sociological science

Supplement to:

Felton, Chris. 2023. "*13 Reasons Why* Probably Increased Emergency Room Visits for Self-Harm among Teenage Girls." Sociological Science 10: 930-963.

Online Supplement for "*13 Reasons Why* Probably Increased Emergency Room Visits for Self-Harm among Teenage Girls"

Chris Felton

May 4, 2023

Contents

A Coding Self-Harm Visits

This section lists the ICD-9 and ICD-10 codes I use to calculate the number of self-harm visits. For the main analysis, I only use codes listed under "self-inflicted harm" in Healthcare Cost and Utilization Project (2021). Observations contribute to the total count if any diagnosis code (i.e., whether primary or secondary) matched one of these codes. This excludes suicidal ideation and "late effects" ("sequelae").[1] This also excludes the ICD-10 code T14.91 ("Suicide attempt"), since it is unavailable for earlier years (but see Section B for counts of visits coded as suicide attempts). For ICD-10, I only use codes that specify the visit was for the "initial encounter." For a full list of codes (e.g., the codes used for accidental injuries), see the `R` script `diagnosis_codes.R` at `https://github.com/cmfelton/13rw`. Researchers who do not use `R` can still open the script in a text editor and locate the lists of diagnosis codes.

- Intentional self-harm codes:
  - ICD-9: All injury codes beginning with E950, E951, E952, E953, E954, E955, E956, E957, and E958.
  - ICD-10: All T and X codes specific to intentional self-harm coded as "initial encounter."
- Intentional cutting:
  - ICD-9: All injury codes beginning with E956.
  - ICD-10: X780XXA, X781XXA, X782XXA, X788XXA, and X789XXA.
- Accidental cutting:
  - ICD-10: W25XXXA, W260XXA, W261XXA, W262XXA, W268XXA, and W269XXA.
- Cutting, undetermined intent:
  - ICD-10: Y280XXA, Y281XXA, Y282XXA, Y288XXA, and Y289XXA.
- Intentional poisoning:
  - ICD-9: All injury codes beginning with E950, E951, or E952.
  - ICD-10: All T codes specific to intentional poisoning coded as "initial encounter."

[1] Initial analyses accidentally included E959, "late effects," for the ICD-9 years. Counts and treatment effect estiamtes were extremely similar.

- Accidental poisoning:
  - ICD-10: All T codes specific to accidental poisoning coded as "initial encounter."
- Poisoning, undetermined intent:
  - ICD-10: All T codes specific to poisoning with undetermined intent coded as "initial encounter."

B Detailed Results

This section contains detailed results from alternative model selection procedures and alternative time series (e.g., self-harm among girls 10–17 rather than 10–19). The results are all consistent with the study's main finding. It also contains ARIMA output tables, alternative block sizes for placebo tests, etc. Data and code for replicating all analyses can be found at `https://github.com/cmfelton/13rw`.

B.1 *p*-value Tables

This section contains tables of conformal p-values on the post-treatment period (Tables B.1 to B.3). That is, they test the null hypothesis of no treatment effect in the post-treatment period.

Each table contains p-values from a different model specification and estimation procedure, as outlined in Section E.1. Conformal inference poses two choices: the length of the post-treatment period and the block size (Chernozhukov et al., 2021). I report p-values for a range of different combinations of post-treatment-period lengths and block sizes.

		Length of Post-Treatment Period		
		1	2	3
Block Size	1	0.0081		
	2	0.0081	0.0081	
	3	0.0325	0.0081	0.0080
	4	0.0244	0.0242	0.0080

Table B.1: Full specification (all pre-treatment data) conformal *p*-values.

		Length of Post-Treatment Period		
		1	2	3
Block Size	1	0.0081		
	2	0.0081	0.0081	
	3	0.0488	0.0081	0.0080
	4	0.0244	0.0242	0.0080

Table B.2: Split specification (all pre-treatment data) conformal *p*-values.

		Length of Post-Treatment Period		
		1	2	3
Block Size	1	0.0161		
	2	0.0323	0.0159	
	3	0.0806	0.0476	0.0156
	4	0.1290	0.0635	0.0469

Table B.3: Split specification (estimation set) conformal p-values.

B.2 Treatment Effects

Figure B.1 plots treatment effects and 95% conformal confidence intervals across three different block sizes and three different model selection procedures as described in Section E.1. Briefly, *Full* entails selecting the model using the entire pre-treatment time series and fitting it using that same series; *Split* entails selecting the model using the first half of the pre-treatment series and fitting it using the second half; and *Split Analysis, Full Estimation (SAFE)* using the first half of the pre-treatment series to select a model and the entire pre-treatment series to fit the model.[2]

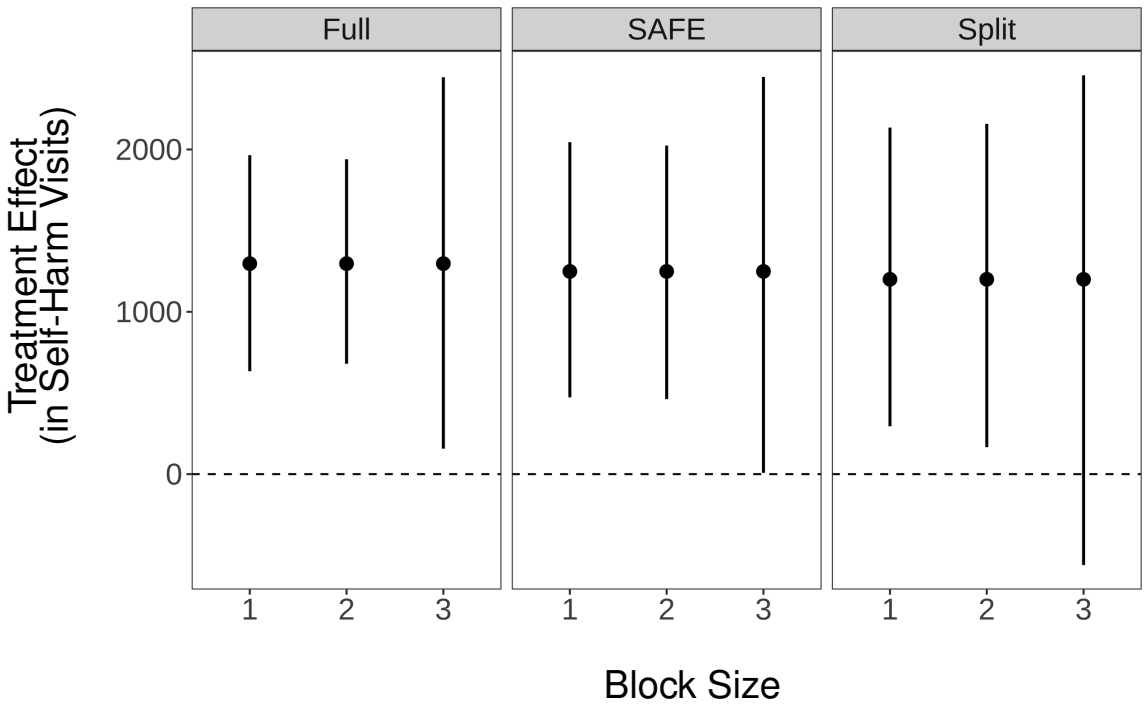

Figure B.1: Results by model-selection strategy and block size.

[2]I borrow the term *SAFE* from Faraway (2016).

B.3 Placebo Test Results

I repeat the placebo test described in the main text across three block sizes (1, 2, and 3) using 90% conformal confidence intervals. Figures B.2–4 show the results. 90% confidence intervals with block size = 3 appear conservative, but I avoid using these intervals in the main text because block sizes greater than 1 appear to produce strange results. For instance, intervals with block size = 2 provide much worse coverage than with block size = 1.

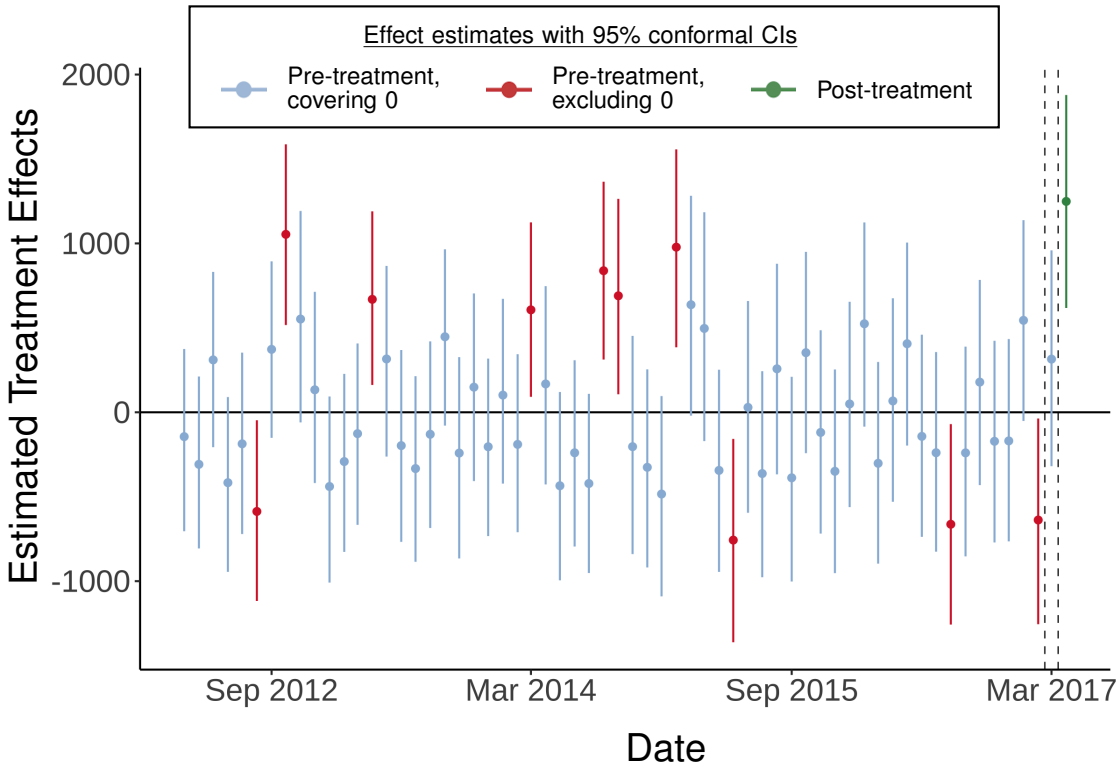

Figure B.2: Repeated placebo test procedure with 90% conformal confidence intervals and a block size of 1 with an expanding training period.

I repeat the procedure another three times, again across block sizes 1, 2, and 3. This time, instead of expanding the training period each time, I shift the period over by one month. This means that the sample size remains constant across tests. The results are much worse: confidence intervals are extremely unstable, particularly with block sizes greater than 1. The width of the intervals varies wildly across periods—even consecutive periods. 95% confidence intervals, unreported here, show even more instability.

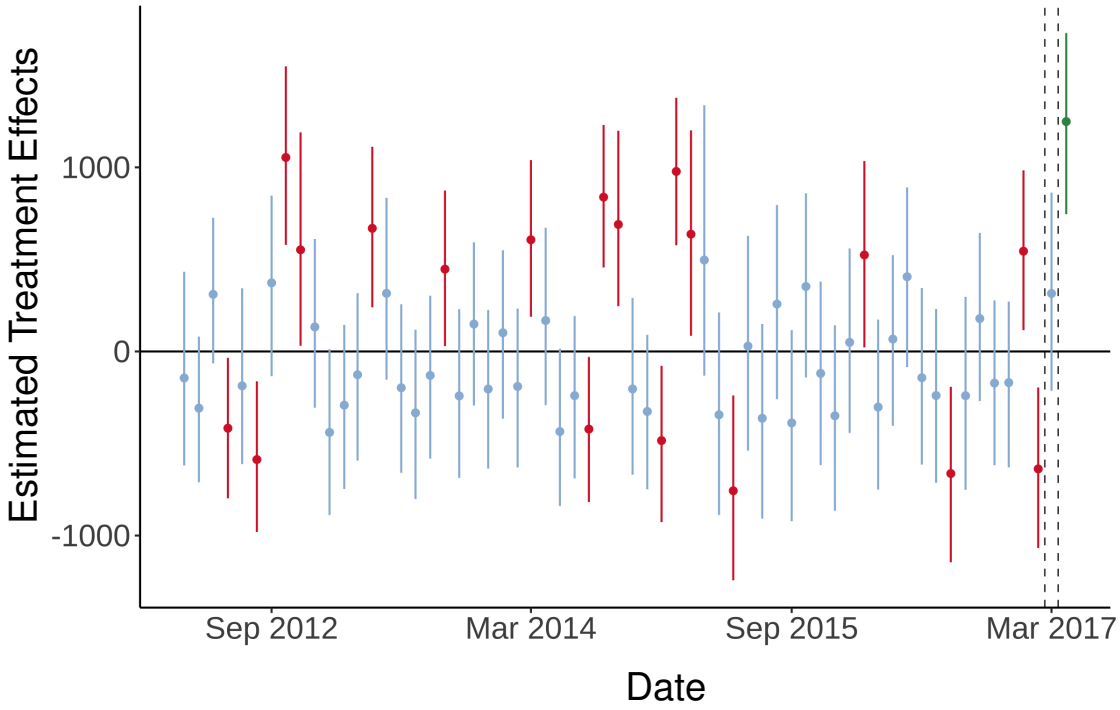

Figure B.3: Repeated placebo test procedure with 90% conformal confidence intervals and a block size of 2 with an expanding training period.

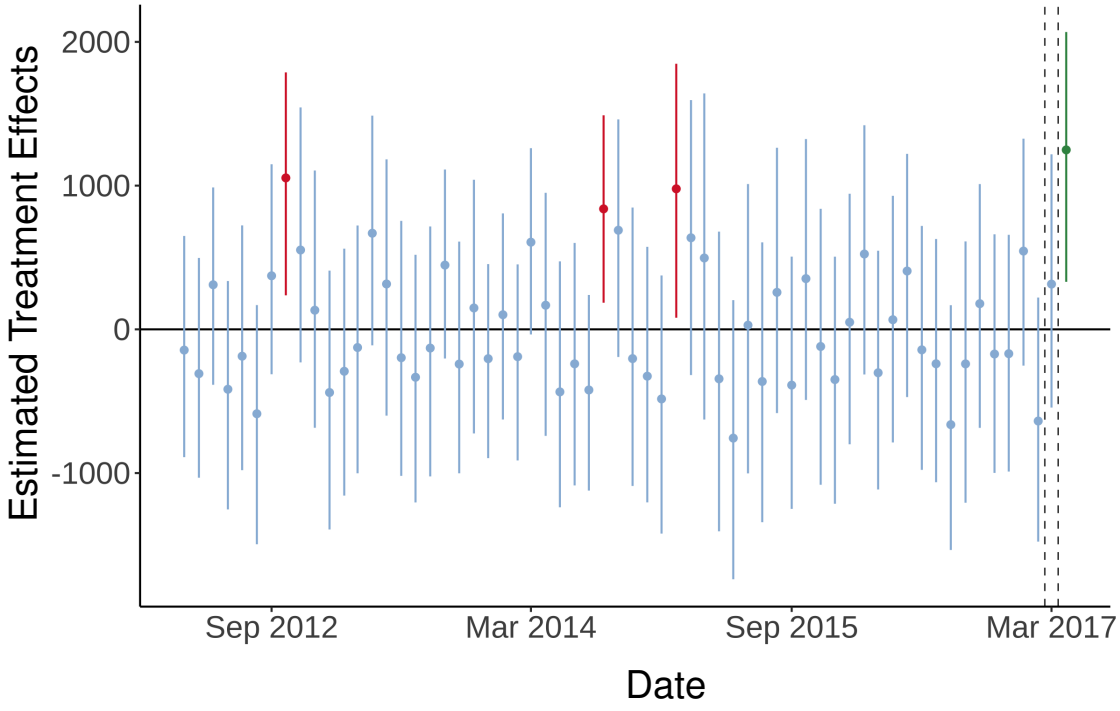

Figure B.4: Repeated placebo test procedure with 90% conformal confidence intervals and a block size of 3 with an expanding training period.

