Targeted Synthetic Control Method

Yuxin Wang*, Dennis Frauen, Emil Javurek, Konstantin Hess, Yuchen Ma, Stefan Feuerriegel
LMU Munich
Munich Center of Machine Learning (MCML)

Abstract

The *synthetic control method* (SCM) estimates causal effects in panel data with a single-treated unit by constructing a counterfactual outcome as a weighted combination of untreated control units that matches the pre-treatment trajectory. In this paper, we introduce the **targeted synthetic control (TSC)** method, a new two-stage estimator that directly estimates the counterfactual outcome. Specifically, our TSC method (1) yields a targeted debiasing estimator, in the sense that the targeted updating refines the initial weights to produce more stable weights; and (2) ensures that the final counterfactual estimation is a convex combination of observed control outcomes to enable direct interpretation of the synthetic control weights. TSC is flexible and can be instantiated with arbitrary machine learning models. Methodologically, TSC starts from an initial set of synthetic-control weights via a one-dimensional targeted update through the weight-tilting submodel, which calibrates the weights to reduce bias of weights estimation arising from pre-treatment fit. Furthermore, TSC avoids key shortcomings of existing methods (e.g., the augmented SCM), which can produce unbounded counterfactual estimates. Across extensive synthetic and real-world experiments, TSC consistently improves estimation accuracy over state-of-the-art SCM baselines.

1 Introduction

The *synthetic control method* (SCM) is a widely used approach for estimating causal effects in panel data settings with a single treated unit and multiple untreated control units observed over a long pre-treatment period [2, 4]. For this, the SCM method constructs a counterfactual outcome for the treated unit as a convex combination (i.e., weighted average) of the control units based on the pre-treatment history. The estimand of interest is the causal effect on the treated unit after treatment, defined as the difference between the observed outcome of the treated unit and the outcome of the synthetic control (see Figure 1).

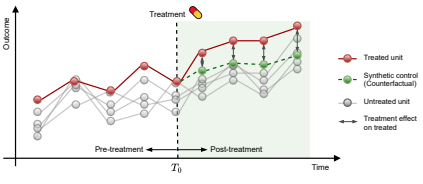


Figure 1: **Synthetic control setting**. Panel data with outcomes for single-treated unit (red) and multiple control units (green). Treatment is assigned at T_0, so that the pre-treatment history of the treated unit is used to learn synthetic-control weights based on other pre-treatment histories of control units (gray), while the post-treatment period is used to estimate the treatment effect between the treated unit and the synthetic control (dark arrow).

Example. *A canonical application of SCM is the evaluation of California's tobacco control program (Proposition 99), enacted in 1989 [4]. Because the policy was adopted by a single state, simple before–and–after comparisons are confounded by national smoking trends. The SCM addresses this by constructing a "synthetic" California under no tobacco control program as a weighted combination of other U.S. states that did not adopt the program,*

*Correspondence to: Yuxin.Wang1@lmu.de

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

with weights chosen to match California's pre-1989 cigarette sales. The difference between the observed cigarette consumption after 1989 and the outcome of synthetic control then provides an estimate of the effect of the policy intervention.

In this work, we propose the ***targeted synthetic control*** (TSC) method, a new estimator for the single-treated unit setting. Our TSC method is flexible and can be instantiated with arbitrary machine learning (ML) models. TSC is inspired by targeted maximum likelihood estimation (TMLE) to directly estimate the synthetic-control weights through a targeted updating procedure tailored to the single-treated unit setting. As a result, TSC has three favorable properties:

1. *Interpretable synthetic-control weights.* TSC ensures, by construction of the estimation procedure, that the estimated counterfactual outcome is a convex combination of observed control outcomes. The counterfactual is defined as a weighted average with nonnegative weights that sum to one, so the weights can be directly interpreted as relative contributions of control units.
2. *Boundedness of estimation.* Our updating step yields bounded estimates while preserving interpretable weights: nonnegative, normalized weights keep the predicted counterfactual within the support of control outcomes and avoid extreme extrapolation.
3. *Estimation accuracy.* By using flexible machine learning and enforcing the residual-correcting term to zero through a one-dimensional TMLE-style targeted update, TSC calibrates the synthetic-control weights using residuals of control units between outcome-regression and observed outcomes to reduce error arising from imperfect pre-treatment weight estimation. We find that this improves finite-sample stability and yields consistently improved estimation accuracy in our numerical experiments.

Methodologically, TSC follows a two-stage estimation procedure[2]. In the first stage, TSC obtains an initial set of synthetic-control weights (e.g., obtained via the classical or regularized SCM [2, 4]) and fits an outcome-regression model trained on the pre-treatment history for predictions of post-treatment outcomes. In the second stage, TSC refines the weights through a one-dimensional *"targeted"* update. Following principles from targeted maximum likelihood estimation (TMLE) [53, 52], we adopt a targeted update that leverages residual information from the control units to mitigate error arising from imperfect pre-treatment weight estimation. The resulting estimator remains a weighted combination of observed control outcomes, thereby preserving the interpretation of synthetic-control weights as relative contributions of control units.

As a result, TSC offers several practical advantages over existing methods (see Table 1). • Unlike the **classical SCM** [2, 4], TSC updates the synthetic-control weights using residuals from a flexible machine-learning outcome-regression model, improving robustness of weight estimation. • Unlike the **augmented SCM** [8], TSC preserves the synthetic control structure: the final counterfactual remains a convex combination of observed control-unit outcomes, with nonnegative weights that sum to one. In contrast, augmented SCM applies an additive regression correction, which can produce the counterfactual outcome that leaves the convex hull of the control outcomes and becomes unbounded.

We make three key **contributions**[3]: **(1)** To the best of our knowledge, we are the first to link SCM to a TMLE-style targeted updating perspective while retaining interpretable synthetic-control weights. **(2)** We propose a novel TSC method that can be combined with flexible machine learning models while ensuring bounded counterfactual predictions. **(3)** Through extensive experiments, we demonstrate state-of-the-art performance of our TSC.

2 Related work

ML for causal inference: There has been a rapidly growing interest in leveraging ML for causal inference from observational data, primarily in 'standard' settings with many treated units and sufficient treatment overlap [e.g., 52, 13, 44], where ML offers a flexible strategy to estimate nuisance components, such as outcome-regressions and propensity scores, even in high-dimensional settings. Modern causal ML estimators are often based on a two-stage framework, which combines flexible nuisance estimation with a second stage based on the efficient influence function to remove plugin bias [25]. Examples include AIPTW [41], TMLE [53, 52], the DR-learner [24], and R-learner [34]. *However, such methods for synthetic controls are underexplored.*

[2] In the causal inference literature, procedures of this form are often referred to as *meta-learners*, as they can be instantiated with arbitrary machine learning models for nuisance estimation, and then pass the estimated nuisance function into a second-stage regression, typically by regressing a constructed pseudo-outcome on covariates, to estimate the estimand of interest [26, 41, 16].

[3] Code is available at: `https://github.com/Yuxin217/targeted_synthetic_control`

Aspects	**Classical SCM** [2, 4]	**Augmented SCM** [8]	**TSC** *(ours)*
Core idea	Pre-treatment balancing weights	Outcome-model augmentation	TMLE-style weights tilting
Counterfactual form	Convex combination of control outcomes	Prediction + additive adjustment	Convex combination of control outcomes
ML model	✗ (*weights only*)	✓ (*regression*)	✓ (*regression*)
Interpretable weights	✓ (*weighted average*)	✗ (*mixture format*)	✓ (*weighted average*)
Boundedness (e.g., binary)	✓ (*convex hull*)	✗ (*prediction and additive term*)	✓ (*convex hull*)

Table 1: **Overview of key synthetic control estimators.** The table compares classical SCM, augmented SCM, and TSC in terms of how they construct the counterfactual and what properties are preserved by the estimator's final form.

A related literature stream extends causal ML methods for panel data, such as under difference-in-differences (DiD) designs [6, 27]. We provide an extended literature review in Appendix A. *However, these approaches typically focus on settings with multiple treated units and rely on strong identifying assumptions, such as parallel trends. As a result, they are not directly applicable to the synthetic control setting.*

Synthetic control method (SCM): The literature on the SCM originates from economics and statistics literature [2, 4][4], and constructs the counterfactual of the treated unit as a convex combination of outcomes from untreated control units. Here, the counterfactual outcome is a weighted average, so the weights directly reflect the relative contribution of each control unit. Several papers have extended the SCM to different settings, such as censored data [15], differential privacy [40], multiple outcomes [51], and conformal prediction [12]. More recent work relaxes parametric assumptions by adopting more flexible identification strategies [45, 46]. *However, in the classical SCM, the counterfactual is a weighted average based on the weights estimated by matching pre-treatment histories, with the limitation: the estimator is sensitive to finite-sample error in the estimated weights.*

Augmented SCM. The augmented SCM aims to reduce bias by combining synthetic-control weighting with an additional outcome model, which can be estimated flexibly via ML [8]. Here, we mitigate bias from imperfectly estimated pre-treatment weights using a regression-based correction constructed from the control units. While this approach can improve accuracy, it alters the structure of the estimator (see Table 1): *the resulting counterfactual is no longer a convex combination of observed control outcomes and can become unbounded. Consequently, synthetic-control weights can no longer be interpreted in the same way as in classical SCM, where they directly quantify the contribution of each control unit to the counterfactual.*

3 Preliminaries

3.1 Setup

We consider the classical SCM setting [2, 4] with $i = 1, 2, \ldots, N$ units observed for $t \in \mathcal{T} = \{1, 2, \ldots, T_0, \ldots, T\}$ time periods (see Figure 1). Throughout the paper, we assume both N and T are fixed. We consider a population $\left(Z, \{Y_t(0)\}_{t=1}^T, \{Y_t(1)\}_{t=1}^T, A\right) \sim \mathbb{P}$, where $Z \in \mathcal{Z} \subseteq \mathbb{R}^p$ are time-invariant baseline covariates, $A \in \mathcal{A} = \{0, 1\}$ denotes the treatment, $\{Y_t(0)\}_{t=1}^T$, and $\{Y_t(1)\}_{t=1}^T \in \mathcal{Y} \subseteq \mathbb{R}^T$ are potential outcome trajectories. We further have an observed panel dataset $\{(X_i, \bar{A}_{iT}, Y_{iT})\}_{i=1}^N \sim \mathbb{P}$ with:

- **Treatment:** Let $\bar{A}_{iT}$ indicate the deterministic treatment scheme $\bar{A}_{iT} = \{A_{it}\}_{t=1}^T$, where $A_{it} = \mathbf{1}\{i = 1\}\mathbf{1}\{t > T_0\}$. It represents that unit 1 is treated at time T_0, where other units never receive the treatment and serve as untreated control units (sometimes also referred to as "donors" in the SCM literature). We assume w.l.o.g. that unit $i = 1$ is the one being treated.
- **Outcomes:** We adopt the potential outcomes framework [42]. Let $Y_{i\tilde{t}}(a_i) \in \mathbb{R}$ denote the potential outcome at post-treatment time $\tilde{t} > T_0$ and treatment assignment $A_{i\tilde{t}} = a_{i\tilde{t}}$. We have observed outcomes follow $Y_{i\tilde{t}} = A_{i\tilde{t}}Y_{i\tilde{t}}(1) + (1 - A_{i\tilde{t}})Y_{i\tilde{t}}(0)$. For the treated unit, we write $Y_{1\tilde{t}}(1) = Y_{1\tilde{t}}(0) + \tau_{\tilde{t}}$, where $\tau_{\tilde{t}}$ is the treatment effect at time $\tilde{t} > T_0$.
- **Covariates:** To ease notation, for each unit i, let $X_i = \left(Z_i, \{Y_{it}\}_{t=1}^{T_0}\right)$ denote concatenated vector of time-invariant baseline covariates Z_i and pre-treatment outcome history $\{Y_{it}\}_{t=1}^{T_0}$, where $Y_{it} = A_{it}Y_{it}(1) + (1 - A_{it})Y_{it}(0)$, for $t = 1, \ldots, T_0$.

