Time-Aware Synthetic Control

Saeyoung Rho[*] **Cyrus Illick**[*] **Samhitha Narasipura**[*]
Alberto Abadie[†] **Daniel Hsu**[*] **Vishal Misra**[*]
Columbia University[*] Massachusetts Institute of Technology[†]

Abstract

The synthetic control (SC) framework is widely used for observational causal inference with time-series panel data. Despite its success across diverse applications, existing SC methods typically treat pre-intervention time indices as exchangeable, meaning they may fail to exploit temporal structure when strong trends are present. We propose Time-Aware Synthetic Control (TASC), a method that addresses this limitation by adopting a state-space model with a constant trend component while preserving the low-rank structure of the signal. TASC uses the Kalman filter and the Rauch–Tung–Striebel smoother in two steps: it first fits a generative time-series model with expectation–maximization and then performs counterfactual inference. We evaluate TASC on simulated and real-world datasets spanning policy evaluation and sports prediction. Our results demonstrate that TASC offers advantages in settings with high observation noise and long prediction horizons.

1 INTRODUCTION

Synthetic Control (SC) is a popular method in observational causal inference. Often described as a natural extension of the Difference-in-Differences (D-in-D, Card and Krueger, 2000), SC aims to evaluate the effects of an intervention more accurately by creating synthetic counterfactual data. The first application was measuring the economic impact of the 1960's terrorist conflict in Basque Country, Spain (a *target unit*) by combining GDP data from other Spanish regions (*donor units*) prior to the conflict to construct a *synthetic* GDP data for Basque Country in the counterfactual world without the conflict (Abadie and Gardeazabal, 2003). Unlike D-in-D, which compares the changes in outcomes over time between a treated group and a comparison group, SC builds a synthetic comparison unit as a weighted combination of donors. SC is becoming increasingly popular with an expanding range of applications, including economics (Abadie and Gardeazabal, 2003; Abadie et al., 2010; Abadie and L'Hour, 2021), political sciences (Abadie et al., 2015; Kreif et al., 2016), social sciences (Robbins et al., 2017; Vagni and Breen, 2021), and healthcare (Thorlund et al., 2020; Vagni and Breen, 2021; Shen et al., 2025).

SC methods assume that time-series panel data arise from a latent variable model, without restricting a relationship among the time-varying latent factors. The linear factor model, widely adopted in SC literature (Abadie and Gardeazabal, 2003; Abadie et al., 2010, 2015), is one example. This model is both flexible and versatile; for example, it can be extended to incorporate autoregressive components. However, this same flexibility leads SC methods built on top of the model to produce identical estimations when the pre-intervention time indices are permuted. While such flexibility avoids imposing strong structural assumptions, it also prevents the model from capturing predictive signals when a learnable trend exists. A key insight is that time-series data often exhibit stable trends, which we explicitly incorporate into the model.

Another key property of time-series panel dataset is that, as the data size increases, the resulting data matrix tends to be approximately low-rank. This phenomenon, well analyzed by Udell and Townsend (2019), becomes more pronounced when temporal trends are stronger, limiting the movement of latent factors across time points. Building on this insight, numerous SC variants have been proposed to leverage the data's low-rank structure. These methods typically rely on spectral analysis of the data matrix: for example, Amjad et al. (2018) employs principal com-

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

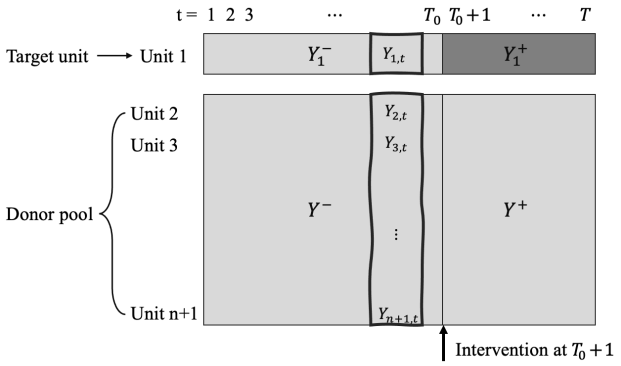

Figure 1: General data structure for synthetic control

ponent regression, while Athey et al. (2021) frames SC as a nuclear-norm minimization problem. However, these approaches are also time-agnostic because shuffling of time indices does not affect the spectrum of a matrix.

Our contribution lies in embedding SC panel data within a state-space model to simultaneously harness *both the low-rank and time-series properties* of the data. We provide 1) the TASC model, a state-space generative model for panel dataset, 2) TASC algorithms to learn the TASC model. Our paper is organized as follows. In Section 2, we present necessary background knowledge in synthetic control methods and common modeling assumptions. In Section 3, we introduce the TASC model, a time-series generative model based on a state-space model. Section 4 outlines expectation-maximization (EM) style TASC algorithms, based on Kalman filtering and Rauch–Tung–Striebel (RTS) smoothing. In Sections 5 and 6, we apply TASC to simulated and real-world datasets and demonstrate when our approach is favorable.

2 RELATED WORK

2.1 Synthetic Control Methods

The time-series panel dataset for SC consists of the following components. Let $Y_{i,t} \in \mathbb{R}$ be the observation from i-th unit (row) at time t (column). The first row corresponds to the treated target unit with index 1, while the n untreated donor units occupy rows $i \in \{2, \ldots, n+1\}$. This setup yields a total of $N = n + 1$ rows. The *untreated* observation matrix Y is of size $N \times T$, where the target unit's values after $t > T_0$ is missing due to the treatment happening after time T_0. Figure 1 illustrates the general structure of an SC dataset, where the superscripts $-$ and $+$ denote the pre- and post-intervention periods, respectively.

Based on this data structure, we define SC family of methods (Algorithm 1) as follows. SC first learns the relationship between the target unit and donors using the pre-intervention data. For example, $\mathcal{M}$ can be a *vertical* regression where the donor's pre-intervention column vectors $Y_{2:n+1,t}$ become input features for the label $Y_{1,t}$, for all $t \in [T_0]$. Then, SC uses this knowledge (f) to project the post-intervention donor data Y^+ and predict $\hat{Y}_1^+$. Finally, the causal effect of the intervention on the target unit is estimated as $Y_1^+ - \hat{Y}_1^+$.

Algorithm 1 Synthetic Control Family of Methods

Data: Target unit's pre-intervention data $Y_1^- \in \mathbb{R}^{T_0}$, Donor data $Y = [Y^-, Y^+] \in \mathbb{R}^{n \times T}$
Result: Counterfactual prediction $\hat{Y}_1^+$, SC weights f
1. Learn $f = \mathcal{M}(Y_1^-, Y^-, Y^+)$
2. Project $\hat{Y}_1^+ = f(Y^+)$
3. Infer the estimated causal effect of the intervention for the target is $Y_1^+ - \hat{Y}_1^+$

$\mathcal{M}$ refers to the SC learning algorithm. Much of the SC literature has adopted a least squares predictor over the convex scan of Y^-: $f = \arg\min_f ||Y_1^- - f^\top Y^-||^2$ where $\sum_{i=1}^n f_i = 1, 0 \le f \le 1 \; \forall i \in [n]$[1] (Abadie and Gardeazabal, 2003; Abadie et al., 2010, 2015). The convex scan condition can be replaced by Lasso ($||f||_1$) or Ridge ($||f||_2^2$) regularization (Doudchenko and Imbens, 2016; Amjad et al., 2018). Some approaches use PCA to keep only the top few singular values in data prior to the optimization step Amjad et al. (2018, 2019). Other variations of synthetic control algorithms focused on issues such as handling multiple treated units (Dube and Zipperer, 2015; Xu, 2017), dealing with a large the number of donors (Abadie and L'Hour, 2021; Rho et al., 2025), and correcting biases (Ben-Michael et al., 2021). See Abadie (2021) for a detailed survey of these techniques.

2.2 Latent Variable Models for Synthetic Control

The first SC algorithm suggested by Abadie and Gardeazabal (2003) assumes a linear factor model

$$Y_{i,t} = \delta_t + \theta_t Z_i + \lambda_t \mu_i + \epsilon_{i,t}, \tag{1}$$

where δ_t is a time trend, $Z_i \in \mathbb{R}^p$ and $\mu_i \in \mathbb{R}^q$ are vector of observed and unobserved predictors, with coefficients θ_t and μ_i, and $\epsilon_{i,t}$ is the noise. This model implies that the signal component of the matrix has a rank no more than $p + q + 1$. When this quantity is considerably smaller than the matrix's full rank, the

[1] Multivariate time-series and other covariates can be used with relative importance weighting, but this paper focuses on univariate time-series.

observation matrix becomes approximately low rank. This is indeed a common case for a factor model, verified in many real-world data (Stock and Watson, 1999; Gregory and Head, 1999; Forni et al., 2000). This has inspired a range of SC algorithms to utilize the approximately low-rank structure of data. Several SC algorithms employ simplex constraints or regularizers to minimize the number of active donor units (Abadie and Gardeazabal, 2003; Abadie et al., 2010; Doudchenko and Imbens, 2016; Chernozhukov et al., 2021); Rho et al. (2025) introduces donor selection step to reduce the number of donors in the first place; Amjad et al. (2018, 2019) uses principal component regression; and Athey et al. (2021) frames SC as a problem of nuclear norm minimization.

Another characteristic of the panel data used in SC is its time-series nature. Despite this, many SC algorithm variants remain invariant to permutations of time indices in pre-intervention data. Although the permutation-invariant approach provides robustness by accommodating a wide range of temporal trends, it discards ordering information, whereas explicit modeling strategies can exploit additional structure when meaningful temporal patterns exist. Some researchers addressed this by adopting state-space models (Brodersen et al., 2015; Klinenberg, 2023; Shao et al., 2022). The focus of these approaches was to allow the relationship between the target and the donor pool to change over time by defining the SC weights as latent variables (i.e., hidden states).

3 STATE-SPACE MODEL FOR TASC

Let y_t be the t-th column of the *untreated* outcomes Y. The TASC approach assumes the following state space model:

$$x_t = Ax_{t-1} + q_{t-1}, \;\; q_{t-1} \sim \mathcal{N}(0, Q), \tag{2}$$
$$y_t = Hx_t + r_t, \;\; r_t \sim \mathcal{N}(0, R), \tag{3}$$

where we assume the initial hidden state $x_0 \sim \mathcal{N}(m_0, P_0)$. The hidden states x_t is d-dimensional, whereas y_t is a $N = n+1$ dimensional vector. To keep the low-rank structure, we require $d \ll min(n, T)$.

The model parameters are $\theta = \{A, H, Q, R, m_0, P_0\}$, where $A \in \mathbb{R}^{d\times d}$, $H \in \mathbb{R}^{N\times d}$, $Q \in \mathbb{R}^{d\times d}$, $R \in \mathbb{R}^{N\times N}$, $m_0 \in \mathbb{R}^d$, and $P_0 \in \mathbb{R}^{d\times d}$. This is a classical linear Gaussian model, and we set all covariance matrices Q, R, and P_0 be positive definite. If desired, we may constrain the noise covariance matrices Q and R to be diagonal with non-zero diagonal entries to reduce the number of parameters.

3.1 Comparison to Other Models

The classical SC (Abadie and Gardeazabal, 2003) assumes a linear factor model with observed and unobserved factors as in Equation (1). This can be reformulated as state-space models: all time-dependent variables define the latent state $x_t = (\delta_t, \theta_t, \lambda_t)$ and the mapping between the latent state and individual observations (i.e., i-th row of H) encodes $h_i = (1, Z_i, \mu_i)$. With this formulation, the hidden state dimension becomes $d = 1 + p + q$. However, linear factor models do not explicitly assume a trend matrix A. This absence can be encoded either by allowing a time-varying trend A_t at each time point or by setting $A = 0$ and modeling $x_t = q_t, ; q_t \sim \mathcal{N}(0, Q_t)$. The key distinction from TASC is that TASC enforces a stable relationship among x_t, whereas linear factor models do not. This suggests that while TASC can achieve improved predictive performance under correct specification, it is also more susceptible to misspecification when temporal dynamics are complex.