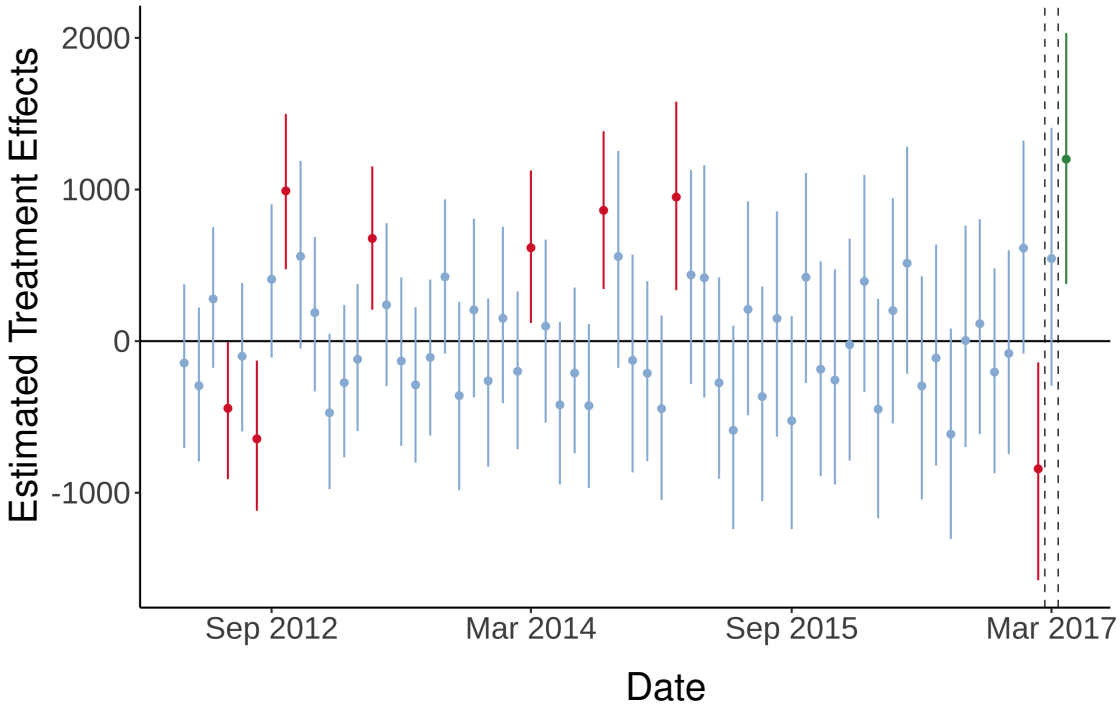

Figure B.5: Repeated placebo test procedure with 90% conformal confidence intervals and a block size of 1 with a sliding training period.

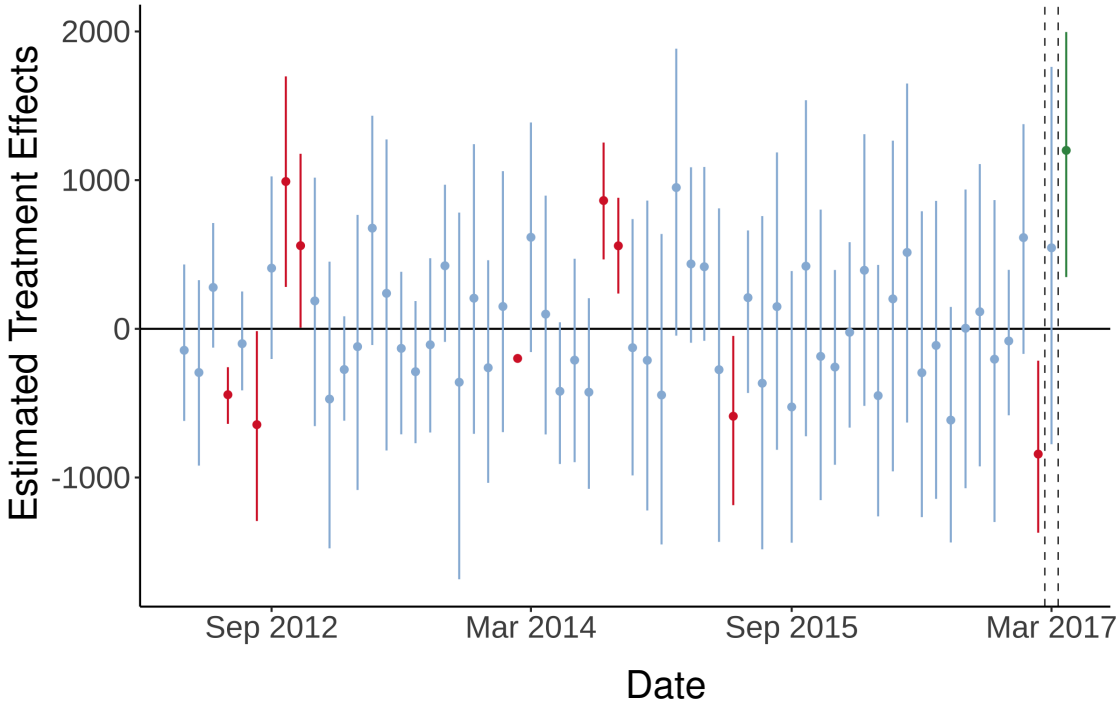

Figure B.6: Repeated placebo test procedure with 90% conformal confidence intervals and a block size of 2 with a sliding training period.

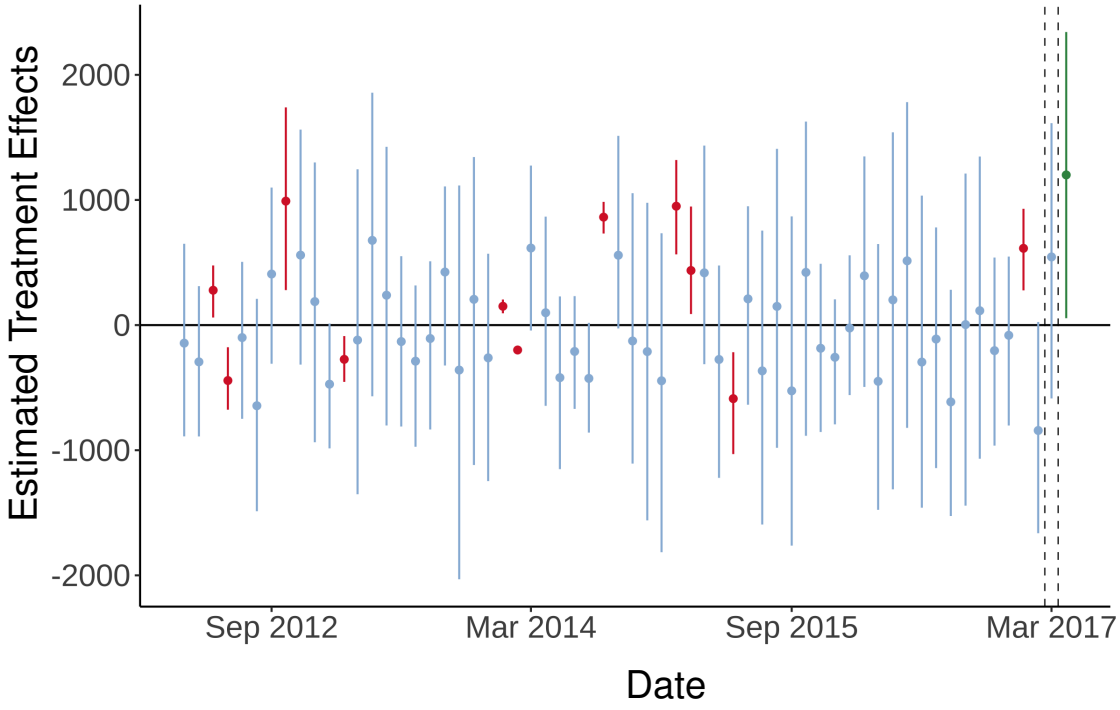

Figure B.7: Repeated placebo test procedure with 90% conformal confidence intervals and a block size of 3 with a sliding training period.

B.4 Residual Plots for Alternative Model Specifications

Figure 13 in the main text shows residuals from the main model specification used in the paper applied to the entire time series (including nine post-treatment periods). It showed that the residuals for April and May 2017 are much larger than all other residuals. We might worry, however, that this pattern is model-dependent—i.e., that other plausible model specifications produce different patterns. Figures B.8–11 show that alternative model specifications produce the same pattern. Across model selection procedures, `auto.arima()` selects fairly simple models, so Figures B.10 and B.11 consider more complex model specifications.

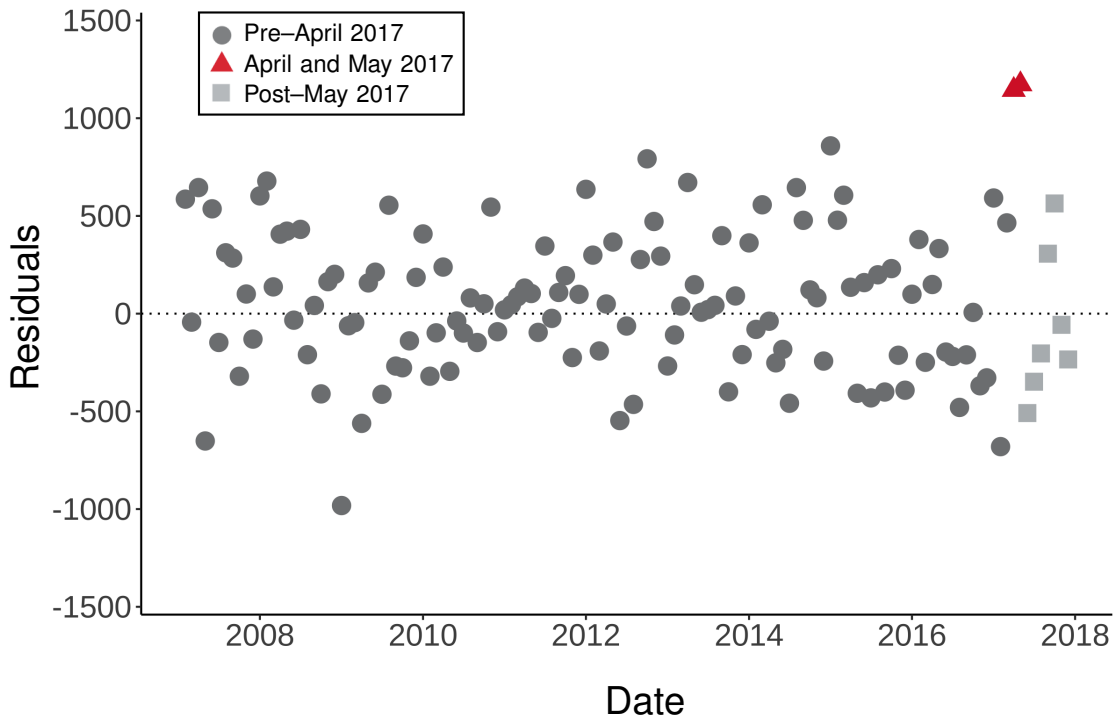

Figure B.8: Residuals from ARIMA selected using the full time series. This model was selected by running `auto.arima` on the full time series, not just the pre-treatment data. If anything, we should expect it to overfit April and May 2017. The specification is $(0, 1, 2)(0, 1, 1)_{12}$.

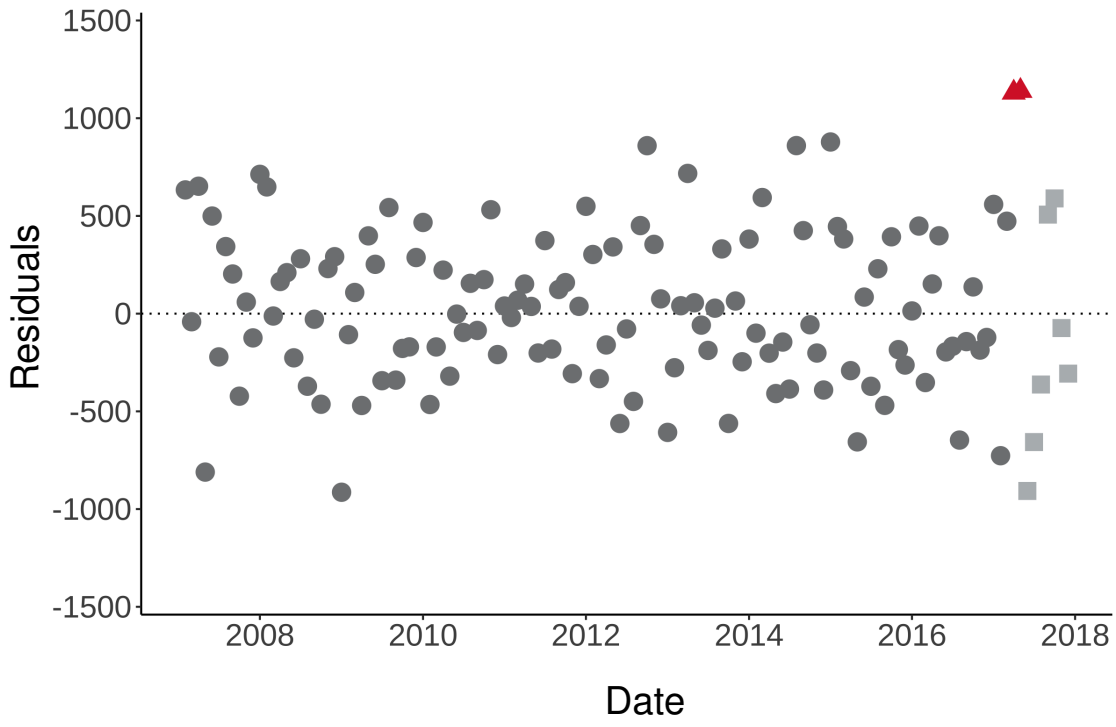

Figure B.9: Residuals from *Split* ARIMA specification run on the full time series. An ARIMA$(2,1,0)(1,1,0)_{12}$.

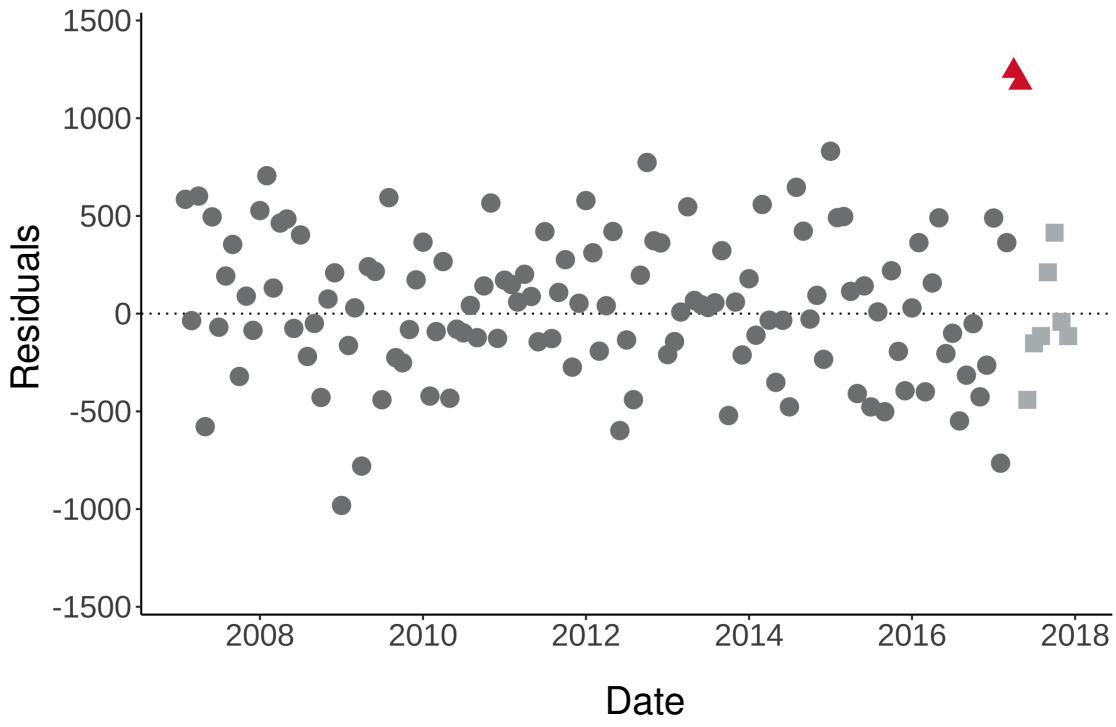

Figure B.10: Residuals from a more complex ARIMA specification run on the full time series. An ARIMA$(3,1,1)(1,1,0)_{12}$.