[4] A related line of work studies prediction-based approaches in settings similar to synthetic control, often referred to as "synthetic twins" [e.g., 39, 9, 7, 43]. However, these methods focus on forecasting the treated unit's trajectory but typically are *model-specific*.

Nuisance components: We write the *synthetic-control weights* as $\mathbf{w} = (w_2, \ldots, w_N) \in \Delta_{N-1}$, where $w_j \in [0,1]$ quantifies the contribution of unit j to the synthetic control, with constraint $\sum_{j=2}^{N} w_j = 1$. We further define the *outcome-regression function* $m_{\tilde{t}}(x) = \mathbb{E}[Y_{\tilde{t}} \mid X = x, A = 0]$. In our method, we treat both $\mathbf{w}$ and $m_{\tilde{t}}(x)$ as *nuisance components*, which are quantities that must be estimated from data but are not themselves the target of inference.

Estimation task: Our estimand of interest is the average treatment effect on treated at the post-treatment time $\tilde{t} > T_0$, i.e., $\tau_{\tilde{t}} = \mathbb{E}[Y_{\tilde{t}}(1) - Y_{\tilde{t}}(0) \mid X = X_1, A = 1]$.

In the single-treated-unit setting, we observe the treated realization $Y_{1\tilde{t}}$, while the main estimation challenge is to estimate the unobserved counterfactual $\mathbb{E}[Y_{1\tilde{t}}(0) \mid X = X_1, A = 1]$.

Identifiability: We follow the standard SCM assumptions [2, 8, 45] to ensure identifiability of $\tau_{\tilde{t}}$:

Assumption 3.1. For all $x \in \mathcal{X}$, $a \in \mathcal{A}$, it holds:

(i) *No interference:* For all units $i = 1, \ldots, N$ and times $t \in \mathcal{T}$, the potential outcome of unit i depends only on its own treatment status, $Y_{it}(a_{1t}, \ldots, a_{Nt}) = Y_{it}(a_{it})$.

(ii) *No anticipation:* For all control units $j \in \{2, \ldots, N\}$ and all times $t \in \mathcal{T} = \{1, \ldots, T\}$, we have $Y_{jt} = Y_{jt}(0)$. For the treated unit $i = 1$, there are no pre-treatment effects and consistency, so that $Y_{1t} = Y_{1t}(0)$ for $t \le T_0$ and $Y_{1\tilde{t}} = Y_{1\tilde{t}}(1)$ for $\tilde{t} > T_0$.

(iii) *Conditional mean exchangeability:* For post-treatment time $\tilde{t} \ge T_0$, we have $\mathbb{E}[Y_{\tilde{t}}(0) \mid X = x, A = 1] = \mathbb{E}[Y_{\tilde{t}}(0) \mid X = x, A = 0]$.

Assumption 3.1 is standard in synthetic control literature [2, 8, 45]: (i) imposes no interference, thus ensuring that the observed outcomes coincide with the relevant potential outcomes; (ii) rules out anticipation effects and requires control units to never be treated, so that control outcomes satisfy $Y_{jt} = Y_{jt}(0)$ for all t, and the treated unit satisfies $Y_{1t} = Y_{1t}(0)$ for $t \le T_0$ and $Y_{1\tilde{t}} = Y_{1\tilde{t}}(1)$ for $\tilde{t} > T_0$. Finally, (iii) assumes an invariant untreated outcome mechanism conditional on covariates X, which allows the conditional mean function $m_{\tilde{t}}(x) = \mathbb{E}[Y_{\tilde{t}} \mid X = x, A = 0]$ to be learned from control units and transported to the treated unit.

Under Assumptions 3.1, our primary estimation target for $\tilde{t} > T_0$ is the untreated conditional mean of the treated unit,

$$\psi_{\tilde{t}} := m_{\tilde{t}}(X_1). \tag{1}$$

Note that we observe the post-treatment outcome of the treated unit $Y_{1\tilde{t}}$, so the statistical estimand to be estimated is $\psi_{\tilde{t}} := m_{\tilde{t}}(X_1)$, where $m_{\tilde{t}}(\cdot)$ only depends on the population of control units. The resulting treatment effect estimate is $Y_{1\tilde{t}} - \hat{\psi}\tilde{t}$, which need not equal the realized individual treatment effect. Thus, $m_{\tilde{t}}(\cdot)$ can be estimated from control outcomes and subsequently evaluated at X_1.

3.2 Background on existing methods

Classical SCM: The classical SCM constructs the counterfactual trajectory of the treated unit as an explicit convex combination of untreated control trajectories [4, 3]. The synthetic-control weights are chosen to balance pre-treatment histories and, optionally, additional covariates. Specifically, the classical SCM estimates weights $\mathbf{w} \in \Delta^{N-1}$ in the simplex

$$\Delta^{N-1} := \{\mathbf{w} \in \mathbb{R}^{N-1} : w_j \ge 0,\ \sum_{j=2}^{N} w_j = 1\} \tag{2}$$

by solving a constrained matching problem

$$\hat{\mathbf{w}}^{\text{SC}} \in \arg\min_{\mathbf{w} \in \Delta^{N-1}} \left\| X_1 - \sum_{j=2}^{N} w_j X_j \right\|_V^2, \tag{3}$$

where $V \succeq 0$ is a user-chosen importance matrix that prioritizes particular control units. Given the estimated weights, the synthetic control estimator (of the counterfactual) is

$$\hat{\psi}_{\tilde{t}}^{\text{SC}} := \sum_{j=2}^{N} \hat{w}_j^{\text{SC}}\, Y_{j\tilde{t}}, \quad \tilde{t} > T_0, \tag{4}$$

and the corresponding treatment effect estimator is $\hat{\tau}_{\tilde{t}}^{\text{SC}} := Y_{1\tilde{t}} - \hat{\psi}_{\tilde{t}}^{\text{SC}}$. In practice, the treated unit's pre-treatment trajectory cannot be perfectly approximated by the synthetic-control weights, meaning that estimation error can propagate to the post-treatment period and bias the counterfactual estimate.

Plug-in estimator: We next introduce the so-called plug-in estimator. In the single-treated-unit setting, a direct plug-in estimate of $\psi_{\tilde{t}} = m_{\tilde{t}}(X_1)$ reduces to predicting the treated unit's untreated outcome:

$$\hat{\psi}_{\tilde{t}}^{\text{plug-in}} = \hat{m}_{\tilde{t}}(X_1), \tag{5}$$

where $m_{\tilde{t}}(\cdot)$ is the control-side outcome-regression at time $\tilde{t} > T_0$. A practical advantage of Eq. 5 is its flexibility: $\hat{m}_{\tilde{t}}(\cdot)$ can be learned from the control unit data using a wide range of predictive models with state-of-the-art ML models, such as representation learning [39], recurrent neural network [9], dynamical systems [7], or controlled differential equation [43]. However, because Eq. 5 relies on extrapolating the learned regression to the treated unit, any estimation error in $\hat{m}_{\tilde{t}}(\cdot)$ translates into bias in $\hat{\psi}_{\tilde{t}}^{\text{plug-in}}$, which can be non-negligible in finite samples (see discussion in [25]).

Augmented SCM: Unlike classical SCM, augmented SCM (ASCM) [8] introduces an outcome-regression adjustment to correct for imperfect pre-treatment fit. Although the original SCM weights remain constrained to the simplex, this adjustment allows the counterfactual estimate to fall outside the convex hull of the observed control outcomes. Specifically, augmented SCM incorporates an outcome-regression model, $\hat{m}_{\tilde{t}}(x)$, and constructs the counterfactual estimate by combining the treated unit's plug-in prediction with a weighted residual correction from the control units:

$$\hat{\psi}_{\tilde{t}}^{\text{ASC}} = \hat{m}_{\tilde{t}}(X_1) + \sum_{j=2}^{N} \hat{w}_j \{Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)\}. \tag{6}$$

Here, $\hat{w}_j$ denotes the original SCM weights, which remain constrained to the simplex. The residual correction allows the resulting ASCM estimate to fall outside the convex hull of the observed control outcomes.

4 Targeted synthetic control method

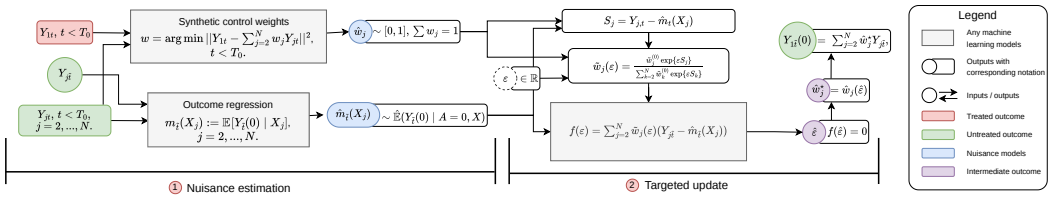


Figure 2: **Overview of TSC**. Our method consists of two stages: ① a *nuisance component estimation* and ② a *targeted update* for debiasing. In the nuisance stage, we (i) compute initial synthetic-control weights $\hat{w}$ by matching pre-treatment histories and (ii) fit an outcome-regression $\hat{m}_{\tilde{t}}(x)$ on control units using any machine-learning backbone. In the debiasing stage, we construct residual scores S_j of control units and perform a one-dimensional exponential-tilting update $\tilde{w}(\varepsilon)$. The parameter $\hat{\varepsilon}$ is obtained by solving the condition $f(\hat{\varepsilon}) = 0$. The resulting targeted weights $\hat{w}^\star = \tilde{w}(\hat{\varepsilon})$ yield the final counterfactual $\hat{\psi}_{\tilde{t}} = \sum_{j=2}^{N} \hat{w}_j^\star Y_{j\tilde{t}}$ (and thus the treatment effect) while preserving interpretable synthetic-control weights.

Inspired by TMLE from classical causal inference, we now develop a two-stage estimator for the single-treated unit setting equipped with a customized TMLE-style targeting step for debiasing. Our TSC (i) debiases the initial synthetic-control weights by exponentially tilting the control-unit weights so that the weighted residuals from the outcome-regression average to zero, while keeping the weights on the simplex and therefore interpretable, and (ii) keeps the estimation of TSC in the convex hull of the outcomes of control units.

4.1 Overview of TSC

Our TSC proceeds in two stages (see Figure 2):

① *Nuisance component estimation.* We estimate the nuisance components with a flexible ML model. This includes (i) a component for the synthetic-control weights that reweights control units to approximate the treated unit's covariate distribution (e.g., via pre-treatment history matching) and (ii) the outcome-regression $m_{\tilde{t}}(x)$ with cross-fitting.

② *Targeted update.* We start with the initial synthetic-control weights via a one-dimensional, TMLE-style targeted update. Specifically, we update the weights along an exponential-tilting submodel, which provides a controlled adjustment of the first-stage weights while preserving the simplex constraint (non-negativity and sum-to-one). As a result, the updated counterfactual remains a convex combination of observed control outcomes and is therefore bounded within the convex hull of control outcomes.

The above procedure yields a *targeted* estimator of $\psi_{\tilde{t}} = \mathbb{E}[Y_{\tilde{t}} \mid X = X_1, A = 1]$. Further, it supports interpretable weights and boundedness of the estimation. The full procedure is summarized in Algorithm 1. Below, we first introduce the intuition behind debiasing (§4.2), which we finally turn into a targeted update using a weight-tilting submodel (§4.3).

4.2 Intuition behind TSC

We start from the classical synthetic control approach. Specifically, we construct *synthetic-control weights* $w \in \Delta^{N-1}$ by matching the controls' pre-treatment history to that of the treated unit (e.g., pre-treatment outcomes or other chosen features), yielding $\hat{\psi}_{\tilde{t}}^{\text{SC}} = \sum_{j=2}^{N} \hat{w}_j \, Y_{j\tilde{t}}$. When accurate pre-treatment balance is infeasible, however, the resulting weights can be poorly estimated in finite samples, which makes this weights-only approximation sensitive and motivates the subsequent targeted update.

