ERASMUS UNIVERSITY ROTTERDAM

Erasmus School of Economics

MSc Econometrics and Management Science

Master Thesis

Modelling The Counterfactual: A Comparison of Bayesian Synthetic Control Methods and Bayesian Structural Time Series Models in Evaluating Causal Impact

Ruoyu Zhi
424107

Supervisor: Prof. Dr. R. Paap
Second assessor: Dr. R. Lange

February 25, 2021

Contents

Abstract

The synthetic control method (SCM) is an important observational methods of evaluating causal impact. In this paper, three extant Bayesian models that belong to the SCM family are studied and compared, namely, BSCM-Spike-and-Slab, BSCM-Horseshoe, and Causal Impact model (CIM). Unlike the standard SCM, the three Bayesian models do not put restrictions on regressors' weights, and can perform variable selection as well as providing valid statistical inferences such as the credible intervals. By a Monte Carlo simulation study we show that BSCM-Horseshoe is more reliable when there is no treatment effect in the data generating process; CIM is computationally fast and more optimistic to make causal claims; BSCM-Spike-and-Slab is slow and might be subject to overfitting issues when dealing with high-dimensional problems. In an empirical study of the economic cost of German reunification in 1990, negative overall cumulative impacts of the reunification on West Germany's growth rate of per capita GDP were found. For an empirical study on the webshop data, by using the three Bayesian models it is found that the treatment under investigation can increase the average online page views of the treated products on the webshop.

1 Introduction

As one of the important topics in econometrics, measuring causal effects has obtained researchers' attention for the past decades. The gold standard of assessing causality — randomized controlled trials (RCTs) — has been applied widely in many fields. By randomly assigning each subject to either of the control and the treatment groups, RCTs have proven successes in establishing causality. The popularity can be seen from the more than 1,000 publications about international development from 1981 to 2012 that are related to RCTs (Cameron et al., 2016). Sometimes, however, RCTs are not applicable due to financial or ethical issues, or simply because they are not possible to design. Such issues are addressed by many practitioners. Among others, Gordon et al. (2019) explain why there are few online ad campaigns using RCTs; Amjad et al. (2018) stress the failures of designing RCTs when it comes to policy interventions. And the above papers suggest that in those cases, instead of using RCTs, one can apply observational or quasi-experimental methods to evaluate causal effects.

A core concept of those observational or quasi-experimental methods is the "counterfactual". Suppose we observe the (time-series) outcome after an intervention (e.g. GDP of a state after a policy change), then the counterfactual for this outcome is what would have happened had no intervention occurred, and the effect of treatment on the treated is defined as the difference between the observed outcome and the counterfactual. Since the counterfactual is not observable, the vital step is then how to estimate it. In a study of the effect of the terrorist conflict in the late 1960's on per capita GDP in the Basque

Country in Spain, Abadie and Gardeazabal (2003) proposed the synthetic control method (SCM), which constructs a "hypothetical" Basque Country using a convex combination of the per capita GDPs of other Spanish regions that did not have the terrorist conflict. The weights of the convex combination are calculated such that the weighted average of the pre-conflict per capita GDPs of other regions best matches the pre-conflict per capita GDP of the Basque Country. The idea is that if the terrorist conflict did not happen in the Basque Country at all, the pre-conflict relations between the Basque Country's per capita GDP and the other Spanish regions' per capita GDPs should remain the same in the post-conflict periods. Thus, supposing the terrorist conflict in the Basque Country did not influence the per capita GDPs of other Spanish regions, it makes sense to assume that the "hypothetical" Basque Country can be regarded as an estimate of what would have happened had no terrorist conflict, i.e., the counterfactual. Therefore, the impact of the conflict is the difference between the per capita GDPs of the "hypothetical" and the true Basque Country for the post-conflict periods.

The SCM (Abadie and Gardeazabal, 2003; Abadie et al., 2010), which is referred to as the standard SCM in the remaining text, is related to the matching approaches (Samartsidis et al., 2019). Moreover, the standard SCM can be seen as a generalization of the famous difference-in-differences method (see Wing et al., 2018, for an introduction for difference-in-differences methods), and is said to be 'arguably the most important innovation in the policy evaluation literature in the last 15 years' (Athey and Imbens, 2017). One pleasing aspect of the standard SCM is that it makes use of only outcomes of the untreated units. Some "classic" methods such as the difference-in-difference or latent difference-in-difference methods (Samartsidis et al., 2019) rely on also other covariates and require large samples for estimating, and are often "ill-suited" when measuring the effect of infrequent treatments on aggregate units (Abadie, 2019). Although it is also possible to include other covariates in the standard SCM, it is preferred to incorporate only the outcomes of control units. This is because of the concerns about overfitting (Powell, 2018), and that the outcomes of control units can provide high predictive power (Doudchenko and Imbens, 2016) and can render all other covariates irrelevant (Kaul et al., 2015).