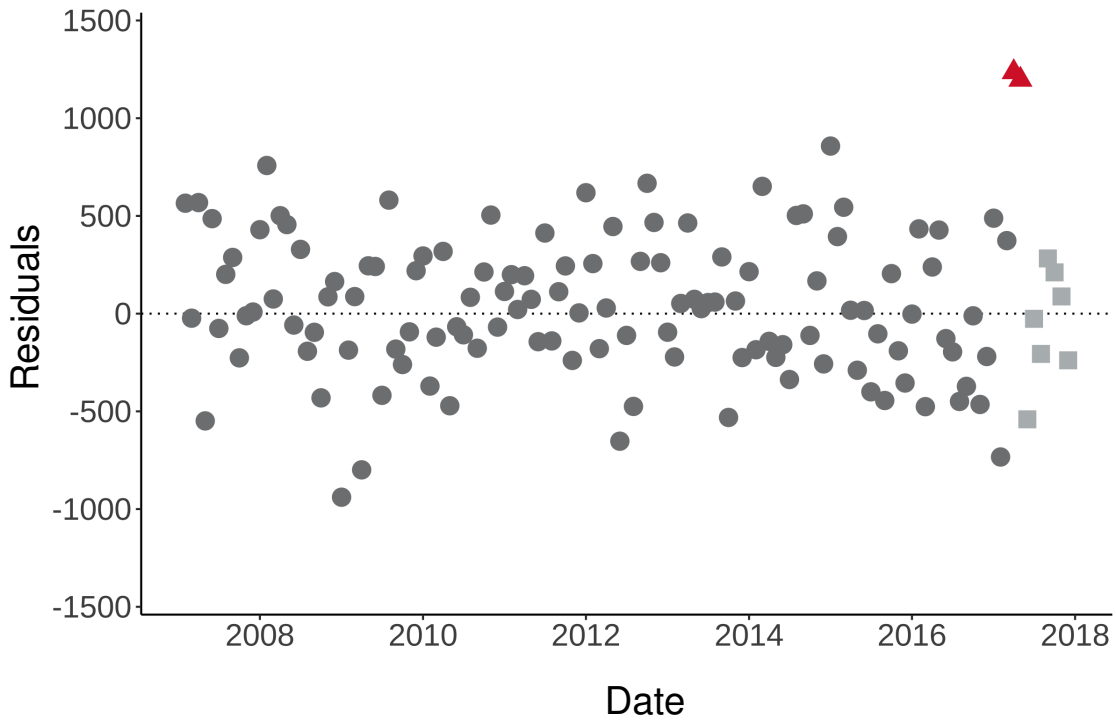

Figure B.11: Residuals from another, more complex ARIMA specification run on the full time series. An ARIMA$(4, 1, 2)(1, 1, 0)_{12}$.

B.5 ARIMA Tables

Tables B.4 to B.7 show ARIMA output tables for the different ARIMA models used to estimate treatment effects for girls 10–19.

ARIMA$(2, 1, 0)(1, 1, 0)_{12}$			
	y_{t-1}	y_{t-2}	y_{t-12}
Coef.	-0.7670	-0.4037	-0.6763
$\widehat{\text{SE}}$	(0.1241)	(0.1272)	(0.0970)
Obs. (pre-differencing)			74
Obs. (post-differencing)			61
$\hat{\sigma}^2 = 93530$ log likelihood $= -438.14$			
AIC $= 884.28$ AICc $= 885$ BIC $= 892.73$			

Table B.4: Split specification (training set).

ARIMA$(2,1,0)(1,1,0)_{12}$			
	y_{t-1}	y_{t-2}	y_{t-12}
Coef.	-0.3623	-0.1223	-0.2678
$\widehat{\text{SE}}$	(0.1282)	(0.1341)	(0.1447)
Obs. (pre-differencing)			74
Obs. (post-differencing)			61
$\hat{\sigma}^2 = 170163$	log likelihood $= -452.89$		
AIC $= 913.78$	AICc $= 914.5$		BIC $= 922.22$

Table B.5: Split specification (estimation set).

ARIMA$(2,1,0)(1,1,0)_{12}$			
	y_{t-1}	y_{t-2}	y_{t-12}
Coef.	-0.5280	-0.2192	-0.4528
$\widehat{\text{SE}}$	(0.0894)	(0.0912)	(0.0855)
Obs. (pre-differencing)			135
Obs. (post-differencing)			122
$\hat{\sigma}^2 = 139390$	log likelihood $= -895.67$		
AIC $= 1799.33$	AICc $= 1799.68$		BIC $= 1810.55$

Table B.6: Split specification (all pre-treatment periods). This corresponds to the *SAFE* model.

ARIMA$(0,1,1)(0,1,1)_{12}$		
	$\hat{u}_{t-1}$	$\hat{u}_{t-12}$
Coef.	-0.6501	-0.5080
$\widehat{\text{SE}}$	(0.0782)	(0.0867)
Obs. (pre-differencing)		135
Obs. (post-differencing)		122
$\hat{\sigma}^2 = 124772$ log likelihood $= -889.96$		
AIC $= 1785.92$ AICc $= 1786.12$ BIC $= 1794.33$		

Table B.7: Full specification (all pre-treatment periods).

B.6 Model-Free Plots

Figure B.12 shows the entire time series, with a separate line for each year to make the consistent seasonality more apparent. Figure B.13 shows that self-harm visits are typically higher in March than February, and Figure B.14 shows that self-harm visits are usually lower in April than March. Figure B.15 plots the first-differenced time series by month, with 2017 observations colored red (pre-treatment) and green (post-treatment). Figure B.16 shows the proportion of self-harm visits (among teen girls) that are for cutting sequentially.

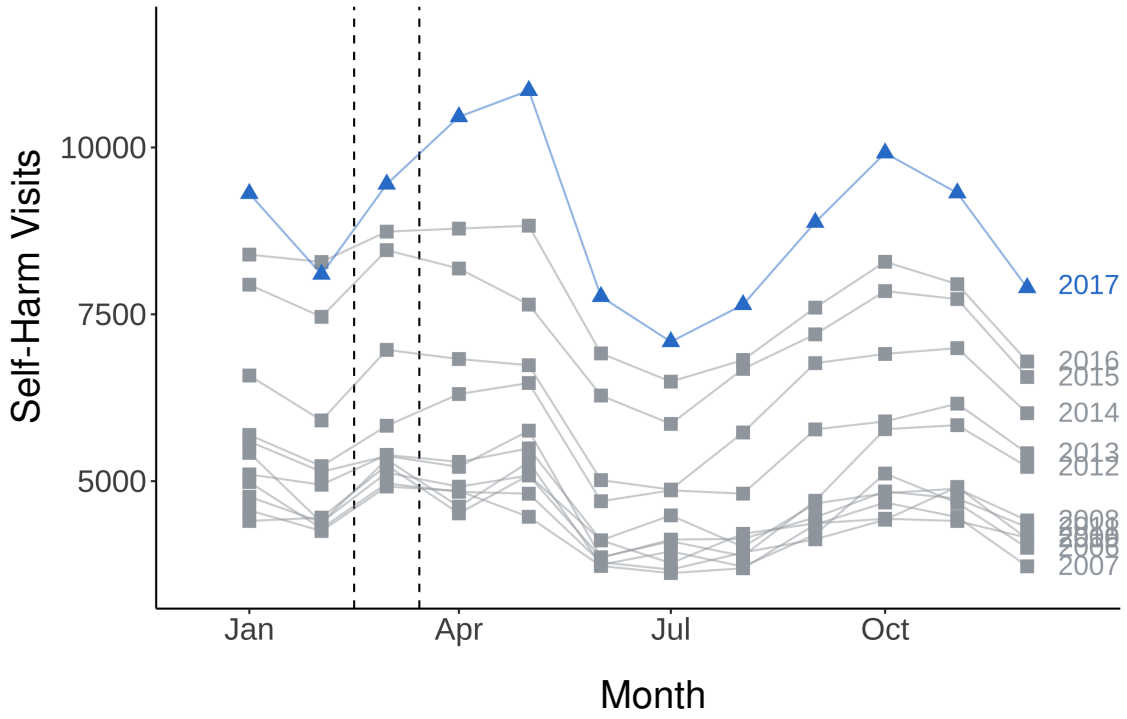

Figure B.12: Self-harm visits for the full time series. The series exhibits seasonality and an upward trend starting in 2012. The rise from December to January and fall from January to February matches what we see in all-group suicide mortality data. The broader seasonal pattern, however, is the opposite: typically suicide mortality is higher in the summer, but self-harm visits for teen girls are lowest during the summer months.

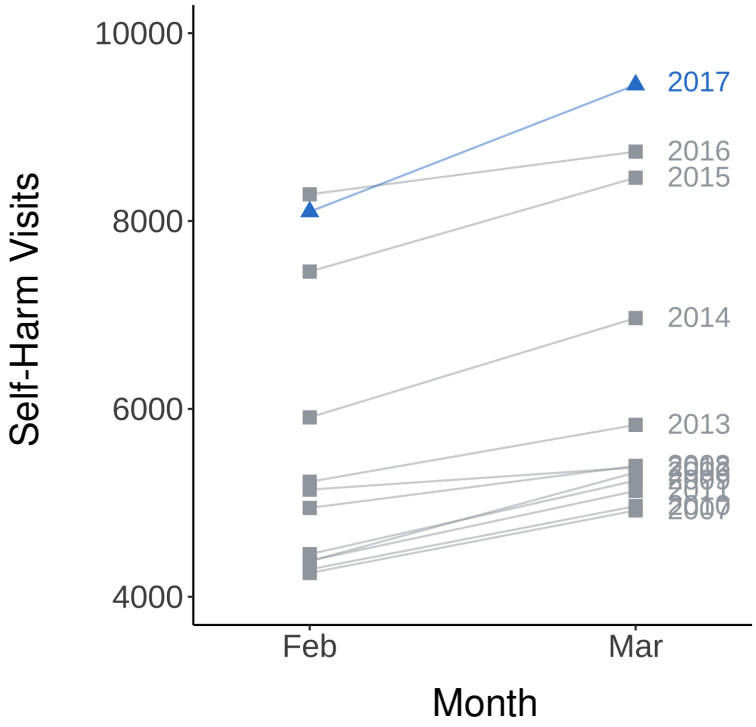

Figure B.13: Self-harm visits across all Februaries and Marches. The count of self-harm visits is typically higher in March.

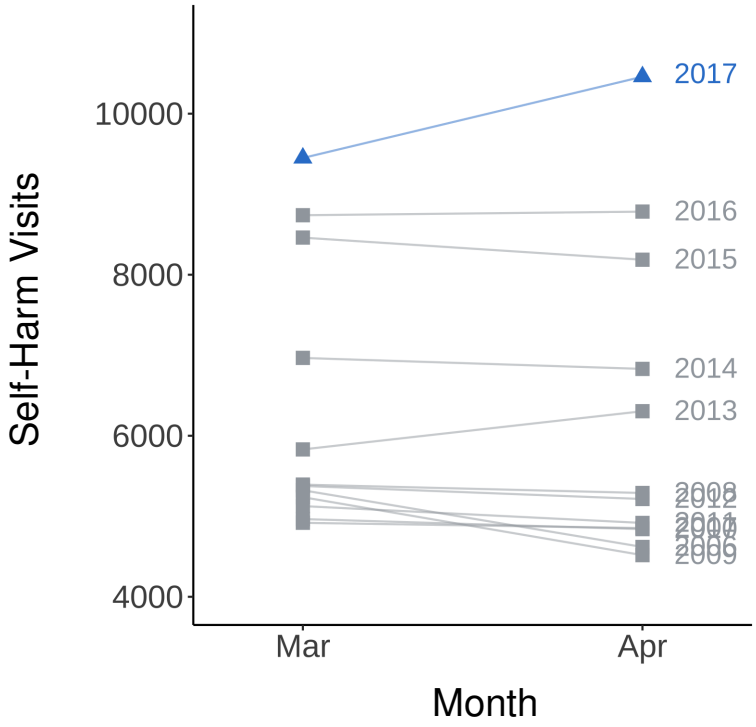

Figure B.14: Self-harm visits across all Marches and Aprils. The count of self-harm visits is typically lower in April, except in 2013 and 2017.

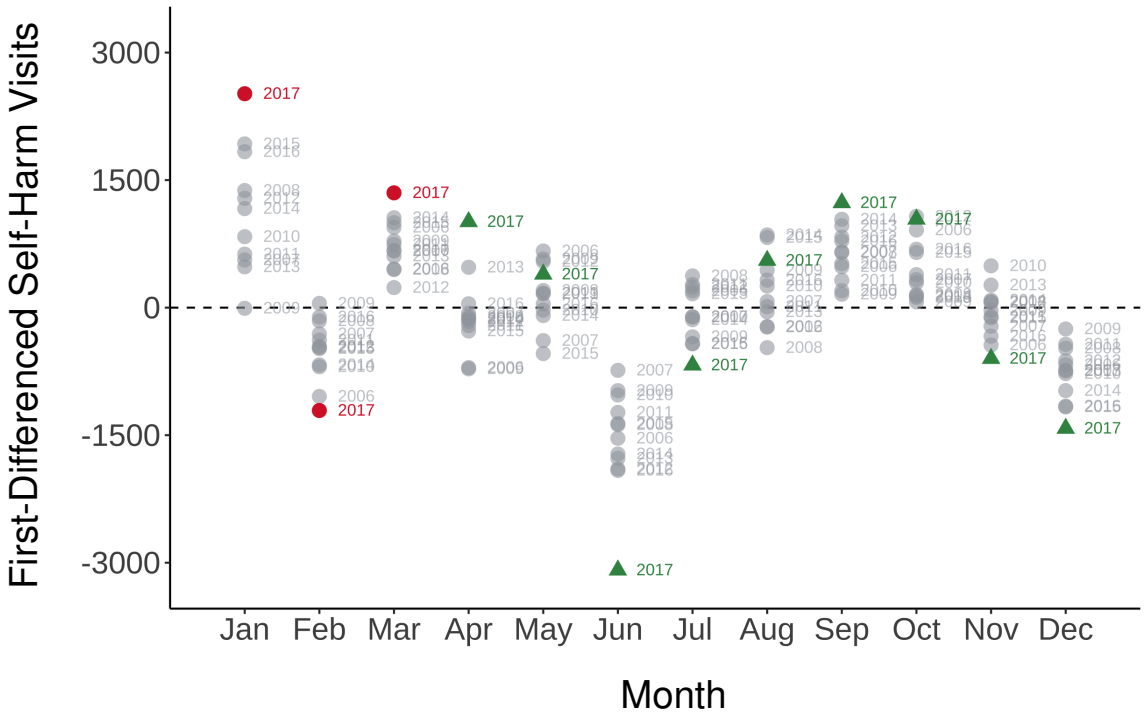

Figure B.15: First-differenced self-harm visits for the entire time series. April 2017 is unusual relative to other Aprils—most are negative, and the two positive observations are much closer to zero. While January 2017 is larger than other Januaries, these values show larger variance and tend to be positive. In this respect, it's not quite as unusual as April 2017. The large negative value for June 2017 is also noteworthy—there's a large drop between May 2017 and June 2017, indicating a fading of the treatment effect.