To motivate our subsequent debiasing step, it is useful to express the corresponding estimator in a debiasing form by adding and subtracting an outcome-regression $\hat{m}_{\tilde{t}}(\cdot)$:

$$\hat{\psi}_{\tilde{t}}^{\text{SC}} = \underbrace{\sum_{j=2}^{N} \hat{w}_j \, \hat{m}_{\tilde{t}}(X_j)}_{\text{weighted average}} + \underbrace{\sum_{j=2}^{N} \hat{w}_j \big\{ Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j) \big\}}_{\text{residual correction}}. \tag{7}$$

Here, the first term is the classical-SCM-style weighted average of outcome-regression predictions of the control units, while the second term is the residual term over the control units, which we later calibrate to zero via updating the weights.

Why we need $\hat{m}_{\tilde{t}}(X)$: A remaining limitation of Eq. 4 is that it predicts the treated outcome at $\tilde{t} > T_0$ solely as a weighted average of observed control outcomes. Although SCM uses pre-treatment information to estimate the weights, it does not explicitly exploit the predictive relationship between pre-treatment characteristics and post-treatment outcomes across control units. As a result, the estimator is sensitive to finite-sample error in the estimated weights. To reduce this sensitivity, we fit an outcome-regression $\hat{m}_{\tilde{t}}(\cdot)$ on the control units to predict $Y_{j\tilde{t}}$ from X_j. This regression captures the systematic component of $Y_{j\tilde{t}}$ associated with X_j, so the remaining residuals $Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)$ are closer to mean-zero noise and can be more stably aggregated by the weighted correction term.

4.3 Derivation of the targeted update

Inspired by TMLE [53, 52], we now introduce a targeting update. The key idea of targeting is to take an initial estimate (here an initial set of synthetic-control weights) and then perform a one-dimensional update guided by Eq. 7 over a submodel, so that the resulting estimator satisfies the residual correction equation with interpretable synthetic-control weights. Below, we first specify (i) how we obtain the initial weight estimates, (ii) the submodel to ensure the interpretability of the synthetic-control weights, and (iii) the targeted update.

(i) Initial synthetic-control weight estimates. The starting point is a standard synthetic control estimator for the treated unit at the post-treatment time T:

$$\hat{\psi}_{\tilde{t}} \;=\; \sum_{j=2}^{N} \hat{w}_j^0 \, Y_{j\tilde{t}}, \qquad \tilde{t} > T_0, \tag{8}$$

Algorithm 1 Targeted Synthetic Control Method

Require: Control units $\{(X_j, Y_{j\tilde{t}})\}_{j=2}^{N}$, treated unit X_1, iterations R, step size $\eta > 0$

/* Stage 1: Nuisance estimation */
$\hat{w}^0 \leftarrow \arg\min_{w \in \Delta^{N-1}} \|X_1 - \sum_{j=2}^{N} w_j X_j\|_V^2$
$\hat{m}_{\tilde{t}}(x) \leftarrow \mathbb{E}[Y_{\tilde{t}} \mid X = x, A = 0]$
/* Stage 2: Targeted update */
$S_j \leftarrow Y_{j,t} - \hat{m}_t(X_j)$
Initialize $\varepsilon \leftarrow 0$
for $r = 1$ **to** R **do**
  $\hat{w}_j(\varepsilon) \leftarrow \dfrac{\hat{w}_j^0 \exp(\varepsilon S_j)}{\sum_{k=2}^{N} \hat{w}_k^0 \exp(\varepsilon S_k)}$
  $g(\varepsilon) \leftarrow \sum_{j=2}^{N} \hat{w}_j(\varepsilon) \big(Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)\big)$
  $\varepsilon \leftarrow \varepsilon - \eta \, g(\varepsilon)$
end for
$\hat{w}^\star \leftarrow \hat{w}(\hat{\varepsilon})$
$\hat{\psi}_{\tilde{t}}^{\text{TSC}} \leftarrow \sum_{j=2}^{N} \hat{w}_j^\star Y_{j\tilde{t}}$
return $\hat{w}^\star$, $\hat{\psi}_{\tilde{t}}^{\text{TSC}}$

where $\hat{w}^0 = (\hat{w}_2^0, \ldots, \hat{w}_N^0)$ denotes an initial set of synthetic-control weights learned by Eq. 2 (e.g., obtained by pre-treatment matching under simplex constraints).

(ii) Weight-tilting submodel. We define a one-dimensional exponential tilting (softmax) submodel around $\hat{w}^0$, which provides a smooth, one-dimensional way to update the initial weights $\hat{w}^0$ while preserving the simplex constraints (non-negativity and unit sum). We thus define the, for $j = 2, \ldots, N$,

$$\hat{w}_j(\varepsilon) \;=\; \frac{\hat{w}_j^0 \exp(\varepsilon S_j)}{\sum_{k=2}^N \hat{w}_k^0 \exp(\varepsilon S_k)}, \tag{9}$$

guaranteeing $\hat{w}_j(\varepsilon) \geq 0$ and $\sum_{j=2}^N \hat{w}_j(\varepsilon) = 1$ for all $\varepsilon \in \mathbb{R}$. Here, S_j represents *relative* model-implied fit: control units with $\hat{m}_{\tilde{t}}(X_j)$ above (below) the initial weighted average tend to receive more (less) weight, making the update focus on correcting mismatch in model-predicted outcomes. In practice, we use a centered choice based on an outcome-regression model, e.g., $S_j = Y_{j,t} - \hat{m}_t(X_j)$, so that the tilt direction is mean-zero under the initial weights, which prevents a uniform drift in the weights and makes the update depend only on relative differences across control units. This centering improves numerical stability by keeping ε on a reasonable scale and reducing sensitivity to the overall level of $\hat{m}_{\tilde{t}}(x)$.

(iii) TMLE-style targeted update. We now describe how the targeted update is implemented in practice by solving the residual-correction equation. Given the tilted weights $\hat{w}(\varepsilon)$, our targeting goal is to update ε so that the weighted control residuals,

$$f(\varepsilon) \;=\; \sum_{j=2}^N \hat{w}_j(\varepsilon)\Big(Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)\Big),\; \hat{\varepsilon} = \arg\min_{\varepsilon} \log\big[\sum_{j=2}^N \hat{w}_j^0 \exp(\varepsilon(Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j))\big], \tag{10}$$

are calibrated to zero, i.e., $\hat{\varepsilon} : f(\hat{\varepsilon}) = 0$. Since ε is one-dimensional, we can compute $\hat{\varepsilon}$ efficiently. In practice, we obtain $\hat{\varepsilon}$ by solving the equivalent convex optimization problem in Eq. 10, for which the first-order derivative with respect to ε is the same as Eq. 10. In other words, $\hat{\varepsilon}$ can be viewed as the maximum likelihood estimate of the fluctuation parameter in the weight-tilting submodel for the control weight distribution.

Targeted estimator. We now write down our final estimator. Let $\hat{\varepsilon}$ denote the root of the estimating equation $f(\varepsilon) = 0$. We denote the *targeted* synthetic-control weights by $\hat{w}^\star = \hat{w}(\hat{\varepsilon})$. The final TSC estimator is then given by

$$\hat{\psi}_{\tilde{t}}^{\text{TSC}} = \sum_{j=2}^N \hat{w}_j^\star \, Y_{j\tilde{t}}. \tag{11}$$

This estimator thus performs *targeting* by updating the initial synthetic-control weights, so that the residual-correction equation is satisfied while ensuring that the estimator is a weighted combination of observed control outcomes.

Intuitively, the above construction has several advantages. (i) It preserves the simplex constraint on the weights and therefore maintains the interpretation of synthetic-control weights as relative contributions to the counterfactual outcome. (ii) It allows us to use flexible ML methods to estimate the nuisance components, including the outcome-regression. (iii) It enforces the vanishing of the first-order estimation error of the outcome-regression model (see next section).

4.4 Properties of TSC

Below, we state two favorable properties of our TSC method, namely, robustness of misspecification of the outcome-regression model and bounded outcomes.

Robustness to misspecification of outcome-regression model and imperfect fit of weight estimation: Our approach follows a TMLE-style targeting principle, but targets the synthetic-control weights rather than the outcome regression. Specifically, we update the initial weights along an exponential-tilting submodel and select $\hat{w}^\star$ to (approximately) satisfy a residual-balancing condition on the controls, i.e., $\sum_{j=2}^N \hat{w}_j^\star\{Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)\} = \sum_{j=2}^N \hat{w}_j^\star Y_{j\tilde{t}} - \sum_{j=2}^N \hat{w}_j^\star \hat{m}_{\tilde{t}}(X_j) \approx 0$. By calibrating the weights using control-unit residuals, this targeted update mitigates bias from imperfect pre-treatment weight estimation, and the final estimator is robust to the misspecification of $\hat{m}_{\tilde{t}}(\cdot)$ by enforcing Eq. 10 to zero.

Theorem 4.1 (Boundedness)**.** *Assume $Y_{j\tilde{t}} \in [a, b]$, where $a, b \in \mathbb{R}$, $a < b$, for all control units $j = 2, \ldots, N$ and the targeted weights satisfy $\hat{w}_j^\star \geq 0$, $\sum_{j=2}^N \hat{w}_j^\star = 1$. Then our TSC is bounded, $a \leq \hat{\psi}_{\tilde{t}}^{\text{TSC}} \leq b$.*

Proof. See Appendix D.

Remark. In classical TMLE, the targeting step typically is a one-dimensional update of the outcome-regression. For binary outcomes, this is commonly implemented on the logit-scale to ensure the updated predictions remain bounded. In the single-treated-unit setting, we do not directly tailor the TMLE idea to the new setting, but tilt the synthetic-control weights, so the final counterfactual is always a convex combination of observed control outcomes. Consequently, boundedness is inherent, even for binary outcomes, without requiring a logit link or any additional transformation for the outcome-regression model.

Asymptotic properties: Establishing asymptotic guarantees in the single-treated-unit setting remains challenging, as standard semiparametric arguments do not directly apply. In Appendix E, we consider a population-level extension under a joint large-N_0, large-T_0 regime and establish asymptotic normality for a corresponding orthogonal estimator under stronger assumptions on stability, transportability, and overlap, together with suitable regularity conditions. Importantly, this result concerns the population-level estimator rather than the original TSC estimator, whose asymptotic properties require further investigation.

4.5 Differences to augmented SCM

In this section, we compare TSC to the augmented SCM to clarify the limitations of the latter. The augmented SCM estimator follows a "prediction + residual correction" decomposition:

$$\hat{\psi}_{\tilde{t}}^{\mathrm{ASC}} = \underbrace{\hat{m}_{\tilde{t}}(X_1)}_{\text{plugin term}} + \underbrace{\sum_{j=2}^{N} \hat{w}_j \left\{ Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j) \right\}}_{\text{residual correction term}}. \tag{12}$$

The augmented SCM in Eq. (6) can be interpreted as a plug-in prediction $\hat{m}_{\tilde{t}}(X_1)$ plus a regression correction. The final outcome does *not* have the boundedness guarantee, since, even if each term individually lies in $[a, b]$, the outcome model prediction in Eq. (6) may *not* be in $[a, b]$.

Why our targeted update is beneficial over augmented SCM: A simple additive residual correction can work well in practice, but it debiases by adding an augmentation term to the plug-in estimator, rather than by updating the synthetic-control weights. Hence, the targeted update has two important advantages: **(1)** it preserves the weight-based structure of synthetic control estimators, so the contributions of control units are interpretable through weights, and **(2)** it enforces the final counterfactual estimate remains a convex combination of observed synthetic control estimators, so outcomes remain within their natural bounds.

5 Experiments

We follow best practice in causal ML (e.g., van der Laan and Rubin [53], Ben-Michael et al. [8], Shi et al. [45]) to perform experiments that demonstrate the effectiveness of TSC across different types of outcomes using synthetic data. Synthetic data has the advantage of providing access to the ground-truth treatment effect and thus allows for direct comparison against oracle estimates. We instantiate all models with the same neural network architectures and hyperparameters. Implementation details and additional experimental results are provided in Appendix F.2 and Appendix G, respectively.