In Robust Synthetic Control (RSC, (Amjad et al., 2018)), matrix entries are assumed to follow a latent variable model $Y_{i,t} = g(\theta_i, \rho_t) + \epsilon_{i,t}$ where θ_i and ρ_t are d dimensional[2] latent vectors characterizing i-th unit and t-th time, and $\epsilon_{i,t}$ is observation noise. This is a more generalized expression that can include the linear factor model in Equation (1). RSC's learning algorithm can be interpreted as learning θ_i and ρ_t by treating g as a dot product and employing PCA: $Y = \sum_{l=1}^{min(n,T)} s_l u_l v_l^\top = \sum_{l=1}^{d} s_l u_l v_l^\top + \sum_{l=d+1}^{min(n,T)} s_l u_l v_l^\top$, where s_l is singular values in decreasing order. By defining $\tilde{U}$ to have $s_l^{1/2} u_l$ for $l \le d$ as columns and $\tilde{V}^\top$ to have $s_l^{1/2} v_l$ for $l \le d$ as rows, the rows of $\tilde{U}$ can be interpreted as θ_i and the columns of $\tilde{V}$ as ρ_t. Similarly, the TASC model suggests a decomposition $Y = HX + E$, where the columns of X are hidden states x_t and the columns of E are observation noise r_t. Here, the rows of H are analogous to θ_i and the columns of X are to ρ_t. Both $\tilde{U}\tilde{V}^\top$ and HX are exactly low-rank matrices, but they differ in how we separate the noise. The difference comes from the learning objectives: RSC's approach minimizes the size of the noise matrix (in terms of spectral norm), whereas TASC focuses on making sure the time-features x_t evolve gradually over time with a constant trend A. As a result, the noise filtered by RSC algorithm becomes rank $\min(n, T) - d$, whereas E is almost surely full rank (omnidirectional).

Some researchers have introduced algorithms that assume temporal trend by utilizing state-space models. Brodersen et al. (2015) designed the state vector to include elements such as SC weights, local lin-

[2]Dimension of θ_i and ρ_t may vary.

ear trends, and seasonality. Klinenberg (2023); Shao et al. (2022) further simplified this structure and only take SC weights as a latent state. The central concept of these approaches is to allow SC weights to change over time, defining the target unit's time series as an observation (scalar) and the SC weights (potentially alongside additional components) as latent states (at least n-dimensional vector, where n is the number of donors). Therefore, these modeling approaches do not necessarily ensure that the observation matrix is approximately low-rank. Furthermore, by not explicitly modeling the stochasticity of the donor pool, these models may not fully leverage the information available in the donor pool.

3.2 When TASC Model Is Advantageous

A distinctive feature of the TASC approach is its explicit modeling of the trend A. This design choice offers several advantages, though it may introduce limitations in certain cases. First, incorporating A enhances the *interpretability* of the TASC model, as it captures the underlying trend in time-series data. Second, if trend A persists beyond the intervention point, it provides TASC with strong predictive power, which is particularly advantageous over longer time horizons. Third, by modeling A, TASC becomes *sensitive to the time orders*, rather than being agnostic to them. Overall, TASC leverages temporal structure when a consistent trend exists but offers limited benefits when the trend is weak or absent (e.g., $A = 0$).

Another benefit of TASC is the approximately low-rank structure, represented as $Y = HX + E$, where HX is an exactly low-rank signal and E denotes noise. This low-rank property commonly appears in real-world datasets, especially as the data size increases. In such cases, Principal Component Analysis (PCA) is a widely adopted technique for extracting low-rank signals. It has been applied in various domains such as image processing, speech and audio analysis, genomics, finance, and sensor data analysis, and has also been utilized in synthetic control methods by (Amjad et al., 2018). However, the performance of PCA deteriorates in the presence of substantial observational noise, as it assumes that the learned principal components are noise-free in the selected directions. In contrast, TASC may offer improved robustness in extracting signals under noisy conditions by assuming omnidirectional noise (i.e., the noise matrix is full rank).

We formalize this intuition using data processing inequality in Appendix B.

Algorithm 2 $\mathrm{EM}_{\mathrm{pre}}(Y_{\mathrm{pre}}; N_1)$

Data: Y_{pre} where (i,j)-th element is $y_{i,t}$ $\forall (i,t) \in [1 : n+1] \times [1 : T_0]$ (pre-intervention data from the target and donors)
Result: $\theta = \{A, H, Q, R, m_0, P_0\}$
Initialize $\theta^{(0)}$
for $i \leftarrow 1$ **to** N_1 **do**
  for $k \leftarrow 1$ **to** T_0 **do**
    Update m_k, P_k via Kalman filtering with $\theta^{(i-1)}$ `/* filtering pass of E-Step */`
  end
  for $k \leftarrow T_0 - 1$ **to** 0 **do**
    Update m_k^s, P_k^s, G_k via RTS Smoothing with $\theta^{(i-1)}$ `/* smoothing pass of E-step */`
  end
  Update $\theta^{(i)}$ using M-step (Algorithm 7) with $T = T_0$ `/* M-step */`
end
return $\theta^{(N_1)}$

4 TASC ALGORITHMS

In this section, we present TASC algorithms to learn TASC model parameters and make counterfactual predictions. First, TASC uses pre-intervention data for parameter learning. We take an Expectation-Maximization (EM) approach, and the update on M-step has a closed form solution for the exact maximizer of the expected complete-data log-likelihood. Then, TASC performs counterfactual estimation by running additional Kalman Filtering and RTS Smoothing passes. Since the post-intervention target data is deemed missing, we set the variance of observation noise for the target coordinate as infinity so that the associated Kalman gain is set to zero.

4.1 Learning from Pre-Intervention Data

TASC learns the model parameters using the pre-intervention data, running $\mathrm{EM}_{\mathrm{pre}}$ as a subroutine. We can take the classical EM approach for a linear gaussian state-space model, where the E-step comprises of a filtering pass (Kalman filtering) and an smoothing pass (RTS smoothing). This gives us estimates m_k^s and P_k^s to define a lower bound for the posterior probability distribution. Algorithm 2 shows the main EM algorithm for parameter estimation based on pre-intervention data.

For the M-step of Algorithm 2, we compute the maximizer of the expected complete log-likelihood (Q-function) by using the fact that y_t follows a multivariate Gaussian distribution. This approach has a closed-form solution, as shown in Algorithm 7. If desired, this step can be replaced by a gradient ascent

over the Q-function. This gradient-based approach is preferable if additional modeling parameters are required and no close-form solutions can be computed. For implementation, one can define a neural network with the same E-step as a forward pass, and perform gradient ascent.

4.2 Counterfactual Inference with Post-Intervention Data

With the model parameters learned from EM_{pre} (Algorithm 2), TASC uses another pass of Kalman filter and RTS smoother to perform counterfactual inference. However, this is impossible without a special treatment since the first element of y_k (which belongs to the target unit) is missing. To handle this, we deem that the target unit's data is missing, and separate the donor portion of the data and parameters: $y_t = \begin{bmatrix} y_{t,1} \\ y_{t,2} \end{bmatrix}, r_t = \begin{bmatrix} r_{t,1} \\ r_{t,2} \end{bmatrix}, H = \begin{bmatrix} h_1^\top \\ H_2 \end{bmatrix}$, and $R = \begin{bmatrix} r_1 & 0 \\ 0 & R_2 \end{bmatrix}$, where $y_{t,2}, r_{t,2} \in \mathbb{R}^n$, $H_2 \in \mathbb{R}^{n \times d}$, and $R_2 \in \mathbb{R}^{n \times n}$. Then, we can rewrite the observation model for the donors as $y_{t,2} = H_2 x_t + r_{t,2}$, where $r_{t,2} \sim \mathcal{N}(0, R_2)$. With this new model, the post-intervention observations will not inform the target-related parameters: h_1 and r_1. This is equivalent to setting $r_1 \to \infty$ in the original model.

With the infinite variance, post-intervention target time series do not affect the outcome of Kalman filtering, hence it can be set to any value. The RTS Smoothing remains the same, as it does not use R or y_k as an input. As a result, this only changes the Kalman filter part in the post-intervention time steps from Algorithm 4 (Original Kalman filter) to Algorithm 5 (Kalman filter with $r_1 \to \infty$). The complete description of TASC is provided in Algorithm 3. The EM_{pre} in the first line requires $O(N_1 T_0 N^3)$ time, where the N^3 and d^3 terms arise from matrix inversion using the naive algorithm. The full TASC procedure incurs an additional $O(TN^3)$, but assuming $T \ll N_1 T_0$, the overall time complexity is dominated by the EM_{pre} part.

Algorithm 3 TASC($Y; N_1$)

Data: $y_{i,t}\ \forall (i,t) \in [1 : n+1] \times [1 : T_0]$ and $\forall (i,t) \in [2 : n+1] \times [T_0 + 1 : T]$
Result: $\hat{\theta} = \{A, H, Q, R, m_0, P_0\}, \hat{y}_{1,T_0+1}, \ldots, \hat{y}_{1,T}$
Learn $\theta^{N_1} \leftarrow \text{EM}_{\text{pre}}(Y_{\text{pre}}; N_1)$
for $k \leftarrow 1$ **to** T_0 **do**
  Update m_k, P_k via Algorithm 4 with θ^{N_1} `/* pre-intervention filtering */`
end
for $k \leftarrow T_0 + 1$ **to** T **do**
  Update m_k, P_k via Algorithm 5 with θ^{N_1} `/* filtering with infinite variance */`
end
for $k \leftarrow T - 1$ **to** 0 **do**
  Update m_k^s, P_k^s via RTS Smoothing with $\theta^{(i-1)}$ `/* smoothing pass */`
end
Define $H = \begin{bmatrix} h_1^\top \\ H_2 \end{bmatrix}$
for $k \leftarrow T_0 + 1$ **to** T **do**
  $\hat{y}_{1,t} \leftarrow h_1^\top m_t^s$ `/* counterfactual inference */`
end
return $\theta^{N_1}, \hat{y}_{1,T_0+1}, \ldots, \hat{y}_{1,T}$

5 EVALUATION ON SIMULATION DATA

In this section, we test the performance of TASC on simulated datasets and compare against the classical Synthetic Control (SC) by Abadie and Gardeazabal (2003), Robust Synthetic Control (RSC) by Amjad et al. (2018), and Causal Impact Model (CIM) by Brodersen et al. (2015). SC is implemented with the relative importance matrix[3] $V = I$ in Algorithm 8, and RSC in Algorithm 9. For CIM, we used the authors' Python implementation[4].

The dataset is generated following the state-space model in Equations (2) and (3) with random parameters. The parameters are generated in the following fashion. A is an orthonormal matrix obtained from a QR decomposition of a random matrix with i.i.d. standard normal entries. The rows of H are d-dimensional vectors randomly drawn from a Dirichlet distribution with parameters $\alpha_i \sim \text{Uniform}([0, 1))$ for $i \in [d]$. Q is generated by first making a random matrix $Z \in \mathbb{R}^{d \times d}$ with all entries drawn from $\text{Uniform}([a, b))$ and defining $\tilde{Q} = \frac{1}{d} ZZ^\top + 10^{-6} I_d$. Then, we randomly flip the sign of off-diagonal entries while keeping it symmetric. R and P_0 are generated in a similar fashion (but with $N = n + 1$ instead of d for R). Finally, all entries of m_0 are generated by $\text{Uniform}([0, 1))$. With these parameters, we generate the true signal $y_t^* = Hx_t$ as well as the noisy observation $y_t = y_t^* + r_t$. The model only sees y_t, but the prediction accuracy was measured by comparing against the true signal y_t^*.

This data generation scheme also falls under the linear factor model as well. Recall that the TASC model is a subset of the linear factor model, where we explic-

[3] See Abadie et al. (2010) for more information, it is equivalent to linear regression with simplex constraint introduced in Section 2.1

[4] `https://github.com/google/tfp-causalimpact`

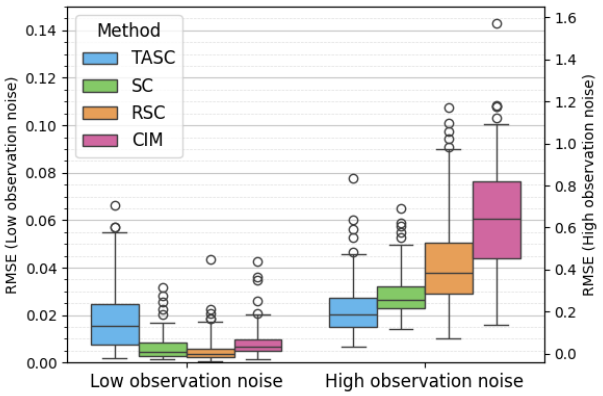

Figure 2: Post-intervention RMSE on datasets generated with a strong trend (small Q)

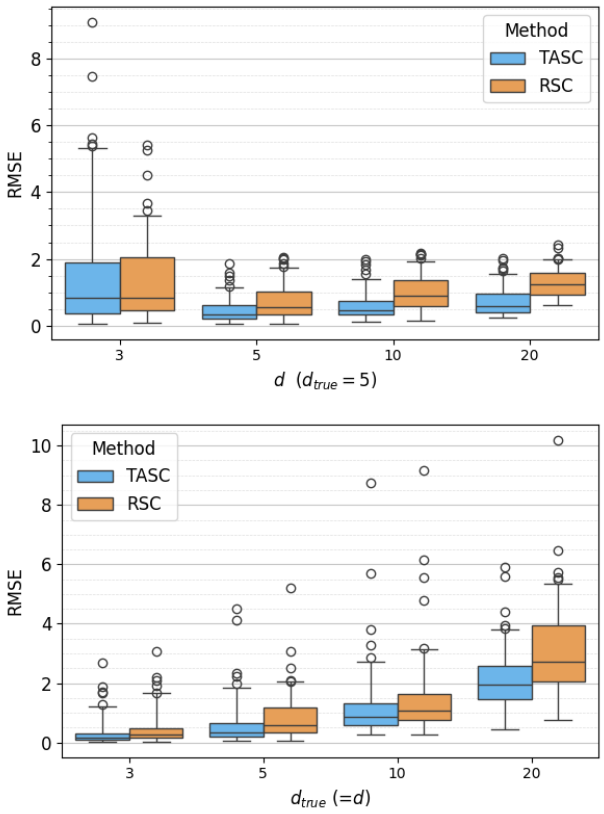

Figure 3: Post-intervention RMSE across different values of d (top, with fixed $d_{\text{true}} = 5$) and d_{true} (bottom, with fixed $d = d_{\text{true}}$)

itly assume the trend A in the evolution of the hidden states x_t. By varying the covariance Q of the hidden state perturbation q_t, we can regulate the time series behavior: a smaller Q enforces a stronger trend A, while a larger Q weakens the trend, making the model closer to the general linear factor framework without assumptions on the trend A. That being said, when Q is small, this data generation scheme may mechanically favor TASC.

