Perils and Pitfalls in the Use of Synthetic Control Methods to Study Public Safety Interventions*

Aaron Chalfin
University of Pennsylvania and NBER

Zubin Jelveh
University of Maryland

This version: May 30, 2024

Abstract

The method of synthetic controls, pioneered by Abadie et al. (2010), has generated a paradigm shift in the analysis of case studies. The method selects an appropriate *synthetic* comparison group by identifying a weighted set of units that closely match the treated unit on the basis of pre-intervention levels and trends. Since Abadie's seminal paper, there has been a proliferation of research expanding and refining the method and a corresponding litany of software packages that provide the means to estimate these models. We show that there can be a shocking lack of correspondence between the estimates produced by commonly used software packages. Even the seemingly innocent choice between using R or Stata to estimate SCM can lead to a meaningful difference in estimated treatment effects. We demonstrate this surprising finding, invoking a recent debate in criminological research concerning a paper on the effects of "de-prosecution" by Hogan (2022) which has been criticized by Kaplan et al. (2022).

*We are deeply indebted to John MacDonald for helpful comments on an earlier version of the draft of what later became this paper. All remaining errors are our own. Correspondence: Chalfin: achalfin@sas.upenn.edu; Jelveh: zjelveh@umd.edu.

1 Introduction

The method of synthetic controls, pioneered by Abadie et al. (2010), has led to a paradigm shift in the analysis of case studies – a research scenario in which there is a single treated unit and a large pool of potential comparison units to choose from. In evaluating the effects of a policy that is implemented in a single city or county – a common setting in criminal justice policy research – a key question is how to select a comparison group against which that city or county should be compared. In the past, researchers appealed to geographic proximity or baseline covariate overlap in order to motivate a comparison group. In other words, select a theoretically-motivated comparison group and then pray for something resembling parallel trends, the partially-testable core identifying assumption of differences-in-differences estimation.[1] A considerable virtue of synthetic controls is that it dispenses with the need for prayer, providing a roadmap to select a comparison group for which pre-intervention trends are as closely matched as possible.[2] The method is also notable for being data-driven, reducing the need for researcher discretion and therefore potentially offering a means of making case study research more reliable and less subject to the potentially devastating effects of selective reporting of results (Iyengar and Greenhouse, 1988; Ioannidis et al., 2014; Simonsohn et al., 2014) and p-hacking (Benjamin et al., 2018; Coker et al., 2021).

Due to its attractive qualities, its transparency, and its easy accessibility for applied researchers (thanks to off-the-shelf implementations for R, Stata and Python), SCM has become an increasingly popular method of causal inference in case study settings across the social sciences.[3] Within criminology, synthetic controls has been used to study the link between immigration and crime (Chalfin and Deza, 2020), the effects of police turnover (Mourtgos et al., 2022), police use of force (Goh, 2021), the impact of death penalty moratoriums (Oliphant, 2022), the effect of labor market shifts on crime (Mitre-Becerril and Chalfin, 2021), a variety of place-based interventions (Saunders et al., 2015; Robbins et al., 2017; Rydberg et al., 2018; Piza et al., 2020; Lawrence et al., 2022; Buggs et al., 2022), prosecutorial reforms (Hogan, 2022; Wu and McDowall, 2023; Zhou et al., 2023), marijuana liberalization (Wu and Cullenbine, 2022; Harper and Jorgensen, 2023) and the effect of gun control policies (Donohue et al., 2019), among other topics. Synthetic controls methods have also taken root in related social science disciplines including economics

[1] The parallel trends assumption is formally untestable as it is a counterfactual assumption about what would have happened in the absence of the intervention. However, a test of pre-intervention trends provides some assurance that treated and comparison units were not experiencing different trends prior to the intervention.

[2] As is noted in a recent working paper by Pickett et al. (2022), it is not necessarily the case that minimizing pre-intervention differences between a treatment unit and its synthetic counterpart will minimize bias. Researchers could potentially overfit by matching on noise, a problem which is intended to be addressed by using penalized regression estimators like Ridge regression (Ben-Michael et al., 2021; Abadie and L'Hour, 2021).

[3] The original paper by Abadie et al. (2010) has, to date, generated nearly 3,000 citations.

(Billmeier and Nannicini, 2013; Bohn et al., 2014; Grier and Maynard, 2016) and political science (Abadie et al., 2015; Kikuta, 2020; Gilens et al., 2021). We illustrate this observation in Figure 1 which plots changes in the number of synthetic controls papers identified using a directed Google Scholar keyword search.[4] As is evident from the figure, use of the methodology in criminology has increased markedly during the last decade and has featured prominently in some of the discipline's most highly-cited journals.

Alongside the increased use of SCM by applied researchers, there has been a corresponding proliferation of methodological research which has expanded and refined the methodology. Recent innovations have addressed a number of important issues with SCM which can arise in applied settings. For example, in some applications traditional SCM will not yield a sufficiently "good" pre-intervention match if the treatment unit's pre-intervention characteristics lie outside of the common support of the available comparison units. When this happens, SCM is potentially biased due to the absence of common trends. Bias corrected synthetic controls estimators proposed by Ben-Michael et al. (2021) and Abadie and L'Hour (2021) offer principled approaches to constructing counterfactual estimates for the treated group that extrapolate away from the pre-intervention characteristics of the control units. These approaches are also intended to make the approach more robust and to guard against the problem of overfitting where an analyst ends up matching on noise rather than true signal.

Alongside these methodological innovations are a growing number of software packages written for *R*, Stata and Python that provide applied researchers with the tools to estimate the newest synthetic controls models with relative ease. In addition to `Synth`, the original package written by the authors of Abadie et al. (2010) for both *R* and Stata, there is `AugSynth`, written for *R* and `allsynth`, written for Stata, which implement the method of bias corrected synthetic controls proposed by Ben-Michael et al. (2021) and Abadie and L'Hour (2021), respectively. There is also the `scpi` package written for *R* which allows researchers to flexibly select a means of bias correcting and which uses a method proposed by Cattaneo et al. (2021) to capture uncertainty.[5] Each of the packages offers a degree of flexibility, allowing researchers to change some of the default settings in order to test the robustness of their estimates to choices made during the research process. However, as we show, the default settings of these packages as well as choices made by the package's designers that cannot be easily changed by end users can have critically important implications for the resulting estimates.

While a full accounting of best practices in the implementation of SCM is beyond the scope of this paper – and is premature as the literature continues to evolve – we identify several different and

[4]We collected annual results from Google Scholar using the search terms "synthetic control" and "criminology".

[5]This method recognizes that there are two sources of uncertainty in SCM estimates: one that is derived from uncertainty about the weights themselves and the other that arises from sampling variability.

seemingly innocuous features of commonly-used software packages that can lead to large and substantively meaningful differences in estimated treatment effects. To our knowledge, most of these issues remain undiscussed – or, at a minimum, heavily underreported – in the extant literature. As a result, applied researchers do not appear to be aware that their estimates can be so sensitive to the seemingly arbitrary choice of a software package, let alone the role that default settings can play.

We focus, in particular, on three features that are baked into available software packages and that influence the relative weights assigned to matching variables in the process of constructing a synthetic control group. First, while discussion of SCM weights has typically been about weights that are assigned to the donor units for constructing the synthetic control group (e.g., synthetic Philadelphia is comprised of 50% NYC, 30% New Orleans and 20% Detroit), there is a second set of weights that matters in SCM – the weights that are assigned to the variables that describe each unit. That is, in generating a synthetic control group, how should each covariate be weighed in relation to each other as well as in relation to pre-intervention values of the outcome variable?[6] We explore the way in which these covariate weights (commonly referred to in the literature as $\mathbf{V}$-weights) are assigned. We show that some software packages optimize these weights to facilitate the closest possible match, while differing in the objective function optimized. Other software packages take a more agnostic approach and assign uniform weights to all covariates.

The second software feature that can influence the relative weighting of covariates is the manner in which the covariates are rescaled. To ensure that differences in scale (i.e., each variable will have a different mean and degree of dispersion around that mean) do not influence the construction of the synthetic control group, two of the packages we study rescale by dividing the value of each matching variable by its standard deviation while another package rescales all covariates to the variance of the outcome. One of the packages we study, `scpi`, does not rescale covariates at all. We investigate the extent to which seemingly innocuous package design choices related to covariate scaling can impact the resulting estimates.

The third source of software variation we consider are differences in how bias correction – a methodological innovation which potentially makes better use of covariates – is implemented in the most recent SCM estimators. In the two packages we study that implement bias correction (`AugSynth` which is an implementation of Ben-Michael et al. (2021) and `allsynth` which implements a version of bias correction that is closely related to that of Abadie and L'Hour (2021)), the user must specify an estimation method for the outcome model which governs how all matching variables are used to improve pre-intervention balance. While Ben-Michael et al. (2021) motivate Ridge regression as a principled way of extrapolating away from the

[6] In this paper, we use the term *covariate* to refer to non-outcome variables that are used to construct the synthetic control. We will use the phrase *matching variables* when collectively referring to covariates *and* pre-intervention values of the outcome.

non-penalized SCM weights, each package allows users to select from a menu of options including penalized estimators like Ridge and Lasso regression. Interestingly, Stata's `allsynth` package uses ordinary least squares (OLS) as its default setting meaning that a user must manually select the option for penalization. As it turns out, this can have profound implications for the resulting estimates when this package is used.

Across all of the software studied, we demonstrate that the default settings and the choices made by the authors of each package that cannot be easily edited by end users can matter a great deal. Especially troubling is that `AugSynth`, written for *R*, and `allsynth`, written for Stata, which both purport to implement the very similar bias corrected methods of Ben-Michael et al. (2021) and Abadie and L'Hour (2021) can generate markedly different estimates even for the original synthetic controls model proposed by Abadie et al. (2010). A typical end user would not realize this unless he or she checked their work by running the `Synth` package. These differences between packages are magnified further when they are used to perform bias correction in SCM, which is their primary purpose.

We demonstrate these findings by invoking a recent debate in criminological research concerning a paper on the effects of "de-prosecution" in Philadelphia by Hogan (2022) which has been criticized in a response paper by Kaplan et al. (2022). Though the debate between the authors covers a wide range of empirical issues, the central point of contention is the claim made by Kaplan et al. (2022) that the large estimated treatment effect – a 30% increase in homicides by 2019– in Hogan (2022) goes away, even reversing in sign, when a different variant of synthetic controls estimation is employed.

While the purpose of this paper is not to adjudicate the debate between the two sets of authors, we re-estimate models presented by Hogan (2022) and Kaplan et al. (2022) and show that estimated treatment effects do, in fact, vary considerably depending upon the software package that is used and the method that is employed to do bias correction. Critically though, while we use these data to demonstrate the sensitivity of SCM to researcher (and software implementer) degrees of freedom, we note that the sensitivity of the estimates need not lead us to throw out SCM with the bathwater. While there is little theory to guide some choices made by package designers, we use insights from recent theoretical work on the use of covariates in synthetic controls to argue that some analytic choices are better motivated than others. With respect to the paper by Hogan (2022), we find that, conditional on the available data and the credibility of the identification strategy, a considerable majority of the model space, including that which is consistent with the modest recommendations that we make in this paper, is in line with a notable increase in homicides in Philadelphia, relative to other U.S. cities, beginning in 2015.[7]

[7]We note that an examination of how different packages conduct inference on synthetic control estimates is beyond the scope of this paper, thus our work does not speak to the statistical significance of the range of estimates presented here.

2 SCM and Statistical Software

In this section, we describe several fundamental differences across four software implementations of SCM, including bias corrected SCM. We begin with a basic outline of the method and a description of a particularly common setting in which the method has been applied. We then briefly describe several important extensions and refinements of SCM that have been made in the past few years and introduce the various software packages that have been written to implement these methods.

2.1 The Synthetic Control Method

Consider a dataset which contains pre- and post-intervention information for one treated unit and a donor pool of N comparison units that have not been subject to any intervention. The dataset spans T time periods with T_0 denoting the number of periods prior to the intervention. For each control unit, n, we observe Y_{nt}^0, or the value of the outcome variable at time t. The same quantity for the treated unit is denoted as Y_t^1. Without loss of generality, we assume in this section that the number of post-treatment periods is one. In order to estimate the treatment effect τ_T for the treated unit, we would ideally compute:

$$\tau_T = Y_T^{(I)} - Y_T^{(U)} \tag{1}$$

In Equation 1, I represents the observed outcome for the treated unit and U represents the unobserved (untreated) potential outcome for the treated unit. The goal of SCM is to identify a sparse vector of weights for each unit in the donor pool, $\mathbf{W}$, or $(w_1,\cdots,w_N)$, that best approximates the counterfactual outcome values for the treated unit. We also note that the identification assumptions for SCM also require good pre-intervention fit on covariates that influence the outcome. This counterfactual is referred to as the "synthetic" control unit as it does not represent a unit that exists in reality, but rather a weighted average over the actual untreated units. The following counterfactual can then be used to estimate the treatment effect:

$$\hat{\tau}_T = Y_T^1 - \sum_{n=1}^{N} w_n Y_{nT}^0 \tag{2}$$

Abadie et al. (2010) show that when the data generating process can be described by a linear factor model, this treatment effect estimate is unbiased as the number of pre-intervention periods grows. However, in most empirical applications the number of pre-intervention periods used for constructing a synthetic control is far from infinity, a point which we will later turn to in our discussion of reducing specification search.