5.1 Synthetic data

Prediction horizon	Methods	Linear	Hinge	Quadratic	Time-varying
1	Classical SCM	0.922 ± 0.146	0.988 ± 0.163	1.030 ± 0.186	0.888 ± 0.126
	Plug-in estimator	1.053 ± 0.245	1.118 ± 0.292	1.434 ± 0.401	0.987 ± 0.212
	Augmented SCM	1.111 ± 0.245	1.186 ± 0.292	1.458 ± 0.401	1.049 ± 0.212
	TSC (*ours*)	$\mathbf{0.892 \pm 0.134}$	$\mathbf{0.924 \pm 0.139}$	$\mathbf{1.022 \pm 0.188}$	$\mathbf{0.879 \pm 0.126}$
5	Classical SCM	0.975 ± 0.233	0.964 ± 0.230	1.044 ± 0.226	0.935 ± 0.236
	Plug-in estimator	0.993 ± 0.346	1.001 ± 0.364	1.152 ± 0.422	1.012 ± 0.365
	Augmented SCM	1.039 ± 0.348	1.051 ± 0.367	1.211 ± 0.425	1.046 ± 0.376
	TSC (*ours*)	$\mathbf{0.970 \pm 0.231}$	$\mathbf{0.937 \pm 0.235}$	$\mathbf{1.035 \pm 0.230}$	$\mathbf{0.934 \pm 0.233}$
10	Classical SCM	0.810 ± 0.203	0.771 ± 0.210	0.866 ± 0.228	0.871 ± 0.231
	Plug-in estimator	1.107 ± 0.185	1.113 ± 0.180	1.032 ± 0.156	1.127 ± 0.191
	Augmented SCM	1.113 ± 0.191	1.115 ± 0.182	1.080 ± 0.164	1.141 ± 0.197
	TSC (*ours*)	$\mathbf{0.778 \pm 0.175}$	$\mathbf{0.727 \pm 0.178}$	$\mathbf{0.806 \pm 0.196}$	$\mathbf{0.849 \pm 0.207}$

(a) Continuous outcomes

Prediction horizon	Methods	Linear	Hinge	Quadratic	Time-varying
1	Classical SCM	0.706 ± 0.119	0.840 ± 0.063	0.618 ± 0.106	0.703 ± 0.082
	Plug-in estimator	0.776 ± 0.153	1.158 ± 0.392	1.075 ± 0.309	1.638 ± 0.453
	Augmented SCM	0.850 ± 0.148	1.194 ± 0.349	1.112 ± 0.316	1.650 ± 0.429
	TSC (*ours*)	$\mathbf{0.423 \pm 0.039}$	$\mathbf{0.524 \pm 0.074}$	$\mathbf{0.386 \pm 0.067}$	$\mathbf{0.486 \pm 0.079}$
5	Classical SCM	0.588 ± 0.073	0.653 ± 0.044	0.606 ± 0.071	0.575 ± 0.090
	Plug-in estimator	0.592 ± 0.145	1.103 ± 0.271	0.725 ± 0.132	0.874 ± 0.270
	Augmented SCM	0.506 ± 0.183	1.096 ± 0.281	0.738 ± 0.160	0.925 ± 0.317
	TSC (*ours*)	$\mathbf{0.316 \pm 0.041}$	$\mathbf{0.478 \pm 0.081}$	$\mathbf{0.451 \pm 0.061}$	$\mathbf{0.312 \pm 0.063}$
10	Classical SCM	0.593 ± 0.090	0.452 ± 0.086	0.694 ± 0.103	0.462 ± 0.079
	Plug-in estimator	0.558 ± 0.096	0.920 ± 0.165	0.766 ± 0.109	0.767 ± 0.180
	Augmented SCM	0.678 ± 0.144	0.902 ± 0.164	0.815 ± 0.094	0.781 ± 0.185
	TSC (*ours*)	$\mathbf{0.420 \pm 0.088}$	$\mathbf{0.448 \pm 0.051}$	$\mathbf{0.562 \pm 0.073}$	$\mathbf{0.408 \pm 0.062}$

(b) Binary outcomes

Table 2: **Results for experiments with synthetic data.** Mean $\pm$ standard error across five runs, reported for prediction horizons of 1, 5, 10 steps ahead. $\Rightarrow$ *Our TSC performs best across all settings.*

Dataset: We simulate samples from multiple data-generating functions, including (1) *linear*, (2) *hinge*, (3) *quadratic*, and (4) *time-varying* outcome functions. Details are in Appendix F.1. For each function, we compare the root mean squared error and report the mean $\pm$ standard error across five runs for prediction horizons of 1, 5, and 10 steps ahead. Altogether, we generate over 50 different datasets under varying scenarios. **Results:** We compare our TSC against the **classical SCM** [4], a **plug-in** estimator (Eq. 5), and the **augmented SCM** [8]. The results are in Table 2 (a) (for continuous outcomes) and Table 2 (b) (for binary outcomes). *Across all datasets, TSC consistently achieves the best performance.* The improvement for binary outcomes (Table 2 (b)) is especially noteworthy. Here, our TSC reduces the RMSE relative to the augmented SCM by $\sim 18\%$. We attribute this improvement to the targeted update, which enforces bounded predictions and avoids unstable extrapolation.

Interpretability: We further demonstrate the interpretability of the weights produced by TSC. Figure 3 compares the initial synthetic control (light colors) with the final weights (dark colors). *Importantly, we confirm that the targeted weights remain nonnegative and sum to one, allowing them to be directly interpreted as relative contributions of control units to the constructed counterfactual.*

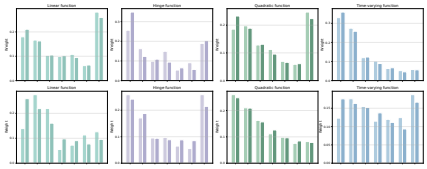
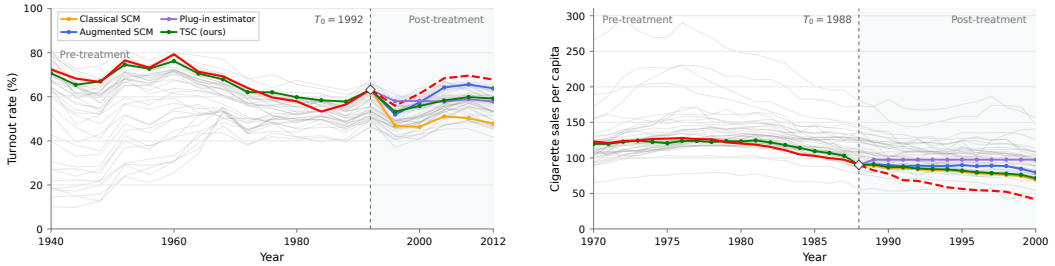


Figure 3: **Insights of weights.** Bars show the weights of control units used to construct the counterfactual (*top row*: continuous; *bottom row*: binary). Shown are the initial synthetic control weights (light colors) and the final weights (dark colors).

Boundedness: Table 3 empirically validates the boundedness property of our method (Theorem 4.1) and thus highlights a key advantage of TSC over the augmented SCM: the targeted update guarantees that counterfactual predictions remain bounded. This boundedness ensures that the original interpretation of synthetic-control weights as defined in the classical SCM. In contrast, the augmented SCM produces many out-of-bounds estimates.

5.2 Real-world data

Data: We conduct case studies using two real-world datasets. **(a) Turnout data:** We use the Turnout rate dataset [48], which records U.S. state-level elections and voter turnout rates from 1920 to 2000. As the targeted unit, we focus on New Hampshire, which adopted same-day voter registration in 1996, and use other states as controls. **(b) California tobacco data:** We use the California smoking dataset from Abadie et al. [4], which records annual per-capita cigarette sales across U.S. states from 1970 to 2000. The targeted unit is California, which enacted Proposition 99 (i.e., a larger health reform, which also included a tax increase on tobacco) in 1988, and we use 38 other states as controls.



(a) Turnout data (b) California tobacco data

Figure 4: **Real-world experiments.** The trajectories of the control units are shown in gray, and the treated unit in red. We plot counterfactual estimates from augmented SCM (blue), classical SCM (orange), the plug-in estimator (purple), and TSC (green). The vertical dashed line indicates the treatment time T_0. TSC matches the pre-treatment history closely in both datasets.

Results: In Figure 4, we plot the predicted counterfactual trajectory of the treated unit under no treatment for both datasets. **(a) Turnout data:** Same-day registration is expected to increase voter turnout, reflected in the observed increase in the treated unit after treatment (shown in red). Our TSC (green) yields a stable counterfactual trajectory, implying a large positive treatment effect. In contrast, the classical SCM exhibits a systematic downward deviation, while the augmented SCM shows an upward deviation,

Bound	Method	Linear	Hinge	Quadratic	Time-varying
Upper bound	Augmented SCM	3.33%	13.33%	20.00%	26.67%
	TSC *(ours)*	0.00%	0.00%	0.00%	0.00%
Lower bound	Augmented SCM	13.33%	40.00%	13.33%	13.33%
	TSC *(ours)*	0.00%	0.00%	0.00%	0.00%

* Violation rates above zero are shown in red.

Table 3: Percentage of counterfactual estimations outside $[0, 1]$ on binary outcomes. $\Rightarrow$ *TSC achieves 0.00% violation rate; augmented SCM does not.*

indicating bias in the constructed controls. **(b) California tobacco data:** Proposition 99 is expected to reduce cigarette consumption. Our TSC (green) produces a stable counterfactual that remains within the range of control outcomes, suggesting a substantial negative treatment effect after 1988 in line with the finding from the classical SCM in Abadie et al. [4]. The plug-in and augmented SCM both show less plausible results in the post-treatment period, whereas TSC tracks the pre-treatment trajectory closely and yields bounded predictions throughout. We provide further discussion of the theoretical properties and limitations of TSC in Appendix H.

Acknowledgement

Our research is supported by the DAAD programme Konrad Zuse Schools of Excellence in Artificial Intelligence, sponsored by the Federal Ministry of Research, Technology and Space. This work has also been supported by the German Federal Ministry of Education and Research (Grant: 01IS24082).

References

[1] A. Abadie. Using synthetic controls: Feasibility, data requirements, and methodological aspects. *Journal of economic literature*, 59(2):391–425, 2021.

[2] A. Abadie and J. Gardeazabal. The economic costs of conflict: A case study of the Basque Country. *American Economic Review*, 93(1):113–132, 2003.

[3] A. Abadie and J. L'Hour. A penalized synthetic control estimator for disaggregated data. *Journal of the American Statistical Association*, 116(536):1817–1834, 2021.

[4] A. Abadie, A. Diamond, and J. Hainmueller. Synthetic control methods for comparative case studies: Estimating the effect of California's tobacco control program. *Journal of the American Statistical Association*, 105(490):493–505, 2010.

[5] D. Arkhangelsky and G. Imbens. Causal models for longitudinal and panel data: A survey. *The Econometrics Journal*, 27(3):C1–C61, 2024.

[6] D. Arkhangelsky, S. Athey, D. A. Hirshberg, G. W. Imbens, and S. Wager. Synthetic difference in differences. *arXiv preprint*, arXiv:1812.09970, 2018.

[7] A. Bellot and M. van der Schaar. Policy analysis using synthetic controls in continuous-time. In *ICML*, 2021.

[8] E. Ben-Michael, A. Feller, and J. Rothstein. The augmented synthetic control method. *Journal of the American Statistical Association*, 116(536):1789–1803, 2021.

[9] I. Bica, A. M. Alaa, J. Jordon, and M. van der Schaar. Estimating counterfactual treatment outcomes over time through adversarially balanced representations. In *ICLR*, 2019.

[10] D. Card and A. Krueger. Minimum wages and employment: A case study of the fast food industry in New Jersey and Pennsylvania. Technical Report NBER Working Paper 4509, National Bureau of Economic Research, Cambridge, MA, 1993.

[11] J. Che, X. Meng, and L. Miratrix. Caliper synthetic matching: Generalized radius matching with local synthetic controls. *arXiv preprint*, arXiv:2411.05246, 2024.

