}_t)$, thereby completing the proof.

□

D. Temporal Convolutional Network

Temporal Convolutional Networks (TCN) (Oord *et al.*, 2016) represent a class of neural network architectures tailored for the effective processing of time-series data. Figure 5 illustrates the TCN's fundamental structure with one-dimensional convolutional layers. These layers are architected to enable the prediction at any given time step to be contingent exclusively on the input information up to that point, precluding any inadvertent inclusion of future data. TCN incorporates Dilated Convolutions to wide the receptive field with a dilation factor. Formally, given a one-dimensional sequence input $\mathbf{z} \in \mathbb{R}^n$ and a filter function $f : \{0, \dots, k-1\} \to \mathbb{R}$, the dilation convolution operation F at a sequence point t is delineated as:

$$F(t) = (\mathbf{z} *_d f)(t) = \sum_{i=0}^{k-1} f(i) \cdot \mathbf{z}_{t-d \cdot i}, \tag{24}$$

where d is the dilation factor, k is the size of the filter, and the term $t - d \cdot i$ captures the historical reach of the convolution.

Following the methodology of (Bai *et al.*, 2018), our TCN architecture incorporates residual connections, an approach empirically validated to benefit deep networks. As delineated by (He *et al.*, 2016), a residual block includes a bypass that adds its output to the block's input $\mathbf{z}$:

$$\mathbf{o} = \text{Activation}(B(\mathbf{z}) + F(\mathbf{z})). \tag{25}$$

In this context, B serves as the identity map when the input $\mathbf{z}$ is dimensionally consistent with the output $F(\mathbf{z})$. Conversely, should their dimensions vary, B assumes the role of a 1×1 convolution, aligning the dimensions for compatibility.

E. Datasets Description and Additional Results

E.1. Fully-synthetic tumor growth dataset

In the study by (Geng *et al.*, 2017), the volume of tumor growth for a period of $t+1$ days following a cancer diagnosis is modeled by the Tumor Growth (TG) simulator, with the outcome being one-dimensional. The simulator incorporates two binary treatment strategies: radiotherapy ($\mathbf{A}_t^r$) and chemotherapy ($\mathbf{A}_t^c$). The treatments are applied in the model as follows: radiotherapy has an instantaneous effect $d(t)$ on the subsequent outcome when administered to a patient. Chemotherapy, on the other hand, influences multiple future outcomes with a diminishing effect $C(t)$ described by the equation:

$$\mathbf{Y}_{t+1} = \left(1 + \rho \log\left(\frac{K}{\mathbf{Y}_t}\right) - \beta_C C_t - (\alpha_r d_t + \beta_r d_t^2) + \epsilon_t\right) \mathbf{Y}_t, \tag{26}$$

where the simulation parameters $\rho, K, \beta_C, \alpha_r, \beta_r$ are specified, and the noise ϵ_t is modeled as an independent sample from a normal distribution $N(0, 0.01^2)$. The individual patient response is characterized by the parameters $\beta_C, \alpha_r, \beta_r$, which are drawn from a mixture of three truncated normal distribution components. Consult the code implementation [3] for the precise parameter values. The mixture component indices serve as static covariates. A biased treatment assignment introduces time-varying confounding for both treatments, expressed as:

$$\mathbf{A}_t^r, \mathbf{A}_t^c \sim Bernoulli\left(\sigma\left(\frac{\gamma}{D_{max}}\left(\bar{D}_{15}(\bar{\mathbf{Y}}_{t-1}) - D_{max}/2\right)\right)\right), \tag{27}$$

with $\sigma(\cdot)$ representing a sigmoid activation function, D_{max} the maximum tumor diameter, $\bar{D}_{15}(\bar{\mathbf{Y}}_{t-1})$ the average tumor diameter over the past 15 days, and γ the confounding parameter. By adjusting γ, the level of confounding can be controlled. With $\gamma = 0$, treatment assignments are fully randomized, whereas higher values of γ increase the extent of time-varying confounding.

In our implementation, RMSNs are configured with two direct binary treatments. For all other methods, we operationalize treatment as a one-hot encoded variable, selecting a unique category from the set $\{(\mathbf{A}_t^r = 0, \mathbf{A}_t^c = 0), (\mathbf{A}_t^r = 1, \mathbf{A}_t^c = 0), (\mathbf{A}_t^r = 0, \mathbf{A}_t^c = 1), (\mathbf{A}_t^r = 1, \mathbf{A}_t^c = 1)\}$.

At each time step for every patient within the test cohort, a series of counterfactual paths are generated, contingent on τ. For predictions that are one step ahead, the full quartet of possible one-step-ahead counterfactual outcomes Y_{t+1} is simulated. These mirror the tumor volume for every treatment combination possible. When predicting multiple steps ahead, the total count of possible outcomes for $Y_{t+2}, \dots, Y_{t+\tau_{\max}}$ proliferates exponentially in line with the projection horizon $\tau_{\max}$. Hence, we employ two distinct approaches as per (Melnychuk *et al.*, 2022):

[3] Please refer to our supplementary materials.