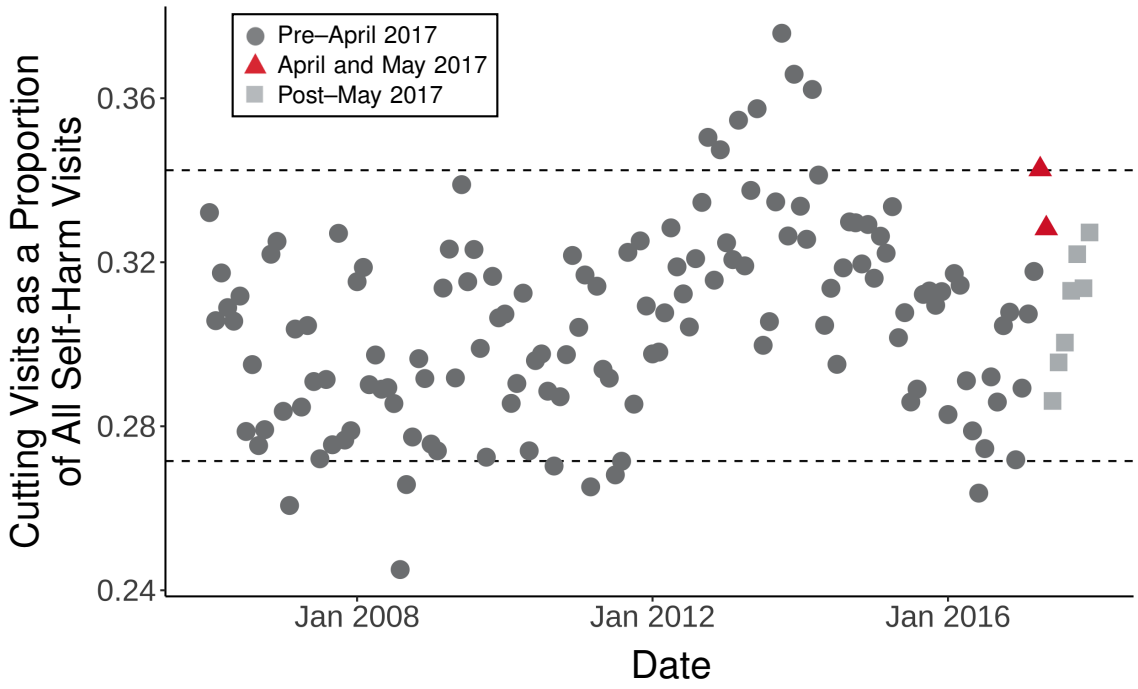

Figure B.16: Proportion of Self-Harm Visits for Cutting. Proportion of ER visits for intentional-self harm for teenage girls that are for cutting. The red triangles show the proportions for April and May of 2017. Dashed lines show 5th and 95th percentiles of the distribution.

B.7 ITS Plots with Alternative Series

Figures B.17–21 show ITS plots for alternative time series. For each series, I re-run `auto.arima()` on the full pre-treatment series and construct 95% conformal prediction intervals. Figure B.17 uses Bridge et al.'s (2020) age grouping (10–17) rather than Niederkrotenthaler et al.'s (2019). Figure B.18 looks only at self-harm visits for cutting or poisoning, excluding rarer forms. Because the time series exhibits a strong upward trend (and more noticeable seasonality) starting in 2013, I re-run the analysis starting the time series at January 2013 in Figure B.19, and again starting at January 2014 in Figure B.20. Figure B.21 shows results using unweighted counts. All show a large increase in self-harm visits in the post-treatment period, with 95% conformal prediction intervals excluding the observed count.

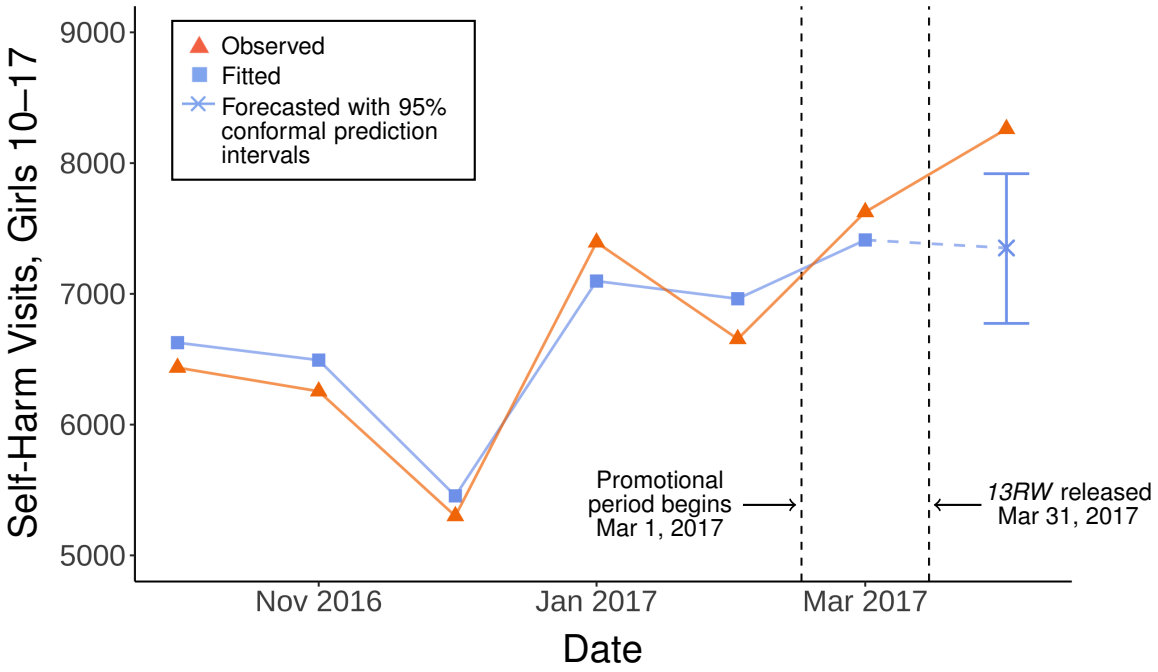

Figure B.17: ITS Analysis Using Self-Harm Visits for Girls 10–17. The ARIMA model is selected using `auto.arima()` on the full pre-treatment time series. The forecasted point is shown with 95% conformal prediction intervals and block size = 1.

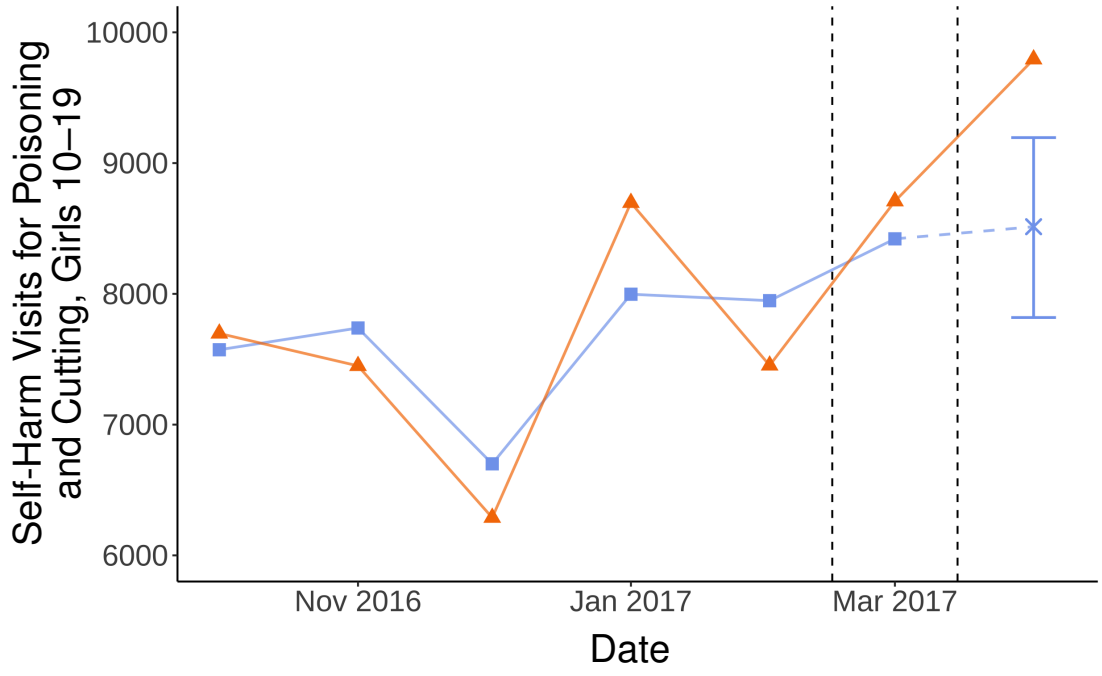

Figure B.18: ITS Analysis Using Self-Harm Visits for Poisoning and Cutting. The ARIMA model is selected using `auto.arima()` on the full pre-treatment time series. The forecasted point is shown with 95% conformal prediction intervals and block size = 1.

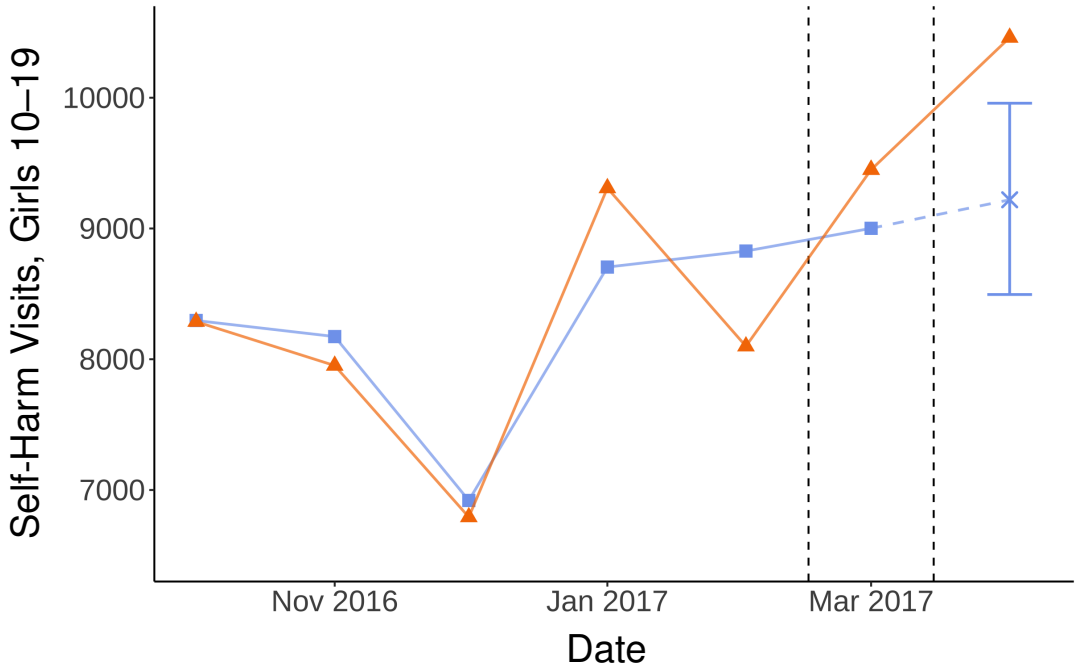

Figure B.19: ITS Analysis Using Shorter Pre-Treatment Series. The ARIMA model is selected using `auto.arima()` on the pre-treatment series starting in January, 2013 to account for a potential structural break between 2012 and 2013. The forecasted point is shown with 95% conformal prediction intervals and block size = 1.

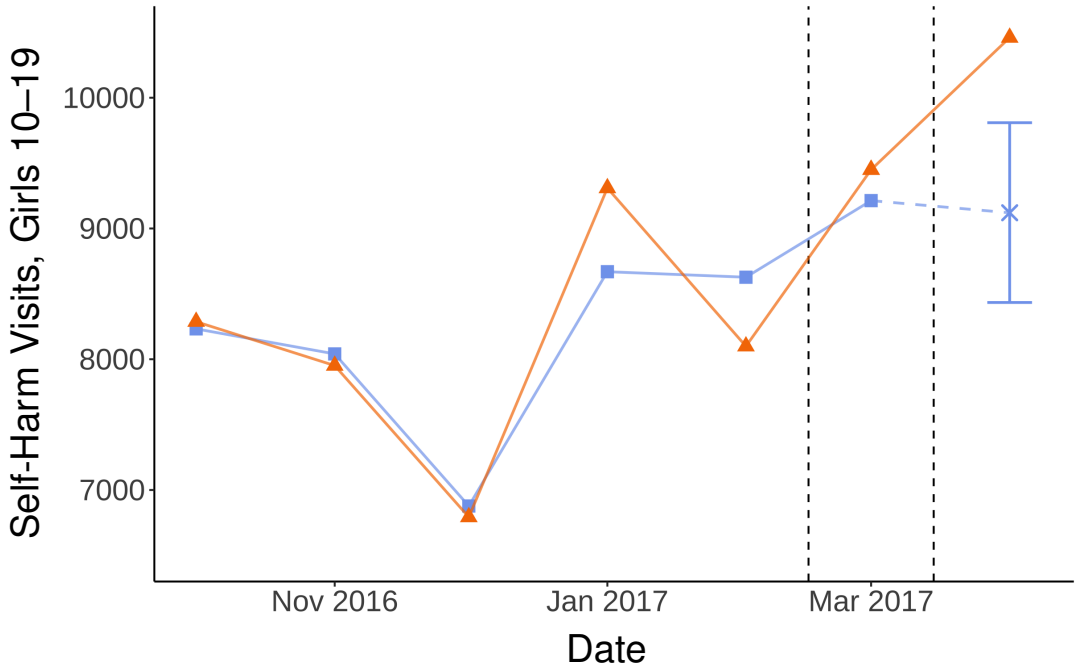

Figure B.20: ITS Analysis Using an Even Shorter Pre-Treatment Series. The ARIMA model is selected using `auto.arima()` on the pre-treatment series starting in January, 2014 to account for a potential structural break between 2013 and 2014. The forecasted point is shown with 95% conformal prediction intervals and block size = 1.

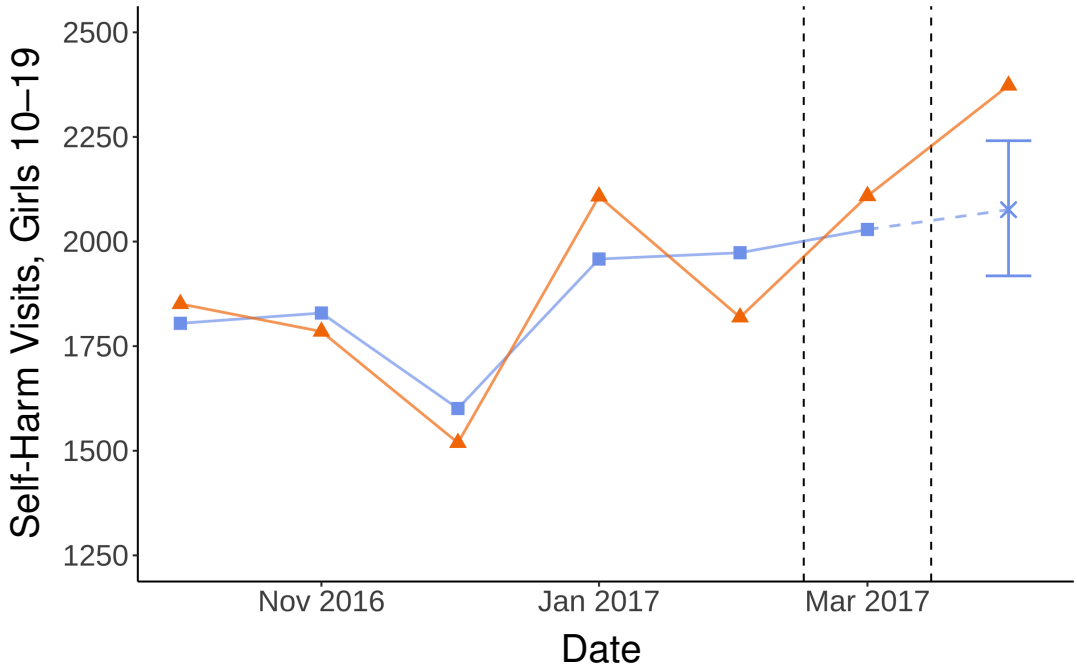

Figure B.21: ITS Analysis Using Unweighted Counts of Self-Harm Visits. The ARIMA model is selected using `auto.arima()` on the full pre-treatment series. The forecasted point is shown with 95% conformal prediction intervals and block size = 1.