[12] V. Chernozhukov, K. Wüthrich, and Y. Zhu. An exact and robust conformal inference method for counterfactual and synthetic controls. *arXiv preprint*, arXiv:1712.09089, 2017.

[13] V. Chernozhukov, D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. Newey, and J. Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 2018.

[14] V. Chernozhukov, K. Wüthrich, and Y. Zhu. Debiasing and t-tests for synthetic control inference on average causal effects. *arXiv preprint*, arXiv:1812.10820, 2018.

[15] A. Curth, H. Poon, A. V. Nori, and J. González. Cautionary tales on synthetic controls in survival analyses. In *CLeaR*, 2024.

[16] D. J. Foster and V. Syrgkanis. Orthogonal statistical learning. *The Annals of Statistics*, 51(3): 879–908, 2023.

[17] D. Frauen, M. Schröder, K. Hess, and S. Feuerriegel. Orthogonal survival learners for estimating heterogeneous treatment effects from time-to-event data. In *NeurIPS*, 2025.

[18] J. Fry. A method of moments approach to asymptotically unbiased synthetic controls. *arXiv preprint*, arXiv:2312.01209, 2024.

[19] J. Fry. Robust inference when nuisance parameters may be partially identified with applications to synthetic controls. *arXiv preprint*, arXiv:2507.00307, 2025.

[20] K. Hess and S. Feuerriegel. Stabilized neural prediction of potential outcomes in continuous time. In *ICLR*, 2025.

[21] K. Hess, V. Melnychuk, D. Frauen, and S. Feuerriegel. Bayesian neural controlled differential equations for treatment effect estimation. In *ICLR*, 2024.

[22] K. Hess, D. Frauen, M. van der Schaar, and S. Feuerriegel. Overlap-weighted orthogonal meta-learner for treatment effect estimation over time. In *ICLR*, 2026.

[23] M. Kellogg, M. Mogstad, G. A. Pouliot, and A. Torgovitsky. Combining matching and synthetic control to tradeoff biases from extrapolation and interpolation. *Journal of the American Statistical Association*, 116(536):1804–1816, 2021.

[24] E. H. Kennedy. Towards optimal doubly robust estimation of heterogeneous causal effects. *arXiv preprint*, arXiv:2004.14497, 2020.

[25] E. H. Kennedy. Semiparametric doubly robust targeted double machine learning: a review. *arXiv preprint*, arXiv:2203.06469, 2022.

[26] S. R. Künzel, J. S. Sekhon, P. J. Bickel, and B. Yu. Meta-learners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 116(10):4156–4165, 2019.

[27] H. Lan, H. Chang, E. Dillon, and V. Syrgkanis. A meta-learner for heterogeneous effects in difference-in-differences. In *ICML*, 2025.

[28] R. Li, S. Hu, M. Lu, Y. Utsumi, P. Chakraborty, D. M. Sow, P. Madan, J. Li, M. Ghalwash, Z. Shahn, and L.-w. 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*, pages 282–299. PMLR, 2021.

[29] B. Lim. Forecasting treatment responses over time using recurrent marginal structural networks. In *NeurIPS*, 2018.

[30] J. Liu, E. T. Tchetgen, and C. Varjão. Proximal causal inference for synthetic control with surrogates. In *AISTATS*, 2024.

[31] H. Ma, D. Frauen, and S. Feuerriegel. DeepBlip: Estimating conditional average treatment effects over time. In *ICML*, 2026.

[32] J. Märkle-Huß, S. Feuerriegel, and D. Neumann. Contract durations in the electricity market: Causal impact of 15 min trading on the EPEX SPOT market. *Energy Economics*, 69:367–378, 2018.

[33] V. Melnychuk, D. Frauen, and S. Feuerriegel. Causal transformer for estimating counterfactual outcomes. In *ICML*, 2022.

[34] P. Morzywolek, J. Decruyenaere, and S. Vansteelandt. On weighted orthogonal learners for heterogeneous treatment effects. *arXiv preprint*, arXiv:2303.12687, 2023.

[35] A. O. R. Nazaret, C. Shi, and D. Blei. On the misspecification of linear assumptions in synthetic controls. In *AISTATS*, 2024.

[36] X. Nie and S. Wager. Quasi-oracle estimation of heterogeneous treatment effects. *arXiv preprint*, arXiv:1712.04912, 2017.

[37] M. O'Riordan and C. M. Gilligan-Lee. Spillover detection for donor selection in synthetic control models. *arXiv preprint*, arXiv:2406.11399, 2024.

[38] C. Park and E. J. Tchetgen Tchetgen. Single proxy synthetic control. *Journal of Causal Inference*, 13(1), 2025.

[39] Z. Qian, Y. Zhang, I. Bica, A. Wood, and M. van der Schaar. Synctwin: Treatment effect estimation with longitudinal outcomes. In *NeurIPS*, 2021.

[40] S. Rho, R. Cummings, and V. Misra. Differentially private synthetic control. In *AISTATS*, 2023.

[41] J. M. Robins, A. Rotnitzky, and L. P. Zhao. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American Statistical Association*, 89(427): 846–866, 1994.

[42] D. B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, 66(5):688, 1974.

[43] N. Seedat, F. Imrie, A. Bellot, Z. Qian, and M. van der Schaar. Continuous-time modeling of counterfactual outcomes using neural controlled differential equations. In *ICML*, 2022.

[44] U. Shalit, F. D. Johansson, and D. Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *ICML*, 2017.

[45] C. Shi, D. Sridhar, V. Misra, and D. Blei. On the assumptions of synthetic control methods. In *AISTATS*, 2022.

[46] X. Shi, K. Li, W. Miao, M. Hu, and E. T. Tchetgen. Theory for identification and inference with synthetic controls: A proximal causal inference framework. *arXiv preprint*, arXiv:2108.13935, 2021.

[47] J. Spiess, G. Imbens, and A. Venugopal. Double and single descent in causal inference with an application to high-dimensional synthetic control. In *NeurIPS*, 2023.

[48] M. J. Springer. *How the States Shaped the Nation: American Electoral Institutions and Voter Turnout, 1920-2000*. Chicago Studies in American Politics. University of Chicago Press, Chicago, IL, 2014. ISBN 978-0-226-11421-7.

[49] Y. Sun, H. Xie, and Y. Zhang. Difference-in-differences meets synthetic control: Doubly robust identification and estimation. *arXiv preprint*, arXiv:2503.11375, 2025.

[50] W. Tian. The synthetic control method with nonlinear outcomes: Estimating the impact of the 2019 anti-extradition law amendments bill protests on Hong Kong's economy. *arXiv preprint*, arXiv:2306.01967, 2023.

[51] W. Tian, S. Lee, and V. Panchenko. Synthetic controls with multiple outcomes. *arXiv preprint*, arXiv:2304.02272, 2023.

[52] M. J. van der Laan and S. Rose. *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer Series in Statistics. Springer New York, New York, NY, 2011. ISBN 978-1-4419-9781-4 978-1-4419-9782-1.

[53] M. J. van der Laan and D. Rubin. Targeted maximum likelihood learning. *The International Journal of Biostatistics*, 2(1), 2006.

[54] M. J. van der Laan, D. Benkeser, and W. Cai. Efficient estimation of pathwise differentiable target parameters with the undersmoothed highly adaptive lasso. *arXiv preprint*, arXiv:1908.05607, 2019.

[55] A. W. van der Vaart. *Asymptotic Statistics*. Cambridge University Press, 2000. ISBN 978-1-107-26844-9.

[56] X. Wang, S. Lyu, C. Luo, X. Zhou, and H. Chen. Variational counterfactual intervention planning to achieve target outcomes. In *ICML*, 2025.

[57] J. Zeitler, A. Vlontzos, and C. M. Gilligan-Lee. Non-parametric identifiability and sensitivity analysis of synthetic control models. In *CLeaR*, 2023.

A Extended related work

Semiparametric inference and orthogonal learning: The concept of Neyman orthogonality is deeply rooted in semiparametric efficiency theory [25, 55]. Neyman-orthogonal and efficient influence function-based estimators have a long tradition in causal inference, primarily for the estimation of average treatment effects. Examples include the AIPTW estimator [41], TMLE [53, 54], and the Double ML framework [13]. Recently, the concept of Neyman orthogonality has been extended to HTEs [16], which allowed the construction of various orthogonal learners, including the DR- and R-learner for conditional average treatment effects [24, 34, 36]. *None of these works are tailored to the single-treated unit setting.*

Heterogeneous treatment effects estimation over time: There has been much research on the estimation of heterogeneous treatment effects (HTE) over time, often leveraging several backbone architectures. Early approaches build on recurrent and state-space sequence models that learn latent disease dynamics and simulate counterfactual treatment trajectories [29, 28]. More recent methods leverage representation learning and deep generative modeling, including variational and adversarial formulations, to capture complex, high-dimensional temporal confounding and treatment responses [9, 33, 43]. In parallel, transformer-style and other attention-based backbones have been proposed to improve long-horizon modeling and to better exploit irregular, multivariate patient histories, but with various strategies for confounder adjustment [21, 22, 20, 17, 56, 31].

CATT estimation under other settings: A growing literature develops more flexible, non-linear approaches for estimating CATT [e.g., 35, 50, 47], with broad applications in public policy [e.g., 32]. Existing methods can be broadly grouped into three strands: (a) Synthetic twin approaches [e.g., 39, 9, 7] predict the treated unit from trajectories and the control pool, effectively turning the problem into a flexible forecasting task. However, these models provide no formal identifiability guarantee for the prediction. (b) Difference-in-differences (DiD) style methods [e.g., 10, 27, 6, 49] leverage non-linear learners but still rely on strong identifying restrictions, most notably a parallel-trends assumption. (c) Proxy causal inference approaches [e.g., 46, 30, 38, 37, 57] attempt to recover causal effects by introducing proxy variables, and identifiability hinges on additional proxy assumptions. *Overall, while these non-linear methods can improve predictive fit, their causal interpretation typically requires additional, method-specific assumptions.*

More synthetic control variants: A related line of work localizes control construction by combining synthetic control with matching-style restrictions. For instance, caliper/radius-based procedures form *local* control pools via distances and adaptive calipers and then fit synthetic controls within these neighborhoods [11]. Complementarily, matching–synthetic-control hybrids use cross-validated model averaging to balance the extrapolation bias of matching against interpolation bias of synthetic controls [23]. These methods can improve finite-sample stability by restricting or reweighting the control set, but they remain primarily weight-based constructions and are typically not derived from Neyman-orthogonal estimating equations. In parallel, recent work develops asymptotically unbiased or debiased synthetic-control estimators [18, 19, 14], largely focusing on inference for averages of post-treatment effects rather than period-specific targets.

B One-step debiasing vs. TMLE

We consider a simple example of estimating the ATE in a randomized controlled trial setting to illustrate the difference between the one-step debiasing estimator (AIPW) and TMLE. Let $Z = (X, A, Y) \sim P$ denote observed data, where $X \in \mathcal{X} \subseteq \mathbb{R}^q$ are observed covariates, $A \in \mathcal{A} = \{0, 1\}$ is binary treatment, $Y \in \mathcal{Y} \subseteq \mathbb{R}$ represents outcomes. We assume that we have an observed dataset $\mathcal{D} = \{(x_i, a_i, y_i)\}_{i=1}^n$ of size $n \in N$ are sampled i.i.d from $\mathbb{P}$. We use the potential outcomes framework [42] and denote $Y(a)$ as the potential outcome corresponding to a treatment $A = a$. We define the *response functions* as $\mu_a(x) = \mathbb{E}[Y \mid X = x, A = a]$ for $a \in \{0, 1\}$ and the *propensity score* as $\pi(x) = \mathbb{P}[A = 1 \mid X = x]$. We refer to these functions as *nuisance functions*, denoted by $\eta = (\mu_1, \mu_0, \pi)$. Besides, we impose standard causal inference assumptions [53, 13], including consistency, overlapping assumption, and ignorability, to ensure the identification of the ATE.