1. *Single sliding treatment.* This approach simulates trajectories with a sole treatment that is sequentially shifted across a timeframe extending from t to $t + \tau_{\max} - 1$, to affirm the precision in treatment timing. This process yields $2(\tau_{\max} - 1)$ distinct trajectories.

2. *Random trajectories.* In this instance, a set number of trajectories, specifically $2(\tau_{\max} - 1)$, are generated with randomized treatment allocations.

This former approach is a replication of the one used in (Bica *et al.*, 2020b). The inclusion of the latter approach aims to encompass a broader spectrum of trajectories involving multiple treatments.

Across varying levels of confounding γ, we generate 10,000 patient trajectories for the training phase, 1,000 for validation purposes, and another 1,000 for the testing phase. The trajectory duration is capped at a maximum of 60 time steps, acknowledging that certain patients may exhibit shorter trajectories due to recovery or demise.

Consistent with previous works (Bica *et al.*, 2020b; Melnychuk *et al.*, 2022), we report the normalized RMSE, which is normalized with respect to the maximum tumor volume $V_{\max} = 1150$ cubic centimeters.

E.2. MIMIC-III datasets

Medical Information Mart for Intensive Care III (MIMIC-III) database (Johnson *et al.*, 2016) provides an extensive collection of de-identified electronic health records from intensive care unit patients. MIMIC-III integrates a wide range of data types, including vital signs, medications, laboratory measurements, observations and notes taken by care providers, fluid balance, procedure codes, diagnostic codes, imaging reports, hospital length of stay, survival data, and more. This information is collected from various hospital information systems which have been standardized to facilitate secondary use in research. It serves as a critical resource for validating the performance of analytical models in intricate and authentic clinical settings. In the implementation described below, MIMIC-extract (Wang *et al.*, 2020) was employed, which applies a standardized preprocessing pipeline to the MIMIC-III dataset.

Real-world data. Following the studies in (Hatt and Feuerriegel, 2021; Kuzmanovic *et al.*, 2021; Melnychuk *et al.*, 2022), we extract 25 patients covariates and 3 static features from MIMIC-III, treat diastolic blood pressure as the predictive target, and choose two treatment interventions: the use of vasopressors and mechanical ventilation.

Our analysis incorporates 25 vital sign indicators and 3 static attributes, which are also represented as one-hot-encoded variables for categorical features. These elements, encompassing both dynamic covariates and invariant characteristics, are acknowledged as potential confounding factors. Our study leverages a pair of binary treatments: the administration of vasopressors and the application of mechanical ventilation. Subsequently, we ascertain the actual outcome concerning (diastolic) blood pressure, which may be subjected to either augmentation or diminution due to the aforementioned treatments, posing a critical consideration for clinicians regarding the expected progression of patient trajectories under such interventions.

The experimental dataset comprised a selected group of 5,000 individuals, each having been admitted to the intensive care unit (ICU) for a minimum duration of 30 hours, with a cap at 60 hours for ICU stay. This cohort was then distributed into training, validation, and testing datasets in a 70%/15%/15% proportion. The study's methodology was adapted to the length of the forecast horizon τ. Specifically, (i) for predictions just one step ahead, the entirety of test set trajectories was utilized. (ii) For predictions that spanned τ steps, where $\tau \geq 2$, the process was as follows: we defined $\tau_{\max} \geq \tau$ as the furthest horizon of projection. From here, sub-trajectories extending no less than $\tau_{\max} + 1$ were isolated, adopting a rolling origin technique, and excising vital sign readings from the initial time steps up to $\tau^{(i)} - \tau_{\max} + 1$, effectively precluding any foresight in the prediction process.

Semi-synthetic data. In alignment with our real world dataset, we extracted 25 distinct vital signs to serve as time-varying covariates, along with three static covariates: gender, ethnicity, and age. Our semi-synthetic data simulation adheres to the framework established by (Melnychuk *et al.*, 2022), initially generating outcome trajectories that exhibit both inherent and external dependencies, subsequently integrating treatment interventions in a sequential manner. We posit that the interplay among treatments, outcomes, and dynamic covariates is minimal, with outcomes predominantly shaped by a select few of these elements.

Step 1: We select a sample of 1,000 patients from those with a minimum of 20 hours in intensive care, limiting the duration to a maximum of 100 hours, thus setting $T^{(i)}$ within the 20 to 100-hour range.

Step 2: For each patient i, simulate d_y untreated outcomes $\mathbf{Z}_t^{j,(i)}$, $j = 1, \ldots, d_y$, incorporating: (1) a B-spline(t) and random function $g^{j,(i)}(t)$ for endogenous effects; (2) an external dependency $f_Z^j(\mathbf{X}_t^{(i)})$ on a subset of current covariates; and (3) independent noise ε_t. This yields:

$$\mathbf{Z}_t^{j,(i)} = \alpha_S^j B\text{-spline}(t) + \alpha_g^j g^{j,(i)}(t) + \alpha_f^j f_Z^j(\mathbf{X}_t^{(i)}) + \varepsilon_t, \tag{28}$$

In this stage, noise is introduced as $\varepsilon_t \sim N(0, 0.005^2)$, and weights α_S^j, α_g^j, and α_f^j are assigned. The B-spline(t) is constructed from three cubic splines representing different ICU stay trends. Patient-specific randomness $g^{j,(i)}(t)$ arises from a Gaussian process with a Matérn kernel, while the external effect $f_Z^j(\cdot)$ is modeled using a random Fourier features (RFF) approach, avoiding the complex Cholesky decomposition. This setup captures multi-scale endogenous patterns and selective exogenous influences on time-variant covariates.