To best approximate the counterfactual outcome value of the treated unit, the standard SCM approach of Abadie et al. (2010) proposes to identify a set of weights that minimize the difference between pre-intervention characteristics of the treated unit and the weighted sum of the same values for the control units. These characteristics include some function of the pre-intervention values of the outcome variable, but it is possible to include other matching variables ("covariates") as well. For the control units, we observe a set of K matching variables, X^0_{n1},...,X^0_{nK}.[8] The matching variables for the treated unit are denoted as X^1_1,...,X^1_K. The structure of the dataset that serves as input into the SCM method is then:

$$\left[\begin{array}{ccc|ccc} X^1_1 & \cdots & X^1_K & Y^1_{T_0+1} & \cdots & Y^1_T \\ \hline X^0_{11} & \cdots & X^0_{1K} & Y^0_{1(T_0+1)} & \cdots & Y^0_{1,T} \\ \vdots & & \vdots & \vdots & & \vdots \\ X^0_{N1} & \cdots & X^0_{NK} & Y^0_{N(T_0+1)} & \cdots & Y^0_{N,T} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{X^1} & \mathbf{Y^1} \\ \hline \mathbf{X^0} & \mathbf{Y^0} \end{array}\right]$$

where $\mathbf{X^1}$ is the 1 x K vector of pre-intervention values for the treated unit, $\mathbf{Y^1}$ is the 1 x $(T-T_0)$ vector of post-intervention outcome values for the treated unit, $\mathbf{X^0}$ is the N x K matrix of pre-intervention values for the control units, and $\mathbf{Y^0}$ is the N x $(T-T_0)$ matrix of post-intervention outcomes values for the control units.

The objective function that SCM minimizes is:

$$\left(\sum_{k=1}^{K} v_k \left(X^1_k - w_1 X^0_{1k} - \cdots - w_N X^0_{Nk}\right)^2\right)^{\frac{1}{2}} \tag{3}$$

In Equation 2, v_k is a vector of weights that captures the relative importance of each matching variable in estimating the pre-intervention outcome values for the treated unit. A uniform $\mathbf{V}$-vector would mean that each matching variable is equally relevant for predicting these pre-intervention outcome values. A non-uniform $\mathbf{V}$-vector requires a data-driven method of identifying a set of matching variable weights.

During the optimization process, each of the donor unit weights, w_n, is constrained to be greater than or equal to zero and the weights must sum to one. These constraints have two important effects. First, they produce weights that are relatively interpretable. Second, they produce a synthetic control that does not extrapolate away from the support of the data. Avoiding bias due to extrapolation does not, however, mean that the optimization process is unbiased. SCM with non-extrapolating weights may still exhibit bias from interpolation (Kellogg et al., 2021) as a result of overfitting to matching variables that are not genuinely predictive of the outcome ("noise") or if the relationship between the pre-intervention predictors for the treatment and control units is not linear, as is assumed by SCM.

[8] A common approach taken by researchers is to set the first T_0 of the K matching variables as the pre-period values of the outcome.

2.2 Recent Extensions to SCM

We next describe two related extensions of SCM proposed by Abadie and L'Hour (2021) and Ben-Michael et al. (2021) that are relevant for our discussion below. Each of these extensions represents an attempt to improve the robustness of SCM by better leveraging the presence of additional covariates and utilizing penalized regression methods and cross-validation to reduce the likelihood of interpolation bias. Each of the extensions is implemented by different software packages for R and Stata which, as we show, can lead to substantively different outcomes some of which are due to choices made by package designers that are seemingly innocuous and indeed unrelated to the ways in which the bias correction is being done.

2.2.1 Bias Correction and Penalization

A synthetic control model with inadequate pre-treatment alignment can both bias the estimate of the treatment effect and call into question the validity of the synthetic unit as a credible counterfactual. A proposed remedy involves correcting for bias from poor alignment by adjusting the treatment effect estimated from standard SCM via the following procedure which utilizes the concept of penalization.

First, a model that forecasts the post-period values of the outcome for the donor pool is estimated:

$$\mathbf{Y^0} = f(\mathbf{X^0}) + \epsilon^0 \tag{4}$$

where $f(\cdot)$ is the function to be learned, typically referred to as the "outcome model", and ϵ represents the approximation error. Then the bias-corrected estimate of the counterfactual outcome is computed as:

$$\hat{Y}_T^{(N)} = \sum_{n=1}^{N} \hat{w}_n Y^0 + \left(\hat{f}(\mathbf{X}^1) - \sum_{n=1}^{N} \hat{w}_n \hat{f}(\mathbf{X}_n^0) \right) \tag{5}$$

Under the assumption that f is consistent for $E[\mathbf{Y}|\mathbf{X}]$, the second term on the right-hand side of equation 5 represents the discrepancy between the actual and potential outcome due to remaining imbalance in the matching variables (Abadie and Imbens, 2002).

Both Abadie and L'Hour (2021) and Ben-Michael et al. (2021) propose methods for bias correction, with the latter showing that bias-correction with an outcome model that uses Ridge regression can be reformulated as a synthetic control problem with a penalty term that penalizes deviations from the unregularized synthetic control weights. Thus, while the method proposed by Abadie and L'Hour does not alter the original SCM weights, since it is formulated as a synthetic control problem, the method proposed by Ben-Michael et al. allows for the updating of the SCM weights.[9]

[9]Another difference between the two methods is in the cross-validation procedure used to find the optimal value

2.2.2 The Use of Non-Outcome Matching Variables

As mentioned previously, one of the identification assumptions of synthetic controls is that there is good pre-period balance on relevant non-outcome covariates. This intuition is derived from a result in Abadie et al. (2010) which shows that for a linear factor model of the form, $Y_{nt} = \delta_t + \theta_t Z_n + \lambda_t \mu_i + \epsilon_{nt}$ – where δ are time shocks shared by all units, $\mathbf{Z}$ are unit-specific observed covariates, and μ are unobserved factor loadings – the optimal weights found by SCM must balance pre-period outcomes and covariates, $\mathbf{Z}$. This insight is further explored empirically by Pickett et al. (2022) who, using a series of simulations, find that including covariates tends to lead to reduce bias in treatment effect estimation – thus leading to the recommendation to match on non-outcome covariates in empirical applications.[10]

In short, covariate balance matters. Yet, as is demonstrated by Kaul et al. (2022), common approaches for optimizing the covariate weights, $\mathbf{V}$, will place zero, or very little, weight on non-outcome covariates, even if, in reality, those variables exert a strong independent influence on the outcome. In this paper, in addition to considering the impact of differences in software, we also build upon prior work by investigating the sensitivity of pre-period fit and estimated treatment effects to the ways in which covariates are entered into the SCM optimization process. We do so using the debate between Hogan (2022) and Kaplan et al. (2022) over the effects of "de-prosecution" on homicides in Philadelphia.

2.3 Software Implementations

Having briefly introduced several extensions of SCM, we now describe the differences in how four software packages have implemented the original synthetic controls estimator of Abadie et al. (2010) and the recent extensions proposed by Abadie and L'Hour (2021) and Ben-Michael et al. (2021). We focus on the following software packages:

1. `Synth` (R/Stata): This package implements the original synthetic controls estimator proposed by Abadie et al. (2010).

for the penalty term, λ. Since it is reformulated as an SCM problem, the cross-validation approach employed by Ben-Michael et al. finds the penalty value that minimizes error between the treated unit's values of pre-intervention matching variables and the predicted values from the synthetic control. The approach in Abadie and L'Hour computes error between the actual and predicted post-intervention outcome values of the donor pool units.

[10] A related contribution by Botosaru and Ferman (2019) shows that a lack of covariate balance leads to bias and that bounding this bias requires many pre-intervention time periods, an uncommon occurrence in many criminal justice applications.For this reason, researchers will sometimes rely on theory to identify non-outcome matching variables that will be used to construct a synthetic control group. In considering the implications of this expansion of the number of researcher degrees of freedom, Ferman et al. (2020) show how the lack of guidance in choosing matching variables leaves open the room for specification search and p-hacking even when the pre-intervention imbalance is small and recommend that researchers show results from a number of specifications.

2. `allsynth` (Stata) developed by Wiltshire (2022): This package adds functionality to the `Synth` Stata package, updating it to allow researchers to estimate the bias-correction for synthetic controls proposed by Abadie and L'Hour (2021).

3. `AugSynth` (R): This package provides an implementation of the augmented (bias corrected) synthetic controls method proposed by Ben-Michael et al. (2021) that is written by the authors.

4. `scpi`[11] (*R*/Stata/Python): This package, developed by Cattaneo et al. (2022), is focused on improved techniques for uncertainty quantification, which is out of the scope of the present work. Instead we focus on the component of the package that estimates standard and penalized SCM weights.

We focus on the following implementation choices and how they vary across each of the four software packages discussed above:

- **Rescaling**: If matching variables are on different scales, then, other things equal, changes in the value of the objective function will be driven more by changes in variables with relatively larger, rather than smaller, scales. For example, height measured in inches would, other things equal, receive more weight than height measured in feet. To avoid bias induced by scale differences, it is common to standardize predictors so that they are on the same scale. As shown in the third column of Table 1, software packages differ in how this scaling is performed and even whether it is performed at all. The `Synth` and `allsynth` packages standardize all predictors so that their variances equal one (unit variance). The `AugSynth` package instead rescales non-outcome matching variables so that they have the same variance as the outcome values for the donor pool. The `scpi` package does not perform any rescaling of matching variables.

- **Covariate weights**: In the canonical version of SCM, Abadie et al. (2010) describe a procedure that selects $\mathbf{V}$ weights so that the "mean squared prediction error of the outcome variable is minimized for the pre-intervention periods" for the resulting synthetic control. However, Abadie (2021) recommends that "researchers should aim to demonstrate that their results are not overly sensitive to particular choices of $\mathbf{V}$." As we show below, the values of $\mathbf{V}$ weights are critical in driving differences in treatment effect estimates in the replication of Hogan (2022) by Kaplan et al. (2022). We focus on two common approaches for setting these weights in the available software packages:

    1. **Uniform weights**: Uniform $\mathbf{V}$ weights mean that each covariate receives equal weight in generating the synthetic controls weight of each donor pool unit. This is the default option

[11] scpi is an acronym for "Synthetic Control Prediction Intervals."

for `AugSynth` and `scpi`.

2. **Regression weights**: The default option for the Stata version of `Synth`, these weights are estimated by regressing a matrix of pre-intervention outcomes on a matrix of pre-intervention matching variables for all units. [12,13]

3. **Nested Optimization**: The default option for the *R* version of `Synth`, this approach involves finding the $\mathbf{V}$ weights which minimize pre-intervention imbalance on the outcomes. The default approach for the *R* version of `Synth` uses the regression-derived weights as input into the nested-optimization procedure. As shown in the fourth column of Table 1, the `allsynth` and `Synth` packages use regression weights, while the *R* version of `Synth` uses nested optimization. Finally, `AugSynth` and `scpi` use uniform $\mathbf{V}$ weights and thus do not have the ability to optimize $\mathbf{V}$.

- **Bias Correction**: Column five of Table 1 notes that `allsynth` and `AugSynth` are the two packages which implement bias correction with two important differences across the packages. First, each algorithm has a different set of available outcome models. Second, each of the packages differs in how the outcome models are used.

3 Empirical Application

We demonstrate the sensitivity of SCM estimates to seemingly innocuous choices made by applied researchers including the statistical software used (using *R* versus Stata) and the choice of a software package (e.g., `AugSynth` versus `allsynth`), invoking a recent empirical debate in criminology. The debate concerns a 2022 paper – that of Hogan (2022) – published in *Criminology & Public Policy*, the flagship policy journal of the American Society of Criminology. The paper studies the effect of "de-prosecution" on homicides in Philadelphia and shows that homicides rose by approximately 30% after 2015, the year in which Hogan pinpoints the beginning of what has since become a large decline in felony prosecutions in the city. To identify a comparison group for Philadelphia, Hogan used SCM to identify a synthetic comparison group for Philadelphia – a weighted average of US cities which, prior to 2015, had similar homicide trends but did not have a "progressive prosecution" regime. In Hogan's paper, the comparison group for Philadelphia is

[12]This latter matrix also includes a constant term. The regression produces a matrix of coefficients, $\beta_{k,t}$ where the rows represent each matching variable (K) and the columns represent the number of pre-intervention time periods (T_0). The final regression-based estimates for $\mathbf{V}$ are computed as $\frac{\sum_t \beta_{k,t}^2}{\sum_t \sum_k \beta_{k,t}^2}$.

[13]Following Pickett et al. (2022), who recommend against the use of this approach for selecting weights, we do not report results for this approach. We did not find meaningful differences in the results presented in this paper between using regression weights versus nested optimization.

comprised of three cities: Detroit, New Orleans and NYC. Hogan's paper uses five years of pre-intervention data (spanning 2010-2014) and, in addition to pre-intervention measures of homicide, includes three additional covariates: city population, the number of cleared homicides and the homicide clearance rate.

In Figure 2, we present Figures 3 and 4 from Hogan (2022) which present the relevant SCM analyses for the number of homicides in Philadelphia compared to comparison cities. In the figure, the top panel plots the number of homicides in Philadelphia (solid line) against the estimated number of homicides in synthetic Philadelphia (dashed line). The bottom panel is the corresponding placebo plot, with Philadelphia represented using the red line and each donor unit represented by a gray line. According to the analysis, by 2019, four years after the year which Hogan asserts represents the beginning of the intervention, Philadelphia is estimated to have experienced approximately 100 more homicides than its synthetic counterpart.