Efficient influence functions (EIFs): Our targeted causal estimand is the ATE, and according to standard causal inference assumptions, we convert the causal estimand to a statistical estimand:

$$\underbrace{\mathbb{E}[Y(1) - Y(0)]}_{\text{causal estimand}} = \mathbb{E}\left[\mathbb{E}[Y(1) - Y(0) \mid X = x]\right] = \underbrace{\mathbb{E}[\mu_1(x) - \mu_0(x)]}_{\text{statistical estimand}}. \tag{13}$$

Since $\psi = \mathbb{E}[\mu_1(x) - \mu_0(x)]$ has a so-called von Mises or distributional Taylor expansion [55], we can derive the efficient influence function

$$\phi = \mu_1(X) - \mu_0(X) + \frac{A}{\pi(X)}(Y - \mu_1(X)) - \frac{1 - A}{1 - \pi(X)}(Y - \mu_0(X)) - \psi. \tag{14}$$

One-step debiased estimators (AIPW). Given initial nuisance estimates $\hat{\eta}$, a generic EIF-based *one-step* debiasing estimator takes the form

$$\hat{\psi}^{\text{one-step}} = \psi(\hat{P}) + \frac{1}{n}\sum_{i=1}^{n} \phi(Z_i; \hat{\eta}), \tag{15}$$

where $\psi(\hat{\mathbb{P}})$ is a plug-in estimator based on an estimated distribution $\hat{\mathbb{P}}$ (equivalently, based on fitted nuisance functions), $\psi(\hat{P}) = \frac{1}{n}\sum_{i=1}^{n}[\hat{\mu}_1(x_i) - \hat{\mu}_0(x_i)]$. Plugging Eq. (14) into Eq. (15) yields the one-step debiasing, i.e., the augmented inverse probability weighting (AIPW) estimator

$$\hat{\psi}^{\text{AIPW}} = \frac{1}{n}\sum_{i=1}^{n}\left[\hat{\mu}_1(x_i) - \hat{\mu}_0(x_i) + \frac{a_i}{\hat{\pi}(x_i)}\big(y_i - \hat{\mu}_1(x_i)\big) - \frac{1 - a_i}{1 - \hat{\pi}(x_i)}\big(y_i - \hat{\mu}_0(x_i)\big)\right]. \tag{16}$$

Under suitable regularity conditions (often paired with cross-fitting), the AIPW estimator is asymptotically normal and achieves the semiparametric efficiency bound when the nuisance estimators are sufficiently accurate. Moreover, it enjoys the familiar *double robustness* property: it remains consistent if either the propensity model π or the outcome models $\{\mu_a\}_{a\in\{0,1\}}$ are consistently estimated (with appropriate positivity).

Targeted maximum likelihood estimation (TMLE). TMLE is another EIF-based debiasing way by *targeting* a nuisance function through a low-dimensional update. Starting from initial nuisance function estimates $\hat{\eta}$, TMLE defines a parametric fluctuation (submodel) $\{\hat{\eta}_\epsilon : \epsilon \in \mathbb{R}^d\}$ through $\hat{\eta}$ and chooses $\hat{\epsilon}$ by minimizing a loss (e.g., negative log-likelihood) in a way that enforces the EIF estimating equation:

$$\frac{1}{n}\sum_{i=1}^{n} \phi(Z_i; \hat{\eta}_{\hat{\epsilon}}) \approx 0. \tag{17}$$

We take the response functions as an example. Start with

$$\mu_A^{\varepsilon}(X) = \hat{\mu}_A(X) + \varepsilon \hat{r}(A, X), \tag{18}$$

where $\hat{r}(A, X) = \frac{A}{\hat{\pi}(X)} - \frac{1-A}{1-\hat{\pi}(X)}$ is the clever covariate, $\varepsilon \in \mathbb{R}$. The fluctuation parameter $\hat{\varepsilon}$ is obtained by maximizing the likelihood along the chosen fluctuation submodel. Equivalently, one can estimate $\hat{\epsilon}$ by fitting a regression of Y on the clever covariate $\hat{r}(A, X)$ using $\hat{\mu}_A(X)$ as an offset and imposing no intercept term. When the outcome Y is bounded (e.g., binary), it is common to use a

link function such as the logit link in the targeting step so that the updated regression $\hat{\mu}^{\hat{\epsilon}}(X)$ remains within the admissible range. Then, the targeted response functions are $\hat{\mu}_a^\star(X) := \hat{\mu}_a^{\hat{\epsilon}}(X)$.

Finally, the TMLE estimator is

$$\hat{\psi}^{\text{TMLE}} = \frac{1}{n}\sum_{i=1}^{n} \hat{\mu}_1^\star(x_i) - \hat{\mu}_0^\star(x_i). \tag{19}$$

Why TMLE can be preferable in practice. Both the one-step estimator and TMLE share the same EIF, which means both of them can attain semiparametric efficiency under comparable nuisance functions fitting conditions (e.g., cross-fitting). However, TMLE often has two practical advantages. *First*, as a likelihood-based substitution estimator, TMLE inherits the natural parameter constraints of the chosen model: by updating the nuisance fit *within* a *constrained* likelihood submodel (e.g., via a logit link), the resulting predictions remain in the admissible range, so bounded outcomes yield bounded estimated means or probabilities. *Second*, TMLE chooses the fluctuation parameter to (approximately) satisfy the EIF estimating equation in Eq. (17), absorbing the first-order bias correction into a targeted nuisance update rather than a single additive adjustment; this typically leads to improved numerical behavior and more stable finite-sample performance. Thus, TMLE is frequently preferred when nuisance models are flexible and finite-sample behavior matters.

C Background on the augmented SCM

The augmented SCM reduces bias by combining a synthetic-control-style weighting with an outcome-regression model fitted on control data [8]. Let $\hat{m}_{\tilde{t}}(\cdot)$ denote a regression estimator trained on the control units to predict $Y_{i\tilde{t}}$ from pre-treatment history, and define residualized outcomes $\tilde{Y}_{i\tilde{t}} = Y_{i\tilde{t}} - \hat{m}_{\tilde{t}}(X_i)$. The augmented SCM estimates weights by matching residualized pre-treatment trajectories, i.e.,

$$\hat{\mathbf{w}}^{\text{ASC}} \in \arg\min_{\mathbf{w}\in\Delta^{N-1}} \left\| X_1 - \sum_{j=2}^{N} w_j X_j \right\|_V^2 + \lambda\mathcal{R}(\mathbf{w}), \quad \forall t \le T_0, \tag{20}$$

where $\mathcal{R}(\mathbf{w})$ is an optional regularizer (e.g., ridge-type), $V \succeq 0$ is a user-chosen importance matrix that prioritizes particular control units, and $\lambda \ge 0$ controls shrinkage. The counterfactual in the augmented SCM takes the regression prediction for the treated unit and adds a weighted correction based on control residuals:

$$\hat{\psi}_{\tilde{t}}^{\text{ASC}} = \hat{m}_{\tilde{t}}(X_1) + \sum_{j=2}^{N} \hat{w}_j^{\text{ASC}} \big(Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)\big), \tag{21}$$

Intuitively, $\hat{m}_{\tilde{t}}(X_1)$ provides a flexible plug-in prediction, while the weighted residual term serves as a residual correction, leveraging observed control outcomes to offset model bias. The augmented SCM can substantially improve performance when pre-treatment fit is imperfect. However, its accuracy can depend on nuisance modeling choices and on how the weighting and regression components interact.

Algorithm 2 Augmented SCM

Require: Control data $\{(X_j, Y_{j\tilde{t}})\}_{j=2}^{N}$, treated covariate X_1, importance matrix V, shrinkage parameter λ

/* Stage 1: Nuisance function estimation */

$\hat{m}_{\tilde{t}}(x) \leftarrow \mathbb{E}[Y_{\tilde{t}}(0) \mid X = x]$

$\hat{w} \leftarrow \arg\min_{w\in\Delta^{N-1}} \left\| X_1 - \sum_{j=2}^{N} w_j X_j \right\|_V^2 + \lambda\mathcal{R}(w)$

$\tilde{Y}_{j\tilde{t}} \leftarrow Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)$ for controls $j = 2, \ldots, N$

/* Stage 2: Augmented synthetic control */

$\hat{\psi}_{\tilde{t}}^{\text{ASC}} \leftarrow \hat{m}_{\tilde{t}}(X_1) + \sum_{j=2}^{N} \hat{w}_j \big(Y_{j\tilde{t}} - \hat{m}_{\tilde{t}}(X_j)\big)$

return counterfactual $\hat{\psi}_{\tilde{t}}^{\text{ASC}}$

D Proofs of Theorem 4.1

Since $\hat{w}^\star \in \Delta^{N-1}$, we have $\hat{w}_j^\star \geq 0$ and $\sum_{j=2}^N \hat{w}_j^\star = 1$. Thus, $\hat{\psi}_{\tilde{t}}^{\text{TSC}}$ is a convex combination of the observed control outcomes at time $\tilde{t}$. When $Y_{j\tilde{t}} \in [a, b]$ for all $j = 2, \ldots, N$, this convex-combination form immediately implies boundedness:

$$\hat{\psi}_{\tilde{t}}^{\text{TSC}} = \sum_{j=2}^N \hat{w}_j^\star m_{\tilde{t}}(X_j) = \sum_{j=2}^N \hat{w}_j^\star Y_{j\tilde{t}} \geq \sum_{j=2}^N \hat{w}_j^\star \cdot a = a, \tag{22}$$

where the inequality uses $\hat{w}_j^\star \geq 0$ and $Y_{j\tilde{t}} \geq a$. Then, we yield

$$\hat{\psi}_{\tilde{t}}^{\text{TSC}} = \sum_{j=2}^N \hat{w}_j^\star m_{\tilde{t}}(X_j) = \sum_{j=2}^N \hat{w}_j^\star Y_{j\tilde{t}} \leq \sum_{j=2}^N \hat{w}_j^\star \cdot b = b, \tag{23}$$

where the inequality uses $Y_{j\tilde{t}} \leq 1$ and $\sum_{j=2}^N \hat{w}_j^\star = 1$.

Therefore, $\hat{\psi}_{\tilde{t}}^{\text{TSC}} \in [a, b]$. More generally, if $Y_{j\tilde{t}} \in [a, b]$, then $\hat{\psi}_{\tilde{t}}^{\text{TSC}} \in [a, b]$ by the same argument.

For the one-step estimator, note that it takes the affine form

$$\hat{\psi}_{\tilde{t}}^{\text{1-step}} = \hat{m}_{\tilde{t}}(X_1) + \sum_{j=2}^N \hat{w}_j Y_{j\tilde{t}} - \sum_{j=2}^N \hat{w}_j \hat{m}_{\tilde{t}}(X_j), \tag{24}$$

which is a difference of two convex combinations plus $\hat{m}_{\tilde{t}}(X_1)$. Even if each term individually lies in $[a, b]$, their signed combination in Eq. (24) need not lie in $[a, b]$. □

E Population-level asymptotic normality

Establishing asymptotic guarantees for TSC is challenging in the standard single-treated-unit setting, as documented in the synthetic control literature [1]. In particular, the target counterfactual is a conditional mean evaluated at a fixed treated unit rather than a population-level functional. Increasing the number of control units does not provide additional independent observations of the treated unit [5], and standard semiparametric asymptotic arguments therefore do not directly apply. In this section, we consider a population-level extension under a more restrictive joint large-N_0, large-T_0 regime, where N_0 denotes the number of control units and T_0 denotes the number of pre-treatment periods. Importantly, the following theoretical results focus on this population-level extension under the stated assumptions, rather than the original single-treated-unit setting.