Although the standard SCM has the advantage of simplicity and 'obvious improvement over the standard methods' (Athey and Imbens, 2017), it is not without drawbacks. Kim et al. (2020) argue that the standard SCM is limited in at least three aspects. Firstly, the standard SCM puts restrictions on the weights that the weights should be non-negative and sum to one, and the intercept should be zero. These restrictions imply that the constructed counterfactual lies in the convex hull of the outcomes of the control units. We can easily see that this restriction is not realistic when the outcome of the treated unit is larger or smaller than all the outcomes of the control unit, i.e lying outside of the convex hull. One might argue that this issue can be resolved by standardization.

However, the convex restriction can still raise concerns that the estimates are biased (Ferman and Pinto, 2016; Carvalhoab et al., 2018). Secondly, Li (2019) points out that the standard SCM lacks formal inference theory. Many studies using the the standard SCM, including Abadie et al. (2010) and (2015), apply the so-called "placebo test" for statistical inference, and its validity is questioned by Hahn and Shi (2017). Thirdly, Kim et al. (2020) conclude that the standard SCM is subject to insufficient ability to deal with the "large p, small n" and the sparsity problems. In a time-series setting, the "large p, small n" problem means that the number of control (outcome) time-series is larger than the number of time points. And the sparsity problem happens when not all outcomes of control units are relevant to the outcome of the treated unit, which would result in zero weights for some outcome time-series.

To resolve the first limitation of the standard SCM, namely the restriction on weights, one can simply relax the convex restrictions (see Amjad et al., 2018), and add the intercept term. And for the third limitation of solving the "large p, small n" and the sparsity problems, researchers have made use of several frequentist regularization methods, including lasso (Carvalhoab et al., 2018), ridge (Amjad et al., 2018) and elastic net (Doudchenko and Imbens, 2016). While the frequentist regularized regression methods can promote sparsity and deal with the high-dimensional problem, they are not able to solve the second limitation of SCM, i.e. they cannot give valid statistical inference such as confidence intervals (Mallick and Yi, 2013). To move one step ahead towards statistical inferences, although in principle we can turn to bootstrap, Kyung et al. (2010) show that when the true parameters are sparse or nearly sparse (the third limitation of SCM), bootstrap sampling would give biased estimates. Finally, Kim et al. (2020) conclude that we have not seen any single frequentist method that can solve all three limitations at the same time.

One path to overcoming all the three limitations is adopting Bayesian regularized regression methods. Pioneers have already proposed the Bayesian equivalences of ridge (Hsiang, 1975), lasso (Park and Casella, 2008), and elastic net (Li and Lin, 2010). By relaxing the restrictions on weights and using the Bayesian counterparts of regularized regression methods, we can resolve the first and third limitations of SCM. Most importantly, Bayesian regularized regression methods provide exact statistical inference via employing Markov Chain Monte Carlo (MCMC) procedure, even for small sample size, as we have samples from the posterior distribution (Kim et al., 2020). As a result, the second limitation of the standard SCM is conquered.

Kim et al. (2020) proposed two new Bayesian Synthetic Control Methods (BSCMs), i.e. BSCM with the Spike-and-Slab prior (BSCM-Spike-and-Slab), and BSCM with the Horseshoe prior (BSCM-Horseshoe). The Spike-and-Slab prior (George and McCulloch, 1993; George and McCulloch, 1997) and the Horseshoe prior (Carvalho et al., 2010) are purely Bayesian variable selection approaches. Kim et al. (2020) state that just like the

Bayesian equivalences of the regularized regression methods, their newly proposed models are not subject to the three limitations of the standard SCM. In addition, motivated by a simulation study, they show that BSCM-Spike-and-Slab and BSCM-Horseshoe have more accurate predictions than the Bayesian counterparts of lasso, ridge, and elastic net.