B.8 ITS Plot for Teen Boys with an Alternative ARIMA Specification

For teen girls, men 40–65, and women 40–65, AIC-guided stepwise selection on the raw time series produces ARIMA specifications that include both first- and seasonal-differencing. But for teen boys, it produces a specification with only seasonal-differencing. The plot in the main paper includes first- and seasonal-differencing, as my general approach to ITS includes performing the differencing first.[3] Figure B.22 shows the results using the model chosen without forced differencing.

[3]This allows us to use forecasters other than ARIMA; in this particular application, I exclusively use ARIMA models since they performed the best in the exploratory dataset.

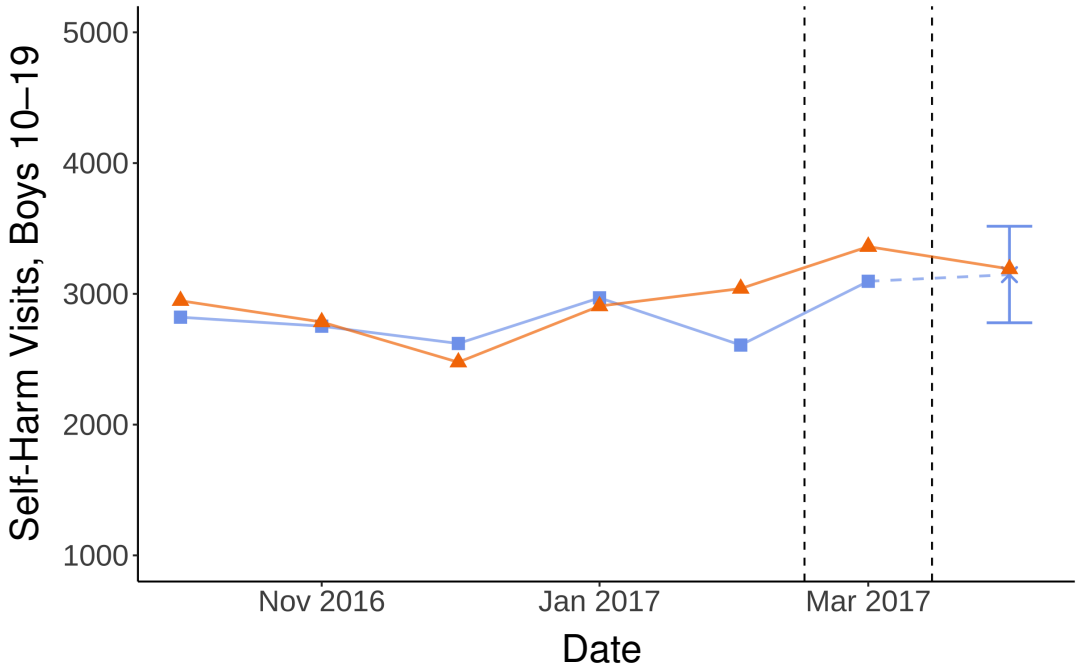

Figure B.22: ITS Analysis for Teen Boys Using Alternative ARIMA Specification. The ARIMA model is selected using `auto.arima()` on the full pre-treatment series without forcing differencing. The algorithm selects seasonal-differencing but not first-differencing. The forecasted point is shown with 95% conformal prediction intervals and block size = 1.

B.9 ER Visits Coded as Suicide Attempts

A relatively small fraction of ER visits for intentional self-harm are coded as "suicide attempts," but only after the introduction of ICD-10 in October 2015. Figure B.23 shows counts of these visits for teen girls. The series is quite noisy with different trajectories across years. Nonetheless, the plot supports the main finding. There is a sharp increase between March and April of 2017, and counts remain elevated in May 2017.

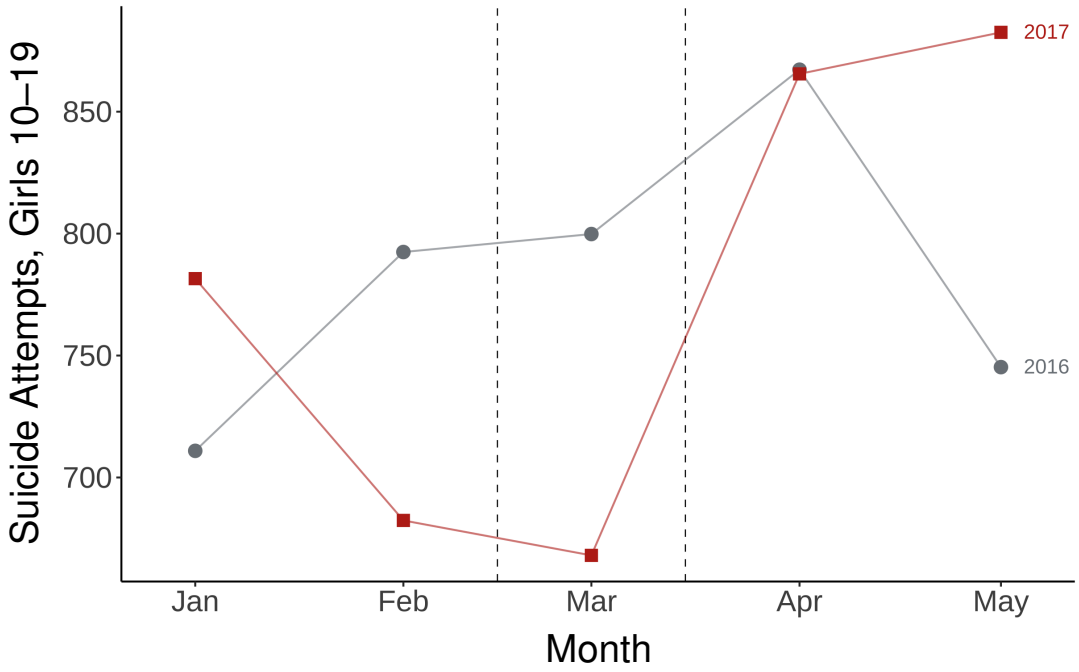

Figure B.23: ER Visits Coded as Suicide Attempts, Girls 10–19. The series is much noisier than the series of self-harm visits. The sharp, post–March 2017 increase in counts is consistent with the main findings.

C Measurement Error

In this section, I provide evidence that measurement error in the time series produces little to no bias in the estimated treatment effect. The evidence is simulation-based, and should thus be taken with a grain of salt: the true measurement error function is unknown, so it is unclear how well the simulated measurement error approximates reality. However, across six different simulated measurement error functions, the estimated bias is extremely close to zero, which should largely assuage concerns about measurement error in the observed time series. Code for replicating all simulations can be found at `https://github.com/cmfelton/13rw`.

There are at least two sources of measurement error in the series. First, the counts are estimated from a sample, not known. Second, many observations are dropped due to missing data on month of visit, age, and gender, as shown in Table C.1, which may bias the estimated counts toward zero. It is also worth noting that, because the NEDS draws a new sample of hospitals each year, the measurement error may vary by year. All six simulated measurement error functions account for this yearly variation.

Year	**Total Obs.**	**Complete Obs.**	**% Dropped**
2006	25,702,597	20,852,495	18.9%
2007	26,627,923	22,085,226	17.1%
2008	28,447,148	23,936,411	15.9%
2009	28,861,047	24,396,152	15.5%
2010	28,584,301	24,091,391	15.7%
2011	28,788,399	24,459,034	15.0%
2012	31,091,020	25,930,759	16.6%
2013	29,581,718	24,490,491	17.2%
2014	31,026,417	26,247,242	15.4%
2015 (Q1–Q3)	29,696,059	27,200,272	8.4%
2015 (Q4)	5,837,947	4,986,489	14.6%
2016	32,680,232	29,078,732	11.0%
2017	33,506,645	29,321,114	12.5%

Table C.1: Number of total observations and complete observations (in gender, month, and age) by year.

Before describing the procedure in detail, I will review the basic logic behind it. The observed time series and estimated treatment effect in the main study will be

treated as the "ground truth." I then bootstrap that series and add additional measurement error in each iteration. If the ITS estimator is able to recover the true simulated treatment effect (on average) in spite of the measurement error, then the simulated measurement error produces no bias in our treatment effect estimates. If the simulated measurement-error-generating process is similar to the true (but unknown) measurement-error-generating process, we should be more confident that the measurement error in the observed time series does not bias the ITS estimator.

The simulation procedure, in broad terms, works as follows:

1. Simulate a time series of potential outcomes under control $Y_t^*(0)$ of length T.
2. For the post-treatment period T, add a hypothetical treatment effect $\tilde{\tau}$ to the simulated potential outcome under control ($Y_T^*(0)$) to obtain a simulated potential outcome under treatment ($Y_T^*(0) + \tilde{\tau} = Y_T^*(1)$). Across all simulations in this appendix, I set $\tilde{\tau}$ to 1,297, the estimated treatment effect in the study.
3. Simulate measurement errors ϵ_t and them to the full simulated time series.
4. Apply the ITS procedure used in the main paper to the simulated time series:
   (a) Run `auto.arima()` on the simulated pre-treatment series to obtain a forecasting model.
   (b) Using the selected and fitted model, forecast the post-treatment potential outcome under control $\widehat{Y}_T^*(0)$.
   (c) Obtain a treatment effect estimate by subtracting the forecasted outcome from the observed outcome: $Y_T^*(1) - \widehat{Y}_T^*(0) = Y_T^* - \widehat{Y}_T^* = \hat{\tau}$.
   (d) Store the estimated treatment effect.
5. Repeat Steps 1–4 $m = 5,000$ times.
6. Compute the bias as the sample mean of $\hat{\tau}$ minus $\tilde{\tau}$.

Next, I describe how I simulate the time series.

C.1 Simulating the Time Series

Simulating time series data for this exercise is not straightforward. One option is to simulate the data from an ARIMA model using a function like `sarima.sim()` from the `astsa` package, but this can result in the simulated time series having drastically different variance from replication to replication, even when the errors are drawn from the same distribution. This will in turn make the simulated treatment effect more or less pronounced across iterations. It can also result in time series that look very different from the series in question.

As an alternative, I use the bootstrapping procedure described in Bergmeir et al. (2016) to simulate time series. The procedure works as follows:

1. Transform the time series using the Box–Cox transformation.
2. Decompose the series into seasonal, trend, and remainder components using the STL procedure of Cleveland et al. (1990).
3. Shuffle blocks of the remainder to get a new remainder series.
4. Add the new remainder series to the trend and seasonal components from Step 2 to get a simulated, Box–Cox-transformed series.
5. Reverse the Box–Cox transformation on the simulated series.
6. Repeat Steps 3–5 $m = 5,000$ times to produce m time series.

I use the `bld.mbb.bootstrap` function from the `forecast` package to generate bootstrapped simulations. I use a block size of 3, but results in Section D suggest the block size matters very little.

I apply this bootstrapping procedure to a partly synthetic time series consisting of (i) the full pre-treatment time series and (ii) the one-step-ahead forecast from the main ARIMA specification used in the paper for the post-treatment period. This partly synthetic time series represents the "observed" series of potential outcomes under control, which I call $\breve{Y}_t(0)$. Each bootstrap replication represents a simulated series of potential outcomes under control $Y_t^*(0)$. The reason I do not bootstrap the full observed time series is that the treatment effect will get picked up as part of the remainder component in the STL decomposition and thus be shuffled into the simulated time series $Y_t^*(0)$.

C.2 Simulating Measurement Errors

I consider six measurement-error-generating processes and report the estimated bias across 5,000 simulations:

1. I draw one measurement error per year from a $\mathcal{N}(0, 200)$ distribution, where $\mathcal{N}(\mu, \sigma)$ represents the Gaussian probability distribution with mean μ and standard deviation σ. The yearly measurement error is applied to each month in that year. I.e., if the first draw is 50, we add 50 to each month in 2006. **Estimated bias:** $1,309.048 - 1,297 \approx$ **12.0 (0.9%)**.
2. I draw one measurement error per year from a $\mathcal{U}_{[-300,300]}$ distribution, where $\mathcal{U}_{[a,b]}$ represents the continuous uniform probability distribution with minimum a and maximum b. The yearly measurement error is applied to each month in that year. **Estimated bias:** $1,303.737 - 1,297 \approx$ **6.7 (0.5%)**.
3. I draw one mean μ_j per year from a $\mathcal{U}_{[-300,300]}$ distribution, and then 12 monthly measurement errors ϵ_t for that year from a $\mathcal{N}(\mu_j, 100)$ distribution. **Estimated bias:** $1,301.337 - 1,297 \approx$ **4.3 (0.3%)**.

4. Same as (1), but the yearly errors are drawn from a $\mathcal{N}(-200, 200)$ distribution. **Estimated bias:** $1,310.312 - 1,297 \approx$ **13.3 (1%)**.

5. Same as (2), but the yearly errors are drawn from a $\mathcal{U}_{[-500,100]}$ distribution. **Estimated bias:** $1,297.978 - 1,297 \approx$ **0.98 (0.08%)**.

6. Same as (3), but the yearly means are drawn from a $\mathcal{U}_{[-500,100]}$ distribution. **Estimated bias:** $1,305.899 - 1,297 \approx$ **8.9 (0.7%)**.

The true measurement-error-generating process will obviously differ from the simulated process. But the fact that the estimated bias is extremely close to zero across all six settings provides us with decent assurance that the true measurement-error-generating process creates little to no bias for the ITS estimator.

D Assessing Type-M and Type-S Error

In this section, I show that type-M and type-S error pose only a small threat to my analysis. High-variance estimators can produce exaggerated effect sizes, conditional on the estimates being statistically "significant." Gelman and Carlin (2014) introduce the *expected type-M error*, where the m stands for $m$agnitude, to quantify this conditional exaggeration. The expected type-M error is calculated by dividing the expectation of the absolute value of an estimate by the true effect size, conditional on the estimate being statistically significant. In practice, we don't know the true effect size, but we can calculate the expected type-M error under different hypothetical values for this quantity. They define the *type-S error rate* as the probability a statistically significant estimate has the wrong $s$ign, which also requires specifying a hypothetical true effect size.

I conduct simulations to estimate the expected type-M error, type-S error, and power under different hypothetical effect sizes. Code for replicating all simulations can be found at `https://github.com/cmfelton/13rw`. The procedure is similar to that used in Section C. It works as follows:

1. Simulate a time series of potential outcomes under control $Y_t^*(0)$ of length T.
2. For the post-treatment period T, add a hypothetical treatment effect $\tilde{\tau}$ to the simulated potential outcome under control ($Y_T^*(0)$) to obtain a simulated potential outcome under treatment ($Y_T^*(0) + \tilde{\tau} = Y_T^*(1)$).
3. Apply the ITS procedure used in the main paper to the simulated time series:
   (a) Run `auto.arima()` on the simulated pre-treatment series to obtain a forecasting model.
   (b) Using the selected and fitted model, forecast the post-treatment potential outcome under control $\widehat{Y}_T^*(0)$.
   (c) Obtain a treatment effect estimate by subtracting the forecasted outcome from the observed outcome: $Y_T^*(1) - \widehat{Y}_T^*(0) = Y_T^* - \widehat{Y}_T^* = \hat{\tau}$.
   (d) Compute a conformal p-value for the observed, simulated outcome Y_T^*.
4. Repeat Steps 1–3 $m = 5,000$ times.
5. Calculate the expected type-M error as follows:
   (a) Subset to estimates for which $p < 0.05$.
   (b) Calculate the mean of $|\hat{\tau}|$ among this subset and divide by the true treatment effect $\tilde{\tau}$.
6. Calculate the type-S error rate as follows:

(a) Subset to estimates for which $p < 0.05$.