Step 3: We simulate synthetic d_a binary treatments $\mathbf{A}_t^l$, $l = 1, \ldots, d_a$, with confounding influenced by both prior treated outcomes $\bar{\mathbf{A}}_{T_l}(\bar{\mathbf{Y}}_{t-1})$ and current covariates, determined as follows:

$$p_{A_t^l} = \sigma(\gamma_A^l \bar{\mathbf{A}}_{T_l}(\bar{\mathbf{Y}}_{t-1}) + \gamma_X^l f_Y^l(\mathbf{X}_t) + b_l), \tag{29}$$

$$\mathbf{A}_t^l \sim \text{Bernoulli}(p_{\mathbf{A}_t^l}), \tag{30}$$

where $\sigma(\cdot)$ denotes the sigmoid function, γ_A^l and γ_X^l represent parameters contributing to confounding, b_l acts as a constant bias term, and $f_Y^l(\cdot)$ is obtained through an RFF-based approximation of a Gaussian process, analogous to $f_Z^j(\cdot)$.

Step 4: Treatments are administered to the initial outcomes, setting $\mathbf{Y}_1$ equal to $\mathbf{Z}_1$. We model each treatment l to impart a sustained influence on a particular outcome j, manifesting its maximum additive impact β_{lj} immediately post-application. The treatment's influence is confined to a specific time frame $t - w_l, \ldots, t$ and its magnitude diminishes over time following an inverse-square law, also being modulated by the treatment's probability $p_{A_t^l}$. Subsequently, the composite impact of various treatments is computed by collating their minimum effects over the treatment duration. The mathematical representation of this process is:

$$E_j(t) = \sum_{i=t-w_l}^{t} \left(\frac{\min_{l=1,\ldots,d_a} \mathbb{1}_{[\mathbf{A}_i^l=1]} p_{A_i^l} \beta_{lj}}{(w^l - i)^2}\right), \tag{31}$$

where β_{lj} signifies the peak impact of treatment l. It's either a consistent value across all outcomes j or null, ensuring the treatment exerts no influence on the outcome.

Step 5: The simulated treatment effect $E_j(t)$ is merged with the untreated outcome, resulting in:

$$\mathbf{Y}_t^j = \mathbf{Z}_t^j + E_j(t). \tag{32}$$

Step 6: Utilizing the established simulator framework, we fabricate our semi-synthetic dataset. Detailed specifications of the simulation parameters are documented in the code implementation. Post the generation of three synthetic binary treatments and two synthetic outcomes, the dataset comprising 1,000 patients is partitioned into training, validation, and testing sets, adhering to a 60%/20%/20% distribution ratio. For one-step-ahead predictions, we compute all 2^3 counterfactual outcomes. When projecting over multiple steps, delineated by $\tau_{\max}$, we generate $\tau_{\max}$ diverse trajectories for each patient at each temporal juncture.

E.3. Continuous intervention synthetic dataset

We construct a synthetic dataset of continuous interventions using an autoregressive process, where the treatment variable $\mathbf{A}_t \in [0, 1]$. The time series is generated iteratively, following the steps outlined below.

The generation of the treatment variable $\mathbf{A}_t$ at time t involves a sequence of operations on historical covariate data. The mean of historical covariates, denoted as $\mathbf{X}_m$, is the average across the time dimension:

$$\mathbf{X}_m = \frac{1}{w} \sum_{i=1}^{w} \mathbf{X}_{t-i}, \tag{33}$$

where w is the number of previous time steps that influence the current value. This mean covariate vector $\mathbf{X}_m$ undergoes a series of non-linear transformations and noise addition to form a decision variable:

$$d_t = \sin(2\pi \mathbf{X}_m^1) + \cos(2\pi \mathbf{X}_m^2) \cdot \mathbf{X}_m^5 + \max(\mathbf{X}_m^3, \mathbf{X}_m^4) + N(0, \sigma_a^2), \tag{34}$$

where σ_a represents the noise scale parameter for $\mathbf{A}$, and the superscript denotes the index of an element within the vector $\mathbf{X}_m$. The decision variable d_t is then mapped to a probability p_t using the sigmoid function: $p_t = \frac{1}{1+\exp(-d_t)}$. Finally, the treatment variable $\mathbf{A}_t$ is sampled from a Beta distribution, with shape parameters influenced by p_t and a scaling parameter γ, formalized as:

$$\gamma_1 = 1 + \gamma \cdot p_t, \quad \gamma_2 = 1 + \gamma \cdot (1 - p_t), \tag{35}$$

$$\mathbf{A}_t \sim \text{Beta}(\gamma_1, \gamma_2). \tag{36}$$

In this process, $\mathbf{A}_t$ is designed to capture the influence of historical covariate information through non-linear, noise-perturbed dynamics and to represent the inherent stochasticity of treatment effects via probabilistic modelling with the Beta distribution.