Suggested by Kim et al. (2020), BSCMs rely on sufficient availability of control units, and in case of lacking information of control units, one can use the Causal Impact model. The Causal Impact model (CIM), proposed by Brodersen et al. (2015), is another approach to predict the counterfactual by constructing a synthetic control unit. It applies the Bayesian structural time-series model, which is the Bayesian version of the state-space model for time-series data. Besides the fact that just like SCM and BSCMs, CIM also makes use of the outcomes of control units, this approach is more flexible since we can add different time-series components, such as local linear trends and seasonality, which makes it preferable when there is insufficient information about control units. And it is able to capture the evolution of states due to the state-space model it utilizes. By using the Spike-and-Slab prior and Bayesian model averaging (Hoeting et al., 1999), CIM tries to avoid overfitting and model uncertainty. Additionally, implemented in a Bayesian framework using MCMC approach, it can provide valid statistical inferences. Although having been applied in many studies of treatment evaluation, CIM has not been compared with the two newly proposed BSCMs by Kim et al. (2020).

In this paper, by a Monte Carlo simulation study I evaluate the abilities of detecting causal effects for the three Bayesian methods, i.e., BSCM-Spike-and-Slab, BSCM-Horseshoe, and CIM, for the cases where a single treated unit is present at a time. The data generating processes (DGPs) of the simulation study reflect different situations, including different effect sizes, sparsity levels, etc. For each DGP, 50 data sets are generated. And 4 statistics are compared between the three Bayesian methods for each DGP, e.g. the average mean squared errors (MSEs), the percentages of correctly detecting pointwise and overall cumulative causal impact, where the later one would be the main criteria to compare the models, as well as the time needed for estimation in seconds.

Furthermore, BSCM-Spike-and-Slab, BSCM-Horseshoe and CIM are applied in two empirical studies. The first empirical study aims to examine the impact of the German reunification in 1990 on the growth rate of West Germany's per capita GDP across 1990 to 2003. The standard SCM has already been used to study the impact of the reunification on West German per capita GDP by Abadie et al. (2015), which did not only utilize the outcome variables of the 16 selected untreated OECD countries but also some other control variables of them, such as the inflation rates. Abadie et al. (2015) show that the 1990 German reunification has negative economic impact on West Germany throughout 1990—2003. Unlike Abadie et al. (2015), in this paper I take into account the non-stationarity of per capita GDPs, which indicates that the results of regressing the pre capita GDP of West Germany on those of other countries is unreliable. Thus, per capita

GDPs are replaced by the growth rates of them. Additionally, I avoid to include in the models other control variables than the growth rates of per capita GDPs of the 16 OECD countries, in order to avoid overfitting (Powell, 2018). The three Bayesian models all detect negative impacts cumulatively from 1990 to 2003, ranging from $-22\%\%$ to -15%. The statistical inferences obtained by the three models, namely the 95% credible intervals, tell us that the overall cumulative impacts obtained by CIM and BSCM-Horseshoe are different from 0, while BSCM-Spike-and-Slab does not converge. The second empirical study uses data from a Dutch webshop and the goal is to understand the impact of a technique called "eagle", which predicts and fills in missing attributes of the products, such as titles and descriptions, on their average online page views. We find that the treatment imposes increases of more than 120% of the average online page views over the treated products, and the cumulative impacts obtained by them are all considered to be different from 0 based on the 95% credible intervals.

The paper proceeds as follows. Section 2 provides a literature review of works that are related to the SCM and CIM. In Section 3, the model setup and methodological details are discussed. Section 3 explains the Monte Carlo simulation study and its results, followed by two empirical studies in Section 5 and Section 6. Section 7 concludes the paper.

2 Literature review

Proposed by Abadie and Gardeazabal (2003) and Abadie et al. (2010), SCM has been used in a large number of studies. It has been employed in the fields including taxation policy (Bollinger and Sexton, 2018; Rojas and Wang, 2017), health policy (Kreif et al., 2016), criminology (Saunders et al., 2015), marketing (Tirunillai and Tellis, 2017) and so on. Besides its applications, SCM has also been developed to difference directions. Hsiao et al. (2012) and Doudchenko and Imbens (2016) propose panel data models for SCM without restricting the weights to be convex. Ben-Michael et al. (2018) introduce the augmented SCM, which corrects the estimate bias caused by imperfect pre-treatment fit. While the standard SCM could only deal with one treated unit, the generalized SCM, developed by Xu (2017), uses an interactive fixed effects model and is able to predict the counterfactuals for more than one treated units at the same time. Powell (2018) compares results in health policy evaluation of models including the generalized SCM and the standard SCM. A framework of SCM aiming to handle data at granular level is created by Robbins et al. (2017). Arkhangelsky et al. (2019) present the Synthetic Difference-in-differences (SDID) methods, which equips the standard SCM with two-way fixed effects and both time and unit weights, and they formally prove that SDID is doubly robust. The robust SCM, proposed by Amjad et al. (2018), is SCM with a de-noising procedure and with the regularization methods such as lasso and ridge, implemented in both the frequentist and Bayesian ways. Amjad et al. (2019) generalize this idea such that it can deal with limited pre-intervention data. Finally, for an implementation of the standard SCM (Abadie and Gardeazabal, 2003; Abadie et al., 2010), the R package **Synth** is developed by Abadie et al. (2011) and is available from the CRAN (Comprehensive R Archive Network).