(b) Calculate the proportion of estimates that have the wrong sign (i.e., are negative) among this subset.

7. Calculate power as follows:

   (a) Calculate the proportion of all estimates for which $p < 0.05$.

8. Repeat Steps 1–7 across five different hypothetical effect sizes $\tilde{\tau}$, described below.

Next, I explain how the simulations work, and then I describe how I chose effect sizes.

D.1 Simulating the Time Series

For the same reasons described in Section C, I use the `bld.mbb.bootstrap` function from the `forecast` package to generate bootstrapped simulations of the time series. I perform the entire procedure across two block sizes: 24 (the software default, 2 × the time series frequency) and 3. Results are extremely similar across the two block sizes, so I explore no other block sizes. As in Section C, and for the same reasons described there, I bootstrap a partly synthetic time series consisting of (i) the full pre-treatment time series and (ii) the one-step-ahead forecast from the main ARIMA specification used in the paper for the post-treatment period.

D.2 Choosing a Hypothetical Effect Size

Once I obtain a simulated control series, I add a hypothetical treatment effect $\tilde{\tau}$ to the simulated post-treatment value. To choose reasonable values for $\tilde{\tau}$, I first looked to Fink et al. (2018)'s study of the effects of Robin Williams's suicide on suicide mortality. Fink et al. (2018) found that in August 2014, suicide mortality among people aged 30–44 was 18.3% higher than in the predicted counterfactual (see Table 1 in Fink et al. (2018)). It is worth noting, however, that Robin Williams died in the middle of August, so this may underestimate the one-month treatment effect. I expect this effect estimate to be less vulnerable to type-M errors for two reasons. First, the authors use population-level CDC mortality data, whereas the self-harm counts I use are estimated from a sample. As a consequence, we should expect that my time series is noisier, and thus my ITS estimator higher variance. Second, because Robin Williams was widely beloved and his death widely covered, a large effect size for his suicide is plausible.

In the simulations, I vary the value of $\tilde{\tau}$ as a function of the Fink et al. (2018) estimate as follows:

1. For some value $0 < \delta \leq 1$, I calculate $0.183 \times \delta \times \breve{Y}_T(0) = \breve{Y}_T(1)$. Recall that $\breve{Y}_T(0)$ is just the one-step-ahead forecast from the ARIMA model used in the main analysis. 0.183 represents the Robin Williams effect size.

2. Calculate $\tilde{\tau} = \breve{Y}_T(1) - \breve{Y}_T(0)$.

I then use this same value $\tilde{\tau}$ across all $m = 5,000$ simulations. I conduct simulations across five values of δ: 0.2, 0.4, 0.6, 0.8, and 1. Figures D.1–6 show the results. Only for $\delta = 0.2$ are the results especially concerning.

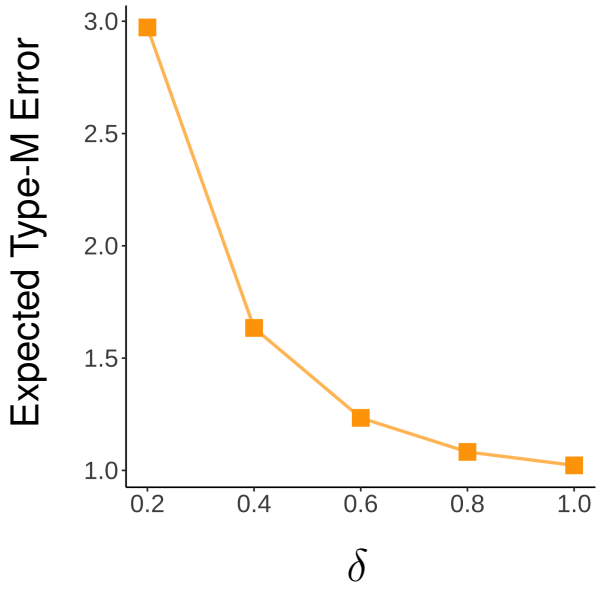

Figure D.1: Expected type-M error as a function of effect size with block size = 24. δ represents the magnitude of the assumed effect size relative to the effect of Robin Williams's death on suicide mortality among people age 30–44.

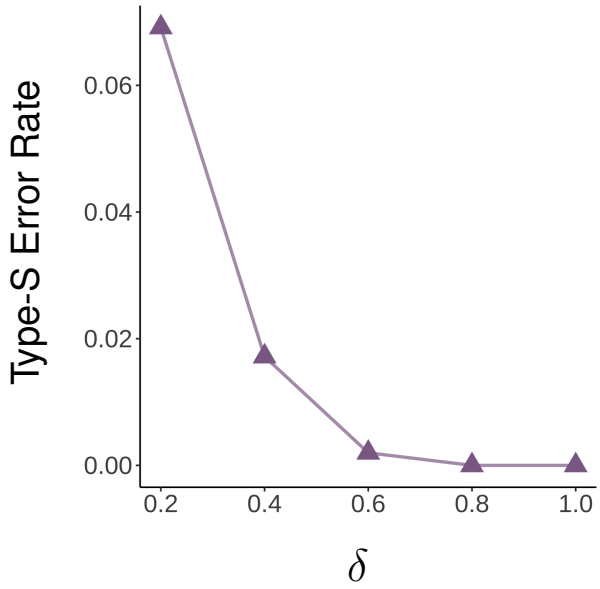

Figure D.2: Type-S error rate as a function of effect size with block size = 24.

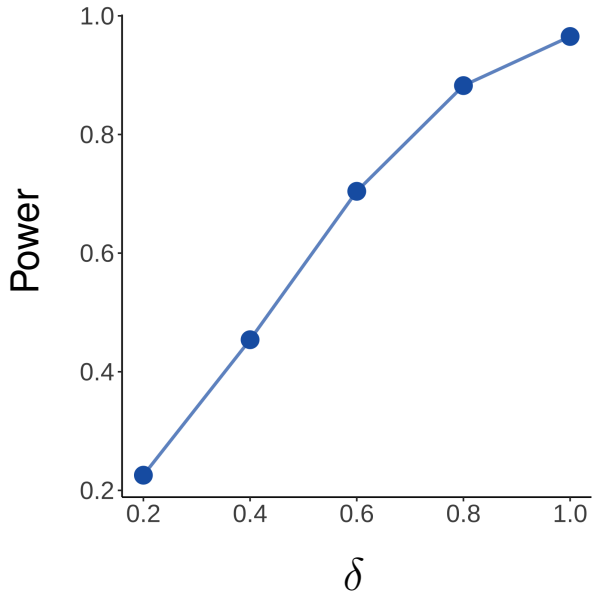

Figure D.3: Power as a function of effect size with block size = 24.

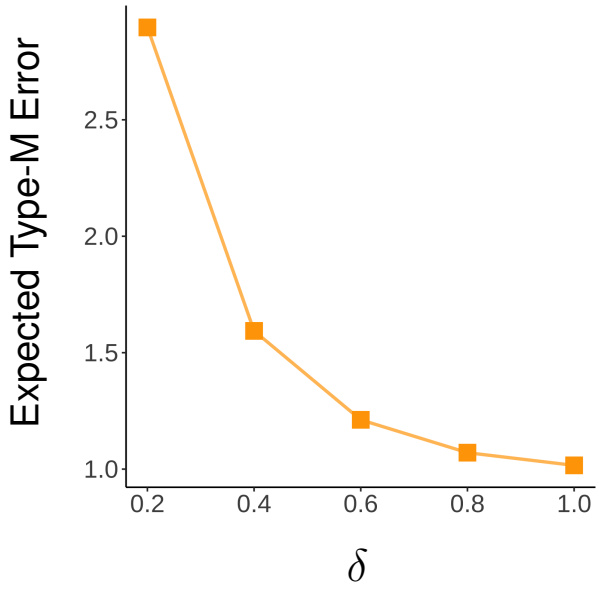

Figure D.4: Expected type-M error as a function of effect size with block size = 3.

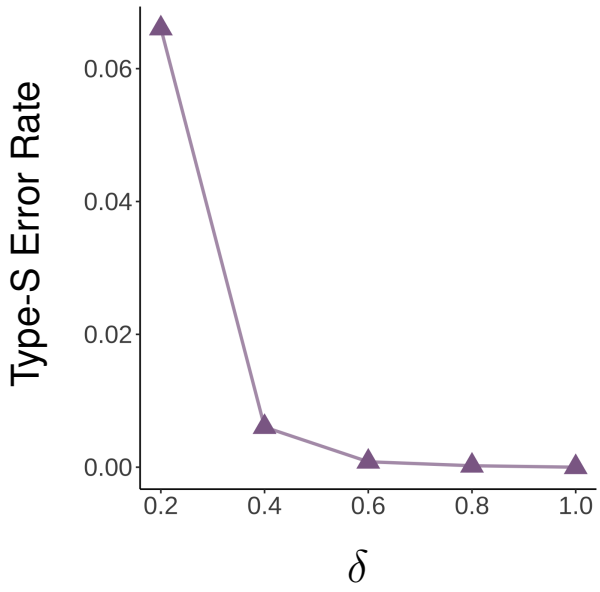

Figure D.5: Type-S error rate as a function of effect size with block size = 3.

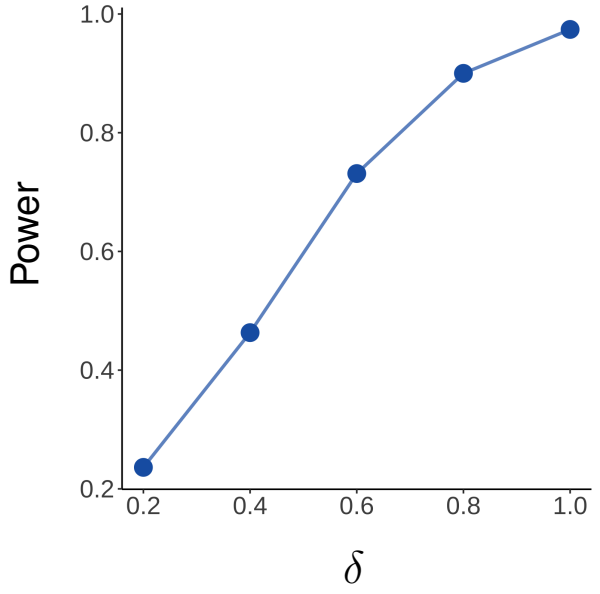

Figure D.6: Power as a function of effect size with block size = 3.

E Methodological Details

In this section, I outline the general approach I use to conduct the ITS analysis. It is a general approach because it can incorporate any forecasting model, from simple autoregressions to gradient-boosted trees and neural networks. The approach seeks to solve three problems that might affect ITS analyses: (i) poor predictive accuracy and invalid inference from non-stationary data; (ii) poor predictive accuracy and invalid inference from data-driven model selection; and (iii) invalid inference from model misspecification. First- and seasonal-differencing solve the first problem. Exploratory datasets and sample-splitting solve the second problem. Conformal inference solves the third problem.

I found that parts of the procedure—like sample-splitting and conformal inference—matter little in this particular application. They might, however, make a difference in other settings where overfitting is more likely or the dataset much larger.

In the next section, I sketch out this approach in detail, described graphically in Figure E.1. In the section that follows, I provide a brief tutorial on how conformal inference works.

E.1 Methodological Approach

As Romer (2020) points out, secular trends, if unaccounted for, can lead to poor out-of-sample predictive accuracy with forecasting models. Furthermore, many inferential methods—including conformal inference—assume some sort of stationary for validity. While certain forecasting methods like ARIMA can difference the data under the hood, most do not. For instance, if we want to use gradient-boosted trees to forecast, we should make sure our data is stationary first. First- and seasonal-differencing often succeed in making data stationary, and we can use a combination of visual inspection and formal statistical procedures to assess stationarity (see Hyndman and Athanasopoulos (2018) for a more thorough discussion). In this study, I use ARIMA models that automatically difference the data and automatically de-difference (i.e., integrate) the predicted values.

It is worth providing more context for the discussion of overfitting. ITS provides us with causal identification when we know the true model for the potential outcomes under control. In practice, we lack knowledge of the true data-generating process. A first thought might be to use our time series to select the model from a large set of models according to some goodness-of-fit metric. But using the same data to both select and estimate a model comes with serious drawbacks. We might overfit the data, leading to poor out-of-sample prediction and invalid prediction intervals (Freedman, 1983; Faraway, 2016; Berk, 2021). This problem can be particularly severe when we use machine learning estimators (or simply select parametric models based on goodness-of-fit metrics that fail to account for model complexity like R^2). I propose two procedures to avoid this problem.

First, we can use a distinct time series with similar trends and seasonality as an

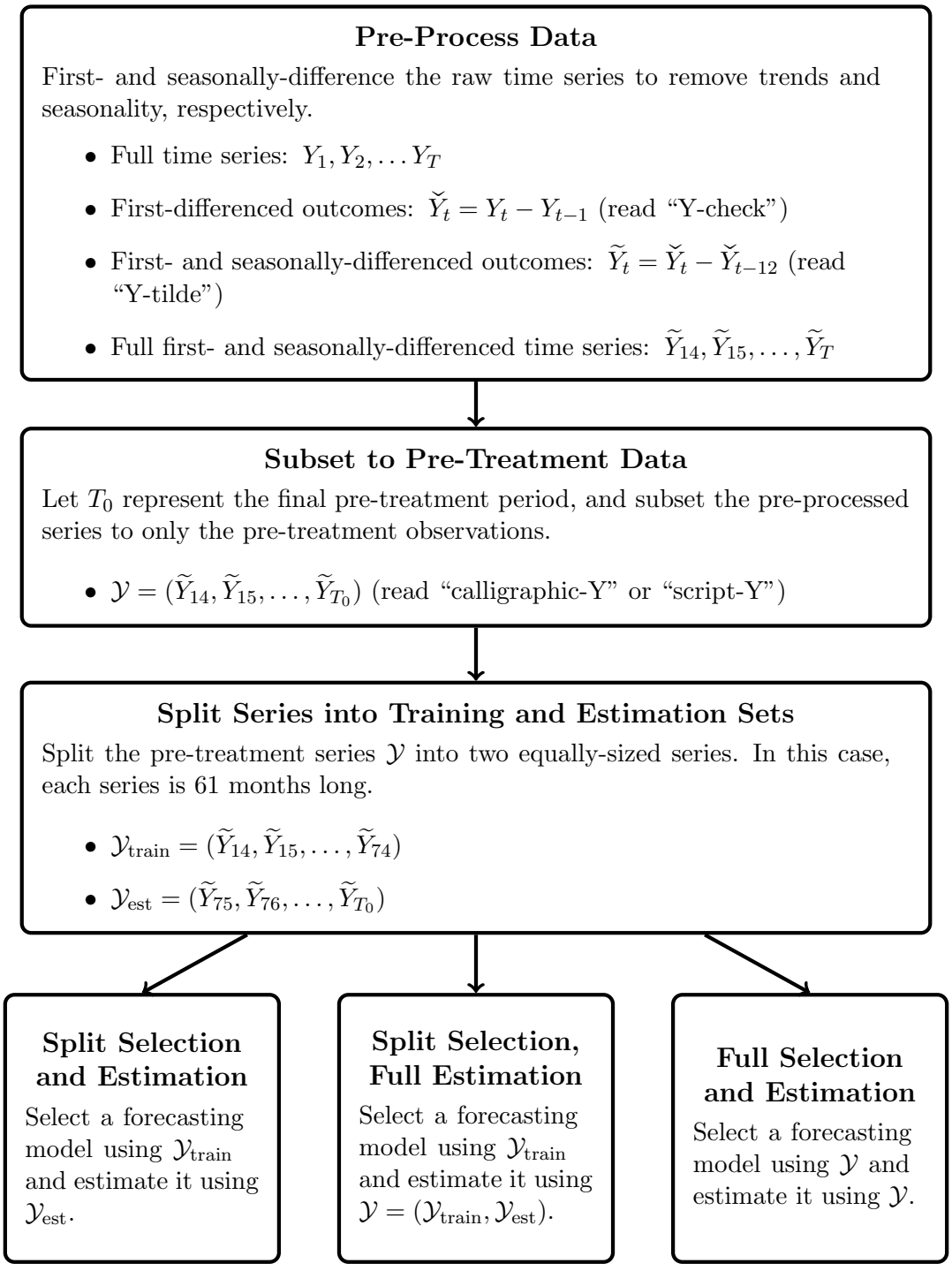

Figure E.1: Diagram illustrating model selection and estimation procedures. The three separate selection–estimation procedures yield extremely similar results in this setting.

exploratory dataset. In this application, I used CDC suicide mortality data (for all people, not just teen girls) to assess a variety of forecasting methods. I used this

data to select a model class, namely, ARIMA models with no additional covariates.[4] I found that using AIC-guided stepwise selection (implemented with the `R` function `auto.arima()` from the `forecast` package) to choose a model order worked well in the CDC data, and in extensive simulations, I found that the threat of overfitting with this procedure was very low.[5]

A second step we can take to avoid overfitting is to split the pre-treatment time series in half (after differencing). For simple parametric models, we can use the first half of the time series to select a model specification and the second half to fit the model. For machine learning estimators, we could use the first half to fit the model and use the second half as inputs to the fitted model. In Section B, I showed results from three model-selection-and-estimation strategies. *Full* uses the entire pre-treatment series to both select and estimate the model; *Split* uses the first half of the pre-treatment series to select the model and the second half to fit the model; *Split Analysis, Full Estimation (SAFE)* uses the first half to select the model and the full pre-treatment set to fit the model (Faraway, 2016).