Shortly after the publication of Hogan (2022), Kaplan et al. (2022) released a working paper which claims to call into question the robustness of Hogan's results. The paper, which was made available via Twitter in July 2022, takes issue with a number of choices made by Hogan (2022) but the central claim in the paper is that the large estimated treatment effect goes away, even reversing in sign, when bias corrected SCM – which Hogan (2022) did not report – is employed.[14] We investigate this claim within a broader framework of using the application to demonstrate the sensitivity of SCM estimates to different packages and settings within those packages.

We use data from Kaplan et al. (2022) who have publicly posted their replication data. We note that we are able to successfully replicate the results reported in the paper.[15] Since we are using the data not to litigate the dispute between the two sets of authors but to make a broader point about the sensitivity of SCM analysis to package design choices, we simplify the exercise and focus on a subset of the data and re-estimate SCM models that condition on three sets of variables: pre-intervention values of the number of homicides (the outcome variable), population, and the number of homicides cleared by an arrest.[16] We re-estimate the models varying 1) the statistical software (Stata versus *R*), 2) the package (`Synth`, `AugSynth`,

[14] Kaplan et al. (2022) also criticizes the length of the pre-intervention period that is used to generate a synthetic match, the choice of start date of the intervention and the use of counts as opposed to rates in comparing across cities. In his reply to Kaplan et al. (2022), Hogan addresses those issues and raises some potential issues with the correctness of clearances and clearance rates used in the replication, suggesting that there may be important data errors. Each of these issues have been litigated extensively by the authors in their replies to one another.

[15] For completeness, we also obtained the original data from Hogan (2022) and replicate key results in Appendix Figure 2 and Appendix Figure 3. The primary discrepancy between the Kaplan et al. (2022) version of the dataset and the Hogan (2022) dataset is that there are differences in the clearance variable. Hogan, in a response to Kaplan et al. (2022), notes that some of the clearance measures for NYC have zero values and that clearance measures for two other relevant cities exhibit unexpected variability. As is evident from Figure 3, when the Hogan data are used, the bias corrected estimates are more homogenous. However, referring to Figure 2, when covariates are employed, estimates continue to vary considerably across different packages.

[16] Adding the homicide clearance rate to the models does not substantively alter the findings.

`allsynth` and `scpi`), 3) whether the $\mathbf{V}$ matrix which generates weights for each pre-intervention time period is optimized or not, 4) the method of rescaling that is used and, where applicable, 5) for the bias corrected estimates, the outcome model used to perform bias correction (OLS versus Ridge regression).

4 Empirical Estimates

4.1 Off-the-Shelf Results

Using the replication data of Kaplan et al. (2022), Figure 3 presents estimates of the effect of de-prosecution in Philadelphia on homicides using the canonical SCM estimator of Abadie et al. (2010). In each panel of the figure, the blue line plots estimates using *R*'s `Synth` package, the red line plots estimates using *R*'s `AugSynth` package and the green line plots estimates using *R*'s `scpi` package. Note that each of the packages reports estimates that purport to conform with the SCM estimator of Abadie et al. (2010) so that researchers can compare newer SCM estimates to the original SCM estimates. The left-hand panel presents estimates when the matching variables include only annual counts of homicides from the pre-intervention period (i.e., matching only on the pre-intervention values of the outcome variable). With no covariates employed, we observe identical performance across the four software implementations.

In the remaining panels, we present estimates which condition on additional covariates including the population, and cleared cases and with both covariates entered simultaneously. The addition of covariates generates striking differences in the estimates. For example, the inclusion of population as a covariate leads to greater imbalance for the `scpi` package, while the inclusion of case clearances causes both the `scpi` package and the `AugSynth` package to generate estimates that differ markedly from the baseline estimates using the `Synth` package of Abadie et al. (2010). The final column presents results when all three additional matching variables are included together. As is evident from the figure, all four packages lead to different pre-period fits and, critically, to substantively different estimated treatment effects. We emphasize that while we are presenting estimates from four different packages, three of which have greater functionality than the original `Synth` package, the additional functionality is not being called upon to generate the estimates in Figure 3. Each of the packages is being used *only to recover results from the synthetic controls estimator* of Abadie et al. (2010). The fact that different packages, written for different statistical software, lead to substantively different SCM estimates is, in our view, striking and motivates an investigation into the drivers of these discrepancies.

We next consider whether different software packages that implement bias corrected synthetic controls – which uses penalization to guard against overfitting – also lead to substantively different estimates. To

do so, we run off-the-shelf bias-correction models.[17] The results from this experiment are presented in Figure 4. The first row of plots presents estimates from Stata's `allsynth` package while the bottom row presents estimates for *R*'s `AugSynth` package. The columns demonstrate the sensitivity of the estimates to the inclusion of covariates. Within each plot, the different colored lines present results when applying no bias correction (red), bias correction using OLS (green), and Ridge bias-correction (blue), respectively.

While the estimates are extraordinarily similar when no non-outcome covariates are included and when only population is included as a covariate, we observe a substantial divergence in the estimates for specifications that include case clearances. Notably, the `allsynth` package's OLS bias correction model yields substantially lower treatment effect estimates than when no bias correction or Ridge bias correction is used. On the other hand, using `AugSynth`, OLS and Ridge bias correction lead to reasonably similar estimates. When both population and case clearances are included as covariates, there is a particularly striking difference between the estimates produced by `allsynth` and `AugSynth` even when both packages use Ridge regression in the outcome model. Whereas the `allsynth` package generates a large positive estimate that is extrapordinarily similar to the original SCM estimator of Abadie et al. (2010) and to the estimates reported by Hogan (2022), the `AugSynth` package generates an estimate that is close to zero.

The divergence in the results are substantively large as different models lead to either a large treatment effect (+ 30% homicides by 2019) or no effect at all. Some of the differences are easily explained by the properties of the models employed – for instance, bias correction via Ridge regression addresses the issue of overfitting but bias correction via OLS does not. However, other results do not lend themselves to a straightforward explanation. Why should `AugSynth`, written for *R*, and `allsynth`, written for Stata, yield such different estimates even when Ridge regression is employed to bias correct in both cases? In order to further investigate the possible causes of these discrepancies, we run a series of experiments where we vary which implementation choices an algorithm has access to. A specification in these experiments is defined by software package, rescaling method, $\mathbf{V}$ weight optimization approach, the set of matching variables allowed, the subset of pre-intervention years to include for each variable, whether bias correction is performed, and whether a penalty term is included when running standard SCM.

4.2 Explaining the Discrepancies

We investigate the discrepancies in estimates from different packages by systematically varying the features of each package. In order to allow a software package to have access to a particular implementation feature, we modify the source code of the three *R* packages to add options that were not initially present. For

[17] We note that, unlike `allsynth`, `AugSynth` does not have an ordinary least squares outcome model, therefore to equate the options provided by both packages, we modify the `AugSynth` source code to add an OLS outcome model.

example, we modified `AugSynth`'s source code so that it would be able to rescale matching variables to unit variance and we altered `Synth` so that it would be able to rescale to the outcome variance. We alter `scpi` to allow it run models with access to both types of rescaling. This allows us to test whether setting the algorithms to rescale data in the same way can explain the discrepancies we observe above. Due to differences in the code base for each algorithm we did not attempt to equalize all packages along all dimensions.[18]

These experiments help us account for many of the discrepancies presented above and further allow us to provide several recommendations when estimating treatment effects for standard and bias-corrected synthetic controls. In what is to follow, we present results from investigating differences in rescaling, estimating $\mathbf{V}$ weights, performing bias correction, and operationalizing how covariates are included in SCM.

4.2.1 Rescaling

To explore the impact of a package's rescaling method on balance and estimates, we modify the source code of each of the three *R* packages so that they are able to either rescale *all matching variables* to either unit variance or to rescale *non-outcome matching variables* to have the same variance as the outcome. We follow `AugSynth`'s implementation and compute the variance of the outcome across all (as opposed to within) pre-intervention years and then rescale the non-outcome matching variables to this value. To account for the differences in $\mathbf{V}$ weight estimation across the packages, we present estimates for specifications where uniform $\mathbf{V}$ weights are employed, turning to the impact of $\mathbf{V}$ weight estimation on balance and treatment effects in the next section.

As above, we vary the set of matching variables that are used to run each package's version of SCM to estimate the impact of "de-prosecution" on homicides in Philadelphia. In Figure 5, the rows represent each of the three software packages used (`AugSynth`, `scpi` and `Synth`) while the columns represent the matching variables used in the specification (outcome only, population, clearances and all of the above). In each plot, the red line represents estimates from specifications in which the data were scaled to the variance of the outcome variable while the cyan line shows the same set of estimates when all covariates were rescaled to unit variance.

We call attention to two main results. First, referring to the top row of the figure, while suffering from the largest variances in pre-period imbalance, estimates from `AugSynth` are not sensitive to changes in the rescaling method. [19,20] Second, we observe that rescaling, which `scpi` does not do when used

[18] For example, while we were able to allow `scpi` to use optimized $\mathbf{V}$ weights, the relatively complex structure of `AugSynth`'s code increased the possibility of introducing a bug.

[19] Appendix Figure 1 shows that this is also true when we perform bias correction with `AugSynth` code.

[20] We are unable to confirm if the same property holds for `Synth`, which rescales all variables so that their variances equal one, as the algorithm fails before completion when clearances that are rescaled to outcome variance are included.

off-the-shelf, generally brings `scpi`'s estimates in line with the other two algorithms. This finding indicates that failing to rescale the matching variables can be a meaningful driver of diveregences in the estimates.

4.2.2 Covariate weights

Next, we explore the impact of **V**-weight optimization on balance and estimates. Here, we modify the source code for the `scpi` package so that the optimal **V** weights found by `Synth`'s optimization routine are used as an input when running `scpi`. We likewise modified the source code of `Synth` to accommodate uniform **V** weights. As `AugSynth`'s code base was relatively more complex, we did not alter its code to accept optimized **V** weights in order to avoid the greater risk of introducing a bug.

In Figure 6, the rows represent the software package used while the columns represent the rescaling method that was employed. Estimates are presented for the specification that uses all matching variables including pre-treatment values of the outcome variable, population and case clearances. In each plot, the red line represents estimates from specifications where the matching variable weights were optimized using `Synth`'s nested optimization procedure, while the cyan line represents runs where the **V** weights were uniform across variables.

The figure establishes that optimizing the **V** weights helps `Synth` maintain a very low degree of pre-intervention imbalance under both types of rescaling. Optimizing **V** weights also helps `scpi` minimize imbalance when variables are scaled to variance one, but greater levels of imbalance still exist with **V** weight optimization and rescaling to outcome variance. As we saw in Figure 5, uniform weights lead to poor imbalance and in the case of outcome variance, a failed run for `Synth`. The good pre-period fit for `Synth`, however, may be a side effect of how pre-intervention outcomes are entered into the models, a feature that we investigate in the next section of the paper.

The result of the previous two experiments combined suggest some causes for the discrepancies observed in Figure 3. First, scaling matters. The results for the `scpi` alter significantly when either unit or outcome rescaling is applied. While this is not surprising, the fact that scaling appears to interact with matching variable inclusion, even when **V** weights are optimized, indicates that researchers should pay close attention to how different settings for all three of these parameters impact estimates.

4.2.3 Using Covariates as Matching Variables

As mentioned above, Kaul et al. (2022) show that when all pre-period outcomes are used as matching variables, the **V** weights for covariates (i.e. non-outcome matching variables) will be driven to zero – even when some or all of the included covariates influence the outcome. Consistent with this result, we

demonstrated in the previous section that when we run `Synth` and `scpi` with optimized $\mathbf{V}$ weights and unit variance rescaling, and when we include all matching variables, we obtain the same estimates as when only pre-intervention outcomes are used as matching variables.

We now stress test this result by altering how matching variables are included in the estimation process. While some authors include functions of matching variables (such as the average value of the outcome over the pre-period years), we focus on the common practice of including every pre-intervention period for all matching variables, including the outcome variable. Specifically, we modify the three *R* packages so, aside from including all years, matching variables can also be entered sparsely. For simplicity, we create a sparse "every other year" specification in which we use matching variable information for two pre-period years: 2011 and 2013. By not using all pre-intervention outcomes, we aim to allow the $\mathbf{V}$ weight optimization to place weights on covariates. Note that this setting is applied uniformly across all matching variables.

In Figure 7, the rows represent software packages and the columns represent the matching variables used in each specification. In order to isolate the importance of differences in matching variables used, all of the models were run with rescaling set to unit variance and with matching variable weights optimized. In each plot, the red and blue lines represent estimates when all and every other year (2011 and 2013), respectively were included for each matching variable in the specification.

When clearances are included in the match, standard SCM produces different pre-period fits and treatment effect estimates. While worse pre-period fit may suggest a worse counterfactual and bias, recall that the assumptions of SCM include good pre-period fit on covariates as well. In this specification, the average absolute error (imbalance) for clearances in the pre-intervention period is 12 for the "every other year" specification but jumps to 119.9 for the "all years" specification. The imbalance in homicides during the pre-intervention years is much closer between the two specifications: The "all years" specification has an average absolute error of 4.8 while the "every other year" specification has an average absolute error of 12.2 homicides per year. However, we see from the second column that when all matching variables are included there is negligible difference between the two specifications studied in this section.