Given a treatment variable $\mathbf{A}_t$, we first apply a series of non-linear transformations to obtain a transformed treatment array, denoted as $T(\mathbf{A}_t)$. This array is then processed through a predefined masking procedure, resulting in a masked transformation matrix $\mathbf{A}_{\text{matrix}}$.

The covariates at time t, $\mathbf{X}_t$, are subsequently generated by integrating $\mathbf{X}_m$ with $\mathbf{A}_{\text{matrix}}$, followed by the addition of Gaussian noise:

$$\mathbf{X}_t = \mathbf{X}_m \times \mathbf{A}_{\text{matrix}} + N(0, \sigma_x^2), \tag{37}$$

where σ_x represents the noise scale parameter for $\mathbf{X}$.

The outcome at time t, $\mathbf{Y}_t$, is subsequently generated by integrating $\mathbf{X}_m$ with $\mathbf{A}_t$, followed by the addition of Gaussian noise:

$$\mathbf{Y}_t = \cos(2\pi \mathbf{A}_t) \cdot \mathbf{X}_m^1 + \mathbf{A}_t^2 \cdot \mathbf{X}_m^4 + \sin(2\pi \mathbf{A}_t) \cdot \mathbf{X}_m^6 + \exp(\mathbf{A}_t) \cdot \mathbf{X}_m^3 + N(0, \sigma_y^2), \tag{38}$$

where σ_y represents the noise scale parameter for $\mathbf{Y}$.

For one-step-ahead predictions, we randomly select five interventions $\mathbf{A}_t$ from a uniform distribution $U(0, 1)$ to compute counterfactual outcomes. In scenarios where the prediction extends over multiple steps, as defined by τ_{max}, we generate a set of τ_{max} diverse trajectories for each patient at every time step.

F. Hyperparameter Tuning

Following the methodology used in CT (Melnychuk *et al.*, 2022), we conduct hyperparameter optimization for all baseline models and ACTIN using random searches. The ranges for the random searches for RMSN, CRN, G-Net, and CT are provided in Tables 6, 7, 8, and 9, respectively. The random search space for ACTIN is outlined in Table 10. In ACTIN, we conduct hyperparameter optimization for two distinct base models, TCN and LSTM, as delineated in Table 10. For TCN, our search encompassed channel sizes, dilation factors, and kernel sizes, whereas for LSTM, we explore various configurations of LSTM hidden units and LSTM layers. It is pertinent to note that, within our experiments, all sub-models within ACTIN utilizes the same base model, although this homogeneity is not a prerequisite.

To ensure a fair comparison, we maintain similar hyperparameter ranges across models. These ranges vary slightly due to differences in the models' input characteristics and convergence behaviors. For instance, compared to CT, ACTIN requires approximately 60% more epochs on the MIMIC-III dataset. This increment is attributed to ACTIN's strategy for learning balanced representations, necessitating larger minibatch sizes, which in turn reduces the number of gradient updates per epoch, thereby requiring a greater number of epochs to achieve convergence.

Table 6. The ranges for hyperparameter tuning of RMSN are specified for various datasets. The symbols N_{Pt}, N_{Ph}, N_E, and N_D denote the Propensity treatment network, Propensity history network, Encoder, and Decoder sub-models, respectively. Unless otherwise stated, it is assumed that all sub-models adhere to the same hyperparameter range.

Hyperparameter	**Range (TG)**	**Range (SS)**	**Range (RW)**
LSTM layers	1	1, 2	1, 2
Learning rate	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001
Minibatch size (N_{Pt}, N_{Ph}, N_E)	64, 128, 256	64, 128, 256	64, 128, 256
Minibatch size (N_D)	256, 512, 1024	256, 512, 1024	256, 512, 1024
LSTM hidden units (N_{Pt})	2, 4, 8, 12, 16	1, 3, 6	1, 2, 4
LSTM hidden units (N_{Ph}, N_E)	2, 4, 8, 12, 16	37, 74, 148	36, 72, 144
LSTM hidden units (N_D)	4, 8, 16, 32, 64	49, 98, 196	47, 94, 188
LSTM dropout rate	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5
Max gradient norm (N_{Pt}, N_{Ph}, N_E)	0.5, 1.0, 2.0	0.5, 1.0, 2.0	0.5, 1.0, 2.0
Max gradient norm (N_D)	0.5, 1.0, 2.0, 4.0	0.5, 1.0, 2.0, 4.0	0.5, 1.0, 2.0, 4.0
Random search iterations (N_{Pt}, N_{Ph}, N_E)	50	50	50
Random search iterations (N_D)	20	20	20
Number of epochs	100	400	200

Table 7. The ranges for hyperparameter tuning of CRN are specified for various datasets. The symbols N_E, and N_D denote the Encoder and Decoder sub-models, respectively. Unless otherwise stated, it is assumed that all sub-models adhere to the same hyperparameter range.