The main analysis in the paper uses the *Full* approach for two reasons. First, simulations suggested using the full time series improves predictive accuracy without harming inference (see Section F below). Second, repeated placebo tests with small pre-treatment series suggested conformal prediction intervals were less stable (see Section B).

Once we have made our data stationary and taken care to avoid overfitting, we can forecast into the post-treatment period and estimate the treatment effect. But a problem remains: our model is likely misspecified, and standard prediction intervals assume the correct specification for validity. Conformal inference provides a solution to this problem (Shafer and Vovk, 2008; Lei et al., 2018; Chernozhukov et al., 2021). In the standard, i.i.d. setting, conformal inference provides exact finite-sample validity under no assumptions other than exchangeability (implied by i.i.d. data) and estimator symmetry (invariant with respect to permutations in the data). In the time series setting, where data points exhibit dependence, we require more assumptions, and we achieve only approximate finite-sample validity. However, these assumptions are weaker than those required for conventional prediction intervals, which lack finite-sample guarantees. I employ Chernozhukov et al.'s (2021) method of permuting blocks of residuals to compute conformal p-values (see also Chernozhukov et al. (2018)). It should also be noted that conformal inference is a general procedure: it can be used with complex machine learning methods that lack default, "model-trusting" prediction intervals.[6]

The conformal prediction intervals I use are approximately valid in finite samples under two sets of conditions (Chernozhukov et al., 2021). The first requires a consistent estimator and stationary, strongly mixing errors. The second requires estimator sta-

[4]Additional covariates might include measures of temperature and economic indicators. I found these provided no improvement to basic ARIMA models trained on the time series.

[5]Some of these simulations are reported in Section F. Many others are unreported but show similar results.

[6]I borrow the term "model-trusting" from Buja et al. (2019).

bility and that the data (not the errors) are weakly stationary and β-mixing with the mixing coefficient β satisfying an inequality described in Chernozhukov et al. (2021). Because I assume all models are misspecified, I focus on the second set of conditions.

Estimator stability requires that forecasts change very little when perturbing a small handful of observations. Weak stationarity in X_t requires that (i) $\mathrm{E}[X_t]$ is independent of t and (ii) $\mathrm{Cov}[X_t, X_{t+h}]$ is independent of t for each h. This means roughly that the mean and variance of the time series is constant over time, and it can be violated by secular trends, seasonality, and abrupt level changes (see Hyndman and Athanasopoulos (2018) for a more thorough discussion with visual examples). The stationarity assumption is made more plausible by differencing the data (although, again, the data is de-differenced for all the plots).[7]

Mixing requires that as the distance between two points increases, some measure of dependence approaches 0—in other words, X_t and X_{t+h} are asymptotically independent in h (Mcdonald et al., 2011). The rough intuition is that two time periods that are close together can be dependent, but time periods that are far apart should be independent. The β in "β-mixing" refers to a specific measure of dependence. See Bradley (2005) for formal definitions of different types of mixing, including β-mixing.

Simulations reporting in Section F show that conventional ARIMA prediction intervals undercover under the null, while conformal intervals provide valid coverage and sometimes overcover. The repeated placebo tests reported in Figure 9 in the main paper, however, suggest that conformal intervals may still undercover in this particular setting.

E.2 How Conformal Inference Works

I illustrate how conformal inference works visually in Figure E.2 and Figure E.3 and formally below.

Let T represent the length of the full time series and T_0 represent the final pre-treatment period. Suppose we have a single pre-treatment period such that $T_0 = T - 1$. We first fit a model to the series of pre-treatment outcomes $\mathcal{Y} = Y_1, Y_2, \ldots, Y_{T_0}$. The goal is to obtain a prediction interval for Y_T, the post-treatment period. The first step in obtaining conformal prediction intervals is computing a conformal p-value for a *candidate* value for Y_T. Call this candidate value Y_T^*. Augment the pre-treatment time series with Y_T^* such that we now have the series $\mathcal{Y}^* = Y_1, Y_2, \ldots, Y_{T_0}, Y_T^*$.

Refit the forecasting model to the augmented series $\mathcal{Y}^*$ and compute the absolute residuals $\hat{u}_t = |\hat{\epsilon}_t|$. Let $\mathcal{R}$ be the vector of absolute residuals $\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_T$, where $\hat{u}_T$ is the absolute residual for Y_T^*. The conformal p-value is the proportion of absolute residuals in $\mathcal{R}$ that $\hat{u}_T$ is less than or equal to:

[7] More exactly, ARIMA automatically differences the data for estimation and then de-differences or *integrates* the fitted values.

$$p = \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}(\hat{u}_T \leq \hat{u}_t),$$

where $\mathbf{1}()$ is the indicator function. Because $\mathcal{R}$ contains $\hat{u}_T$, the p-value can never be 0, as $\mathbf{1}(\hat{u}_T \leq \hat{u}_T) = 1$. Figure E.2 illustrates this procedure.

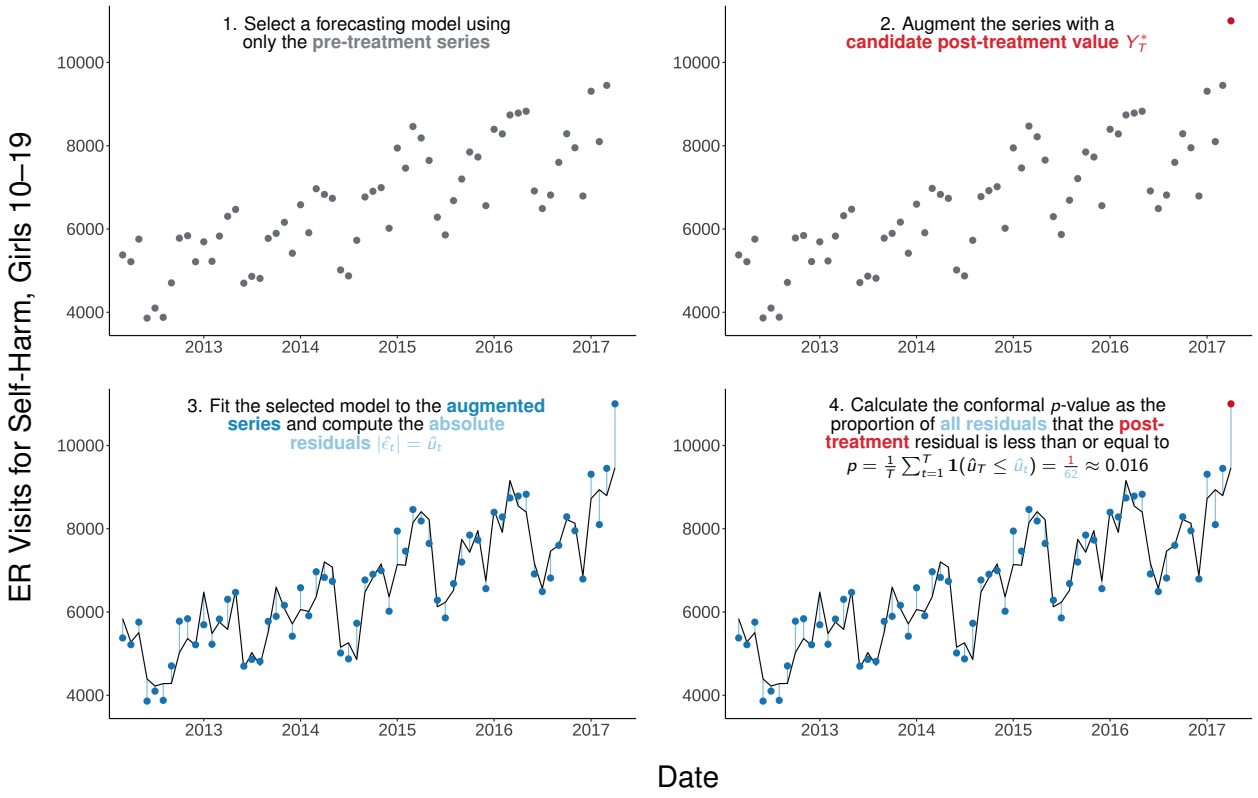

Figure E.2: Grid of plots illustrating how to calculate a conformal p-value for a candidate value for Y_T, denoted Y_T^*. Here $Y_T^* = 11,000$. Note that the set of all residuals includes the residual for Y_T^*, $\hat{\epsilon}_T$.

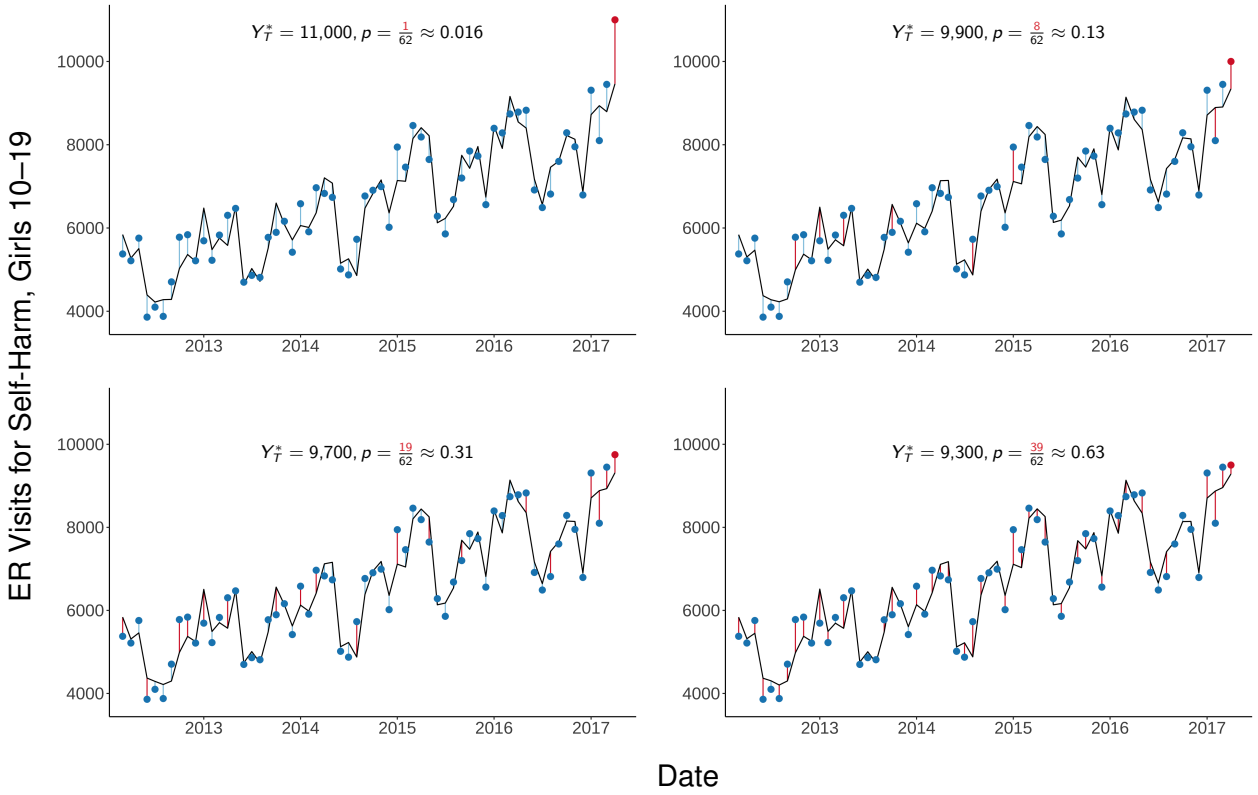

Figure E.3: Grid of plots illustrating a test inversion procedure for generating conformal prediction intervals. For each candidate value Y_T^*, we refit the predictive model and calculate a new p-value. The prediction interval contains all values of Y_T^* for which $p > \alpha$, where α represents the desired probability of falsely rejecting the null hypothesis.

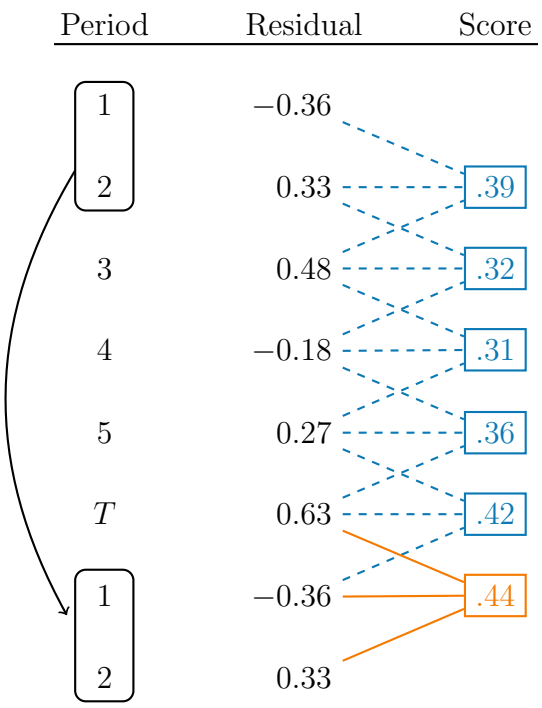

Figure E.4: Toy example illustrating the overlapping block scheme for conformal inference. We have five pre-treatment periods ($t = 1, 2, 3, 4, 5$) and one post-treatment period ($t = T$). The nonconformity score is the mean absolute residual from 3 consecutive periods, with lower values indicating the period better conforms to the model. To calculate a conformal p-value for the treatment period, we calculate the proportion of scores (including both **blue** and **orange** scores) that the the **orange** score is less than or equal to. In this example, the orange score is only less than or equal to 1 out of 6 scores (itself), making the p-value $1/6 = 0.1\overline{6}$.

A word of caution about using conformal inference with ARIMA models. An ARIMA model will automatically difference and un-difference (i.e., integrate) the data. When we difference data, the initial observations will become missing. I.e., if we first-difference a time series beginning on January 2006, we will lack a first-differenced observation for January 2006. ARIMA software will nonetheless return residuals for these missing observations that are extremely close to zero. These should be removed from the set of residuals when conducting conformal inference.