All experiments are designed as a placebo test, where counterfactual estimation algorithms are used to predict outcomes in the absence of intervention (i.e., the counterfactual predictions should closely match the observed outcomes when no intervention occurs). We fix the default data generation setting to have $T = 100$, $N = 50$, $d_{\text{true}} = 5$. For covariance matrix generation, we set $a = 0.1$, $b = 1$ for both Q and R, and $a = 0.01, b = 0.1$ for P_0. The intervention point is set to $T_0 = 50$. Some settings are modified for the ablation study, as detailed in each study. For each experimental setting, we tested on 100 randomly generated datasets. All boxplot whiskers extend from zero to the 95th percentile. We provide a concise summary of the main findings here, with complete experimental details available in Appendix E.

We first examine the effect of permuting time indices on TASC performance (Figure 10). TASC's accuracy degrades when the time order is shuffled, whereas SC and RSC remain unaffected by design.

Next, we conduct an ablation study on the covariance matrices Q and R. A small Q indicates a stronger temporal trend (i.e., stronger influence of A), whereas a large Q generates data that resemble a more general linear factor model where A is not restricted to be constant over time. Similarly, a small R corresponds to low observation noise, while a large R implies higher observational noise. Figure 2 reports the post-intervention RMSE from datasets with small Q (parameters $a = 0.01$, $b = 0.1$). We test with low observation noise (left, R: $a = 0.01$, $b = 0.1$) and high observation noise (right, R: $a = 0.1$, $b = 1$). When observation noise is low, RSC achieves the best prediction accuracy, reflecting the strong performance of PCA. In contrast, under high-noise conditions, TASC performs best. This pattern persisted when Q was large as well (see Figures 11 and 12).

Then, we conduct an ablation study on the hidden state dimension. Among the methods we tested, TASC and RSC explicitly impose a low-rank structure on the data matrix and require the approximate rank d as a hyperparameter. Let d_{true} denote the true data-generating dimension. Figure 3 reports the post-intervention RMSE with varying hyperparameter d while fixing $d_{\text{true}} = 5$ (top) and varying d_{true} while fixing $d = d_{\text{true}}$. We find that both TASC and RSC suffer performance degradation when d is under- or over-estimated. However, TASC remains comparatively robust to overestimation of the true dimension. When we set $d = d_{\text{true}}$, TASC's relative advantage

over RSC increases as the dimensionality of the hidden state grows.

Lastly, we conduct an ablation study on the number of donors n (Figure 15). Performance is generally optimized when $n \approx T_0$, while too few or too many donors increase the post-intervention RMSE. TASC demonstrates particular robustness in small-n settings, with relatively minor performance loss between $n = 10$ and $n = 50$. This likely reflects a mechanical advantage in the simulation design, which aligns with TASC's model assumptions.

6 EVALUATION ON REAL-WORLD DATA

In this section, we demonstrate TASC on three real-world applications: (1) the Proposition 99 case study, (2) cricket game score prediction, and (3) basketball game score prediction. While Appendix F provides detailed plots and analyses, we summarize the main findings here. In each case, we select the hyperparameter d for TASC (hidden state dimension) and RSC (approximate rank) using singular spectrum heuristics, Identifying the "kink" point where the marginal gain in cumulative singular value becomes negligible. For RSC, the ridge coefficient was chosen by testing[5] values in $\{10^{-1}, 10^0, 10^1, \ldots, 10^6\}$, where the ridge coefficient of 10^3 was chosen for both cricket and NBA.

Proposition 99 We revisit the Proposition 99 case study from Abadie et al. (2010). Proposition 99 was a policy enacted in California in 1988 that significantly increased the state's cigarette tax. This policy was followed by a noticeable decline in cigarette sales. To assess whether this decline was causally driven by the policy, synthetic control can be applied to estimate the counterfactual outcome for California—i.e., what it would have looked like had the policy not been implemented. We only use per-capita cigarette sales data to ensure consistency, focusing on comparing methods rather than estimating the policy effect itself.

Figure 4 shows the observed California (black) and estimated counterfactual California by TASC (blue), SC (green), RSC (orange), and CIM (pink). All four methods output broadly consistent predictions, slightly diverging closer to 2000. TASC suggests that the effect of the policy was realized by around 1995 (see Figure 19, the gap between observation and counterfactual estimated by TASC is flat at the end), whereas others continue to predict the gap to increase until later. The pink and blue shades denote the 95% confidence interval for CIM and TASC. CIM's interval is much wider than TASC's, and this pattern was observed in other datasets as well (see Figure 22). Figure 5 shows the post-intervention RMSE from placebo test[6]. TASC achieves the lowest median and the smallest variance of RMSE, suggesting that TASC provided the most reliable estimates among the methods tested in this experimental setting.

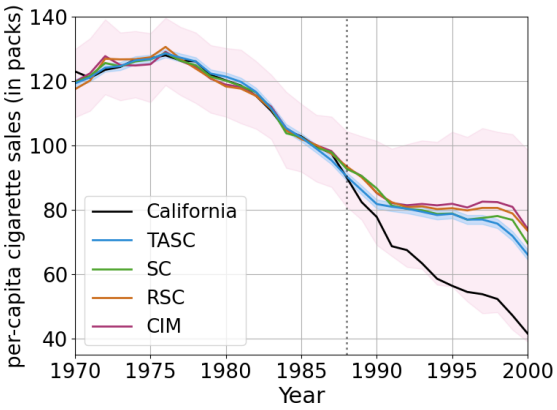

Figure 4: Per-capita cigarette sales in packs in California (black), and estimated counterfactual outcomes by TASC and other benchmarks. Blue and pink shades denote 95% confidence interval from TASC and CIM.

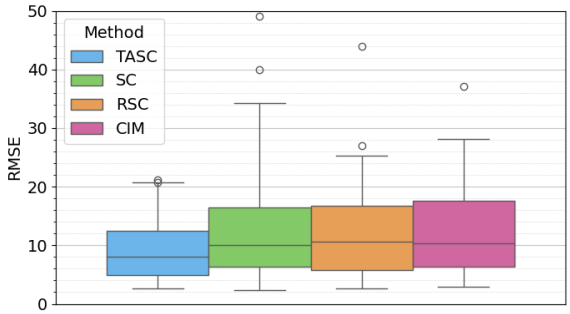

Figure 5: Post-intervention RMSE from placebo test

Cricket Although synthetic control is primarily a tool for counterfactual estimation, it can also be used for prediction with *episodic time series*, such as game score trajectories: by aligning games at their start, we can construct a panel data. Using Indian Premier League[7] cricket data, we evaluated each method on 100 randomly selected matches from 2008 to 2025. We fixed $T_0 = 72$ (end of Over 12) for all experiments.

First, we tested the effect of the number of donors n on

[5]We ran a cross-validation on pre-intervention data: we split the first 80% of pre-intervention data as training, and the following 20% of data as validation set.

[6]A donor state becomes the target, and the remaining donor states serve as the donor pool.

[7]Each game consists of 20 overs of six balls each.

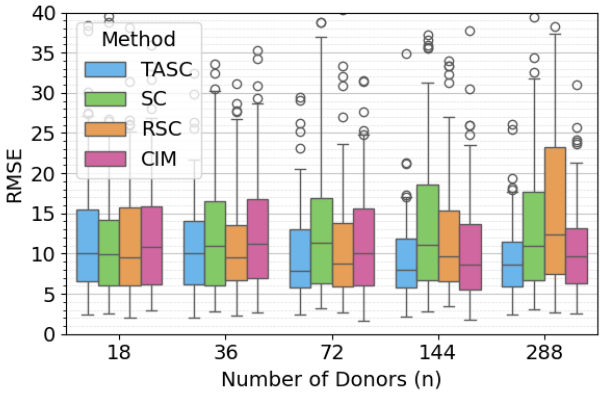

Figure 6: Overall post-intervention RMSE, cricket

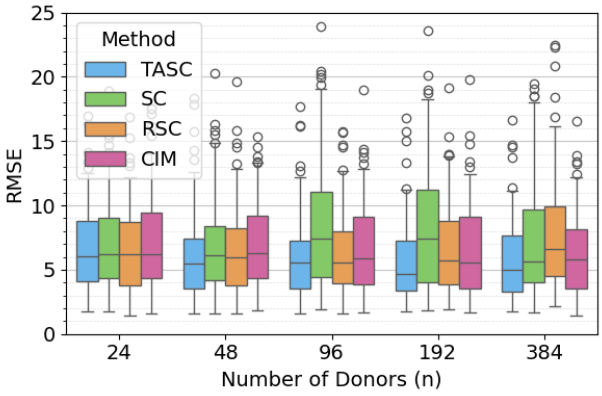

Figure 8: Overall post-intervention RMSE, NBA

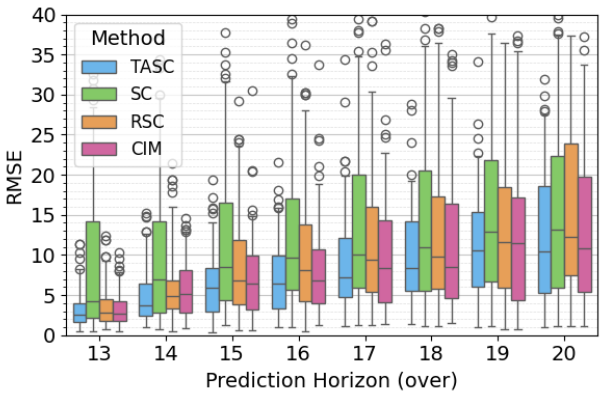

Figure 7: Prediction RMSE per over, cricket

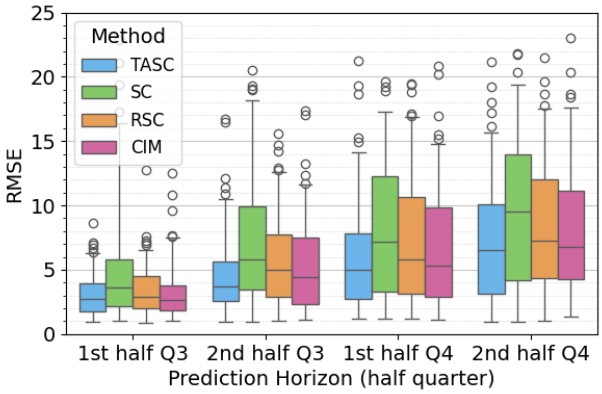

Figure 9: Prediction RMSE per half-quarter, NBA

the post-intervention RMSE (Figure 6). As the number of donors increase, the performance of TASC and CIM improves, indicating that they are using more information from increased donor units. On the other hand, the performance of SC and RSC deterioratesas n increases, possibly due to the curse of dimensionality (i.e., n defines the dimension of a learning problem in SC/RSC formulation). Next, we fix $n = 144$ and plot prediction RMSE per over (Figure 7). Accuracy declines with longer prediction horizon across all methods, but TASC degrades least and achieves the lowest RMSE, followed by CIM. RSC excels in short-term predictions, while SC is stronger in long-term predictions relative to RSC.

Basketball Following the settings from cricket experiments, we evaluate 100 randomly selected NBA games from 2020 to 2023. NBA game score was sampled at 15-second-interval, for four quarters consisting 12 minutes each. We set $T_0 = 96$ at halftime for all experiments. Figure 8 shows the effect of the number of donors n on the post-intervention RMSE. We observe similar trend as in cricket, where TASC improves with larger donor size and RSC suffers from high-dimensional overfitting. Figure 9 shows the prediction RMSE per half-quarter when $n = 192$. Similar to the cricket results, TASC degrades least and achieves the lowest RMSE especially in predictions made for further future.