Hyperparameter	**Range (TG)**	**Range (SS)**	**Range (RW)**
LSTM layers	1	1, 2	1, 2
Learning rate	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001
Minibatch size (N_E)	64, 128, 256	64, 128, 256	64, 128, 256
Minibatch size (N_D)	256, 512, 1024	256, 512, 1024	256, 512, 1024
LSTM hidden units (N_E)	3, 6, 12, 18, 24	37, 74, 148	36, 72, 144
LSTM hidden units (N_D)	d_r^e	d_r^e	d_r^e
BR size d_r^e (N_E)	3, 6, 12, 18, 24	37, 74, 148	36, 72, 144
BR size d_r^d (N_D)	3, 6, 12, 18, 24	49, 98, 196	47, 94, 188
FC hidden units (N_E)	$0.5d_r^e$, $1d_r^e$, $2d_r^e$, $3d_r^e$, $4d_r^e$	$0.5d_r^e$, $1d_r^e$, $2d_r^e$	$0.5d_r^e$, $1d_r^e$, $2d_r^e$
FC hidden units (N_D)	$0.5d_r^d$, $1d_r^d$, $2d_r^d$, $3d_r^d$, $4d_r^d$	$0.5d_r^e$, $1d_r^e$, $2d_r^e$	$0.5d_r^e$, $1d_r^e$, $2d_r^e$
LSTM dropout rate	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5
Random search iterations (N_E)	50	50	50
Random search iterations (N_D)	30	30	30
Number of epochs	100	400	200

Table 8. The ranges for hyperparameter tuning of G-Net are specified for various datasets.

Hyperparameter	**Range (TG)**	**Range (SS)**	**Range (RW)**
LSTM layers	1	1, 2	1, 2
Learning rate	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001
Minibatch size	64, 128, 256	64, 128, 256	64, 128, 256
LSTM hidden units	3, 6, 12, 18, 24	37, 74, 148	36, 72, 144
LSTM output size d_o	3, 6, 12, 18, 24	37, 74, 148	36, 72, 144
FC hidden units	$0.5d_o$, $1d_o$, $2d_o$, $3d_o$, $4d_o$	$0.5d_o$, $1d_o$, $2d_o$	$0.5d_o$, $1d_o$, $2d_o$
LSTM dropout rate	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5
Random search iterations	50	50	50
Number of epochs	50	400	200

Table 9. The ranges for hyperparameter tuning of CT are specified for various datasets.

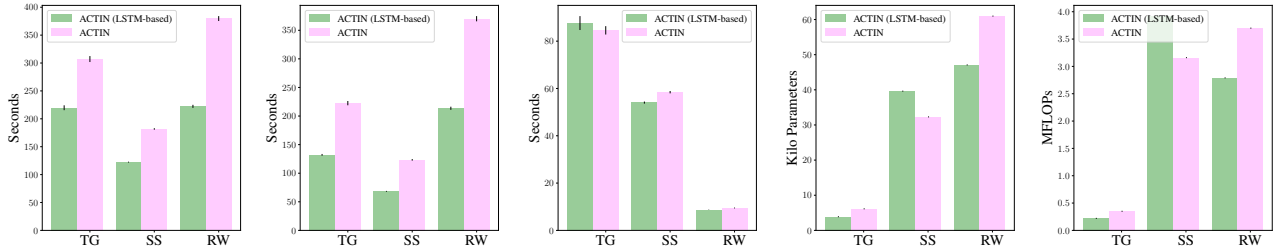
Hyperparameter	**Range (TG)**	**Range (SS)**	**Range (RW)**
Transformer blocks	1	1, 2	1, 2
Learning rate	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001	0.01, 0.001, 0.0001
Minibatch size	64, 128, 256	32, 64	32, 64
Attention heads	2	2, 3	2, 3
Transformer units	4, 8, 12, 16	24, 48, 64	24, 48, 64
BR size d_r	2, 4, 8, 12, 16	22, 44, 88	22, 44, 88
FC hidden units	$0.5d_r$, $1d_r$, $2d_r$, $3d_r$, $4d_r$	$0.5d_r$, $1d_r$, $2d_r$	$0.5d_r$, $1d_r$, $2d_r$
Sequential dropout rate	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5	0.1, 0.2, 0.3, 0.4, 0.5
Max positional encoding	15	20	30
Random search iterations	50	50	50
Number of epochs	150	400	300

Table 10. The ranges for hyperparameter tuning of ACTIN are specified for various datasets.