Once we know how to calculate conformal p-values, we can obtaining conformal prediction intervals through test inversion. We consider a fine grid of candidate values for Y_T^* and calculate p-values for each value in the grid. The interval consists of all values of Y_T^* for which $p > \alpha$, where α represents the desired probability of falsely rejecting the null hypothesis. Figure E.3 illustrates this procedure with four candidate values for Y_T^*. In practice, I use the sequence of values $\hat{Y}_T - 2{,}000, \hat{Y}_T - 1{,}999, \ldots, \hat{Y}_T + 1{,}999, \hat{Y}_T + 2{,}000$ to generate prediction intervals.[8] Confidence intervals on the treatment effect

[8]For the repeated placebo test procedures, I sometimes had to expand the width of this grid. In general, it is good practice to check whether the lower or upper limit of the prediction interval is equal

are obtained by subtracting the upper and lower limits of the prediction interval as in (Chernozhukov et al., 2021).

These prediction intervals are valid with i.i.d. data. In order to generate prediction intervals for dependent data, I use blocks of residuals rather than individual residuals to conduct inference (Chernozhukov et al., 2018), illustrated in Figure E.4. In practice, I find that these intervals with block sizes greater than 1 are somewhat unstable, so the main paper reports only intervals with a block size of 1.[9] As I show in Section B, however, confidence intervals and p-values for the treatment effect are very similar with alternative block sizes.

to the maximum or minimum value in the grid, which would suggest the grid should be wider.

[9]In Chernozhukov et al. (2018) and Chernozhukov et al. (2021), the authors focus on prediction intervals using a block size of 1, and reserve the use of larger block sizes for p-value computation only.

F Sample-Splitting Simulations

This section shows that different model-selection-and-estimation procedures yield similar results. The simulated data and selection procedure are *specifically tailored to this study*. The simulated time series are monthly and exhibit strong seasonality and trends over time, just as the observed series of self-harm visits does. Model selection is always done using the `auto.arima()` function with no additional covariates (e.g., no weather or economic predictors), as no other forecasting models and no other covariates were considered in this study.[10] *These simulations should* ***not*** *be used to guide general ITS practice.* Instead, they serve to assess how the specific model-selection strategies used in this study would perform if repeated across many similar settings. Code for replicating all simulations can be found at `https://github.com/cmfelton/13rw`.

Time series are simulated using the `sarima.sim()` function from the `astsa` package. The function generates a time series from an ARIMA model given a set of model parameters and a simulated error distribution. For the results I show below, I use the specification and parameter estimates from the main model used in the analysis (see Table B.7). In other, unreported simulations, I used different specifications and parameter values, and the results were essentially unchanged. I consider two different sample sizes and three different error distributions. The first sample size is 135, matching the length of the time series for this study, and the second is 51, to see whether using a shorter time series substantively changes the results. I consider errors drawn from a Gaussian distribution, a uniform distribution, and a t distribution with 3 degrees of freedom (i.e., a t_3 distribution). The latter two will produce time series with more outliers than the former. In unreported simulations, I drew errors from Laplace and Cauchy distributions, and the results were essentially the same.

As discussed in Section E.1, I use three different model-selection procedures: *Full*, *Split*, and *SAFE*. *Full* uses the entire pre-treatment series to select a model using `auto.arima()`, then fits this model specification to the entire pre-treatment time series. *Split* cuts the time series in half, using the first half to select a model specification and the second half to estimate the model parameters. The fitted model used for forecasting thus uses a series half as long as the series used in *Full*. I borrow the term *SAFE*—Split Analysis, Full Estimation—from Faraway (2016). *SAFE* uses only the first half of the time series to select a model (like *Split*) but the entire pre-treatment time series to fit the model (like *Full*).

I find that *Full* yields greater predictive accuracy than *Split* or *SAFE* without invalidating inference. Conformal inference with *Split* is always more conservative than with *Full* or *SAFE*—that is, *Split* tends to slightly overcover. This may have more to do with the sample size than the selection procedure, however. As Lei et al. (2018) prove (in the i.i.d. setting), conformal inference can overcover in small samples, and the coverage can be bounded by $1 - \alpha + \frac{1}{n+1}$. (E.g., when $n = 99$, 95% conformal

[10] In other, unreported simulations, I selected autoregressive models with covariates using AIC-guided step-wise selection, where most or all of the covariates are irrelevant. Results were essentially the same.

intervals will cover the truth *at most* 96% of the time because $1 - .05 + \frac{1}{99+1} = .96$. When $n = 49$, they might cover the truth up to 97% of the time.) Furthermore, *Full* with $n = 51$ overcovers more than *Split* with $n = 135$ when errors are Gaussian.

The simulations also suggest that conformal inference provides better coverage than conventional ARIMA intervals, although the improvements are modest. Note also that the assumptions required for conformal validity are satisfied by design in the simulations, but they may not hold in the real world.

The simulation results provide some evidence that, in this study, confidence intervals from *Split* are slightly more conservative and that effect estimates from *Full* are slightly more accurate. As shown in Figure B.1, results estimated treatment effects and their confidence intervals remain largely unchanged across selection procedures.

Approach	MSE	RMSE	ARIMA	Conformal		
			$p < .05$	$p < .05$	$p < .1$	$p < .2$
Full	**1.0442**	**1.0219**	**0.0606**	0.0488	0.1024	0.1974
Split	1.0801	1.0393	0.0646	**0.0454**	**0.0866**	**0.1880**
SAFE	1.0475	1.0235	0.0646	0.0468	0.1008	0.1922

Table F.1: Gaussian errors, $n = 135$.

Approach	MSE	RMSE	ARIMA	Conformal		
			$p < .05$	$p < .05$	$p < .1$	$p < .2$
Full	**1.0656**	**1.0323**	0.0652	0.0378	0.0966	0.1992
Split	1.3259	1.1515	**0.0628**	**0.0362**	**0.0698**	**0.1738**
SAFE	1.3038	1.1418	**0.0628**	0.0390	0.0962	0.1904

Table F.2: Gaussian errors, $n = 51$.

Approach	MSE	RMSE	ARIMA	Conformal		
			$p < .05$	$p < .05$	$p < .1$	$p < .2$
Full	**2.5200**	**1.5875**	**0.0534**	0.0432	0.0964	0.1946
Split	2.6685	1.6335	0.0610	**0.0396**	**0.0874**	**0.1918**
SAFE	2.5475	1.5961	0.0610	0.0424	0.0958	0.1966

Table F.3: t_3 errors, $n = 135$.

Approach	MSE	RMSE	ARIMA	Conformal		
			$p < .05$	$p < .05$	$p < .1$	$p < .2$
Full	**3.3814**	**1.8389**	**0.0714**	0.0396	0.1018	0.2034
Split	4.0660	2.0164	0.0766	**0.0378**	**0.0764**	**0.1928**
SAFE	3.7094	1.9260	0.0766	0.0406	0.1006	0.2022

Table F.4: t_3 errors, $n = 51$.

Approach	MSE	RMSE	ARIMA	Conformal		
			$p < .05$	$p < .05$	$p < .1$	$p < .2$
Full	**0.1471**	**0.3835**	**0.0482**	0.0498	0.0982	0.2040
Split	0.1523	0.3903	0.0512	**0.0476**	**0.0916**	**0.1962**
SAFE	0.1486	0.3854	0.0512	0.0464	0.1022	0.2032

Table F.5: Uniform errors, $n = 135$.

Approach	MSE	RMSE	ARIMA	Conformal		
			$p < .05$	$p < .05$	$p < .1$	$p < .2$
Full	**0.1563**	**0.3953**	0.0644	0.0464	0.1136	0.2102
Split	0.1686	0.4106	**0.0606**	**0.0430**	**0.0796**	**0.1992**
SAFE	0.1657	0.4071	**0.0606**	0.0460	0.1102	0.2042

Table F.6: Uniform errors, $n = 51$.

G Anticipation Effects

This section formalizes the idea of anticipation effects in ITS designs and clarifies why it is problematic. The discussion has two key takeaways. First, in the presence of anticipation effects, estimating the average treatment effect is infeasible, and we must instead target alternative causal estimands. Second, to identify these alternative causal effects, we can use multi-step-ahead forecasting, beginning our forecast prior to the onset of anticipation effects.

We can begin by describing the onset of the promotional period as a separate treatment from the show's release. Let A_t represent an indicator for whether the promotional period for *13 Reasons Why* began in time t, and let D_t represent an indicator for whether the show was released in time t. We can now describe potential outcomes as a function of both treatments. $Y_t(A_t = 1, D_t = 1)$, for instance, represents the potential outcome in time t when both the promotional period and show's release occur in time t. That is, it represents the outcome we *would* observe had both the promotional period and show occurred in the same period. Finally, let $\bar{A}_t = 0$ ($\bar{D}_t = 0$) indicate that the unit has never experienced treatment A (D) as of time t. $Y_t(A_t = 1, \bar{D}_t = 0)$, for instance, represents the outcome we would observe in time t if both (i) the promotional period had began in time t and (ii) the show had not yet been released as of time t.

We typically define the causal effect of the show's release at time t as $Y_t(D_t = 1) - Y_t(D_t = 0)$. But when we allow potential outcomes to be a function of two treatments, this effect is no longer well-defined. Instead, we can consider the following causal contrasts of interest:

1. $Y_t(A_{t-1} = 1, D_t = 1) - Y_t(A_{t-1} = 1, \bar{D}_t = 0)$: the effect of releasing the show in time t having already started promoting the show in time $t - 1$.

2. $Y_t(A_{t-1} = 1, D_t = 1) - Y_t(\bar{A}_t = 0, \bar{D}_t = 0)$: the effect of promoting the show in time $t - 1$ and the show in time t relative to having never promoted or released the show.

Both causal effects correspond to realistic interventions. In the first setting, we can imagine delaying (or cancelling) the release of the show having already started promoting it. In the second case, we can imagine intervening to prevent the production of the show altogether.[11]

Identifying the first causal effect, however, is infeasible: it would require imputing the potential outcome $Y_t(A_{t-1} = 1, \bar{D}_t = 0)$—that is, the outcome we would observe in time t if (i) the promotional period began in time $t - 1$ but (ii) the show had not yet been released as of time t. The problem is we have no observations for which $A_{t-1} = 1$

[11] We might be tempted to ask about the effect of the show having never promoted it, $Y_t(\bar{A}_t = 0, D_t = 1) - Y_t(\bar{A}_t = 0, \bar{D}_t = 0)$, but this is an unrealistic intervention because studios almost always release trailers before releasing movies and series. It is also worth noting that the effects of promotion and the show itself are not necessarily additive: promoting the show might cause more people to watch the show when it comes out, which would in turn produce stronger contagion effects.

and $\bar{D}_t = 0$: the show was released one month after the trailer began airing. The only observation for which $A_{t-1} = 1$ is the post-treatment period T for which $D_t = 1$.

We can, in contrast, estimate the second causal effect: the combined effect of releasing the show in period t and promoting it in period $t-1$ relative to the counterfactual in which we release neither. The reason we can estimate this effect is we have many periods during which neither the show nor trailer had been released (i.e., all pre–March 2017 periods).

The problem of anticipation effects now becomes clearer. Y_{T-1} no longer represents a potential outcome under control $Y_t(\bar{A}_t = 0, \bar{D}_t = 0)$. Instead, it represents the potential outcome under the release of the trailer but not the show: $Y_t(A_t = 1, \bar{D}_t = 0)$.

Period $T-1$, then, poses two problems for us. First, using Y_{T-1} to select a model might select a worse model for the potential outcomes under control $Y_t(\bar{A}_t = 0, \bar{D}_t = 0)$. Second, even if the observation is not used to select a model, an autoregressive model will use Y_{T-1} to forecast Y_T. Our model is built to forecast potential outcomes under control $Y_t(\bar{A}_t = 0, \bar{D}_t = 0)$ using $Y_{t-1}(\bar{A}_{t-1} = 0, \bar{D}_{t-1} = 0)$, but now we are forecasting the potential outcome under control using $Y_t(\bar{A}_t = 1, \bar{D}_t = 0)$.

A simple solution to the problem, then, is to use multi-step-ahead forecasting to impute $Y_t(\bar{A}_t = 0, \bar{D}_t = 0)$ for the post-treatment period, beginning the forecasting period before anticipation effects occur, and without using periods plagued by anticipation effects to select the model. Refer to Section 7.1 in the main paper for results.

References

Bergmeir, Christoph, Rob J. Hyndman, and José M. Benítez. 2016. "Bagging Exponential Smoothing Methods Using STL Decomposition and Box–Cox Transformation." *International journal of forecasting* 32:303–312.

Berk, Richard A. 2021. "Post-Model-Selection Statistical Inference with Interrupted Time Series Designs: An Evaluation of an Assault Weapons Ban in California." *arXiv preprint arXiv:2105.10624* .

Bradley, Richard C. 2005. "Basic Properties of Strong Mixing Conditions: A Survey and Some Open Questions." *Probability Surveys* 2:107–144.

Bridge, Jeffrey A., Joel B. Greenhouse, Donna Ruch, Jack Stevens, John Ackerman, Arielle H. Sheftall, Lisa M. Horowitz, Kelly J. Kelleher, and John V. Campo. 2020. "Association Between the Release of Netflix's 13 Reasons Why and Suicide Rates in the United States: An Interrupted Time Series Analysis." *Journal of the American Academy of Child & Adolescent Psychiatry* 59:236–243.

Buja, Andreas, Lawrence Brown, Richard Berk, Edward George, Emil Pitkin, Mikhail Traskin, Kai Zhang, and Linda Zhao. 2019. "Models as Approximations I." *Statistical Science* 34:523–544.

Chernozhukov, Victor, Kaspar Wüthrich, and Zhu Yinchu. 2018. "Exact and Robust Conformal Inference Methods for Predictive Machine Learning with Dependent Data." In *Conference On Learning Theory*, pp. 732–749. PMLR.

Chernozhukov, Victor, Kaspar Wüthrich, and Yinchu Zhu. 2021. "An Exact and Robust Conformal Inference Method for Counterfactual and Synthetic Controls." *Journal of the American Statistical Association* 116:1849–1864.

Cleveland, Robert B., William S. Cleveland, Jean E. McRae, and Irma Terpenning. 1990. "STL: A Seasonal-Trend Decomposition." *Journal of Official Statistics* 6:3–73.

Faraway, Julian J. 2016. "Does Data Splitting Improve Prediction?" *Statistics and Computing* 26:49–60.

Fink, David S., Julian Santaella-Tenorio, and Katherine M. Keyes. 2018. "Increase in Suicides the Months After the Death of Robin Williams in the US." *PLoS One* 13:e0191405.

Freedman, David A. 1983. "A Note on Screening Regression Equations." *The American Statistician* 37:152–155.

Gelman, Andrew and John Carlin. 2014. "Beyond Power Calculations: Assessing Type S (Sign) and Type M (Magnitude) Errors." *Perspectives on Psychological Science* 9:641–651.

Healthcare Cost and Utilization Project. 2021. "Suicidal Ideation, Suicide Attempt, or Self-Inflicted Harm: Pediatric Emergency Department Visits, 2010–2014 and 2016."

Hyndman, Rob J. and George Athanasopoulos. 2018. *Forecasting: Principles and Practice*. OTexts.

Lei, Jing, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. 2018. "Distribution-Free Predictive Inference for Regression." *Journal of the American Statistical Association* 113:1094–1111.

Mcdonald, Daniel, Cosma Shalizi, and Mark Schervish. 2011. "Estimating Beta-Mixing Coefficients." In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 516–524. JMLR Workshop and Conference Proceedings.

Niederkrotenthaler, Thomas, Steven Stack, Benedikt Till, Mark Sinyor, Jane Pirkis, David Garcia, Ian R.H. Rockett, and Ulrich S. Tran. 2019. "Association of Increased Youth Suicides in the United States with the Release of "13 Reasons Why"." *JAMA Psychiatry* 76:933–940.

Romer, Daniel. 2020. "Reanalysis of the Bridge et al. Study of Suicide Following Release of 13 Reasons Why." *PLoS One* 15:e0227545.

Shafer, Glenn and Vladimir Vovk. 2008. "A Tutorial on Conformal Prediction." *Journal of Machine Learning Research* 9.