Another method employed in this paper, is the Causal Impact model (CIM), which essentially uses a predicting model from Scott and Varian (2014). It is referred to as the Causal Impact model because its R implementation (open-source package available from CRAN) is named **CausalImpact**. CIM is also only able to deal with a single treated unit at a time, but its multivariate version is derived by Menchetti and Bojinov (2020). Originally proposed for marketing evaluations, CIM has been popular and also used in different studies. By applying CIM, the treatment effect of alcohol licensing policies on hospital admission and crime is measured by de Vocht et al. (2017); the effect of pressure from owners on journalistic output is examined for two popular Russian online newspapers by Fredheim (2017); de Vocht (2016) evaluates the effect of mobile phone use on selected brain cancer subtypes; Bruhn et al. (2017) explore the effectiveness of vaccines in a population-level; in finance, Märkle-Huß et al. (2018) study how 15 min trading influences the EPEX SPOT market. Other than being used for treatment evaluation, CIM by its nature can also be utilized for prediction or exploratory means. As examples, Madhavan et al. (2020) predict passenger and cargo demand of the Indian airline industry, and Poyser (2019) explores the determinants of Bitcoin's price using CIM.

3 Methodology

This section starts with the basic setup. Then the standard SCM (Abadie and Gardeazabal, 2003; Abadie et al., 2010) is introduced, which is an optimization problem with constraints, followed by how we relax its convex restrictions. The formulations of BSCMs and CIM are discussed afterwards.

3.1 Basic setup

Suppose we observe $N+1$ units, which means that unit $i = 0, 1, ..., N$. And our observation spans T periods, that is, $t = 1, ..., T$. In our setup only unit $i = 0$ receives the treatment, and units $i = 1, ..., N$ are control units that are untreated. Unit $i = 0$ receives the treatment at period $t = T_0 + 1$, $1 \leq T_0 < T$. Therefore, we call $\{1, ..., T_0\}$ the pre-treatment period, and $\{T_0 + 1, ..., T\}$ the post-treatment period. As is explained before, the treatment effect is defined as the difference between the observed outcome of the treated unit and the unobserved counterfactual outcome of the same unit. Thus, the treatment effect at time t for unit i is written as $e_{it} = Y_{it} - Y_{it}^N$ for $t = T_0 + 1, ..., T$, where Y_{it} is the observed outcome of unit i at t, and Y_{it}^N is the potential outcome if no treatment was received by the same unit at the same time. And therefore,

$$Y_{it} = Y_{it}^N + e_{it} D_{it}, \quad \text{for} \quad t = 1, \dots, T, \tag{1}$$

where D_{it} equals 1 if unit i has been treated at time $t > T_0$ and 0 otherwise. Since only unit $i = 0$ is treated when $t > T_0$, we are particularly interested in

$$e_{0t} = \underbrace{Y_{0t}}_{\text{observed}} - \underbrace{Y_{0t}^N}_{\text{counterfactual}}, \quad \text{for} \quad t = T_0 + 1, \dots, T, \tag{2}$$

where we use that $D_{0t} = 1$ when $t > T_0$. Because Y_{it} are observed, the problem is then how to predict the unobservable counterfactual Y_{0t}^N. The underlying model is

$$Y_{0t}^N = \beta_0 + \sum_{i=1}^{N} \beta_i Y_{it} + \varepsilon_t, \tag{3}$$

for $t = 1, \dots, T$, where ε_t are error terms following i.i.d. Normal$(0, \sigma_\varepsilon^2)$, and where $Y_{0t}^N = Y_{0t}$ for $t = 1, \dots, T_0$.

We use the relations between the outcome of the treated unit and the outcomes of the untreated units, prior to the treatment, to predict Y_{0t}^N for $t = T_0 + 1, \dots, T$. Supposing the estimates for β_i are $\hat{\beta}_i$, the counterfactual estimated is $\hat{Y_{0t}^N} = \hat{\beta}_0 + \sum_{i=1}^N \hat{\beta}_i Y_{it}$, for $t = T_0 + 1, \dots, T$, and the estimated treatment effect for unit $i = 0$ is

$$\hat{e}_{0t} = \underbrace{Y_{0t}}_{\text{observed}} - \underbrace{\hat{Y}_{0t}^N}_{\text{predicted counterfactual}}, \quad \text{for} \quad t = T_0 + 1, \dots, T. \tag{4}$$

Note that, following Imbens and Rubin, we assume that the *stable unit treatment value assumption (SUTVA)* should hold. This means that potential outcomes of a unit are only functions of the treatment assignment of this unit itself, or putting differently, a unit's exposure to the treatment should not have any effect on the potential outcomes of other units.