The results in this section underscore the importance of careful consideration of how matching variables are constructed for inclusion in SCM. Notably, our results emphasize that the traditional focus on minimizing outcome imbalance in the pre-period may lead to higher imbalances for covariates. This perspective is further substantiated by the work of Pickett et al. (2022), , whose simulation studies reveal a low correlation between outcome imbalance and bias in treatment effect estimates.

4.2.4 Bias Correction

Thus far we have shown that the off-the-shelf results vary as a result of the rescaling method used and whether $\mathbf{V}$ weights are optimized. Ben-Michael et al. (2021) argue that bias correction should potentially be employed when pre-intervention balance is not achieved. We now explore whether bias correction can homogenize the estimated treatment effects across different specifications or whether this feature of software packages itself can lead to important differences in estimates.

In Figure 8, each panel presents estimates for a combination of software package used and the type of bias correction that is employed. For simplicity, we only include the specification which matches on all covariates and we focus on the `Synth` and `allsynth` packages, both of which optimize the $\mathbf{V}$ weights and utilize unit rescaling. In other words, the only remaining difference between the models is the manner in which bias correction is employed. We modified `Synth`'s code to perform bias correction in the vein of Abadie and L'Hour (2021).[21]. In each plot, the red lines represent estimates when no bias correction is done, the green lines represent estimates which use OLS to perform bias correction and the blue lines represent estimates when Ridge regression is used to perform bias correction. The figure shows that the manner in which bias correction is performed can matter a great deal. A few findings are notable. First, using the `allsynth` package in Stata, the estimates become much smaller when OLS is used to perform bias correction, a finding which we address in the next paragraph. Second, there is a large discrepancy between the Ridge bias corrections for `Synth` and `allsynth`. This is puzzling given that these two implementations should, in theory, be nearly identical.

Given the large variations in the bias corrected estimates using different software packages, we further investigate the choice of a model to do the bias correction, focusing in particular on the choice between OLS and Ridge regression.[22] We provide an empirical demonstration of this issue, using data from Kaplan et al. (2022) and Stata's `allsynth` package which was used in the Kaplan et al. (2022) paper.

We begin by estimating the original SCM of Abadie et al. (2010) and bias corrected SCM model using baseline data: the 2010-2014 values of the outcome variable, homicides, and the 2010-2014 values of city population. Next, we add fifteen "signal-free" covariates, each of which is normally distributed with a mean of 58 and a standard deviation of 76, to conform with the distribution of the outcome variable. Because the

[21]We implemented OLS and ridge regression outcome models for `Synth`. We trained separate models to predict each year of donor pool outcomes using the matching variables available in a specification. We then predicted for both the treated unit and the donor pool and applied Equation 5.

[22]The `allsynth` package in Stata offers four estimators to specify the outcome model: OLS, Ridge regression, Lasso regression, and elastic net regression. Of the four options, the latter three provide a means to address the problem of overfitting by subjecting OLS to a penalty function that either makes the model sparser (Lasso) or shrinks the resulting coefficients (Ridge, elastic net). Cross-validation is used to ensure that the model predicts out of sample.

covariates are randomly generated, by construction, they have no value in predicting homicides and should not be used to generate a synthetic match for Philadelphia. We investigate how the inclusion of these signal-free covariates affects SCM estimates generated using either OLS or Ridge regression to do bias-correction.

Results are presented in Figure 9 which presents estimates from Kaplan et al. (2022) with and without the addition of fourteen signal-free covariates. The lefthand panel presents estimates where the only matching variables are pre-intervention measures of the outcome variable and population using no bias correction (red), OLS bias correction (green) and Ridge bias correction (blue). As is evident from the figure, without additional covariates, all three versions of bias correction return similar estimates. In the other panel, we add the fifteen signal-free random normal matching variables. Here, we see that when OLS is employed to do bias correction, the presence of the signal-free covariates leads to a different estimate – an indicator that the model is overfitting to covariates that, in reality, offer no true signal. However, when Ridge regression, which uses penalization to distinguish between true signal and noise, is employed, the estimates remain unchanged. Consistent with the intuition of Abadie and L'Hour (2021) and Ben-Michael et al. (2021), the exercise suggests that penalization is an important feature of bias correction.

While this analysis does not establish that the use of OLS to bias correct will lead to similarly large differences in estimates in all cases, the inability of OLS to distinguish signal from noise in our generic scenario suggests that applied researchers should exercise consider considerable caution in reporting bias corrected SCM estimates that use OLS as a bias correction model. This is especially critical as Stata's `allsynth` package uses OLS as the default setting for bias correction.

5 Discussion

In this paper, we show that seemingly trivial differences in the software packages that are used by applied researchers to estimate synthetic controls models can drive meaningful differences in estimated treatment effects. We show this in an empirical context which is broadly relevant to researchers in criminology, where studying the impacts of policies carried out by aggregated units such as cities and coutnies is commonplace. `AugSynth` and `scpi`, two common R packages that are used to estimate synthetic controls do not optimize $\mathbf{V}$ weights meaning that matching variable weights in the pre-intervention period are constrained to be equal. On the other hand, the `Synth` package, available in both R and Stata as well as the `allsynth` package for Stata optimize these $\mathbf{V}$ weights, allowing some matching variables to be weighted more heavily than others in deriving a synthetic comparison group. The packages also differ with respect to how covariates and lagged values of the outcome variable are scaled. Researchers who are using either `AugSynth`, written for R or `allsynth`, written for Stata, to estimate the bias corrected synthetic

controls models of Ben-Michael et al. (2021) and Abadie and L'Hour (2021) must deal with several additional challenges. First, the differences between the `AugSynth` and `allsynth` packages discussed above continue to hold. Second, the packages differ with respect to how cross-validation is done. Finally, researchers must choose a model to do the bias correction. We provide evidence that OLS, the default option for `allsynth`, is not an appropriate choice.

Referring to the debate between Hogan (2022) and Kaplan et al. (2022) about the effects of "de-prosecution" in Philadelphia, we show that choices by package designers and applied researchers can lead to substantively and strikingly different estimates of the impact of de-prosecution on homicides. In order to summarize the extent of the variability, in Figure 10 we provide a specification curve (Simonsohn et al., 2020) plotting the estimated SCM treatment effect using a variety of software packages and varying the choice of covariates, rescaling, $\mathbf{V}$ weights and method of bias correction.[23] The curve plots estimated treatment effects as of 2019, the final post-intervention year studied by Hogan (2022).[24] An initial review of the figure shows that different incarnations of SCM can lead to estimates that range from an increase in homicides of more than 100 (+6.3 per 100K population) to increases that are close to zero, a difference which is clearly large enough to be relevant to policy discussions around the paper. Referring to the figure, over 90% of the model space leads to an estimate of at least 50 additional homicides (+3.2 per 100K population) in Philadelphia in 2019, compared to its synthetic counterpart. Among the models that point to much smaller estimated treatment effects, these specifications are notable for their inclusion of the case clearances variable, a measure which has been a topic of debate between Kaplan et al. (2022) and Hogan (2022). Models estimated using `AugSynth`, which uses uniform $\mathbf{V}$ weights, also tend to be smaller than models estimated using `allsynth` which uses optimized $\mathbf{V}$ weights.

In this final section of the paper, we provide context for these findings and offer several targeted suggestions for applied researchers. We begin by offering some insight into the debate between Hogan (2022) and Kaplan et al. (2022). Using Stata's `allsynth` package, Kaplan et al. (2022) shows that Hogan's substantive result – that there were more homicides in Philadelphia after 2015 compared to other large US cities in the donor pool – falls apart when the augmented synthetic controls model of Ben-Michael et al. (2021) is used. The advantage of using Stata's `allsynth` package here is that it uses the same package design as the `Synth` package which means that package design choices like $\mathbf{V}$ weight optimization, variance rescaling and optimization routines will not affect the estimates. However, it is

[23] As we limit the focus of this paper on estimation and not inference, Figure 10 differs from the specification curve procedure outlined in Simonsohn et al. (2020) in that we do not perform significance tests on the overall specification curve.

[24] In Appendix Figure 4, we present an analog to Figure 10, using data from Hogan (2022). The general pattern in the results is similar but even fewer of the estimates, using his clearance rate variable, are smaller than +50 homicides.

important to note that the bias corrected synthetic control model that most directly challenges Hogan's result uses OLS as a means of bias correction. Given the simulation we present in Figure 9, we believe that this model should be regarded with considerable skepticism. In Kaplan et al. (2022), when Ridge regression is used to bias correct, the estimates are similar to those of Hogan (2022). Interestingly though, *R*'s `AugSynth` package does, in fact, lead to smaller treatment effect estimates even when Ridge regression is used to bias correct. The magnitude of Hogan's estimates thus does appear to depend on the manner of bias correction, including the method of cross-validation, as well as whether the $\mathbf{V}$ weights were optimized.

More generally, our advice is as follows. First, we recommend that researchers begin by estimating the original SCM of Abadie et al. (2010), in particular to optimize $\mathbf{V}$ weights using nested optimization, options that are available in the `Synth` package (in either *R* or Stata) or the `allsynth` package in Stata. These estimates will reflect the version of SCM proposed by Abadie et al. (2010). Second, it is important to note that Rs `AugSynth` package implements a different variant of SCM than the one that was proposed in Abadie et al. (2010). That package does not optimize the $\mathbf{V}$ weights and also rescales the variances of control variables using a different procedure. As a result, estimates generated using `AugSynth`, while highly valuable when there is a poor pre-intervention match, should nevertheless be compared with caution to estimates using `Synth` to generate inferences about the value of bias correction. Third, understanding the importance of matching on non-outcome covariates, researchers should consider specifying a model in which the outcome model is matched more sparsely (i.e., not using all pre-intervention values to match), in order to allow covariates to be properly matched, particularly if there is theoretical justification for including covariates. Finally, in cases in which researchers believe a bias correction model will be helpful (understanding that there is a tradeoff between interpolation and extrapolation), researchers should exercise caution in interpreting results in which OLS, the default method of `allsynth`, is used to bias correct as it is not robust to overfitting. Above all, understanding that theory offers incomplete guidance about how synthetic controls models should be estimated, we recommend that synthetic controls papers include a specification plot (or similar Steegen et al. (2016) specification analysis tools) which allows readers to visually check whether estimates are sensitive to the use of different software packages and the researcher decisons that they interact with.

References

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

_ , Alexis Diamond, and Jens Hainmueller, "Synthetic control methods for comparative case

studies: Estimating the effect of California's tobacco control program," *Journal of the American statistical Association*, 2010, *105* (490), 493–505.

_ , _ , **and** _ , "Comparative politics and the synthetic control method," *American Journal of Political Science*, 2015, *59* (2), 495–510.

_ **and Guido Imbens**, "Simple and bias-corrected matching estimators for average treatment effects," 2002.

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

Ben-Michael, Eli, Avi Feller, and Jesse Rothstein, "The augmented synthetic control method," *Journal of the American Statistical Association*, 2021, *116* (536), 1789–1803.

Benjamin, Daniel J, James O Berger, Magnus Johannesson, Brian A Nosek, E-J Wagenmakers, Richard Berk, Kenneth A Bollen, Björn Brembs, Lawrence Brown, Colin Camerer et al., "Redefine statistical significance," *Nature human behaviour*, 2018, *2* (1), 6–10.

Billmeier, Andreas and Tommaso Nannicini, "Assessing economic liberalization episodes: A synthetic control approach," *The Review of Economics and Statistics*, 2013, *95* (3), 983–1001.

Bohn, Sarah, Magnus Lofstrom, and Steven Raphael, "Did the 2007 Legal Arizona Workers Act reduce the state's unauthorized immigrant population?," *The Review of Economics and Statistics*, 2014, *96* (2), 258–269.

Botosaru, Irene and Bruno Ferman, "On the role of covariates in the synthetic control method," *The Econometrics Journal*, 2019, *22* (2), 117–130.

Buggs, Shani A, Daniel W Webster, and Cassandra K Crifasi, "Using synthetic control methodology to estimate effects of a Cure Violence intervention in Baltimore, Maryland," *Injury Prevention*, 2022, *28* (1), 61–67.

Cattaneo, Matias D, Yingjie Feng, and Rocio Titiunik, "Prediction intervals for synthetic control methods," *Journal of the American Statistical Association*, 2021, *116* (536), 1865–1880.

_ , _ , **Filippo Palomba, and Rocio Titiunik**, "scpi: Uncertainty Quantification for Synthetic Control Methods," *arXiv preprint arXiv:2202.05984*, 2022.

Chalfin, Aaron and Monica Deza, "Immigration enforcement, crime, and demography: Evidence from the Legal Arizona Workers Act," *Criminology & Public Policy*, 2020, *19* (2), 515–562.

Coker, Beau, Cynthia Rudin, and Gary King, "A theory of statistical inference for ensuring the robustness of scientific results," *Management Science*, 2021, *67* (10), 6174–6197.

Donohue, John J, Abhay Aneja, and Kyle D Weber, "Right-to-carry laws and violent crime: A comprehensive assessment using panel data and a state-level synthetic control analysis," *Journal of Empirical Legal Studies*, 2019, *16* (2), 198–247.