7 CONCLUSION AND FUTURE WORK

In this work, we introduce TASC, a state-space framework for modeling synthetic control-type data with algorithms for model learning and counterfactual inference. By explicitly modeling the temporal evolution of latent time-factors, TASC achieves strong long-term predictive performance. Experiments on synthetic data and real-world cases show that TASC remains effective under high observational noise, better leverages larger donor pools, is robust to the choice of hyperparameter d, and produces narrower confidence intervals than CIM.

There are limitations to our current work. The TASC

model focuses solely on a single time series, despite the availability of multivariate time series in many real-world settings. Also, TASC imposes a strictly linear and time-invariant trend, which limits its flexibility. Lastly, the current EM-based learning algorithm is slow and sensitive to initialization, often leading to poor fits. Trying multiple initializations may improve fit, but further reduces computational efficiency.

Looking ahead, TASC can be extended in several directions to advance causal inference tools for richer panel data. First is to adopt advanced models, such as more complex latent variable structures and non-linear state-space models. These enhancements would enable TASC to incorporate multiple auxiliary time series and capture non-linear trends. Second, we can improve the learning algorithm, for example, by adopting gradient-based EM or MCMC samplers. This is also linked to developing a more sophisticated model architecture as a future step. Finally, designing diagnostic tools to identify the conditions under which TASC is most effective would help formalize the empirical findings of this work.

Acknowledgements

This work was partially supported by ONR (N00014-24-1-2687, N00014-24-1-2700), and the Columbia-Dream Sports AI Innovation Center PhD Fellowship.

References

Abadie, A. (2021). Using synthetic controls: Feasibility, data requirements, and methodological aspects. *Journal of Economic Literature*, 59(2):391–425.

Abadie, A., Diamond, A., and Hainmueller, J. (2010). 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.

Abadie, A., Diamond, A., and Hainmueller, J. (2015). Comparative politics and the synthetic control method. *American Journal of Political Science*, 59(2):495–510.

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

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

Amjad, M., Misra, V., Shah, D., and Shen, D. (2019). mRSC: Multi-dimensional robust synthetic control. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(2):1–27.

Amjad, M., Shah, D., and Shen, D. (2018). Robust synthetic control. *Journal of Machine Learning Research*, 19(22):1–51.

Athey, S., Bayati, M., Doudchenko, N., Imbens, G., and Khosravi, K. (2021). Matrix completion methods for causal panel data models. *Journal of the American Statistical Association*, 116:1716–1730.

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

Brodersen, K. H., Gallusser, F., Koehler, J., Remy, N., and Scott, S. L. (2015). Inferring causal impact using bayesian structural time-series models. *The Annals of Applied Statistics*, 9(1):247–274.

Card, D. and Krueger, A. B. (2000). Minimum wages and employment: A case study of the fast food industry in New Jersey and Pennsylvania. *American Economic Review*, 90(5):1397–1420.

Chernozhukov, V., Wüthrich, K., and Zhu, Y. (2021). An exact and robust conformal inference method for counterfactual and synthetic controls. *Journal of the American Statistical Association*, 116(536):1849–1864.

Doudchenko, N. and Imbens, G. W. (2016). Balancing, regression, difference-in-differences and synthetic control methods: A synthesis. NBER Working Paper 22791.

Dube, A. and Zipperer, B. (2015). Pooling multiple case studies using synthetic controls: An application to minimum wage policies. IZA Discussion Paper No. 8944.

Durbin, J. and Koopman, S. J. (2012). *Time Series Analysis by State Space Methods*, volume 38 of *Oxford Statistical Science Series*. Oxford University Press, Oxford, 2nd edition.

Forni, M., Hallin, M., Lippi, M., and Reichlin, L. (2000). The generalized dynamic-factor model: Identification and estimation. *Review of Economics and statistics*, 82(4):540–554.

Gregory, A. W. and Head, A. C. (1999). Common and country-specific fluctuations in productivity, investment, and the current account. *Journal of Monetary Economics*, 44(3):423–451.

Klinenberg, D. (2023). Synthetic control with time varying coefficients a state space approach with bayesian shrinkage. *Journal of Business & Economic Statistics*, 41(4):1065–1076.

Kreif, N., Grieve, R., Hangartner, D., Turner, A. J., Nikolova, S., and Sutton, M. (2016). Examination of the synthetic control method for evaluating health

policies with multiple treated units. *Health economics*, 25(12):1514–1528.

Rho, S., Tang, A., Bergam, N., Cummings, R., and Misra, V. (2025). Clustersc: Advancing synthetic control with donor selection. In *International Conference on Artificial Intelligence and Statistics*, pages 109–117. PMLR.

Robbins, M. W., Saunders, J., and Kilmer, B. (2017). A framework for synthetic control methods with high-dimensional, micro-level data: evaluating a neighborhood-specific crime intervention. *Journal of the American Statistical Association*, 112(517):109–126.

Shao, J., Yin, M., Cai, X., and Valeri, L. (2022). Generalized synthetic control method with state-space model. In *NeurIPS 2022 Workshop on Causality for Real-world Impact*.

Shen, D., Agarwal, A., Misra, V., Schelter, B., Shah, D., Shiells, H., and Wischik, C. (2025). Obtaining personalized predictions from a randomized controlled trial on alzheimer's disease. *Scientific Reports*, 15(1):1671.

Stock, J. H. and Watson, M. W. (1999). Forecasting inflation. *Journal of monetary economics*, 44(2):293–335.

Thorlund, K., Dron, L., Park, J. J., and Mills, E. J. (2020). Synthetic and external controls in clinical trials–a primer for researchers. *Clinical epidemiology*, pages 457–467.

Udell, M. and Townsend, A. (2019). Why are big data matrices approximately low rank? *SIAM Journal on Mathematics of Data Science*, 1(1):144–160.

Vagni, G. and Breen, R. (2021). Earnings and income penalties for motherhood: estimates for British women using the individual synthetic control method. *European Sociological Review*, 37(5):834–848.

Xu, Y. (2017). Generalized synthetic control method: Causal inference with interactive fixed effects models. *Political Analysis*, 25(1):57–76.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Time-Aware Synthetic Control: Supplementary Materials

A More Advanced Models for TASC

In this section, we show some examples of more advanced state-space model formulations that can be adopted by the TASC learning algorithms with minimal modifications.

A.1 Allowing time-invariant portion of latent states

Without loss of generality, we can add a constant state x^* and let only x'_t part to change over time.

$$x'_t = Ax'_{t-1} + q_{t-1}, \;\; q_{t-1} \sim \mathcal{N}(0, Q), \tag{4}$$
$$x_t = x^* + x'_t \tag{5}$$
$$y_t = Hx_t + r_t, \;\; r_t \sim \mathcal{N}(0, R). \tag{6}$$

The additional model parameter $x^* \in \mathbb{R}^d$ is required, and this model can be easily adopted with our algorithms with minimal modifications.

A.2 Allowing seasonality

The current model in Equations (2) and (3) do not capture the seasonality. Fortunately, the TASC model can be easily modified to accommodate the seasonality. Let $\mathbf{1}$ denote a column vector with all entries equal to 1. To incorporate seasonality, we define $s_t \in \mathbb{R}$ as the seasonal effect at time t that is constant across units, and specify the model parameters as $\theta = \{A, H, Q, R, m_0, P_0, s_1, \ldots, s_T\}$.

$$x_t = Ax_{t-1} + q_{t-1}, \;\; q_{t-1} \sim \mathcal{N}(0, Q), \tag{7}$$
$$y_t = Hx_t + s_t\mathbf{1} + r_t, \;\; r_t \sim \mathcal{N}(0, R). \tag{8}$$

With this formulation, s_t can either set by the user as a hyperparameter, or learned in the M-step of the EM algorithm.

A.3 Allowing multiple time series

When we have m time series observed at time t for unit i, we can stack them vertically to redefine y_t. Let $y_t^{(1)}, \ldots, y_t^{(m)} \in \mathbb{R}^n$ be the m time series observed from n units at time t. Then, we treat as if we observe nm units at time t and define $y_t = [{y_t^{(1)}}^\top, \ldots, {y_t^{(m)}}^\top]^\top \in \mathbb{R}^{nm}$.

$$x_t = Ax_{t-1} + q_{t-1}, \;\; q_{t-1} \sim \mathcal{N}(0, Q), \tag{9}$$
$$y_t = \begin{bmatrix} y_t^{(1)} \\ \vdots \\ y_t^{(m)} \end{bmatrix} = Hx_t + r_t, \;\; r_t \sim \mathcal{N}(0, R). \tag{10}$$

If desired, one can model the connection among the m time series from the same unit by designing another layer of latent lookup table.

B Theoretical Justification of TASC

In this section, we elaborate more on the theoretical justifications behind TASC. We make the intuition outlined in Section 3.2 precise by showing that classical SC and many of its variants, being permutation-invariant, may overlook predictive information that TASC's structure is designed to recover. This connection can be formalized through the data processing inequality.

A central claim of our framework is that the classical synthetic control method and many of its variants are *permutation-invariant* over time, and therefore cannot exploit predictive information embedded in temporal ordering. In contrast, our TASC approach leverages a state-space model and the Kalman filter to extract *minimal sufficient statistics* for forecasting future outcomes. We now formalize this claim using the data processing inequality. Let $\mathcal{F}_{T_0}^{\text{seq}}$ denote the σ-algebra generated by the *ordered* pre-treatment trajectories $\{y_t\}_{t=1}^{T_0}$, and $\mathcal{F}_{T_0}^{\text{bag}}$ the σ-algebra generated by the same data treated as an *unordered multiset* of columns. Let $\hat{x}_{T_0|T_0}$ and $\hat{P}_{T_0|T_0}$ denote the filtered hidden state and covariance at T_0 based on the observation up to T_0.

Proposition B.1. *Consider the panel data following Equations* (2) *and* (3)*. Then, the following holds.*

1. *(**Kalman Sufficiency**) The filtered state $\hat{x}_{T_0|T_0}$ and covariance $\hat{P}_{T_0|T_0}$ obtained from the Kalman filter are* minimal sufficient statistics *for predicting $\{y_t\}_{t>T_0}$. That is,*

$$y_t \perp\!\!\!\perp \mathcal{F}_{T_0}^{\text{seq}} \mid (\hat{x}_{T_0|T_0}, \hat{P}_{T_0|T_0}), \quad t > T_0.$$

2. *(**Information Loss by Permutation Invariance**) Since $\mathcal{F}_{T_0}^{\text{bag}} \subset \mathcal{F}_{T_0}^{\text{seq}}$ for any $T_0 > 1$, the data processing inequality (i.e., Blackwell's ordering of information structures) implies that,*

$$I\Big(\{y_t\}_{t>T_0}; \mathcal{F}_{T_0}^{\text{bag}}\Big) \;\le\; I\big(\{y_t\}_{t>T_0}; \mathcal{F}_{T_0}^{\text{seq}}\big)\,,$$

   with strict inequality whenever $A \neq 0$ and Q is finite (i.e., the temporal dynamics carry a predictive signal).

3. *(**Dominance**) Consequently, the Bayes-optimal predictor based on Kalman sufficient statistics $\hat{y}_{t,1}^{\text{KF}} = \mathbb{E}[y_{t,1} \mid \hat{x}_{T_0|T_0}, \hat{P}_{T_0|T_0}]$ achieves a mean squared error that is theoretically no larger than that of permutation-invariant predictors $\hat{y}_{t,1}^{\text{bag}}$, measurable with respect to $\mathcal{F}_{T_0}^{\text{bag}}$, under these assumptions.*

$$\mathbb{E}\big[(y_{t,1} - \hat{y}_{t,1}^{\text{KF}})^2\big] \;\le\; \mathbb{E}\Big[(y_{t,1} - \hat{y}_{t,1}^{\text{bag}})^2\Big]\,.$$

Proof Sketch. Let $Y^- \in \mathbb{R}^{N \times T_0}$ be the pre-treatment donor matrix (rows = donors, columns = time indices), and $Y_1^- \in \mathbb{R}^{T_0}$ the pre-treatment vector for the target unit. The classical SC solves

$$f^\star = \arg\min_{f \in \Delta^{n-1}} \|Y_1^- - f^\top Y^-\|^2,$$

where Δ^{n-1} is the n-dimensional unit simplex. For any permutation matrix $\Pi \in \mathbb{R}^{T_0 \times T_0}$ acting on the time indices (columns), we have

$$\|Y_1^-\Pi - f^\top Y^-\Pi\|^2 = \|Y_1^- - f^\top Y^-\|^2,$$

since the Euclidean norm is invariant under reordering. Thus SC and RSC objectives are unaffected by permutations of time, confirming permutation invariance.