Moreover, following Brodersen et al. (2015), as the measure of the relative impact throughout the post-treatment period, in this paper we look at the so-called effect size, which is defined as

$$e = \frac{\sum_{t=T_0+1}^T e_{0t}}{\sum_{t=T_0+1}^T Y_{0t}^N} \times 100\%, \tag{5}$$

where the numerator is called the overall cumulative impact (see Section 4.2). And the estimated effect size is given by $\hat{e} = \frac{\sum_{t=T_0+1}^T \hat{e}_{0t}}{\sum_{t=T_0+1}^T \hat{Y}_{0t}^N} \times 100\%$.

The standard Synthetic Control Method (SCM) proposed by Abadie and Gardeazabal (2003) and Abadie et al. (2010) can be written as the following constrained optimization problem (Kim et al., 2020):

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}\in\Lambda} \sum_{t=1}^{T_0} \left(Y_{0t} - \beta_0 - \sum_{i=1}^{N} \beta_i Y_{it} \right)^2, \tag{6}$$

where β_0 stands for the intercept, which is subject to the convex restrictions

$$\Lambda = \left\{ \boldsymbol{\beta} \in \mathbb{R}^{N+1} : \beta_0 = 0, \beta_i \geq 0 \text{ for } i = 1, \dots N \text{ and } \sum_{i=1}^{N} \beta_i = 1 \right\}. \tag{7}$$

Remember that the convex restrictions (7) correspond to the first limitation of the standard SCM. Thus, relaxing the restrictions on weights implies that β_i can be estimated by OLS.

3.2 Bayesian methods

This subsection discusses the three Bayesian methods, i.e. BSCM-Spike-and-Slab, BSCM-Horseshoe, and CIM. Before explaining the utilized methods, I first introduce the Bayesian counterparts of the standard SCM. Then BSCM-Spike-and-Slab and BSCM-Horseshoe methods are discussed, followed by CIM, which also uses the Spike-and-Slab prior, but with few deviations from the prior specification of BSCM-Spike-and-Slab.

3.2.1 Bayesian standard SCM

Instead of going to BSCM-Spike-and-Slab directly, it would be helpful for understanding to start with the Bayesian equivalence of the standard SCM, which can be expressed by using uninformative priors (with restrictions):

$$\beta_i \sim \text{uniform}(-\infty, \infty) \text{ for } i = 1, \dots, N,$$

$$\sigma_\varepsilon \sim \text{uniform}(0, \infty), \tag{8}$$

$$\text{s.t, } \beta_0 = 0, \beta_i \geq 0, \text{ and } \textstyle\sum_{i=1}^{N} \beta_i = 1.$$

The idea behind prior (8) is straightforward: we give completely uninformative prior distributions to the parameters and let the data decide their estimation. Again, the way to deal with the convex restriction on weights is simply removing them. And then we want to relax the no-intercept constraint, i.e. $\beta_0 = 0$. Of course, we can simply replace the no-intercept restriction by an uninformative prior, $\beta_0 \sim \text{uniform}(-\infty, \infty)$, which would result in the Bayesian counterpart of the panel data model by Hsiao et al. (2012).

However, a more informative prior for β_0 is adopted in this paper, that is

$$\beta_0 \sim \text{Cauchy}(0, 10). \tag{9}$$

Prior (9) is said to be weakly informative and 'conservative' (Gelman et al., 2008) and is used in the main text of Kim et al. (2020). Kim et al. (2020) conduct a robustness check for more and less informative priors for β_0, and shows that a more informative prior would result in models with better prediction performances, whereas the differences of performances are small. Thus, following the main text of Kim et al. (2020) for both BSCM-Spike-and-Slab and BSCM-Horseshoe, prior (9) for β_0 is employed.

3.2.2 BSCM-Spike-and-Slab and BSCM-Horseshoe

The well-known Spike-and-Slab prior (George and McCulloch, 1993) is a purely Bayesian variable selection method. It can be seen as a mixture of a "spike" and a "slab" normal distributions, where the probability mass of the spike portion is concentrated at 0 in order to decide whether the coefficient is close to 0, and the slab portion is weakly informative in order to determine the values of non-zero coefficients.