Hyperparameter		**Range (TG)**	**Range (SS)**	**Range (RW)**
Linear transformation size		4, 8, 16	16, 32, 64	16, 32, 64
Learning rate l		0.01, 0.002, 0.001	0.01, 0.002, 0.001	0.01, 0.002, 0.001
Learning rate $l_{\mathcal{D}}$		0.001, 0.0002, 0.0001	0.001, 0.0002, 0.0001	0.001, 0.0002, 0.0001
Minibatch size		64, 128, 256	64, 128, 256	64, 128, 256
BR size d_r		8, 12, 16	16, 32, 64	16, 32, 64
λ_X		-	0.1, 0.05, 0.01	0.1, 0.05, 0.01
λ_X		0.01	0.01	0.01
TCN-based	Kernel sizes	2, 3	2, 3	2, 3
	Dilation factors	2, 3	2, 3	2, 3
	Channel size d_c	4, 8, 12, 16	20, 22, 24	28, 32, 36
LSTM-based	LSTM layers	1	1, 2	1, 2
	LSTM hidden units	4, 8, 12, 16	16, 32, 64	16, 32, 64
FC hidden units		16, 32, 64	16, 32, 64	16, 32, 64
Dropout rate		0.1, 0.2, 0.3	0.1, 0.2, 0.3	0.1, 0.2, 0.3
Random search iterations		50	50	50
Number of epochs		150	500	500

	$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$
CRN	2.08±0.45	2.35±0.28	2.75±0.32	3.13±0.27	3.53±0.35	3.89±0.37
CT	2.27±0.37	2.52±0.30	2.92±0.22	3.29±0.21	3.85±0.20	4.25±0.29
ACTIN (LSTM-based)	1.77±0.39	1.99±0.22	2.51±0.21	3.12±0.19	3.70±0.25	4.13±0.29
ACTIN (LSTM-based) *w/o integrating*	1.52±0.34	1.65±0.22	2.10±0.24	2.59±0.21	3.10±0.25	3.58±0.28
ACTIN	1.00±0.16	1.29±0.51	1.78±0.56	2.35±0.59	2.90±0.63	3.38±0.71

Table 11. τ-step-ahead prediction results on the CISD dataset across different τ values. Results are presented as mean $\pm$ standard deviation over five runs.



(a) Running time (b) Training time (c) Inference time (d) Parameters (e) Computational Load

Figure 6. The comparison of running costs and model complexities for ACTIN across various base models on a fully synthetic tumor dataset (Tumor, $\gamma = 0$), the semi-synthetic (SS) dataset derived from MIMIC-III, and the real-world dataset (RW).

G. Additional Results

In the following, we furnish additional results across various datasets. This includes one-step-ahead and multi-step-ahead prediction outcomes for the fully-synthetic tumor dataset under the *single sliding treatment* setting (Table 12), multi-step-ahead prediction outcomes for the fully-synthetic tumor dataset under the *random trajectories* setting (Table 13), as well as one-step-ahead and multi-step-ahead prediction outcomes for the MIMIC-III Real-World (RW) and Semi-Synthetic (SS) datasets (Table 14 and Table 15). It is imperative to note that within the fully-synthetic tumor dataset, the one-step-ahead prediction outcomes under the *single sliding treatment* setting and the *random trajectories* setting were consistent. Consequently, only the results from the former setting are reported.

Experimental results indicate that ACTIN, when employing TCN as the base model, achieved state-of-the-art performance in nearly all evaluations. Moreover, ACTIN instantiated with LSTM as the base model displayed results comparable to, and occasionally surpassing, those of CT, notably on the MIMIC-III Semi-Synthetic (SS) dataset. Figure 6 contrasts the running costs of ACTIN with these two distinct base models across various datasets, revealing a slight edge for LSTM-based ACTIN over its TCN-based counterpart. These findings highlight the superiority of ACTIN in enhancing the efficiency and effectiveness of ostensibly simple base models.

Additionally, Table 11 compares the experimental results of CRN, CT, and ACTIN on the CISD dataset, all of which employ the balancing strategy proposed in this paper. The experimental results clearly show that our dual-module design significantly impacts performance. This is likely because the impact of continuous interventions can be easily overlooked by models, while our design effectively emphasizes this aspect.

Table 12. Under the *single sliding treatment* setting, we present the one-step-ahead and multi-step-ahead prediction results for the fully-synthetic tumor dataset. Shown: RMSE as mean ± standard deviation over five runs.