Ferman, Bruno, Cristine Pinto, and Vitor Possebom, "Cherry picking with synthetic controls," *Journal of Policy Analysis and Management*, 2020, *39* (2), 510–532.

Gilens, Martin, Shawn Patterson, and Pavielle Haines, "Campaign finance regulations and public policy," *American Political Science Review*, 2021, *115* (3), 1074–1081.

Goh, Li Sian, "Did de-escalation successfully reduce serious use of force in Camden County, New Jersey? A synthetic control analysis of force outcomes," *Criminology & public policy*, 2021, *20* (2), 207–241.

Grier, Kevin and Norman Maynard, "The economic consequences of Hugo Chavez: A synthetic control analysis," *Journal of Economic Behavior & Organization*, 2016, *125*, 1–21.

Harper, Alexis J and Cody Jorgensen, "Crime in a time of cannabis: Estimating the effect of legalizing marijuana on crime rates in Colorado and Washington using the synthetic control method," *Journal of Drug Issues*, 2023, *53* (4), 552–580.

Hogan, Thomas P, "De-prosecution and death: A synthetic control analysis of the impact of de-prosecution on homicides," *Criminology & Public Policy*, 2022.

Ioannidis, John PA, Marcus R Munafo, Paolo Fusar-Poli, Brian A Nosek, and Sean P David, "Publication and other reporting biases in cognitive sciences: detection, prevalence, and prevention," *Trends in Cognitive Sciences*, 2014, *18* (5), 235–241.

Iyengar, Satish and Joel B Greenhouse, "Selection models and the file drawer problem," *Statistical Science*, 1988, pp. 109–117.

Kaplan, Jacob, JJ Naddeo, and Tom Scott, "De-prosecution and death," 2022.

Kaul, Ashok, Stefan Klößner, Gregor Pfeifer, and Manuel Schieler, "Standard synthetic control methods: The case of using all preintervention outcomes together with covariates," *Journal of Business & Economic Statistics*, 2022, *40* (3), 1362–1376.

Kellogg, Maxwell, Magne Mogstad, Guillaume A Pouliot, and Alexander Torgovitsky, "Combining matching and synthetic control to tradeoff biases from extrapolation and interpolation," *Journal of the American statistical association*, 2021, *116* (536), 1804–1816.

Kikuta, Kyosuke, "The Environmental Costs of Civil War: A Synthetic Comparison of the Congolese Forests with and without the Great War of Africa," *The Journal of Politics*, 2020, *82* (4), 1243–1255.

Lawrence, Daniel S, Bryce E Peterson, Lily Robin, and Rochisha Shukla, "The impact of correctional CCTV cameras on infractions and investigations: A synthetic control approach to evaluating surveillance system upgrades in a Minnesota prison," *Criminal Justice Policy Review*, 2022, *33* (8), 843–869.

Mitre-Becerril, David and Aaron Chalfin, "Testing public policy at the frontier: The effect of the $15 minimum wage on public safety in Seattle," *Criminology & Public Policy*, 2021, *20* (2), 291–328.

Mourtgos, Scott M, Ian T Adams, and Justin Nix, "Elevated police turnover following the summer of George Floyd protests: A synthetic control study," *Criminology & Public Policy*, 2022, *21* (1), 9–33.

Oliphant, Stephen N, "Estimating the effect of death penalty moratoriums on homicide rates using the synthetic control method," *Criminology & Public Policy*, 2022.

Pickett, Robert EM, Jennifer Hill, and Sarah K Cowan, "The Myths of Synthetic Control: Recommendations for Practice," 2022.

Piza, Eric L, Andrew P Wheeler, Nathan T Connealy, and Shun Q Feng, "Crime control effects of a police substation within a business improvement district: A quasi-experimental synthetic control evaluation," *Criminology & Public Policy*, 2020, *19* (2), 653–684.

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

Rydberg, Jason, Edmund F McGarrell, Alexis Norris, and Giovanni Circo, "A quasi-experimental synthetic control evaluation of a place-based police-directed patrol intervention on violent crime," *Journal of Experimental Criminology*, 2018, *14*, 83–109.

Saunders, Jessica, Russell Lundberg, Anthony A Braga, Greg Ridgeway, and Jeremy Miles, "A synthetic control approach to evaluating place-based crime interventions," *Journal of Quantitative Criminology*, 2015, *31*, 413–434.

Simonsohn, Uri, Joseph P Simmons, and Leif D Nelson, "Specification curve analysis," *Nature Human Behaviour*, 2020, *4* (11), 1208–1214.

_ , Leif D Nelson, and Joseph P Simmons, "P-curve: a key to the file-drawer.," *Journal of Experimental Psychology: General*, 2014, *143* (2), 534.

Steegen, Sara, Francis Tuerlinckx, Andrew Gelman, and Wolf Vanpaemel, "Increasing transparency through a multiverse analysis," *Perspectives on Psychological Science*, 2016, *11* (5), 702–712.

Wiltshire, Justin C, "allsynth:(Stacked) synthetic control bias-correction utilities for Stata," Technical Report, Working paper 2022.

Wu, Guangzhen and Roarke R Cullenbine, "Recreational marijuana legalization and drug-related offenses in Washington State: an interrupted time series analysis with a combination of synthetic controls," *Journal of Experimental Criminology*, 2022, pp. 1–26.

Wu, Sishi and David McDowall, "Does Bail Reform Increase Crime in New York State: Evidence from Interrupted Time-Series Analyses and Synthetic Control Methods," *Justice Quarterly*, 2023, pp. 1–29.

Zhou, Angela, Andrew Koo, Nathan Kallus, Rene Ropac, Richard Peterson, Stephen Koppel, and Tiffany Bergin, "Synthetic Control Analysis of the Short-Term Impact of New York State's Bail Elimination Act on Aggregate Crime," *Statistics and Public Policy*, 2023, (just-accepted), 1–26.

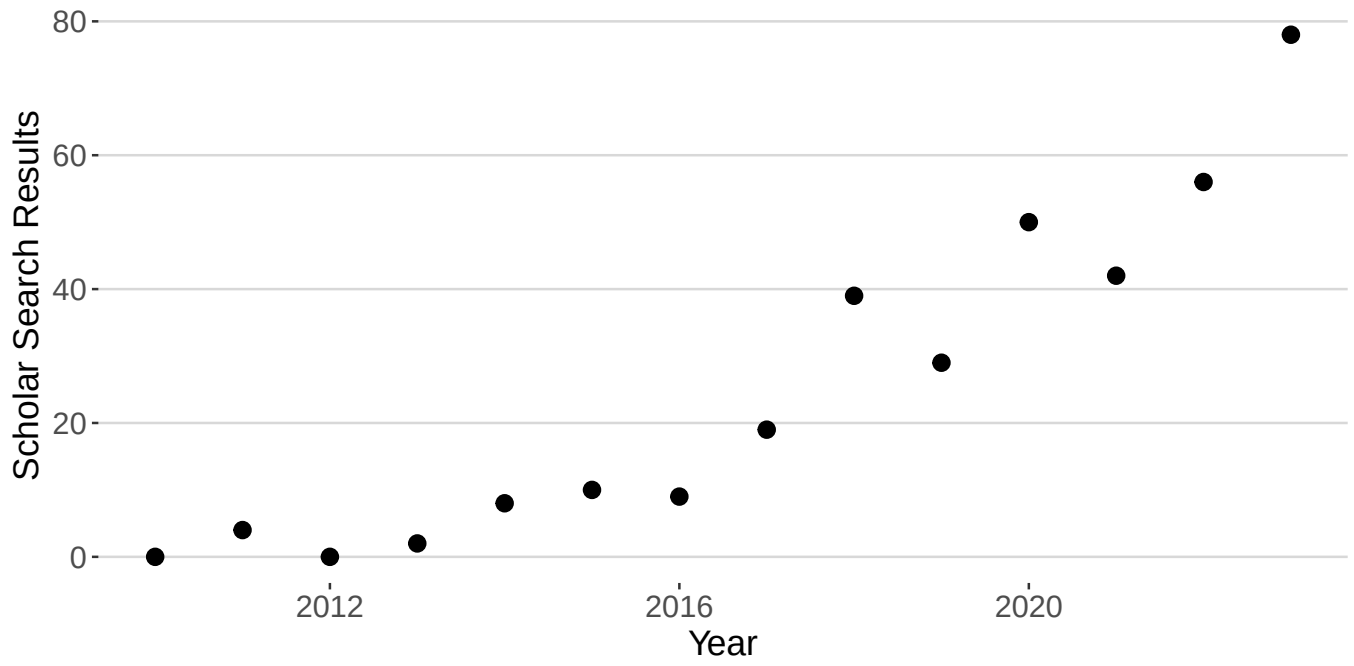
Table 1: Software Implementations

(1)	(2)	(3)	(4)	(5)
Package	Software	Scaling	**V** weights	Outcome Models
allsynth	Stata	Unit	Regression	OLS/Ridge*
Synth	R	Unit	Regression/Optimized	N/A
AugSynth	R	Outcome	Uniform	Ridge*
SCPI	Python/Stata/R	No Scaling	Uniform	N/A

Note: Table shows the default options that are available for four software packages that implement the Abadie et al. (2010) version of SCM.

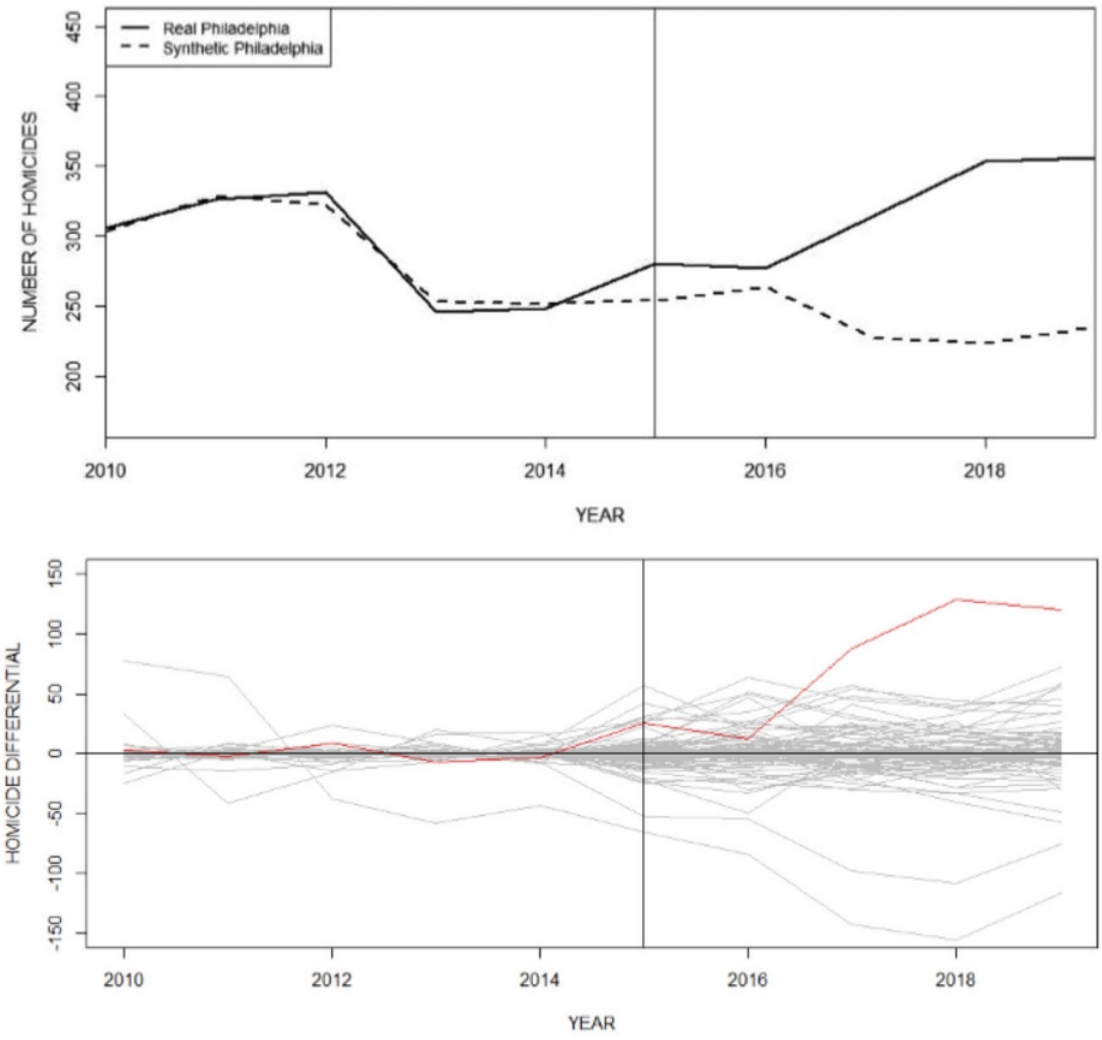
* `AugSynth` and `allsynth` have additional outcome models that are not tested as part of this paper.