Recall that we have N remaining coefficients that are corresponding to the N control variables to estimate, namely $\beta_1, ..., \beta_N$. Define the inclusion parameters γ_i, where $\gamma_i = 1$ if $\beta_i \neq 0$ and $\gamma_i = 0$ if $\beta_i = 0$ for $i = 1, ..., N$. Let $\boldsymbol{\beta} = (\beta_1, ..., \beta_N)'$ and $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_N)'$. The variances of the slab portion is denoted as $\boldsymbol{\tau}^2 = (\tau_1^2, \dots, \tau_N^2)'$. In addition, define θ_i as the probability that β_i is not 0, and $\boldsymbol{\theta} = (\theta_1, ..., \theta_N)'$. Then the Spike-and-Slab prior can be written as

$$\beta_i \mid \gamma_i, \tau_i^2 \sim \underbrace{(\gamma_i) \cdot \text{ Normal } \left(b_i, \tau_i^2\right)}_{\text{slab portion}} + \underbrace{(1-\gamma_j) \cdot \text{ Normal } \left(0, \phi^2\right)}_{\text{spike portion}}, \quad \text{for } i = 1, \dots, N, \tag{10}$$

$$\theta_i \sim \text{uniform}(0, 1), \quad \text{for} \quad i = 1, \dots, N, \tag{11}$$

$$\boldsymbol{\gamma} \mid \boldsymbol{\theta} \sim \prod_{i=1}^{N} \theta_i^{\gamma_i}(1-\theta_i)^{1-\gamma_i}, \tag{12}$$

$$\frac{1}{\tau_i^2} \sim \text{ Gamma } \left(\frac{\nu}{2}, \frac{s}{2}\right), \tag{13}$$

$$\sigma_\varepsilon \sim \text{Cauchy}^+(0, 10), \tag{14}$$

where for (10), the probability mass of the slab portion spreads over a large range of possible values of β_i, while the spike portion is concentrated around 0. Thus, β_i tends to be away from 0 if the inclusion indicator $\gamma_i = 1$, and is likely to be very small when $\gamma_i = 0$. Also in (10), b_i stands for our prior expectation for β_i and is usually set to 0, and ϕ is a fixed small number to ensure that the spike portion is close to 0. Following Kim et al. (2020), I set $\phi^2 = 0.001$. Figure (1) gives an example of how the spike and the slab portions look like. Prior (11) is the prior distribution of the inclusion probability $\boldsymbol{\theta}$. Prior (12) is the prior distribution for the inclusion parameters $\boldsymbol{\gamma}$, which is an independent Bernoulli distribution given $\boldsymbol{\theta}$. Prior (13) is the gamma distribution with mean $\frac{\nu}{s}$, and for BSCM-Spike-and-Slab I follow Kim et al. (2020) that $\nu = s = 1$. Additionally, for (14), $\text{Cauchy}^+(0, a)$ denotes the standard half-Cauchy distribution with parameter a on positive real numbers, which is again 'conservative' and weakly informative (Gelman et al., 2008). And its probability density function for $x \in \mathbb{R}$ is

$$\text{Cauchy}^+(x \mid \mu, \sigma) = \frac{1}{\pi\sigma} \frac{1}{1 + ((x-\mu)/\sigma)^2}, \tag{15}$$

for any $\mu \in \mathbb{R}$ and $\sigma \in \mathbb{R}^+$ (strictly positive real numbers).