Population-level setting: We consider a longitudinal setting with a single treated unit and N_0 control units observed over T_0 pre-treatment periods. For a fixed prediction horizon $h \geq 1$, the corresponding post-treatment time is $t = T_0 + h$. Let $X_{i,s} = Y_{i,s} \in \mathcal{X} \subseteq \mathbb{R}$ denote the state of unit i at time s, and let $Y_{i,s+h}(0)$ denote its untreated potential outcome h periods later. We observe two sources of data: the treated unit's pre-treatment states $\{X_{1,s}\}_{s=1}^{T_0}$ and the control-unit state-outcome pairs $\{(X_{j,T_0}, Y_{j,T_0+h}(0))\}_{j=2}^{N_0+1}$. Let P_1 denote the marginal distribution of the treated unit's pre-treatment state process, and let Q_0 denote the joint distribution of (X, Y_h) in the control population, where X represents the state at the prediction origin and Y_h the untreated outcome h periods later. Let P_0 denote the marginal distribution of X under Q_0. We define the horizon-specific outcome regression as $m_h(x) = \mathbb{E}_{Q_0}[Y_h \mid X = x]$. Throughout, we consider a joint asymptotic regime where $N_0, T_0 \to \infty$ and impose the following additional assumptions.

Assumption E.1 (Population-level assumptions)**.** For a fixed prediction horizon $h \geq 1$:

(i) *Stability*: The process $\{X_{1,s}\}_{s\geq 1}$ is strictly stationary with $X_{1,s} \sim P_1$ and satisfies suitable weak-dependence and moment conditions ensuring a central limit theorem. Moreover, $(X_{j,T_0}, Y_{j,T_0+h}(0)) \sim Q_0$ for all T_0, where Q_0 is invariant across prediction origins.

(ii) *Transportability*: For all s and P_1-almost every $x \in \mathcal{X}$, $\mathbb{E}[Y_{1,s+h}(0) \mid X_{1,s} = x] = m_h(x) = \mathbb{E}_{Q_0}[Y_h \mid X = x]$.

(iii) *Overlap*: $P_1 \ll P_0$, with $\omega(x) = \frac{\mathrm{d}P_1}{\mathrm{d}P_0}(x) \leq C < \infty$ for P_0-almost every $x \in \mathcal{X}$.

Assumption E.1 characterizes the population-level setting: (i) stability ensures that the treated unit's pre-treatment observations follow a stationary, weakly dependent process with marginal distribution P_1, while the control-unit state-outcome distribution remains invariant across prediction origins; (ii) transportability ensures that the horizon-specific conditional untreated outcome regression is shared between the treated and control populations; and (iii) overlap ensures that the treated-state distribution is supported by the control-state distribution, with a bounded density ratio $\omega(x) = \mathrm{d}P_1(x)/\mathrm{d}P_0(x)$.

Population-level estimand: Under Assumption E.1, we consider the population-level target

$$\psi_h = \mathbb{E}_{P_1}[m_h(X)] = \mathbb{E}_{Q_0}[\omega(X)Y_h]. \tag{25}$$

Unlike the counterfactual prediction for a fixed treated unit, this estimand averages over the treated-state distribution. It admits the following orthogonal estimating functional:

$$\Psi_h(\bar{m}, \bar{\omega}) = \mathbb{E}_{P_1}[\bar{m}(X)] + \mathbb{E}_{Q_0}[\bar{\omega}(X)\{Y_h - \bar{m}(X)\}]. \tag{26}$$

This functional has a second-order bias structure in the outcome-regression and density-ratio estimation errors, providing a basis for population-level orthogonal estimation.

Asymptotic normality: Let $\{X_{1,s}\}_{s=1}^{T_0}$ denote the treated unit's pre-treatment states and $\{(X_{j,T_0}, Y_{j,T_0+h}(0))\}_{j=2}^{N_0+1}$ denote the control-unit observations. We consider the following cross-fitted population-level orthogonal estimator:

$$\hat{\psi}_h = \frac{1}{T_0}\sum_{s=1}^{T_0} \hat{m}_h(X_{1,s}) + \frac{1}{N_0}\sum_{j=2}^{N_0+1} \hat{\omega}(X_{j,T_0})\{Y_{j,T_0+h}(0) - \hat{m}_h(X_{j,T_0})\}, \tag{27}$$

where the nuisance estimators are evaluated out of sample using aligned cross-fitting across the treated and control samples, with dependence-aware splitting for the treated-state process. For simplicity, fold-specific superscripts are suppressed.

Theorem E.2 (Asymptotic normality of the population-level orthogonal estimator)**.** *Suppose Assumption E.1 holds. Assume further that (i) the control unit observations are independently and identically distributed according to Q_0 and are independent of the treated state process; (ii) the treated state contribution satisfies a central limit theorem, and the control-unit contribution has finite variance, with both asymptotic variances strictly positive; (iii) the cross-fitted nuisance estimators are consistent in $L^2(P_0)$, with centered nuisance-score empirical-process remainders of order $o_p(N_0^{-1/2} + T_0^{-1/2})$; and (iv) the nuisance estimators satisfy the product-rate condition*

$$\|\hat{m}_h - m_h\|_{L^2(P_0)} \|\hat{\omega} - \omega\|_{L^2(P_0)} = o_p\left(N_0^{-1/2} + T_0^{-1/2}\right), \tag{28}$$

where conditions (iii) and (iv) are understood to hold uniformly over the fixed number of cross-fitting folds. Then, as $N_0, T_0 \to \infty$,

$$\frac{\hat{\psi}_h - \psi_h}{\sqrt{\sigma_{1,h}^2/T_0 + \sigma_{0,h}^2/N_0}} \xrightarrow{d} \mathcal{N}(0,1), \tag{29}$$

where

$$\sigma_{1,h}^2 = \lim_{T_0 \to \infty} \frac{1}{T_0} \mathrm{Var}\left(\sum_{s=1}^{T_0} m_h(X_{1,s})\right), \tag{30}$$

$$\sigma_{0,h}^2 = \mathrm{Var}_{Q_0}\left(\omega(X)\{Y_h - m_h(X)\}\right). \tag{31}$$

Here, $\sigma_{1,h}^2$ denotes the long-run variance of the treated-state contribution, while $\sigma_{0,h}^2$ denotes the variance of the control-unit contribution. Consequently, $\hat{\psi}_h - \psi_h = O_p(N_0^{-1/2} + T_0^{-1/2})$, corresponding to an effective convergence rate of $r_{N_0,T_0} = (N_0^{-1} + T_0^{-1})^{-1/2} \asymp \min\{\sqrt{N_0}, \sqrt{T_0}\}$.

Proof sketch: By the density-ratio identity and the conditional mean-zero property of the control residuals, the population-level estimating functional satisfies

$$\Psi_h(\bar{m}, \bar{\omega}) - \psi_h = -\mathbb{E}_{P_0}\left[(\bar{\omega}(X) - \omega(X))(\bar{m}(X) - m_h(X))\right]. \tag{32}$$

By the Cauchy–Schwarz inequality, the resulting bias is bounded by the product of the two nuisance estimation errors. Let $\mathbb{P}_{1,T_0}$ and $\mathbb{P}_{0,N_0}$ denote the empirical measures of the treated states and control observations, respectively. The estimation error admits the decomposition

$$\hat{\psi}_h - \psi_h = (\mathbb{P}_{1,T_0} - P_1)m_h + \mathbb{P}_{0,N_0}\left[\omega(X)\{Y_h - m_h(X)\}\right] + R_{N_0,T_0}, \tag{33}$$

where the remainder consists of the second-order nuisance-product term and the centered nuisance-score empirical-process terms, interpreted foldwise. Under conditions (iii) and (iv), $R_{N_0,T_0} = o_p(N_0^{-1/2} + T_0^{-1/2})$. The first leading term satisfies a time-series central limit theorem with long-run variance $\sigma_{1,h}^2$, while the second satisfies an independent-sample central limit theorem with variance $\sigma_{0,h}^2$. Their independence and the negligible remainder yield the stated asymptotic normality.

F Implementation details

F.1 Data generation

General setup. We generate panels with one treated unit and 30 control units. Each unit has $p = 12$ covariates, and outcomes are observed for $T = 50$ periods. The first 40 periods are pre-treatment, and treatment starts at period 41. We report results for prediction horizons $h \in \{1, 5, 10\}$. The treated unit is fixed before outcomes are generated.

Continuous outcomes. For each unit i, we draw $X_i \sim \mathcal{N}(0, I_p)$ and define the normalized time index $u_t = t/40$. The untreated outcome is generated as

$$g_i^{(d)}(t) = 0.8X_{i1} - 0.6X_{i2} + 0.35X_{i3} + (0.25X_{i4} - 0.15X_{i5})\left(u_t - \frac{1}{2}\right) + q_i^{(d)}(t), \tag{34}$$

$$Y_{it}(0) = 0.2u_t + 0.1\sin(2\pi u_t) + 0.2g_i^{(d)}(t) + \varepsilon_{it}, \qquad \varepsilon_{it} \sim \mathcal{N}(0, 0.8^2). \tag{35}$$

The four DGPs differ only through the nonlinear component $q_i^{(d)}(t)$:

$$q_i^{(\text{linear})}(t) = 0, \tag{36}$$

$$q_i^{(\text{hinge})}(t) = \{0.8(X_{i1} - 0.25)_+ - 0.6(-X_{i2} - 0.25)_+\}(1 + 0.2u_t), \tag{37}$$

$$q_i^{(\text{quadratic})}(t) = \left\{\frac{0.5(X_{i1}^2 - 1)}{\sqrt{2}} + 0.4X_{i1}X_{i2}\right\}(1 + 0.2u_t), \tag{38}$$

$$q_i^{(\text{time-varying})}(t) = 0.7\sin(X_{i1})\sin(2\pi u_t) + 0.5X_{i2}X_{i3}\cos(\pi u_t), \tag{39}$$

where $(a)_+ = \max\{a, 0\}$. For the treated unit, the observed post-treatment outcome is obtained by adding a constant effect of one to $Y_{it}(0)$.

Binary outcomes. For binary outcomes, covariates are generated around a common cohort center:

$$C \sim \mathcal{N}(0, I_p), \tag{40}$$

$$X_i = C + s_d\eta_i, \qquad \eta_i \sim \mathcal{N}(0, I_p), \tag{41}$$

where $s_d = 0.7$ for the linear DGP and $s_d = 0.35$ for the other DGPs. We use

$$g_i^{(d)}(t) = 0.8X_{i1} - 0.6X_{i2} + 0.35X_{i3} + 0.9\,(0.5X_{i4} - 0.3X_{i5})\left(u_t - \frac{1}{2}\right) + \widetilde{q}_i^{(d)}(t), \tag{42}$$

$$\widetilde{q}_i^{(\text{linear})}(t) = 0, \tag{43}$$

$$\widetilde{q}_i^{(\text{hinge})}(t) = 1.2\,\{0.8(X_{i1} - 0.25)_+ - 0.6(-X_{i2} - 0.25)_+\}\,(1 + 0.3u_t), \tag{44}$$

$$\widetilde{q}_i^{(\text{quadratic})}(t) = 1.2\left\{\frac{0.5(X_{i1}^2 - 1)}{\sqrt{2}} + 0.4X_{i1}X_{i2}\right\}(1 + 0.3u_t), \tag{45}$$

$$\widetilde{q}_i^{(\text{time-varying})}(t) = 1.2\,\{0.7\sin(X_{i1})\sin(2\pi u_t) + 0.5X_{i2}X_{i3}\cos(\pi u_t)\}\,. \tag{46}$$

To add low-rank time variation, let

$$F_t = \left[1 \quad u_t - \tfrac{1}{2} \quad \sin(2\pi u_t) \quad \cos(3\pi u_t)\right]^\top, \tag{47}$$

$$L_i = 0.35X_{i,5:8} + \kappa_d Z_i, \qquad Z_i \sim \mathcal{N}(0, I_4), \tag{48}$$

where $\kappa_d = 0.5$ for the linear DGP and $\kappa_d = 0.2$ otherwise. The untreated probability is

$$M_{it} = 0.3u_t + 0.2\sin(2\pi u_t) + g_i^{(d)}(t) + \frac{1}{2}L_i^\top F_t, \tag{49}$$

$$E_{i1} = \nu_{i1}, \qquad E_{it} = 0.2E_{i,t-1} + \sqrt{1 - 0.2^2}\,\nu_{it}, \qquad \nu_{it} \sim \mathcal{N}(0, 1), \tag{50}$$

$$p_{it}(0) = \operatorname{expit}\left(-0.25 + \frac{M_{it} + 0.8E_{it}}{2}\right), \tag{51}$$

$$Y_{it}(0) = \mathbb{1}\{U_{it} < p_{it}(0)\}, \qquad U_{it} \sim \text{Uniform}(0, 1). \tag{52}$$

For the treated unit after period 40, the treatment shifts the log odds by $1 + 0.5(t - 40)/10$. The same uniform draw is used to construct the observed and untreated potential outcomes.