Figure 1: Longitudinal Trends in the Publication of Synthetic Controls Papers



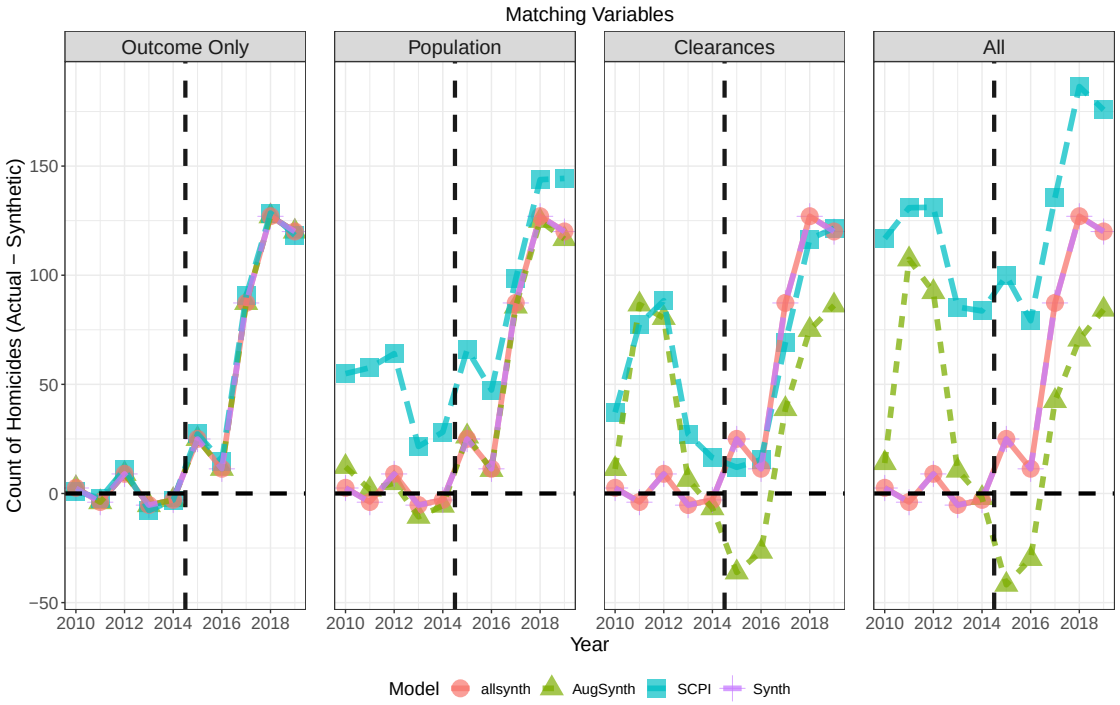
Note: Figure plots the number of social science publications in each year that utilize the method of synthetic controls, identified using a Google Scholar search.

Figure 2: SCM Estimates From Hogan (2022)



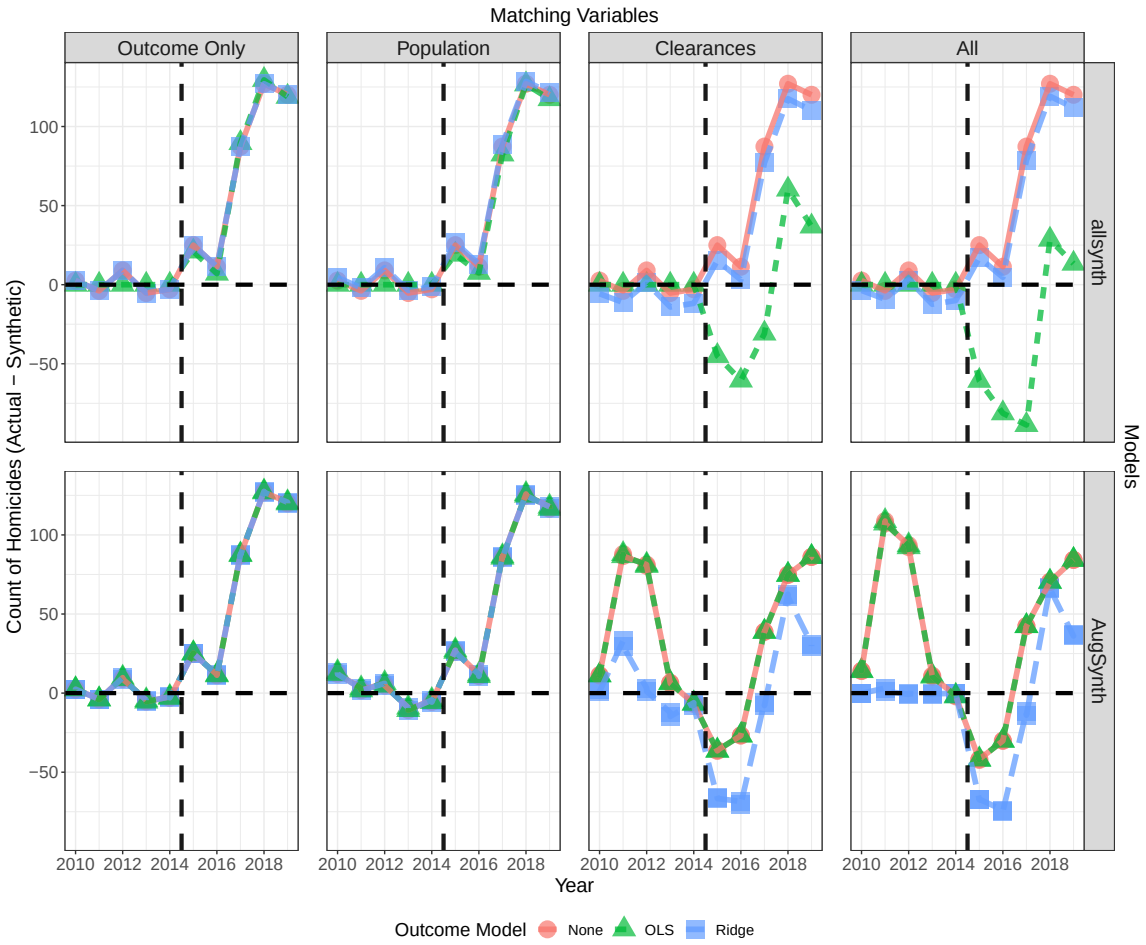
Note: Figure reproduces Figures 3 and 4 from Hogan (2022). The top panel plots the number of homicides in Philadelphia (black line) against the estimated number of homicides in synthetic Philadelphia (dashed line). The bottom panel is the corresponding placebo plot, with Philadelphia represented using the red line and each donor unit represented by a gray line.

Figure 3: Variance in Standard SCM Estimates



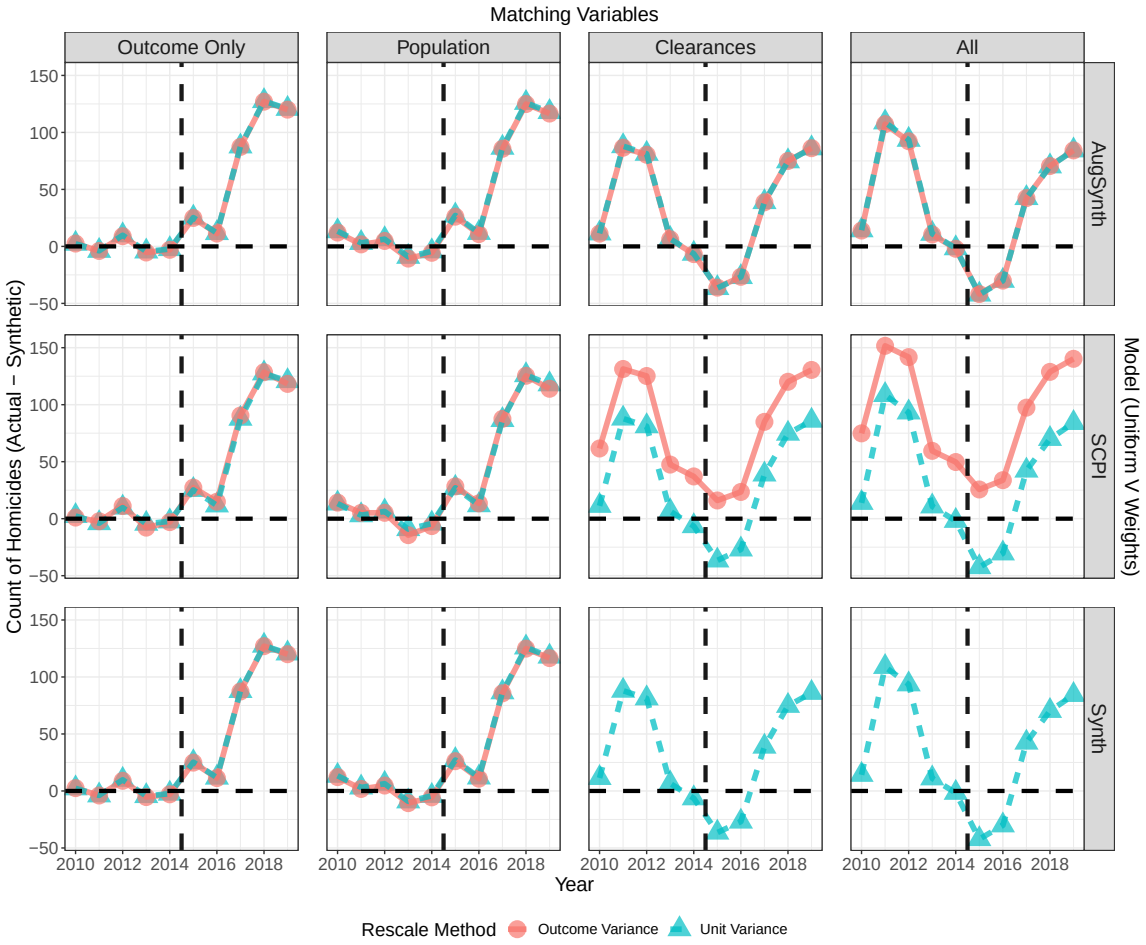
Note: Figure presents the standard SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by four different software packages. Each panel represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention.

Figure 4: Variance in Bias-Corrected SCM Estimates



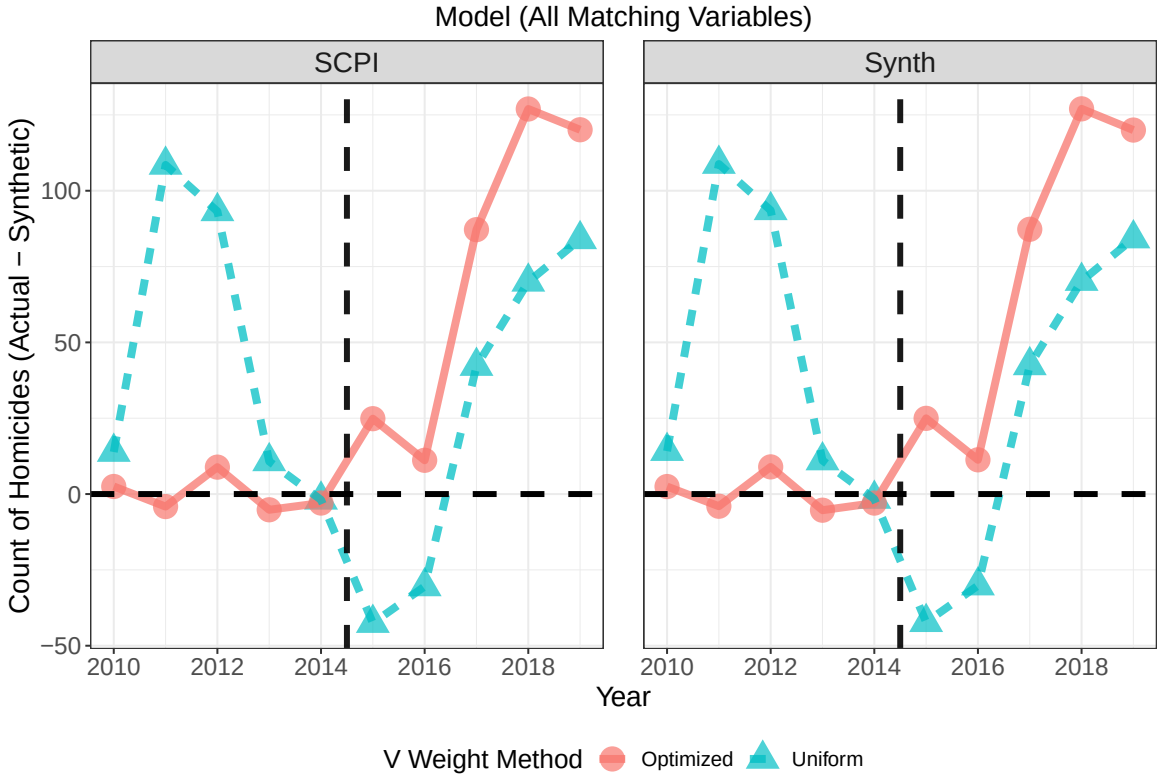
Note: Figure presents the bias-corrected SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by `allsynth` (top row) and `AugSynth` (bottom row). Each column of figures represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention.