		$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$
$\gamma = 0$	RMSN	0.91±0.04	0.81±0.07	0.83±0.08	0.86±0.09	0.88±0.10	0.91±0.09
	CRN	0.78±0.05	0.69±0.05	0.72±0.05	0.75±0.05	0.78±0.07	0.82±0.09
	G-Net	0.83±0.05	0.95±0.13	1.12±0.20	1.21±0.24	1.26±0.26	1.29±0.27
	CT	0.78±0.06	0.69±0.06	0.71±0.05	0.73±0.05	0.76±0.05	0.79±0.05
	ACTIN (LSTM-based)	0.77±0.06	0.69±0.06	0.71±0.06	0.73±0.06	0.75±0.06	0.78±0.06
	ACTIN	**0.75±0.06**	**0.66±0.05**	**0.68±0.05**	**0.70±0.05**	**0.73±0.05**	**0.76±0.05**
$\gamma = 1$	RMSN	1.12±0.10	1.03±0.09	1.05±0.07	1.06±0.07	1.06±0.06	1.07±0.06
	CRN	0.82±0.05	0.73±0.06	0.76±0.05	0.78±0.05	0.81±0.04	0.84±0.04
	G-Net	0.87±0.08	0.98±0.09	1.15±0.13	1.22±0.16	1.26±0.20	1.28±0.23
	CT	0.80±0.08	0.71±0.06	0.75±0.05	0.79±0.05	0.83±0.07	0.86±0.08
	ACTIN (LSTM-based)	0.83±0.04	0.73±0.07	0.75±0.05	0.78±0.04	0.80±0.04	0.82±0.04
	ACTIN	**0.78±0.04**	**0.67±0.04**	**0.70±0.04**	**0.73±0.03**	**0.76±0.04**	**0.78±0.03**
$\gamma = 2$	RMSN	1.14±0.07	1.09±0.23	1.12±0.17	1.14±0.12	1.17±0.11	1.21±0.13
	CRN	0.89±0.07	0.75±0.05	0.84±0.07	0.93±0.10	1.02±0.12	1.09±0.13
	G-Net	1.00±0.06	1.03±0.08	1.18±0.11	1.26±0.15	1.30±0.18	1.33±0.20
	CT	0.87±0.08	0.73±0.06	0.78±0.07	0.84±0.07	0.88±0.08	0.92±0.10
	ACTIN (LSTM-based)	0.90±0.05	0.79±0.08	0.85±0.08	0.90±0.09	0.95±0.11	0.99±0.13
	ACTIN	**0.85±0.07**	**0.68±0.04**	**0.73±0.04**	**0.77±0.06**	**0.81±0.08**	**0.85±0.09**
$\gamma = 3$	RMSN	1.35±0.11	1.22±0.07	1.26±0.12	1.29±0.14	1.33±0.17	1.37±0.19
	CRN	1.13±0.17	1.07±0.37	1.27±0.55	1.42±0.64	1.53±0.68	1.61±0.69
	G-Net	1.30±0.23	1.12±0.09	1.25±0.09	1.28±0.10	1.30±0.11	1.31±0.12
	CT	1.02±0.12	0.80±0.09	0.87±0.11	0.94±0.13	1.00±0.15	1.04±0.16
	ACTIN (LSTM-based)	1.07±0.13	0.86±0.11	0.96±0.15	1.03±0.18	1.08±0.21	1.13±0.23
	ACTIN	**1.01±0.13**	**0.73±0.05**	**0.80±0.07**	**0.85±0.10**	**0.90±0.11**	**0.94±0.12**
$\gamma = 4$	RMSN	1.39±0.18	0.97±0.18	1.10±0.26	1.22±0.35	1.31±0.41	1.38±0.46
	CRN	1.33±0.17	1.06±0.14	1.28±0.25	1.45±0.34	1.56±0.41	1.65±0.46
	G-Net	1.37±0.25	1.16±0.14	1.44±0.18	1.61±0.25	1.73±0.33	1.82±0.42
	CT	1.35±0.24	1.01±0.28	1.13±0.35	1.23±0.40	1.29±0.43	1.33±0.45
	ACTIN (LSTM-based)	1.42±0.23	1.07±0.18	1.18±0.22	1.27±0.25	1.33±0.26	1.37±0.26
	ACTIN	**1.26±0.21**	**0.81±0.12**	**0.93±0.18**	**1.02±0.22**	**1.10±0.25**	**1.17±0.26**

Table 13. Under the *random trajectories* setting, we present the multi-step-ahead prediction results for the fully-synthetic tumor dataset. Shown: RMSE as mean ± standard deviation over five runs.