Part 1. follows from the standard sufficiency of the Kalman filter in linear Gaussian models: $(\hat{x}_{T_0|T_0}, P_{T_0|T_0})$ are sufficient for $p(y_t \mid y_{1:T_0})$ for any $t > T_0$ (see, e.g., Durbin and Koopman (2012)).

Part 2. follows since $\mathcal{F}_{T_0}^{\text{bag}} \subset \mathcal{F}_{T_0}^{\text{seq}}$; by the data processing inequality and Blackwell's ordering, predictors restricted to $\mathcal{F}_{T_0}^{\text{bag}}$ cannot outperform those using $\mathcal{F}_{T_0}^{\text{seq}}$, except in degenerate cases ($A = 0$).

Part 3. then follows directly: the Kalman-based predictor uses sufficient statistics from $\mathcal{F}_{T_0}^{\text{seq}}$, while permutation-invariant SC is restricted to $\mathcal{F}_{T_0}^{\text{bag}}$, implying weak dominance in MSE and strict dominance when temporal structure is present. □

Remark B.2. *This result motivates a natural two-stage approach: (i) extract Kalman sufficient statistics from the temporal structure, then (ii) apply SC to these statistics instead of raw observations. Our TASC formulation can be viewed as a unification of this two-stage Kalman filter and synthetic control procedure into a generative model, with improved robustness under model misspecification and noisy parameter estimation. Empirically, this can be validated via a* permutation stress test*: permutation-invariant methods should show no degradation when pre-treatment time indices are shuffled, while TASC should degrade, confirming that it exploits temporal information absent in classical approaches.*

C Basic Algorithms

In this section, we provide the full pseudocode for the basic algorithms comprising the EM approach: Kalman Filter (Algorithm 4), Kalman Filter with Infinite Variance (Algorithm 5), RTS Smoother (Algorithm 6), the M-step with MLE approach (Algorithm 7).

Algorithm 4 Kalman Filter

Input : $y_k \in \mathbb{R}^{n+1}$, previous estimate m_{k-1}, P_{k-1}, current parameter $\theta = \{A, H, Q, R, m_0, P_0\}$
Output: m_k, P_k
$m_{k|k-1} \leftarrow Am_{k-1}$ /* prediction from the previous timestep $k-1$ */
$P_{k|k-1} \leftarrow AP_{k-1}A^\top + Q$ /* prediction from the previous timestep $k-1$ */
$v_k \leftarrow y_k - Hm_{k|k-1}$
$S_k \leftarrow HP_{k|k-1}H^\top + R$
$K_k \leftarrow P_{k|k-1}H^\top S_k^{-1}$ /* Kalman Gain */
$m_k \leftarrow m_{k|k-1} + K_k v_k$ /* Update after observing y_k */
$P_k \leftarrow P_{k|k-1} - K_k S_k K_k^\top$ /* Update after observing y_k */

Algorithm 5 Kalman Filter with Infinite Variance

Input : $y_k \in \mathbb{R}^{n+1}$ with the target(first) element missing, previous estimate m_{k-1}, P_{k-1}, current parameter $\theta' = \{A, H, Q, R, m_0, P_0\}$, where $R'_{1,1} = \infty$
Output: m_k, P_k
Define h_1, H_2, R_2 from $H = \begin{bmatrix} h_1^\top \\ H_2 \end{bmatrix}$, $R' = \begin{bmatrix} \infty & 0 \\ 0 & R_2 \end{bmatrix}$
$y_k \leftarrow [h_1^\top m_{k|k-1}, y_{1,k}, \ldots, y_{n,k}]^\top$ /* augment target values */
$m_{k|k-1} \leftarrow Am_{k-1}$ /* prediction from the previous timestep $k-1$ */
$P_{k|k-1} \leftarrow AP_{k-1}A^\top + Q$ /* prediction from the previous timestep $k-1$ */
$v_k \leftarrow y_k - Hm_{k|k-1}$ /* the first element is zero */
$S_k \leftarrow HP_{k|k-1}H^\top + R'$
$S_k^{-1} \leftarrow \begin{bmatrix} 0 & 0 \\ 0 & (H_2 P_{k|k-1} H_2^\top + R_2)^{-1} \end{bmatrix}$ /* by Schur Complement */
$K_k \leftarrow P_{k|k-1}H^\top S_k^{-1}$
$m_k \leftarrow m_{k|k-1} + K_k v_k$ /* Update after observing y_k */
$P_k \leftarrow P_{k|k-1} - K_k S_k K_k^\top$ /* Update after observing y_k */

Algorithm 6 Rauch–Tung–Striebel (RTS) Smoother

Input : Kalman filter estimate m_k, P_k, smoothed estimate m^s_{k+1}, P^s_{k+1}, current parameter $\theta = \{A, H, Q, R, m_0, P_0\}$
Output: m^s_k, P^s_k
$m_{k+1|k} \leftarrow Am_k$ /* prediction from Kalman filter estimate m_k */
$P_{k+1|k} \leftarrow AP_kA^\top + Q$ /* prediction from Kalman filter estimate P_k */
$G_k \leftarrow P_kA^\top P^{-1}_{k+1|k}$
$m^s_k \leftarrow m_k + G_k[m^s_{k+1} - m_{k+1|k}]$ /* $m^s_t = m_t$ for the last timestep $t = T_0$ or T */
$P^s_k \leftarrow P_k + G_k[P^s_{k+1} - P_{k+1|k}]G^\top_k$ /* $P^s_t = P_t$ for the last timestep $t = T_0$ or T */
return m^s_k, P^s_k, G_k

Algorithm 7 Parameter Update (M-Step) with MLE Approach

Input : current parameter $\theta = \{A, H, Q, R, m_0, P_0\}$, length of the sequence T, RTS parameters m^s_k, P^s_k, G_k for all $k \in \{0, \ldots, T\}$, observations y_k for all $k \in \{1, \ldots, T\}$
Output: θ'
Define
$\Sigma = \frac{1}{T}\sum_{k=1}^{T} P^s_k + m^s_k {m^s_k}^\top$
$\Phi = \frac{1}{T}\sum_{k=1}^{T} P^s_{k-1} + m^s_{k-1} {m^s_{k-1}}^\top$
$B = \frac{1}{T}\sum_{k=1}^{T} y_k {m^s_k}^\top$
$C = \frac{1}{T}\sum_{k=1}^{T} P^s_k G^\top_{k-1} + m^s_k {m^s_{k-1}}^\top$
$D = \frac{1}{T}\sum_{k=1}^{T} y_k {y_k}^\top$
Update
$A' \leftarrow C\Phi^{-1}$
$H' \leftarrow B\Sigma^{-1}$
$Q' \leftarrow \mathrm{Diag}(\Sigma - 2CA^\top + A\Phi A^\top)$ /* Diag$(\cdot)$ keeps only the diagonal elements of the input */
$R' \leftarrow \mathrm{Diag}(D - 2BH^\top + H\Sigma H^\top)$
$m'_0 \leftarrow m^s_0$
$P'_0 \leftarrow P^s_0 + (m^s_0 - m_0)(m^s_0 - m_0)^\top$
return $\theta' = \{A', H', Q', R', m'_0, P'_0\}$

D Benchmark Synthetic Control Algorithms

In this section, we provide a full algorithm description for the benchmark synthetic control algorithms used in Sections 5 and 6.

The classical Synthetic Control (SC) performs a vertical regression with simplex constraint Abadie et al. (2010). Algorithm 8 shows the classical SC implemented in our study, where the importance matrix V is set to an identity matrix. Note that our implementation in Proposition 99 study (Section F.1) is slightly different from the original analysis (in Abadie et al. (2010)) because we only use the target time series of interest (per capita tobacco sales in packs) without any additional covariates.

Algorithm 8 Synthetic Control Abadie et al. (2010)

Data: Target unit's pre-intervention data $Y_1^- \in \mathbb{R}^{T_0}$, Donor data $Y = [Y^-, Y^+] \in \mathbb{R}^{n \times T}$
Result: Counterfactual prediction $\hat{Y}_1^+$, SC weights f
1. Learn
$f = \arg\min_f ||Y_1^- - f^\top Y^-||^2$ where $0 \leq f \leq 1, \sum_{i=1}^n f_i = 1$ `/* Simplex constraint */`
2. Project $\hat{Y}_1^+ = f(Y^+)$
3. Infer the estimated causal effect of the intervention for the target is $Y_1^+ - \hat{Y}_1^+$

Robust Synthetic Control (RSC, Amjad et al. (2018)) performs hard singular-value thresholding (HSVT) as a pre-processing step to denoise the observation data. Then, it learns a vertical regression model using the pre-intervention portion of the data, and projects with the post-intervention data for counterfactual inference. Algorithm 9 describes our adoption of the original algorithm with the observation probability $p = 1$ (no missing data).

Algorithm 9 Robust Synthetic Control Amjad et al. (2018)

Data: Target unit's pre-intervention data $Y_1^- \in \mathbb{R}^{T_0}$, Donor data $Y = [Y^-, Y^+] \in \mathbb{R}^{n \times T}$, Number of singular values to keep d
Result: Counterfactual prediction $\hat{Y}_1^+$, SC weights f
1-1. Denoise
$Y = \sum_{i=1}^{min(n,T)} s_i u_i v_i^\top$ `/* Singular Value Decomposition (SVD) */`
$\tilde{Y} = \sum_{i=1}^{d} s_i u_i v_i^\top$ `/* Hard Singular Value Thresholding (HSVT) */`
1-2. Learn $f = \arg\min_f ||Y_1^- - f^\top \tilde{Y}^-||^2 + \lambda||f||^2$
2. Project $\hat{Y}_1^+ = f^\top \tilde{Y}^+$
3. Infer the estimated causal effect of the intervention for the target is $Y_1^+ - \hat{Y}_1^+$

E More Results from Empirical Evaluation on Simulated Data

In this section, we demonstrate our method TASC on simulated data and compare against three benchmark algorithms: Synthetic Control (SC) with simplex constraint Abadie and Gardeazabal (2003), Robust Synthetic Control (RSC) with hard singular-value thresholding Amjad et al. (2018), and Causal Impact Model (CIM) with bayesian modeling approach Brodersen et al. (2015).

E.1 Effect of Permuting Time Indices

We tested the effect of permuting time indices on TASC performance. From a randomly generated dataset, we shuffle the pre-intervention and post-intervention indices separately to ensure no mixing between the two segments. Figure 10 shows the post-intervention RMSE when the time indices are kept in their original order (left) and when the indices are permuted (right). TASC performance deteriorates when the time indices are permuted: the mean increases by 48.5% and the standard deviation increases by 25.7%. In contrast, SC and RSC predictions remain unchanged by design, even when the time indices are permuted.

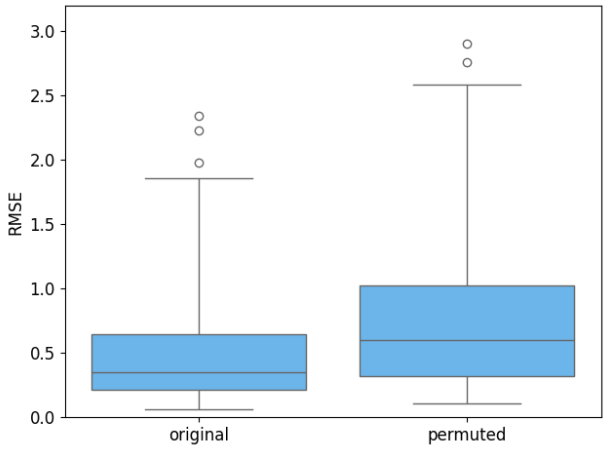

Figure 10: Post-intervention RMSE of TASC when the time indices are kept in their original order (left) and when the indices are permuted (right)

E.2 Ablation Study on the Covariance Matrices

We compare the performance of TASC against three benchmarks under different settings for generating the hidden state perturbation covariance Q and the observation noise covariance R. For both Q and R, we test a small covariance matrix (generated with $a = 0.01$, $b = 0.1$) and a large covariance matrix ($a = 0.1$, $b = 1$). To illustrate how these settings affect the generated time series, the average absolute value of the observation noise is approximately 0.0839 with small R, and 0.8365 with large R. A large Q results in a higher average absolute signal value (2.6624), while a small Q produces a lower value (0.4842). As a result, the signal-to-noise ratio (SNR) is highest in the small R and large Q setting, followed by small R and small Q, large R and large Q, and finally large R and small Q. In practice, a small Q indicates a stronger temporal trend (i.e., stronger influence of A), whereas a large Q generates data that resemble a more general linear factor model where A is not restricted to be constant over time. Similarly, a small R corresponds to low observation noise, while a large R implies higher observational noise.