Figure 5: Impact of Rescaling on Standard SCM Estimates



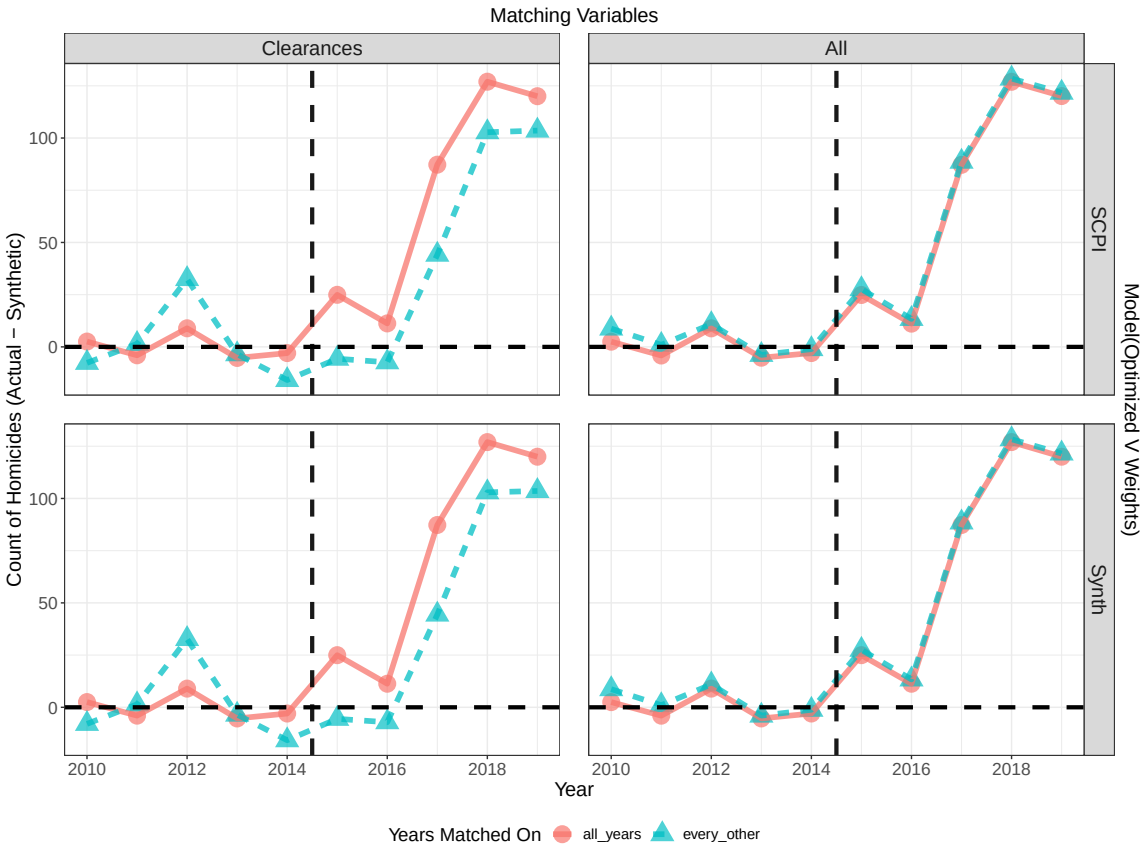
Note: Figure presents the standard SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by `AugSynth` (top row), `scpi` (middle row) and `Synth` (bottom row). Each column of figures represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention. Within each figure, the blue line represents estimates when matching variables where scaled to unit variance, and the red line represents estimates when matching variables were scaled to outcome variance.

Figure 6: Impact of Optimizing $\mathbf{V}$ Weights on on Standard SCM Estimates:



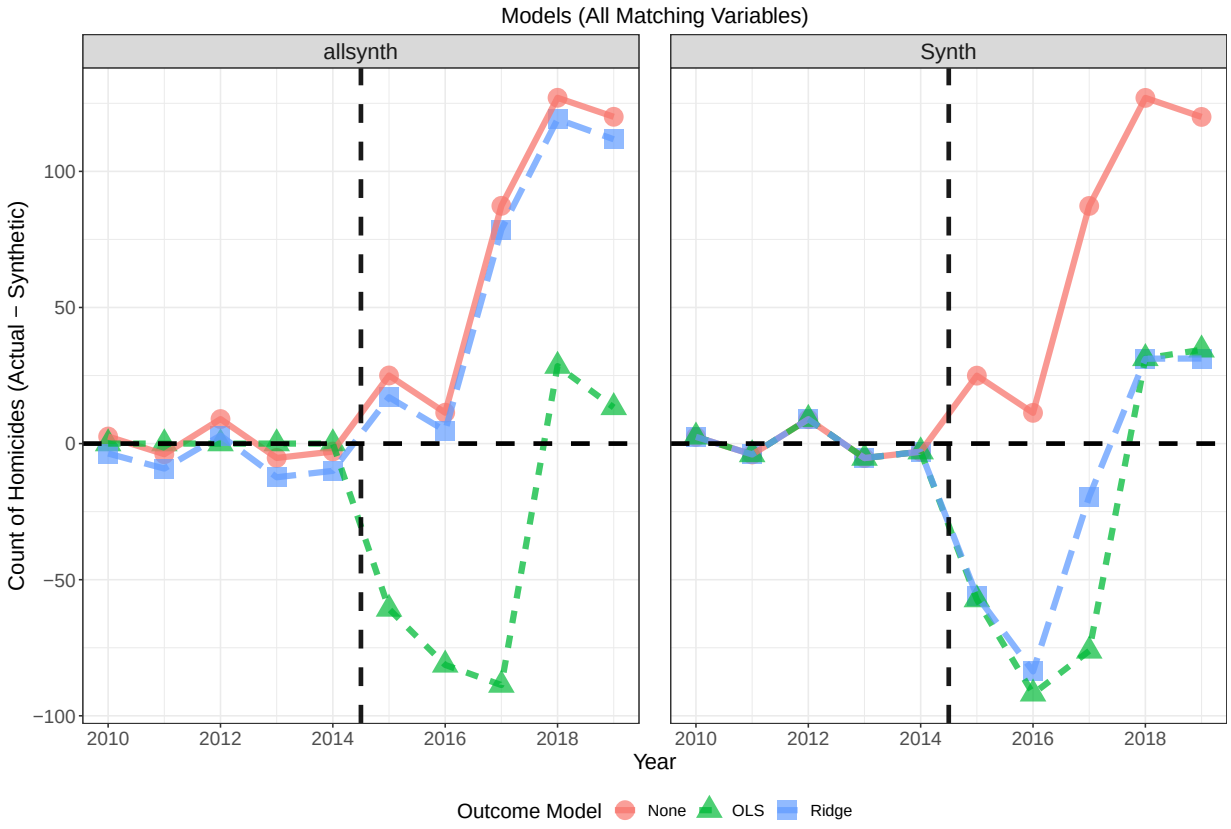
Note: Figure presents the standard SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by `scpi` (left) and `Synth` (right). Each column of figures represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention. Within each figure, the blue line represents estimates when the matching variable weights were uniform while the red line represents estimates when matching variables weights were optimized.

Figure 7: Impact of Sparse Matching Variables



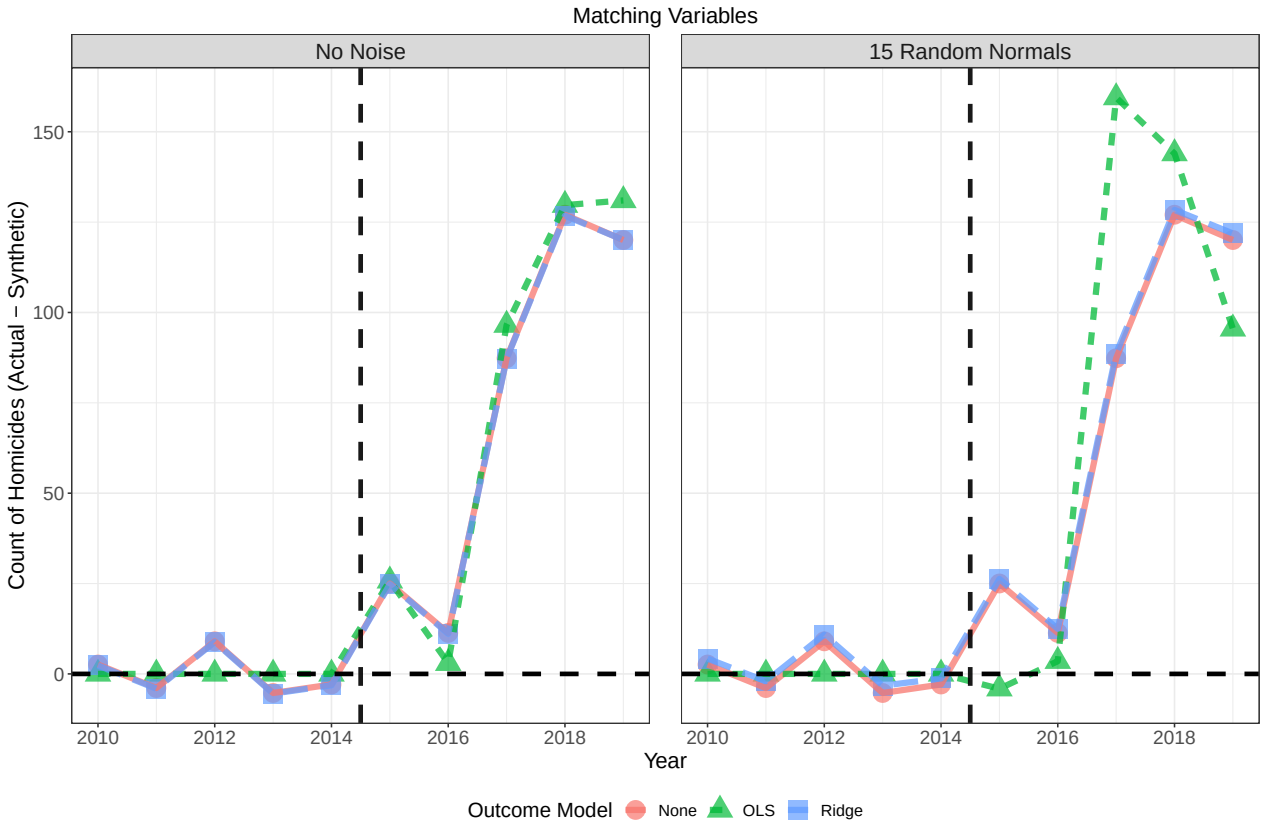
Note: Figure presents standard SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by `scpi` (first row) and `Synth` (second row). Each column of figures represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention. Within each figure, the blue line represents estimates when for each matching variable every other pre-period year (2011 and 2013) was included in the match, while the red line represents estimates when all per-period years were included.

Figure 8: Impact of Bias Correction



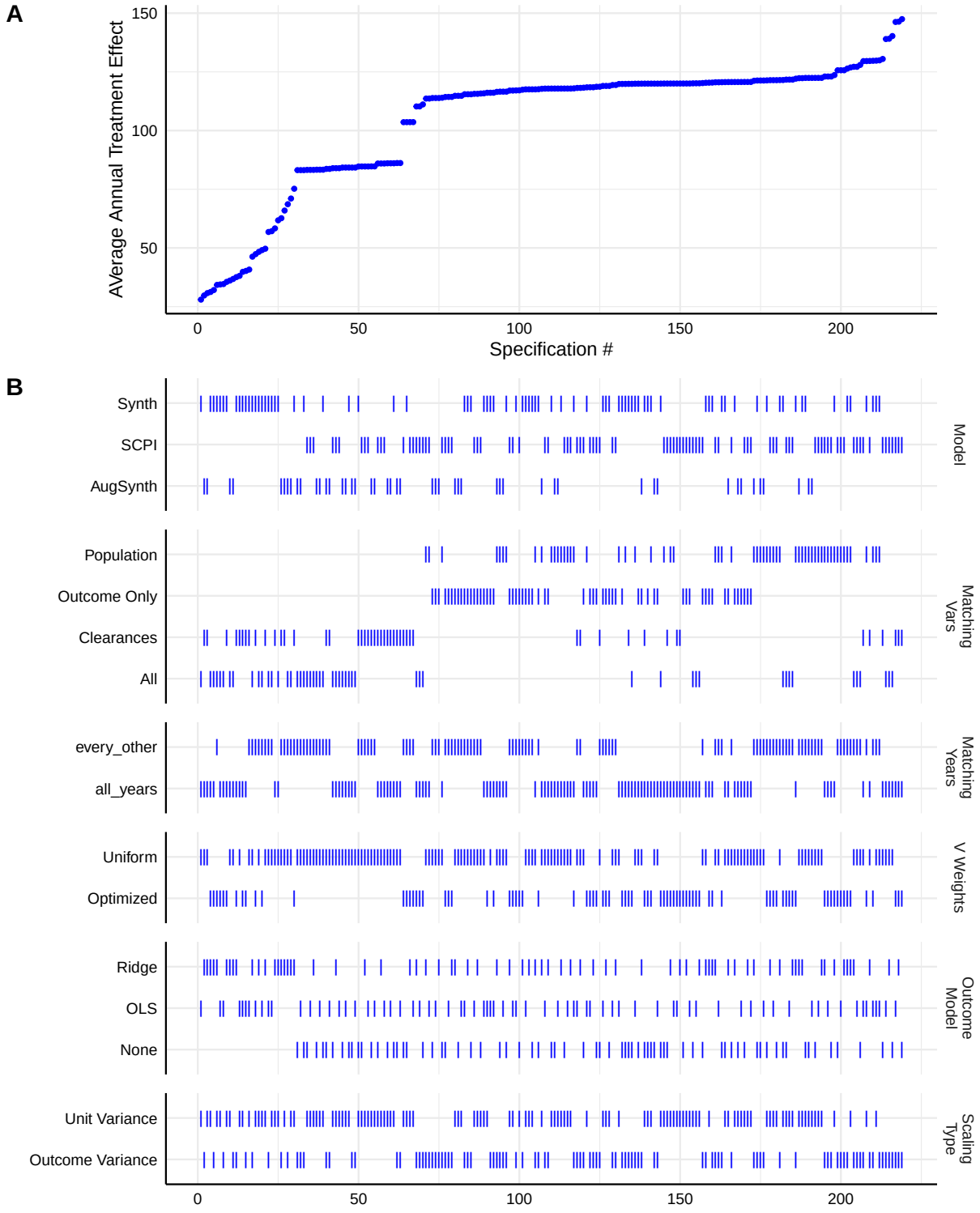
Note:

Figure 9: Sensitivity of Methods of Bias Correction to "Signal-Free" Covariates



Note: Figure presents bias corrected SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 estimated using `allsynth`. The lefthand panel presents model estimates conditioning on pre-intervention values of the outcome variable and population. The righthand panel also conditions on 15 "signal-free" covariates, each of which is distributed random normal. Within each figure, the red line represents estimates in which bias correction is not used, the green line represents estimates in which OLS is used to bias correct and the blue line represents estimates in which Ridge regression is used to bias correct.