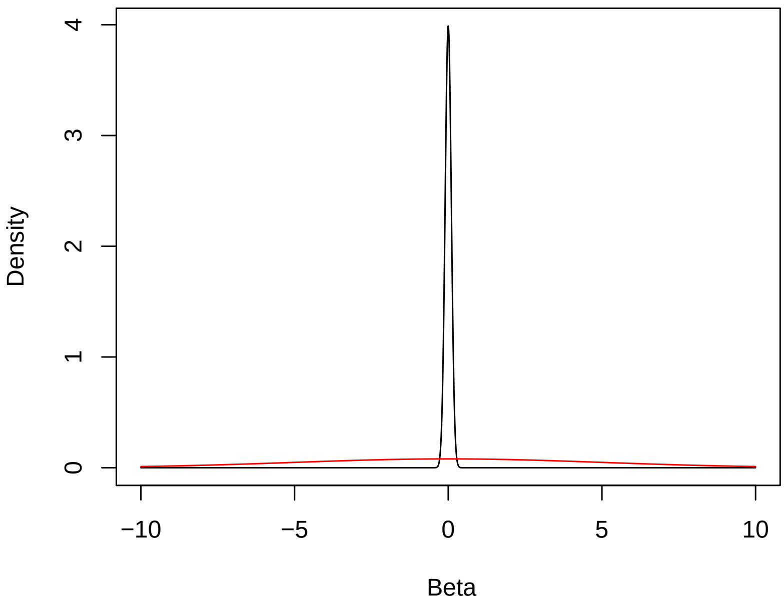


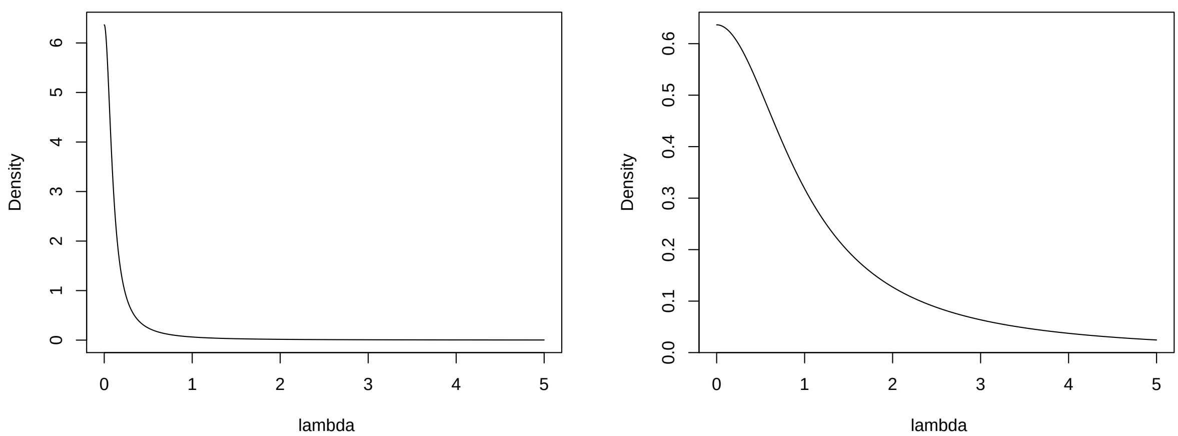
Figure 1: An example of the Spike-and-Slab prior for beta, where the red line is the slab portion which follows Normal(0, 25), and the black line denotes the spike portion which follows Normal(0, 0.001).

Although the Spike-and-Slab prior is clear in how it performs variable selection, Carvalho et al. (2010) argue that it can be computationally demanding once the number of variables becomes larger, as the number of possible models becomes rapidly increasing. In these cases, the Horseshoe prior can be adopted, which requires less computational time while it shows similar predictive power. The Horseshoe prior (Carvalho et al., 2010;

Kim et al., 2020) is given by

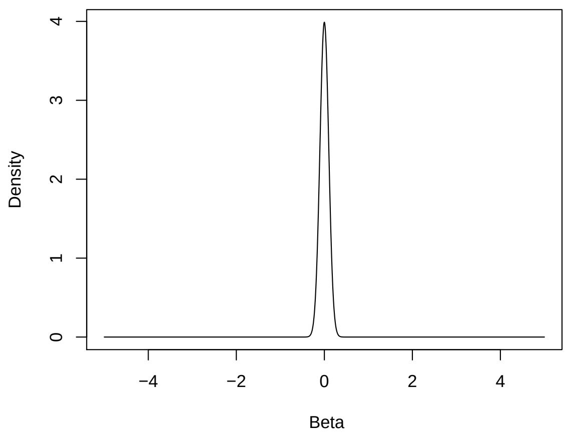
$$\begin{aligned}
\beta_i \mid \lambda_i &\sim \text{ Normal } \left(0, \lambda_i^2\right) \text{ for } i = 1, \ldots, N, \\
\lambda_i \mid \tau &\sim \text{Cauchy}^+(0, \tau), \\
\tau \mid \sigma_\varepsilon &\sim \text{Cauchy}^+(0, \sigma_\varepsilon), \\
\sigma_\varepsilon &\sim \text{Cauchy}^+(0, 10),
\end{aligned} \tag{16}$$

where τ is a common shrinkage parameter for all λ_i and regularizes all coefficients to 0 while some of the local shrinkage parameters λ_i's offer heavy tails, meaning that some β_i's are possible to be away from 0. The smaller the common shrinkage parameter τ is, the smaller each λ_i is likely to be, and then it is more possible that β_i is close to 0 (see Figure 2 and Figure 3). We can see that the Horseshoe prior does not require users for any additional inputs, as all the hyperparameters are fully specified. I also notice that the prior for σ_ε used by Kim et al. (2020), which is half-Cauchy, is different than the one used by Carvalho et al. (2010), which is a diffuse prior. In this paper, we use the same prior as Kim et al. (2020), not only for being consistent, but also because Kim et al. (2020) conducted a robustness check for differently informative priors and the results of prediction accuracy are shown to be robust.