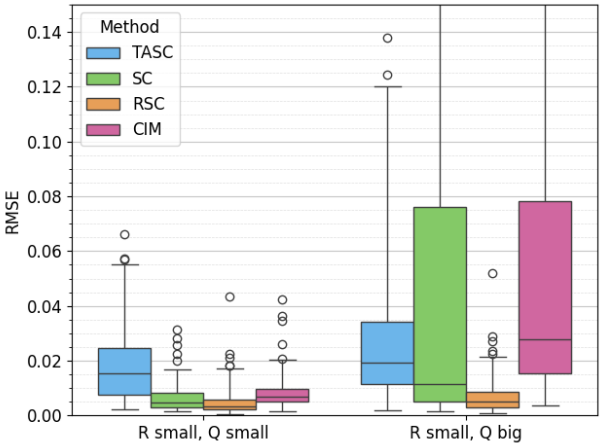

Figure 11: Post-intervention RMSE of TASC and benchmark methods on datasets generated with low observation noise (small R): small Q (left) and large Q (right)

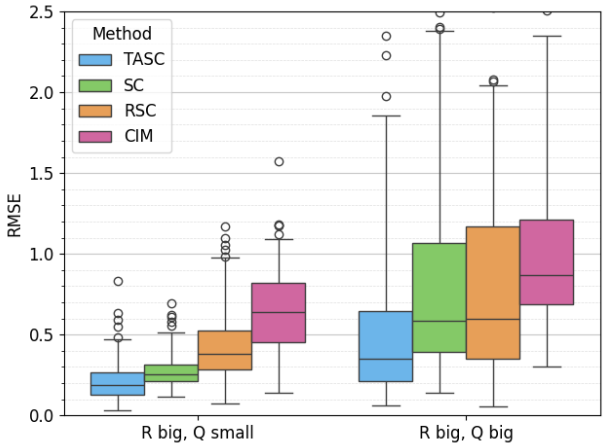

Figure 12: Post-intervention RMSE of TASC and benchmark methods on datasets generated with high observation noise (large R): small Q (left) and large Q (right)

Figure 11 shows the post-intervention RMSE of TASC and benchmark methods on datasets generated with low observation noise (small R). For both small Q (left) and large Q (right), RSC demonstrates the best prediction

accuracy, highlighting the strong performance of PCA when observation noise is low. SC performs comparably to RSC when Q is small, suggesting that the simplex constraint in SC is effective under low observation noise. TASC shows relatively better performance when Q is small, as expected with a more evident trend A, though it still underperforms compared to the other benchmarks in this setting. These results suggest that SC and RSC are able to capture the underlying structure more effectively—even without explicitly modeling the temporal trend A, which is more noticeable when Q is small—under low observation noise.

Figure 12 presents a similar plot, this time under high observation noise (i.e., large R). In this high-noise setting, TASC demonstrates strong performance when Q is small. As Q increases, TASC's performance declines, similar to the low observation noise case, but it still shows the best prediction accuracy among all other benchmark methods. Interestingly, all four methods perform better when Q is small (hence the trend A is evident). Among them, SC proves to be the most reliable in this regime, exhibiting the lowest variance in RMSE. Overall, TASC appears to be a robust choice under high observation noise, regardless of the strength of the temporal trend.

Last, we break down the performance under high observation noise into five evenly divided future time periods and inspect the change in performance as prediction horizon extends. In Figure 13, which shows the results from small Q, the performance of TASC is stable across prediction periods (near future or further future). In Figure 14, which shows the results from large Q, the performance of TASC degrades when the prediction horizon is further away (91˜100) compared to the near future (51˜60). Similar trend is also observed in other benchmark methods as well.

Finally, we break down the performance under high observation noise into five evenly divided future time periods and examine how performance changes as the prediction horizon extends. Figure 13, which shows results for small Q, indicates that TASC maintains relatively stable performance across all prediction periods—whether in the near or more distant future. In contrast, Figure 14, which presents results for large Q, shows that TASC's performance deteriorates more clearly as the prediction horizon extends (91–100) compared to earlier periods (51–60).

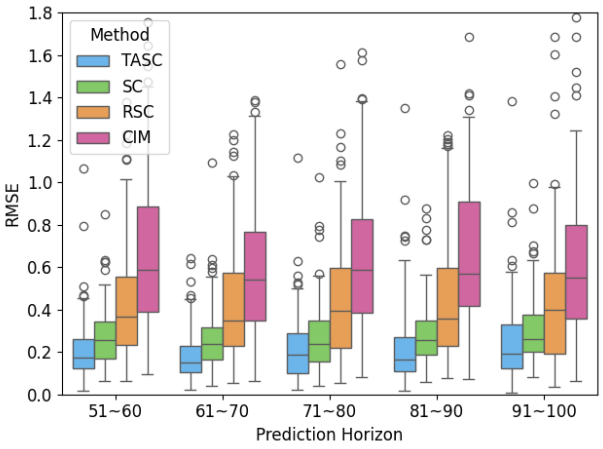

Figure 13: Post-intervention RMSE under low observation noise and small Q, evaluated across five future prediction horizons (10 time periods each)

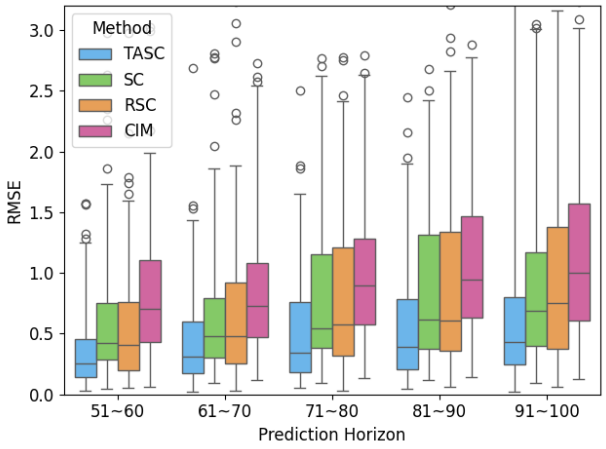

Figure 14: Post-intervention RMSE under low observation noise and large Q, evaluated across five future prediction horizons (10 time periods each)

E.3 Ablation Study on the Number of Donor Units

We tested the effect of increasing the number of donors in the dataset. We varied the number of rows in the panel data, denoted as $N = n + 1$, where n represents the number of donor units and the additional row corresponds to the target unit. The pre-intervention period was fixed at $T_0 = 50$, and the prediction horizon spanned from $T_0+1 = 51$ to $T = 100$. Figure 15 shows the post-intervention RMSE as N varies. Notably, the best performance for TASC is achieved when $N = T_0 = 50$. TASC demonstrates robustness, particularly when N is small, showing minimal performance difference between $N = 10$ and $N = 50$. Other synthetic control benchmarks also perform better when N is close to T_0, while both too many donors (e.g., $N = 200$) and too few (e.g., $N = 10$) are

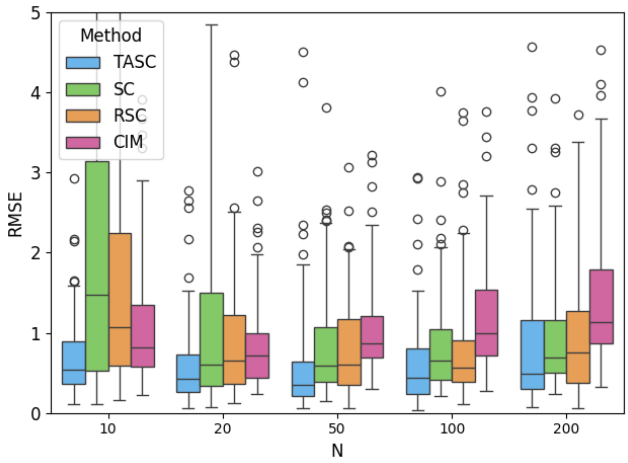

Figure 15: Post-intervention RMSE with varying N (the number of rows in panel data)

detrimental across all methods.

As N increases from 10 to 50, most methods (with the exception of CIM) exhibit improved performance, suggesting that additional training data in this range is beneficial. However, increasing N from 50 to 200 does not lead to further improvements, despite the availability of more training data. In fact, it deteriorates the performance in general. We suspect two possible explanations for this phenomenon: (1) the information content of the random data source saturates beyond a certain point, and (2) the model structure struggles to benefit from the high-dimensional input space when the number of features increases while the number of observations remains fixed (i.e., the curse of dimensionality).

F More Results from Empirical Evaluation on Real-World Data

In this section, we present more results from the empirical evaluation on real-world datasets. We compare TASC against three benchmarks used in the previous section: 1) Synthetic Control (SC, Abadie and Gardeazabal (2003)), 2) Robust Synthetic Control (RSC, Amjad et al. (2018)), and Causal Impact Model (CIM, Brodersen et al. (2015)). Section F.1 demonstrates the classical synthetic control application on evaluating Proposition 99 in California, and Sections F.2 and F.3 apply synthetic control to predict game score trajectories in cricket and basketball games, respectively.

F.1 Evaluating Effect of Proposition 99 in California

In this section, we demonstrate our method using a classic synthetic control application from Abadie et al. (2010): evaluating the effect of Proposition 99. Proposition 99 was a policy enacted in California in 1988 that significantly increased the state's cigarette tax. This policy was followed by a noticeable decline in cigarette sales (black line in Figure 16). To assess whether this decline was causally driven by the policy, synthetic control methods can be applied to estimate the counterfactual outcome for California—i.e., what cigarette sales would have looked like had the policy not been implemented. For economic analyses, multiple auxiliary predictors are often used to improve predictive accuracy. For example, Abadie et al. (2010) incorporate variables such as the average retail price of cigarettes, per capita state personal income, the percentage of the population aged 15–24, and per capita beer consumption, in addition to the target time series (per-capita cigarette sales). However, in our analysis, we intentionally focus on a single predictor, per-capita cigarette sales, to ensure a fair and consistent comparison across different methods. Our goal is not to produce the most accurate estimate of the effect of the policy, but to evaluate the performance of competing methodologies under a controlled setting.

With this in mind, RSC was implemented with the ridge coefficient 0.1 and the approximate rank $d = 2$, and TASC used the same $d = 2$ for hidden state dimension. Figure 16 shows the observation from California in black and all predictions lie above the observed trend, with the gap capturing the policy's effect. The four estimates

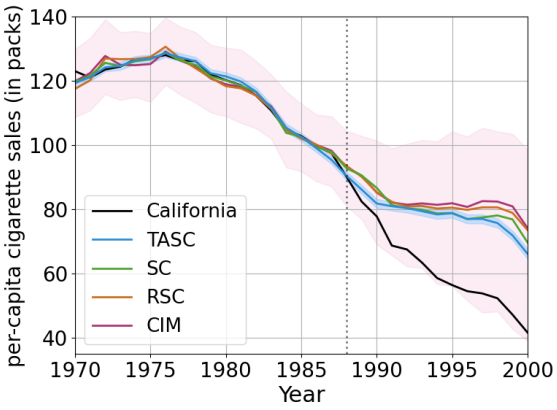

Figure 16: Per-capita cigarette sales in packs in California (black), and estimated counterfactual outcomes by TASC and other benchmarks. Blue and pink shades denote 95% confidence interval from TASC and CIM.

are broadly consistent, diverging slightly near 2000. TASC and CIM present confidence interval estimates, shown in blue and pink shaded areas, respectively. TASC's confidence interval is considerably narrower than that of CIM. Notably, CIM's interval encompasses the observed values for California, indicating that no strong causal conclusions can be drawn from this analysis.

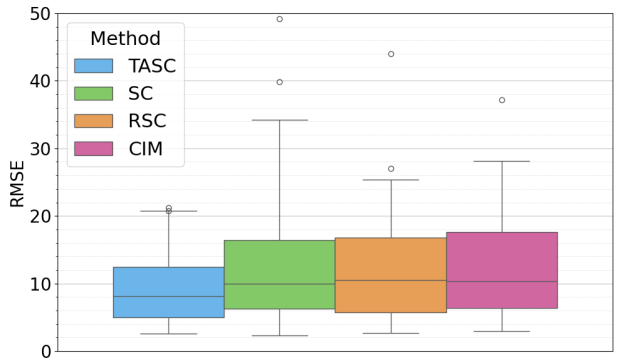

Figure 17: Post-intervention RMSE from placebo test

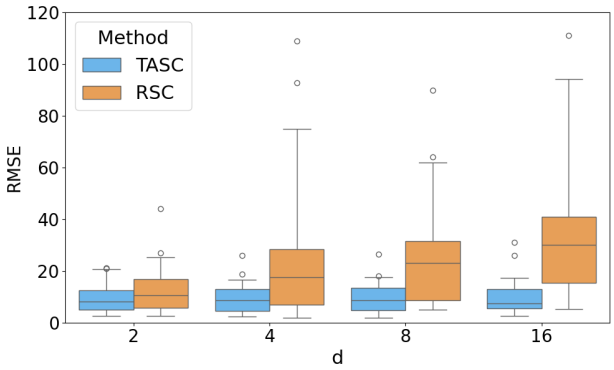

Figure 18: Post-intervention RMSE with varying d.