		$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$
$\gamma = 0$	RMSN	0.90±0.08	0.89±0.08	0.85±0.07	0.78±0.06	0.70±0.05
	CRN	0.78±0.04	0.79±0.05	0.75±0.06	0.69±0.06	0.62±0.06
	G-Net	0.96±0.12	1.02±0.14	0.97±0.14	0.89±0.13	0.79±0.12
	CT	0.77±0.06	0.77±0.05	0.73±0.05	**0.67±0.05**	**0.60±0.04**
	ACTIN (LSTM-based)	0.77±0.03	0.77±0.04	0.73±0.05	**0.67±0.05**	**0.60±0.05**
	ACTIN	**0.75±0.08**	**0.76±0.08**	**0.72±0.09**	**0.67±0.09**	**0.60±0.08**
$\gamma = 1$	RMSN	1.03±0.05	1.02±0.06	0.97±0.06	0.90±0.06	0.82±0.06
	CRN	0.81±0.06	0.82±0.06	0.78±0.06	0.72±0.06	0.65±0.05
	G-Net	0.98±0.09	1.06±0.13	1.01±0.14	0.92±0.15	0.82±0.15
	CT	0.78±0.05	0.80±0.05	0.77±0.05	0.71±0.05	0.65±0.05
	ACTIN (LSTM-based)	0.82±0.10	0.81±0.09	0.77±0.08	0.71±0.08	0.63±0.07
	ACTIN	**0.76±0.05**	**0.77±0.06**	**0.74±0.06**	**0.69±0.05**	**0.62±0.05**
$\gamma = 2$	RMSN	1.09±0.11	1.06±0.07	1.00±0.08	0.91±0.09	0.82±0.10
	CRN	0.81±0.04	0.86±0.08	0.85±0.10	0.81±0.11	0.76±0.11
	G-Net	1.01±0.06	1.06±0.07	1.00±0.09	0.91±0.11	0.82±0.13
	CT	0.83±0.08	0.84±0.11	0.81±0.12	0.75±0.12	0.68±0.12
	ACTIN (LSTM-based)	0.83±0.09	0.84±0.12	0.80±0.13	0.75±0.15	0.68±0.15
	ACTIN	**0.78±0.10**	**0.80±0.14**	**0.77±0.16**	**0.71±0.16**	**0.65±0.16**
$\gamma = 3$	RMSN	1.19±0.11	1.17±0.16	1.11±0.18	1.04±0.18	0.95±0.17
	CRN	0.96±0.19	1.07±0.30	1.07±0.35	1.02±0.34	0.95±0.31
	G-Net	1.12±0.13	1.17±0.17	1.08±0.20	0.98±0.20	0.87±0.19
	CT	0.89±0.14	0.93±0.18	0.90±0.20	0.84±0.20	0.77±0.18
	ACTIN (LSTM-based)	0.93±0.17	0.97±0.21	0.93±0.22	0.87±0.21	0.79±0.19
	ACTIN	**0.83±0.14**	**0.87±0.18**	**0.84±0.19**	**0.79±0.19**	**0.72±0.17**
$\gamma = 4$	RMSN	1.08±0.20	1.16±0.25	1.14±0.26	1.07±0.25	0.97±0.23
	CRN	1.12±0.15	1.26±0.23	1.29±0.26	1.24±0.26	1.15±0.25
	G-Net	1.20±0.17	1.34±0.20	1.32±0.22	1.26±0.23	1.18±0.24
	CT	1.13±0.27	1.17±0.30	1.14±0.30	1.06±0.29	0.96±0.26
	ACTIN (LSTM-based)	1.13±0.20	1.16±0.20	1.11±0.19	1.02±0.18	0.92±0.16
	ACTIN	**0.97±0.16**	**1.05±0.20**	**1.03±0.20**	**0.97±0.20**	**0.88±0.19**

Table 14. One-step-ahead and multi-step-ahead prediction results on the RW dataset. Shown: RMSE as mean ± standard deviation over five runs.

	$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$
RMSN	5.26±0.13	10.02±0.53	11.18±0.71	11.93±1.07	12.56±1.47	13.12±1.82
CRN	4.85±0.06	9.13±0.16	9.77±0.16	10.10±0.17	10.36±0.20	10.58±0.22
G-Net	5.05±0.04	11.89±0.19	12.92±0.25	13.59±0.28	14.09±0.30	14.52±0.38
CT	4.60±0.09	8.99±0.21	9.59±0.22	9.91±0.25	10.14±0.29	10.34±0.32
ACTIN (LSTM-based)	4.67±0.08	9.06±0.15	9.66±0.15	9.97±0.17	10.22±0.19	10.42±0.21
ACTIN	**4.56±0.10**	**8.98±0.18**	**9.56±0.18**	**9.87±0.19**	**10.11±0.21**	**10.30±0.23**

Table 15. One-step-ahead and multi-step-ahead prediction results on the SS dataset. Shown: RMSE as mean ± standard deviation over five runs.

	$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 4$	$\tau = 5$	$\tau = 6$	$\tau = 7$	$\tau = 8$	$\tau = 9$	$\tau = 10$	$\tau = 11$
RMSN	0.23±0.01	0.47±0.01	0.59±0.01	0.70±0.03	0.79±0.06	0.88±0.08	0.95±0.08	1.01±0.07	1.04±0.05	1.07±0.04	1.10±0.03
CRN	0.29±0.01	0.46±0.01	0.57±0.01	0.62±0.01	0.66±0.01	0.69±0.01	0.70±0.02	0.73±0.02	0.75±0.03	0.77±0.03	0.80±0.03
G-Net	0.36±0.01	0.67±0.01	0.84±0.01	0.96±0.02	1.05±0.02	1.13±0.03	1.20±0.04	1.26±0.05	1.31±0.06	1.37±0.07	1.41±0.09
CT	0.20±0.00	0.37±0.00	0.45±0.01	0.49±0.01	0.51±0.01	0.53±0.01	0.55±0.01	0.56±0.02	0.58±0.02	0.59±0.02	0.60±0.02
ACTIN(LSTM-based)	0.17±0.00	0.35±0.00	0.41±0.00	0.45±0.01	0.48±0.01	0.50±0.01	0.51±0.01	0.52±0.01	0.54±0.02	0.55±0.02	0.56±0.02
ACTIN	**0.15±0.00**	**0.33±0.00**	**0.39±0.00**	**0.43±0.00**	**0.46±0.00**	**0.48±0.00**	**0.49±0.00**	**0.50±0.00**	**0.51±0.01**	**0.52±0.01**	**0.54±0.01**