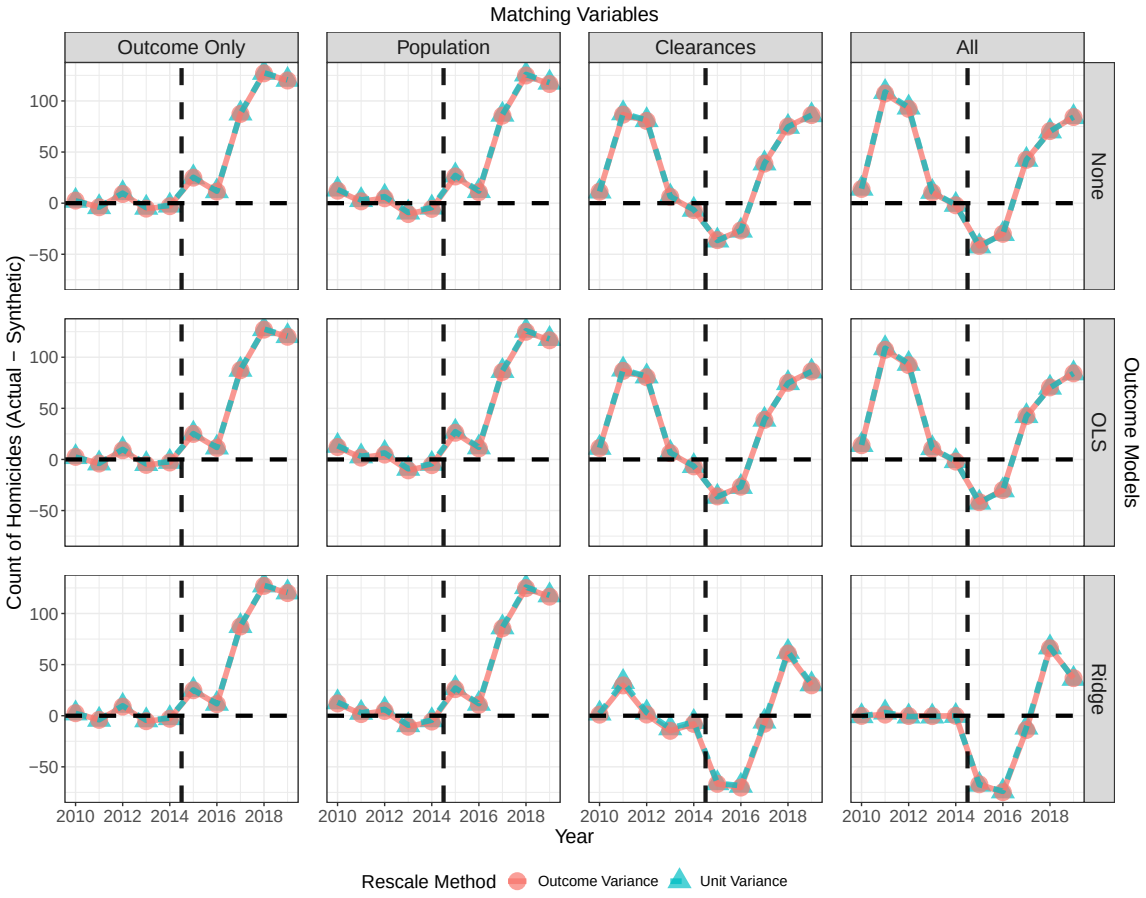
Figure 10: Specification Curve



Note: Figure presents a specification curve, inspired by Simonsohn et al. (2020), which characterizes the sensitivity of estimates to different implementations of SCM.

ONLINE APPENDIX

Appendix Figure 1: Impact of Rescaling and Bias Correction on `AugSynth` SCM Estimates

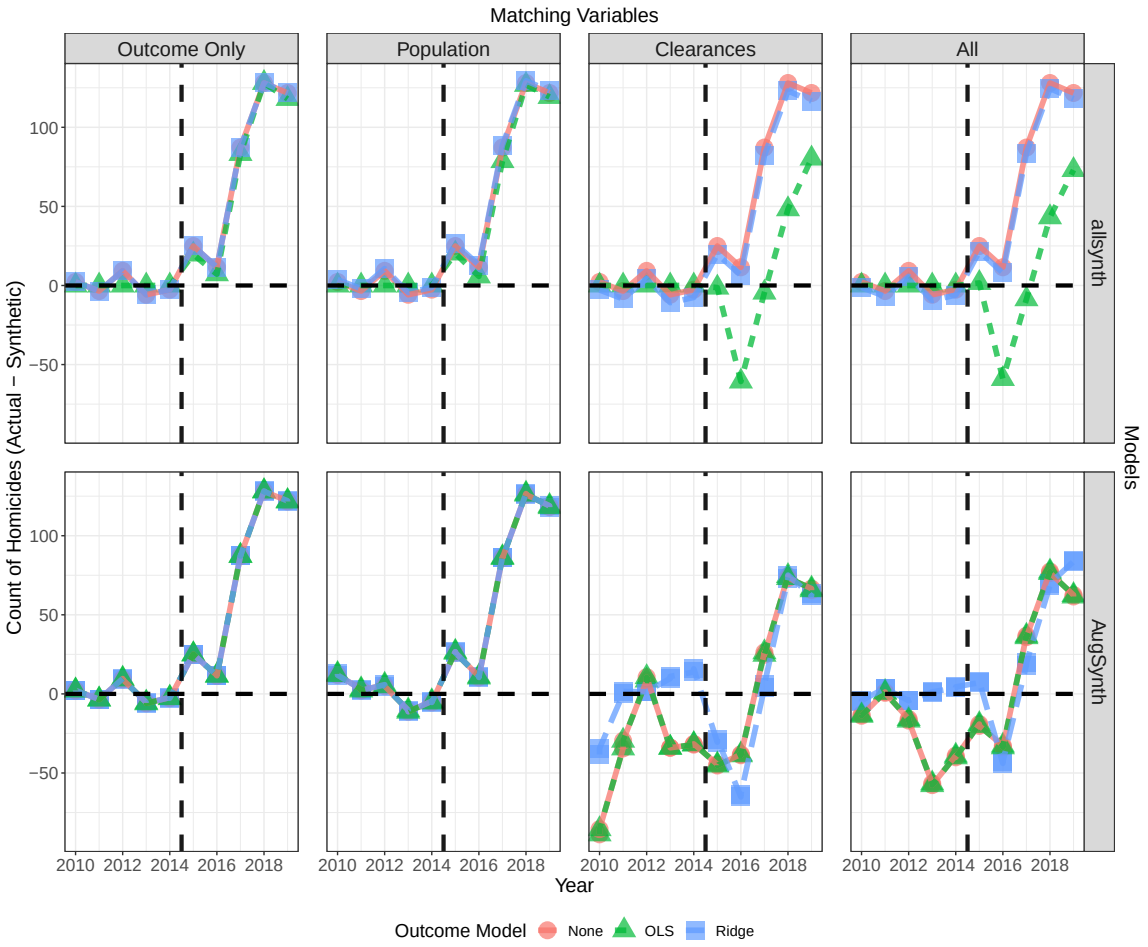


Note: Figure shows standard and bias-corrected SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by `AugSynth`. Each column represents a different set of matching variables used to construct the synthetic control. The first row of figures shows results from `AugSynth`'s implementation of standard SCM, while the second and third rows show bias-corrected estimates when the outcome model is OLS and ridge regression, respetively. The vertical dashed line splits the time period into pre- and post-intervention.

Appendix Figure 2: Variance in Standard SCM Estimates, Hogan (2022) Data

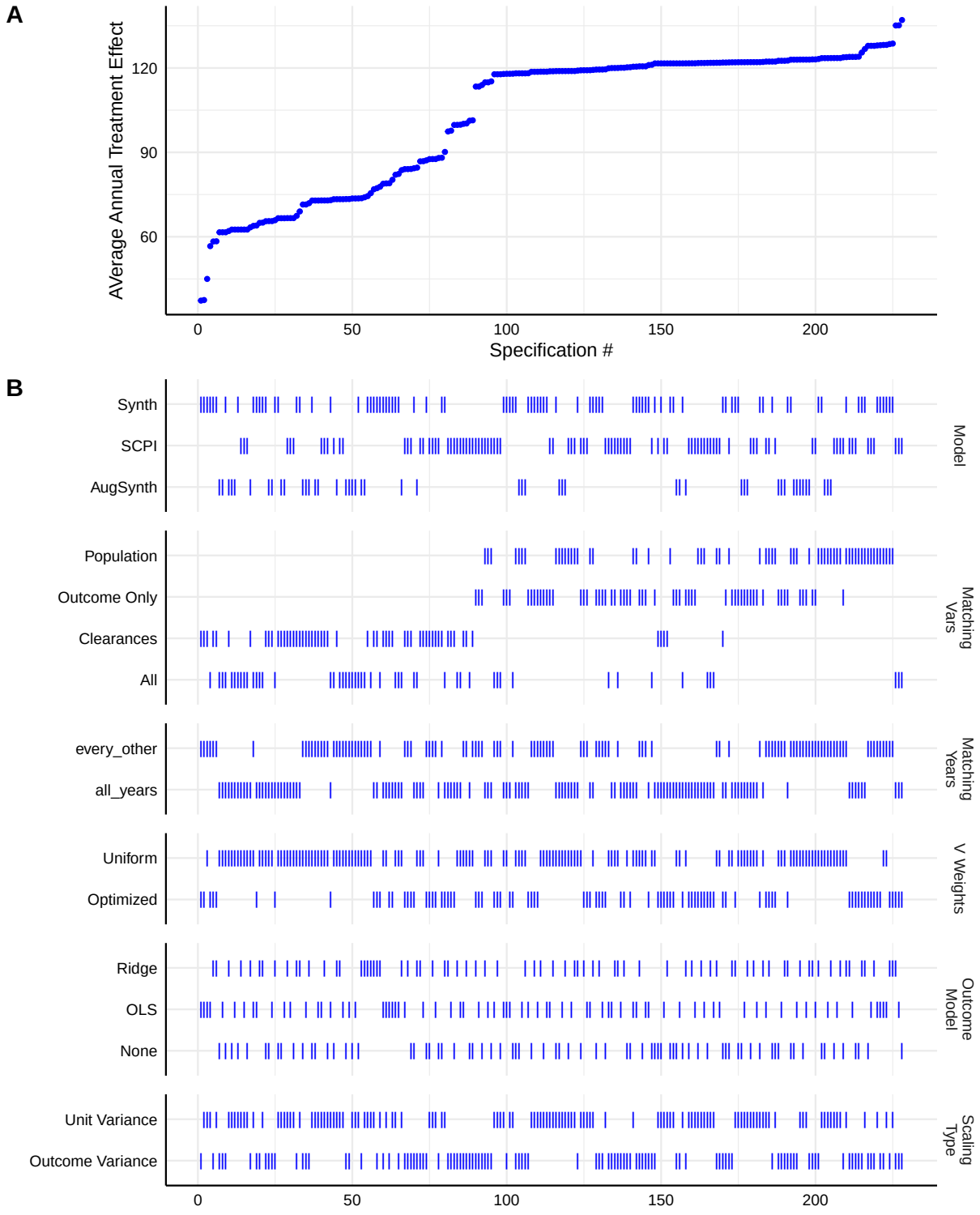
Note: Figure shows the standard SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by four different software packages. Each panel represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention. As compared with Figure 3 which uses data from Kaplan et al. (2022), this figure uses data from Hogan (2022).

Appendix Figure 3: Variance in Bias-Corrected SCM Estimates, Hogan (2022) Data



Note: Figure shows the bias-corrected SCM estimates of homicide counts for actual minus synthetic Philadelphia from 2010 to 2019 as estimated by `allsynth` (top row) and `AugSynth` (bottom row). Each column of figures represents a different set of matching variables used to construct the synthetic control. The vertical dashed line splits the time period into pre- and post-intervention. As compared with Figure 3 which uses data from Kaplan et al. (2022), this figure uses data from Hogan (2022).

Appendix Figure 4: Specification Curve, Hogan (2022) Data



Note: Figure presents a specification curve, inspired by Simonsohn et al. (2020), which characterizes the sensitivity of estimates to different implementations of SCM. As compared with Figure 10 which uses data from Kaplan et al. (2022), this figure uses data from Hogan (2022).