To evaluate the credibility of these counterfactual estimates, we conduct *placebo tests*. Since the true counterfactual is unobservable, we simulate it by treating a donor unit as if it were the target. In each placebo test, we predict a donor unit's time series using the remaining donors and assess whether the synthetic control method can accurately reconstruct the observed outcomes. Figure 17 shows post-intervention root mean squared error (RMSE) from the placebo test with TASC and other benchmarks. Among these, TASC achieves the lowest median RMSE and smallest variance of RMSE, suggesting that TASC provided the most reliable estimates among the methods tested in this experimental setting.

Next, we take a deeper dive into comparing TASC and RSC. Among the methods we tested, TASC and RSC explicitly filter the data to have *exactly* low-rank signals. As a result, both TASC and RSC require the hyperparameter d, which denotes the hidden dimension for TASC and the approximate rank for RSC. To examine how the choice of d affects performance, we evaluate both methods across different values of d. Figure 18 reports the post-intervention RMSE from the placebo test as d varies. The lowest error occurs at $d = 2$ for both methods. While RSC's performance deteriorates rapidly as d increases, TASC remains relatively stable. This aligns with the simulation results, which showed that TASC is resilient to overestimating d (Figure **??**). Hence, this reconfirms that TASC may offer advantages in settings where the true value of d is difficult to estimate.

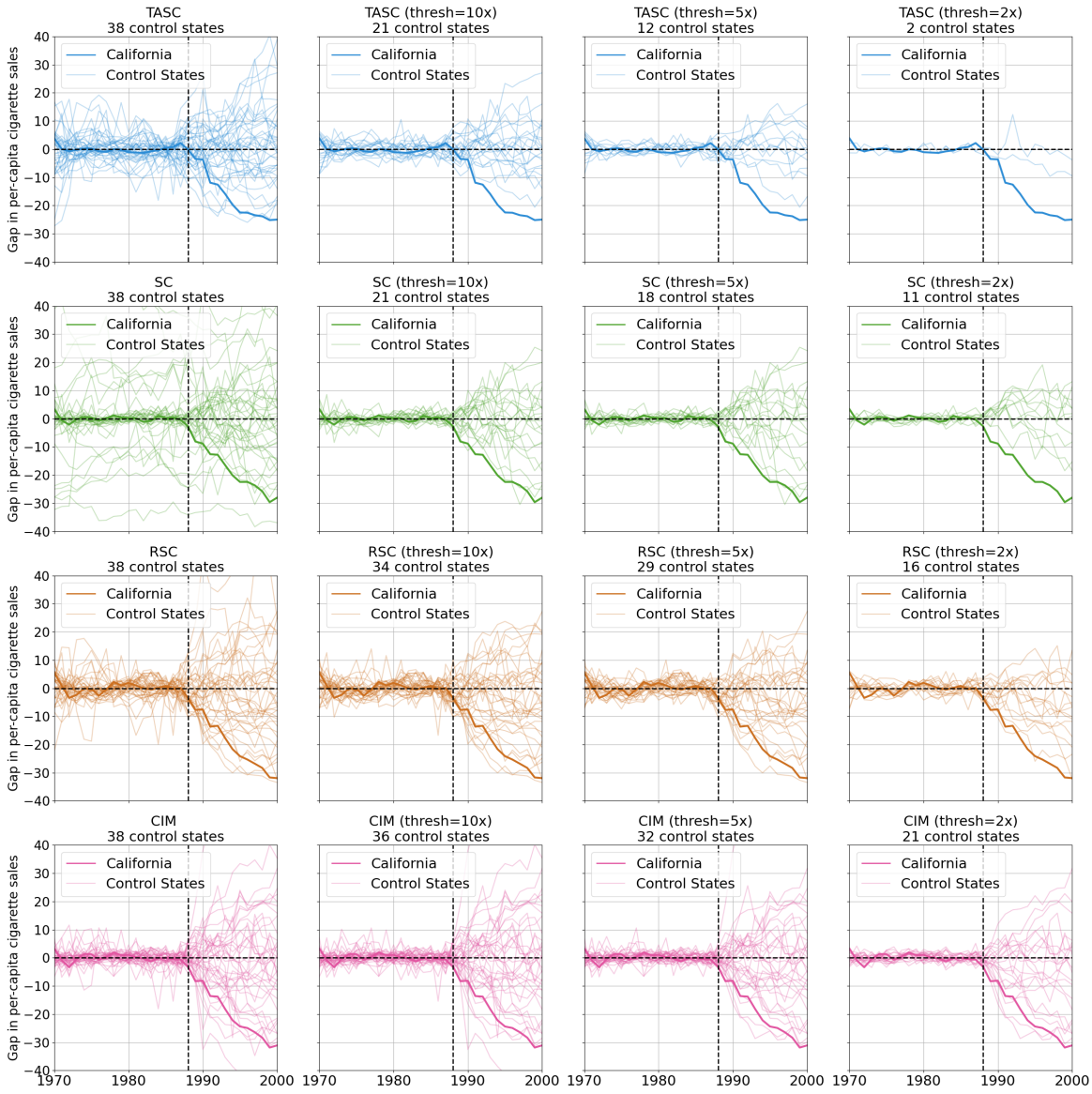

Figure 19: Gap in per-capita cigarette sales (in packs) between the observed outcome and synthetic control predictions (comparable to Figures 4–7 in Abadie et al. (2010)). Each row represents a different algorithm—TASC, SC, RSC, and CIM (from top to bottom)—while each column applies a different threshold for selecting control units: no threshold, at most 10 times California's pre-intervention error, 5 times, and 2 times (from left to right).

Lastly, following the approach from (Abadie et al., 2010), we plot the difference between the observed outcome and the predicted counterfactual for California, alongside those of the control states obtained from the placebo test. The first column of Figure 19 reports the gap between prediction and observation of per-capita cigarette sales (in packs) for California and the 38 control states. Subsequent columns restrict the set of control states based on the relative quality of pre-intervention fit, measured by mean-squared error (MSE). The second column includes only control states whose pre-intervention MSE is no more than 10 times that of California, while the third and fourth columns apply stricter thresholds of 5 and 2 times, respectively.

Notably, the last row (CIM) retains the largest number of control states as stricter thresholds are imposed, indicating that the accuracy of pre-intervention prediction was similar across states. In contrast, the TASC approach (the first row) retains substantially fewer control states under the most stringent threshold compared to SC or RSC. This indicates that the pre-intervention fit for California was more accurate than other states when using TASC. Note that California is one of the most populated states in the US, and hence the collected data (per capita cigarette sales) may have lower variance compared to other states due to averaging effect. In such case,

TASC may have learned smaller observation noise variance (R) and yielded more accurate (pre-intervention) fit for California. Indeed, corresponding variance for California was the smallest (2.58, with median 12.95, standard deviation 36.17 and maximum 170.79). Across all specifications, California consistently displays the largest gap, while it is more apparent in the plots in the top right corner (stricter thresholds, TASC method). The estimate effect of policy is similar across methods, diverging only at the end. TASC and RSC shows flatter estimates closer to 2000, whereas SC and CIM estimates keep increasing.

F.2 Score Trajectory Prediction in Cricket Games

In this section, we demonstrate our method with cricket score trajectory data from Indian Premier League (IPL). The dataset employed in this study is derived from ball-by-ball records of Indian Premier League matches from April 18, 2008 to March 25th, 2025. In the T20 format, a standard match consists of two innings of 20 overs each, with 6 balls bowled per over, and one team batting per inning. While the number of deliveries in an inning can occasionally exceed 120 due to penalty balls, this analysis restricts attention to the first 120 legal deliveries of each inning. This selected 1524 out of 2092 score trajectories in the dataset that have at least 120 balls delivered. The scores have been aggregated cumulatively over the course of each inning per ball.

To simulate a real-world use case, a target match is randomly selected from all 1524 matches, and the donor pool consists of the n most recent matches that occurred prior to the target match, with the choice from $n \in \{18, 36, 72, 144\}$. Each inning is a time series of length $T = 120$, with a fixed intervention point at $T_0 = 72$. The choice of $T_0 = 72$ is to balance pre-intervention data between the first 36 balls of power play and the play that follows. The remaining 48 deliveries ($t = T_0 + 1, \ldots, T$) are used for counterfactual inference and evaluation expecting no intervention effect (placebo test).

We tested TASC against three benchmarks, SC, RSC, and CIM. For all four methods, the data is mean-centered prior to fitting by subtracting the mean score trajectory calculated from the selected donors. For TASC and RSC, the hidden state dimension and the number of singular values to keep were both set to be $d = 5$. RSC is implemented with the ridge coefficient of 10^3, after cross-validation on the pre-intervention data using the values in $\{10^{-1}, 10^0, \ldots, 10^6\}$. For each method and donor pool size, the experiment is repeated 100 times, each time with a newly selected random target match.

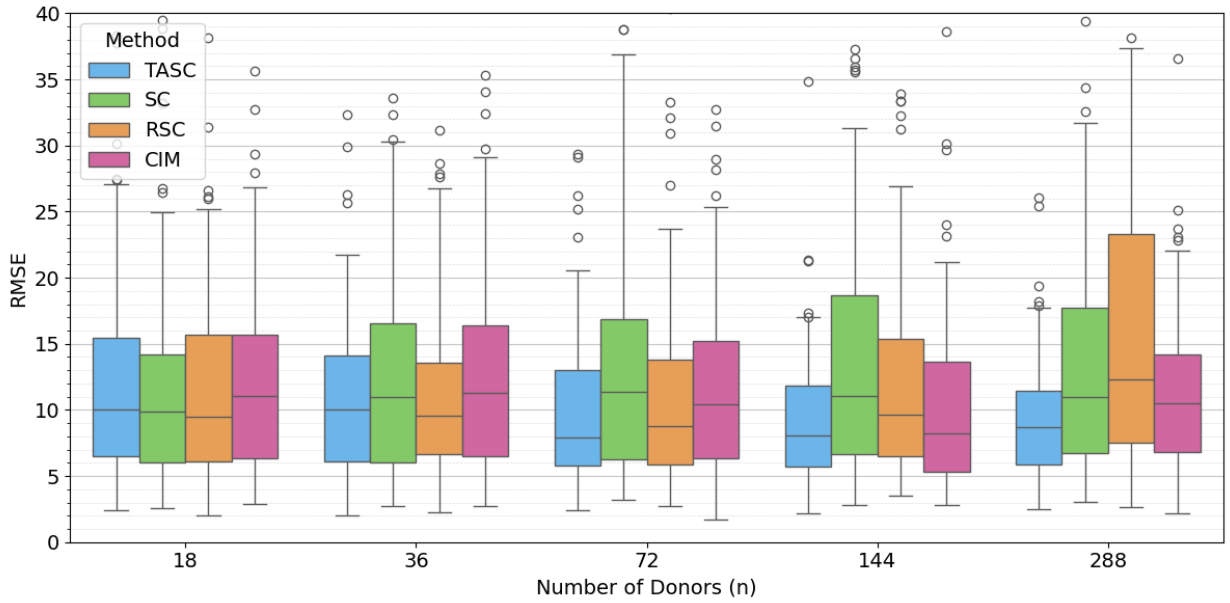

Figure 20: Overall post-intervention RMSE for TASC, SC, RSC, and CIM across varying donor sizes.