F.2 Estimation and computation

The initial SCM weights are obtained by minimizing the pre-treatment squared error over the probability simplex, using all available pre-treatment outcomes. When several solutions have essentially the same loss, we select the solution closest to uniform weights.

At each target period, the outcome regression uses the 12 covariates and information from the last 10 observed outcome periods. For continuous outcomes, the features include the full ten-period history, its mean, standard deviation, most recent value, three-period mean, and linear slope. Binary outcomes use the five summary features, except for the quadratic DGP, which uses the full history. We fit ridge regression with penalty 5 and use five-fold cross-fitting across control units. For the binary linear DGP, the ridge penalty is 20. Predictions for binary outcomes are not clipped.

Let $\widehat{m}_{jt}$ be the out-of-fold prediction for control unit j at the target period and let $R_{jt} = Y_{jt} - \widehat{m}_{jt}$. Starting from the SCM weights $\widehat{w}_j^0$, the targeted weights are

$$\widehat{w}_j(\epsilon, \alpha) = \frac{\widehat{w}_j^0 \exp\{\alpha \epsilon R_{jt}\}}{\sum_{k=1}^{N-1} \widehat{w}_k^0 \exp\{\alpha \epsilon R_{kt}\}}, \tag{53}$$

$$\sum_{j=1}^{N-1} \widehat{w}_j(\epsilon, 1) R_{jt} = 0, \tag{54}$$

$$\widehat{Y}_{1t}^{\mathrm{TSC}}(0) = \sum_{j=1}^{N-1} \widehat{w}_j(\widehat{\epsilon}, \widehat{\alpha}) Y_{jt}. \tag{55}$$

The root is searched over $\epsilon \in [-20, 20]$; if no root exists, we set $\widehat{\epsilon} = 0$. The shrinkage parameter is selected from $\{0, 0.1, 0.25, 0.5, 0.75, 1\}$ using squared prediction error over six pre-treatment pseudo-post periods. For the binary quadratic DGP, we fix $\alpha = 1$.

We compare TSC with classical SCM, a plug-in estimator, and augmented SCM. All methods use the same initial SCM fit and outcome-regression specification. Performance is measured by squared error relative to the realized untreated outcome of the treated unit.

The experiments are implemented in Python 3.11 using NumPy 1.24.3, pandas 2.0.3, SciPy 1.11.1, and scikit-learn 1.3.0. All experiments are run serially with one numerical thread.

G Additional experimental results

This section presents three sets of additional experiments to further evaluate TSC. We first assess its counterfactual prediction accuracy using semi-synthetic experiments based on real-world datasets. We then conduct an oracle ablation to investigate the sensitivity to outcome-regression estimation errors. Finally, we compare TSC with additional baselines and examine its flexibility in incorporating alternative outcome-regression backbones.

G.1 Semi-synthetic experiments

To provide a direct quantitative comparison using real trajectories, we added additional semi-synthetic experiments using both datasets. We retain the observed control units, covariates, and outcome trajectories, and designate untreated units as pseudo-treated. Their observed post-intervention trajectories therefore provide ground truth for evaluating counterfactual prediction accuracy.

On the tobacco panel, TSC achieves an RMSE of 11.728 ± 3.617 at lag 1, 23.616 ± 8.226 at lag 5, and 16.388 ± 5.174 at lag 10. At lag 10, the corresponding RMSE values for classical SCM, augmented SCM, and the plug-in estimator are 39.127 ± 8.652, 27.304 ± 7.359, and 19.240 ± 5.485, respectively. On the binary turnout panel, TSC achieves 0.0390 ± 0.0110, 0.0971 ± 0.0248, and 0.0748 ± 0.0159 across the three lag settings. TSC achieves the lowest RMSE in all six settings.

We note that the differences become larger at longer horizons. This pattern is consistent with the targeting step being particularly useful when inaccuracies in the initial pre-treatment weights have a larger effect on longer-horizon predictions.

These experiments also clarify the role of boundedness. The empirical contribution is not only that TSC remains within the control-outcome range, but that this structural guarantee is obtained without sacrificing predictive accuracy. TSC matches or improves upon the unconstrained augmented SCM.

Method	Lag 1 (1988)	Lag 5 (1992)	Lag 10 (1997)
Classical SCM	14.000 ± 6.092	26.699 ± 8.883	39.127 ± 8.652
Plugin-in estimator	11.872 ± 3.581	24.585 ± 9.054	19.240 ± 5.485
Augmented SCM	11.981 ± 3.746	23.990 ± 10.678	27.304 ± 7.359
TSC (*ours*)	$\mathbf{11.728 \pm 3.617}$	$\mathbf{23.616 \pm 8.226}$	$\mathbf{16.388 \pm 5.174}$

Table 4: Tobacco: Continuous outcome

Method	Lag 1 (1992)	Lag 5 (1996)	Lag 10 (2001)
Classical SCM	0.0495 ± 0.0135	0.0993 ± 0.0249	0.0904 ± 0.0184
Plugin-in estimator	0.0404 ± 0.0145	0.1982 ± 0.0306	0.0968 ± 0.0237
Augmented SCM	0.0402 ± 0.0290	0.1077 ± 0.0584	0.0816 ± 0.0502
TSC (*ours*)	$\mathbf{0.0390 \pm 0.0110}$	$\mathbf{0.0971 \pm 0.0248}$	$\mathbf{0.0748 \pm 0.0159}$

Table 5: Turnout: Binary outcome

G.2 Oracle outcome-regression ablation

To further investigate whether the advantage of TSC arises solely from its convex-hull constraint, we conduct an oracle ablation under the latent-factor data-generating process (DGP). Specifically, we replace the estimated conditional mean $\hat{m}_t$ with the true conditional mean $m_t^\star$ for both Augmented SCM and TSC. We perform 50 repetitions for each prediction horizon $h \in \{1, 5, 10\}$ and evaluate the oracle gain, defined as $G_h = \mathrm{RMSE}_h(\hat{m}_t) - \mathrm{RMSE}_h(m_t^\star)$. A larger oracle gain indicates a greater improvement in predictive accuracy when the true outcome model is available.

Method	$h = 1$	$h = 5$	$h = 10$
Augmented SCM	1.24	1.18	0.61
TSC (*ours*)	0.53	0.54	0.49

Table 6: Oracle gain in RMSE under the latent-factor DGP across different prediction horizons.

As shown in Table 6, the two methods respond differently to outcome-regression quality. Augmented SCM benefits substantially from the oracle outcome model, with gains of up to 1.24, whereas TSC exhibits smaller improvements, with gains of at most 0.54. Once both methods are provided with the true conditional mean, their predictive accuracy becomes comparable. This suggests that the performance gap under estimated outcome models is driven, at least in part, by differences in how outcome-regression estimation errors propagate into counterfactual predictions. In particular, Augmented SCM directly relies on the treated-unit prediction $\hat{m}_t(X_1)$, which can be sensitive to extrapolation errors, whereas TSC avoids evaluating the outcome model at the treated unit.

These findings suggest that the advantage of TSC is not solely attributable to its convex-hull constraint. Rather, its construction also reduces sensitivity to outcome-regression estimation errors in finite samples. We emphasize that the oracle experiment serves as a diagnostic analysis rather than an attainable practical setting, since the true conditional mean is generally unknown. The results also indicate that when the outcome model can be estimated accurately, Augmented SCM remains competitive with TSC.

G.3 Additional Baselines and Backbone Ablation

To further evaluate the predictive performance of TSC, we extend our comparison to three additional baselines: Generalized Synthetic Control (GSC), Matrix Completion (MC), and SyncTwin. We conduct experiments under the synthetic quadratic data-generating process (DGP) and report the RMSE across three prediction horizons, $h \in \{1, 5, 10\}$. As shown in Table 7, TSC consistently achieves the lowest RMSE among the four methods across all three horizons.

Method	Lag 1	Lag 5	Lag 10
GSC	10.612 ± 0.405	10.680 ± 0.468	10.670 ± 0.478
MC	10.668 ± 0.401	10.667 ± 0.418	10.638 ± 0.413
SyncTwin	10.921 ± 1.341	9.825 ± 1.342	9.925 ± 1.343
TSC (*ours*)	$\mathbf{10.150 \pm 1.286}$	$\mathbf{9.733 \pm 1.127}$	$\mathbf{9.923 \pm 1.328}$

Table 7: Comparison with additional baselines under the synthetic quadratic DGP.

Furthermore, TSC is model-agnostic and can incorporate different outcome models within its targeting framework. To examine this flexibility, we replace the default MLP-based outcome estimator $\hat{m}_t(\cdot)$ with GSC, MC, and SyncTwin, while keeping the targeting step unchanged. Table 8 presents the corresponding results under the same quadratic DGP.

Method	Lag 1	Lag 5	Lag 10
TSC + GSC $m_t(X)$	9.991 ± 1.656	10.015 ± 1.662	10.007 ± 1.659
TSC + MC $m_t(X)$	$\mathbf{9.267 \pm 1.901}$	$\mathbf{9.270 \pm 1.902}$	$\mathbf{9.259 \pm 1.898}$
TSC + SyncTwin $m_t(X)$	9.312 ± 0.609	9.599 ± 0.614	10.184 ± 0.596

Table 8: Backbone ablation of TSC under the synthetic quadratic DGP.

Among the alternative backbones, MC consistently yields the lowest RMSE across all three horizons. In particular, at lag 10, incorporating MC reduces the RMSE from 9.923 ± 1.328 with the default MLP backbone to 9.259 ± 1.898. These results demonstrate that TSC is not restricted to a specific outcome-regression architecture and can benefit from alternative prognostic estimators while retaining the same targeting procedure.

H Discussion

Conclusion: We propose a targeted synthetic control (TSC) method that refines initial synthetic-control weights through a TMLE-style targeting procedure while preserving the convex-combination structure of the counterfactual estimator. By incorporating outcome-regression information through residual balancing, TSC maintains interpretable and bounded counterfactual predictions. Our experiments demonstrate that TSC can improve counterfactual prediction accuracy across a range of synthetic and real-world settings.

Limitations: The advantages of TSC are not universal. In particular, TSC can be especially beneficial when outcome models are difficult to estimate accurately, as its final predictions do not directly rely on outcome-model predictions for the treated unit. However, when the outcome model is accurately estimated, augmented SCM can achieve comparable performance. Moreover, our theoretical guarantees primarily concern finite-sample structural properties, including residual balancing and convex-hull preservation. Establishing general asymptotic properties remains challenging in the single-treated-unit setting, where standard large-sample arguments do not directly apply. Asymptotic normality may be established in special regimes with an increasing number of control units and pre-treatment periods, but this would require additional assumptions on temporal stability, transportability, and overlap.

Future directions: An important direction for future research is to establish asymptotic guarantees and develop valid statistical inference for TSC under suitable large-sample regimes. Beyond theoretical developments, the proposed targeting framework could also be extended to other settings, such as single-arm clinical trials in medicine using external or historical controls, where interpretable and bounded counterfactual estimates may be particularly valuable.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We state clearly in the introduction and abstract our contributions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We mentioned limitation in the end of main paper.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide the proofs in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have implementation details in the appendix for the data generation process and models to use.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All the details about the data generation process are in the appendix. And we provide an anonymous GitHub link in the paper.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: It is all included in the implementation detail section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We use the RMSE to report errors with standard errors.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We put the computing source requirement in the implementation details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We used synthetic data and public datasets.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discussed in the introduction.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our method is not related with pre-trained language models, image generators, or scraped datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All the codes and public clinical datasets are properly cited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not introduce new assets. And the code will be made public upon acceptance.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: We do not include with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: We do not include human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLM is only used for writing and formatting purposes.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.