First, we investigate the effect of donor size on the performance of different methods. Figure 20 shows the overall post-intervention RMSE across different numbers of donor units n. TASC shows steady improvement as the number of donors increases up to around $n = 72$, beyond which performance stabilizes. RSC exhibits the strongest sensitivity to donor size: it achieves comparable accuracy with very $n = 18$, but suffers substantial

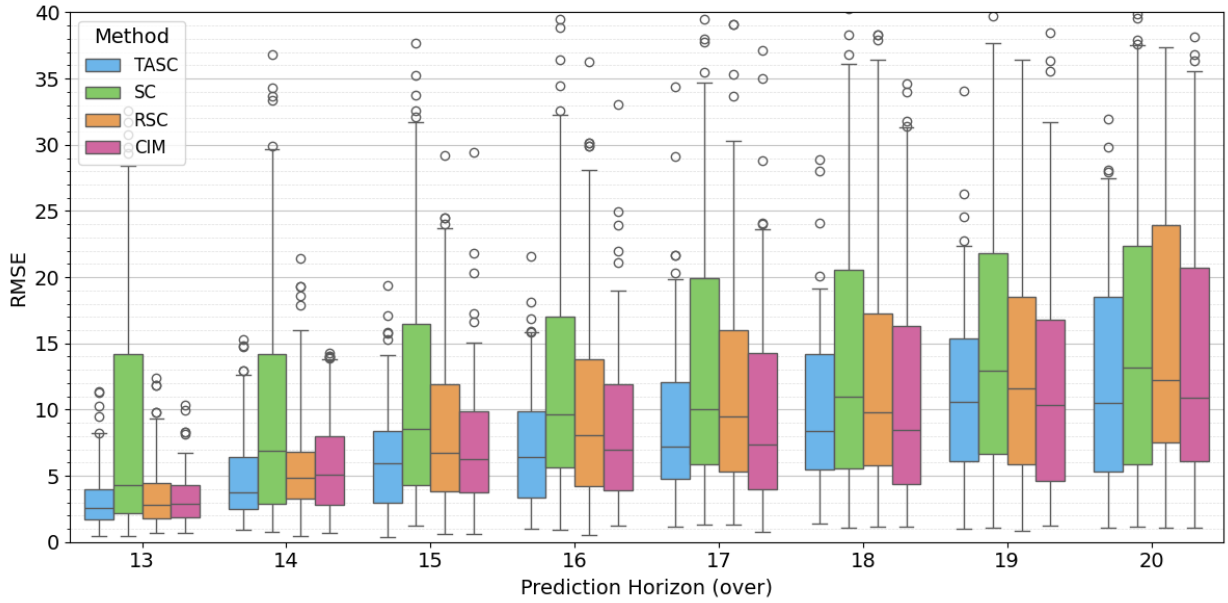

Figure 21: Per-over RMSE by prediction horizon for $n = 144$ donors.

degradation as n grows, likely due to high-dimensional overfitting. In contrast, SC, constrained to simplex weights, maintains stable performance across donor sizes. Similarly, CIM maintains performance across donor sizes, with a marginal improvement at $n = 144$. TASC achieves the lowest median RMSE across methods for donor sizes $n \in \{36, 72, 144\}$ whereas SC achieves the lowest when the donor size is $n = 18$. Across all settings, TASC with $n = 72$ donor units achieves the lowest median RMSE of 7.88.

Next, we examine how predictive performance changes over an extended time horizon. Figure 21 presents RMSE segmented by overs (6-ball intervals), illustrating how forecasting accuracy changes with increasing prediction horizons. The top plot corresponds to $n = 18$ donors, a regime where RSC and SC are competitive. RSC achieves strong short-term accuracy, but its advantage fades as horizon lengthens. SC performs with a lower accuracy in the short-term, but it performs the best as prediction horizon lengthens. The bottom panel corresponds to $n = 72$ donors, where TASC yields the most accurate predictions, with particular strength in more distant future time points. Across both settings, prediction errors grow with the forecast horizon, reflecting the increasing difficulty of long-range forecasting. The performance of TASC and CIM degrades more slowly than that of the other methods, possibly because they explicitly encode the temporal evolution of latent factors: when a learnable trend is present, explicitly modeling it improves long-range forecast accuracy.

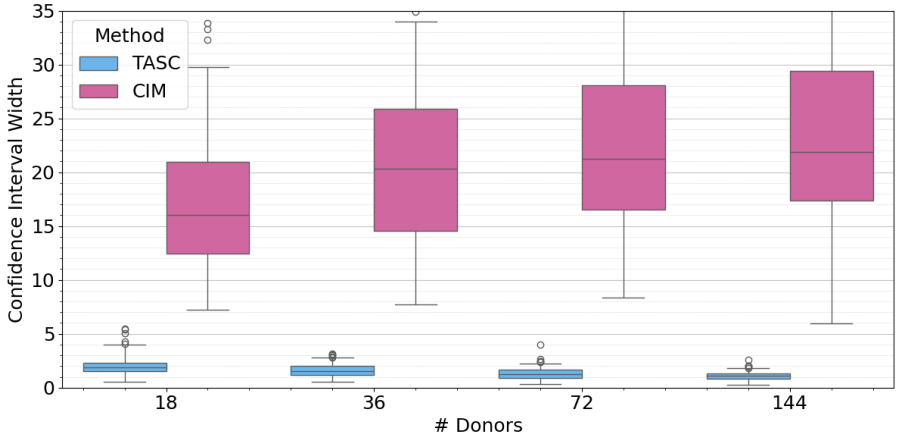

Figure 22: Post-intervention average confidence interval width for varying number of donors n

Lastly, we compare the confidence interval predictions of TASC and CIM. In terms of point estimation (mean), TASC and CIM achieved comparable performance in the short-term range, while TASC performed slightly better in long-term range predictions. Both TASC and CIM not only produce point estimates but also provide confidence interval predictions: TASC derives them from the covariance estimate of the latent state, whereas CIM obtains them from its MCMC sampler. Figure 22 displays the average 95% confidence interval width[8] of post intervention predictions from TASC and CIM, with varying number of donors. Although TASC and CIM show similar mean estimation performance, TASC tends to produce narrower confidence intervals, which may facilitate interpretation of intervention effects. This aligns with the results in Section F.1, where CIM's confidence interval included the observed outcome from California, preventing any causal conclusions from being drawn based on its estimates.

F.3 Score Trajectory Prediction in Basketball Games

In this section, we demonstrate our method with basketball score trajectory data from the National Basketball Association (NBA) games. NBA is a professional basketball league in North America founded in 1946 and currently consists of 30 teams. An NBA game consists of four quarters, each 12 minutes in duration, resulting in a total of 48 minutes of game play, excluding potential overtime periods. We use the play-by-play data from Kaggle[9] to construct a score trajectory panel dataset. The raw event data has been preprocessed into time series format, with cumulative scores sampled at uniform intervals of 15 seconds, yielding the total time series of length $T = 192$ for each game. We only focus on the first four quarters of the game, and choose the games that lasted for at least 48 minutes from January 2nd, 2020 to June 9th, 2023.

Following the same methodology as with Cricket, a target is randomly selected from all 7574 possible game score trajectories and the donor pool consists of the n most recent games that occurred prior to the target game date, with the choice from $n = \{24, 48, 96, 192\}$. Each game score trajectory is a time series of length $T = 192$ (corresponding to 48 minutes of game time measured at 15 second intervals). The intervention point is $T_0 = 96$, which corresponds to the intervention occurring at halftime. The data is mean-centered prior to fitting using selected donor data, for all four methods. Similar to the Cricket analysis, the hidden dimension for TASC and the approximate rank for RSC were both set to be $d = 5$ and RSC ridge coefficient is 10^3, after cross-validation on the pre-intervention data using the values in $\{10^{-1}, 10^0, ..., 10^6\}$. For each method and donor pool size, the experiment is repeated 100 times, each with the same set of random targets.

We test how each method handles growing number of donor data. Figure 23 shows overall post-intervention RMSE using different sizes of donor pool $n \in \{24, 48, 96, 192, 384\}$. TASC achieves the lowest median RMSE across n, with stable performance over changing number of donors. CIM shows similar robustness to increasing n, with slightly higher RMSE compared to TASC. SC and RSC are relatively more sensitive to high-dimensional learning issues with increased n, with RSC being more stable than SC. Overall, TASC and CIM yield gradually lower median RMSE as n increases, whereas SC and RSC behaves in the opposite direction.

With these results, we focus on the case when the lowest post-intervention RMSE is achieved ($n = 192$) and examine how prediction accuracy changes over extended time horizon. Figure 24 presents post-intervention RMSE segmented by half quarter intervals (6 minutes, which comprises 24 time points) with $n = 96$. In most cases, TASC yields the lowest RMSE, followed by CIM, RSC, and SC. Prediction accuracy declines as the forecast horizon increases across all methods, reflecting the inherent challenges of long-term forecasting. CIM and RSC performs comparably to TASC when predicting the near future (the first half of Q3), but the gap widens as the horizon extends, with a more pronounced difference emerging in the forecasts for the second half of Q4.

[8] The confidence interval width is computed per experiment as the average distance between the upper and lower boundaries of the confidence interval across all post-intervention time steps.

[9] `https://www.kaggle.com/datasets/wyattowalsh/basketball/data`

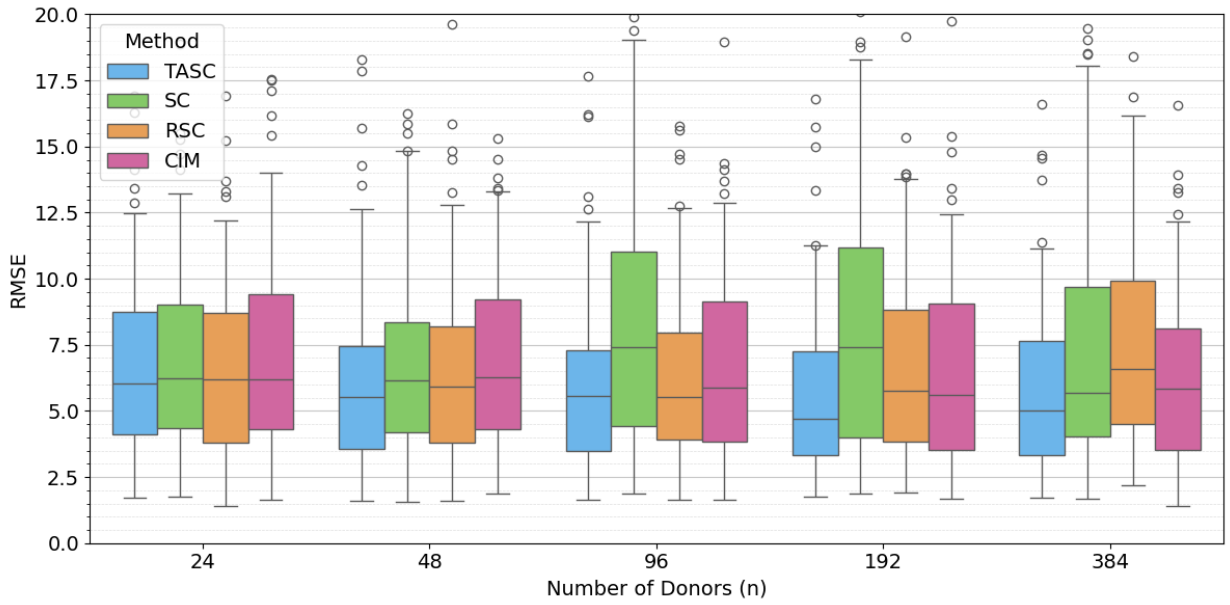

Figure 23: Overall post-intervention RMSE for TASC, SC, RSC, and CIM across varying donor sizes.

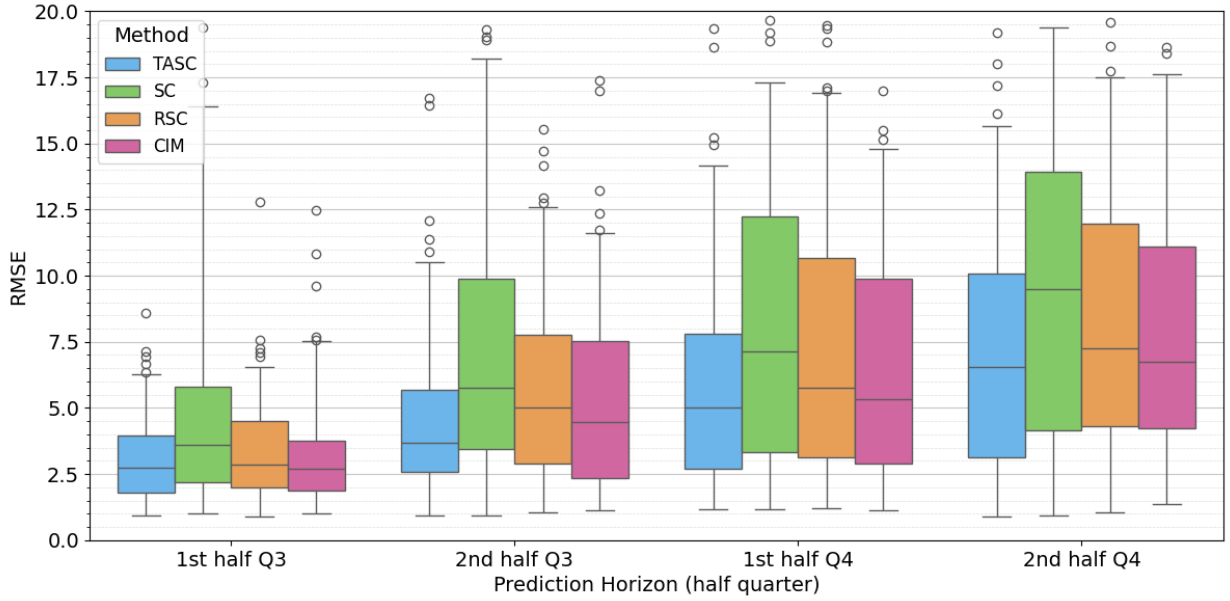

Figure 24: Post-intervention RMSE per half quarter (6 minutes) interval prediction horizon, with $n = 192$.