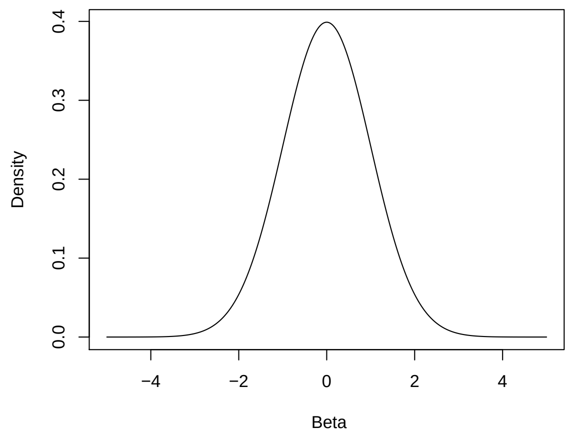


(a) Prior distribution of λ_i, given $\tau = 0.1$ (b) Prior distribution of λ_i, given $\tau = 1$

Figure 2: Prior distributions of λ_i given different values of τ



(a) Prior distribution of β_i, given $\lambda_i = 0.1$



(b) Prior distribution of β_i, given $\lambda_i = 1$

Figure 3: Prior distributions of β_i given different values of λ_i

3.2.3 CIM

The Causal Impact model (CIM), proposed by Brodersen et al. (2015), is sometimes called the Bayesian structural time-series (BSTS) model. This is because that CIM utilizes the BSTS model proposed by Scott and Varian (2014) for constructing synthetic controls and forecasting post-treatment counterfactuals. Brodersen et al. (2015) argue that generally speaking, the sources of information that can be applied for constructing a proper synthetic control lie in three categories. The first category is the pre-treatment time-series characteristics of the outcome of the treated unit itself, such as its local linear trend. The second category is information that comes from the outcomes of untreated units that have predicting power prior to the treatment. Additionally, when we apply Bayesian methods, the third category is the prior knowledge about models, which comes from previous studies or "expert" opinions. Obviously, the BSCM-Spike-and-Slab and BSCM-Horseshoe models are able to utilize the second and third sources of information, while the pre-treatment time-series characteristics of the treated unit's outcome are not incorporated in these two BSCM models. On the contrary, CIM is able to make use of all three kinds of sources of information, and this is done by the Bayesian structural time-series model.

The structural time-series models are essentially state-space models for time-series data. In the paradigm of the state-space model, we can add local linear trend or seasonality to capture the outcome's time-series characteristics. And the relation between the treated and untreated units is utilized by adding a regression component into the state-space model. In addition, to select the control outcomes that have predictive power for the target outcome prior to the treatment, CIM also uses a Spike-and-Slab prior.

Specifically, the structural time-series model employed in this paper can be written as the pair of equations

$$\begin{aligned} Y_{0t} &= \mu_t + \sum_{i=1}^{N} \beta_i Y_{it} + \varepsilon_t, \\ \mu_t &= \mu_{t-1} + u_t, \end{aligned} \tag{17}$$

where $\varepsilon_t \sim \text{Normal}\left(0, \sigma_\varepsilon^2\right)$ and $u_t \sim \text{Normal}\left(0, \sigma_\mu^2\right)$. The pair of equations (17) contains two time-series components — a local level component μ_t and a regression component $\sum_{i=1}^{N} \beta_i Y_{it}$, where the evolution of μ_t is captured by the normally distributed term u_t and the regression parameters β_i are assumed to be constant over time. This setting can be seen as relatively static, because we want to avoid uncertainty raised by more dynamic time-series components, especially when predicting more than several time periods ahead. Brodersen et al. (2015) suggest that using time-series behaviors such as the local level and a static regression component will give a proper balance between capturing dynamics and stable relationships. Therefore, in this paper, CIM with the local level component and the static regression component are utilized.

Although in this paper we tend to use relatively static time-series components, some dynamic components can still be added to the model when they are desired. The first possible time-series component of state is the local linear trend, and its observation and transition equations are

$$\begin{aligned} \mu_{t+1} &= \mu_t + \delta_t + \eta_{\mu,t}, & \eta_{\mu,t} &\sim \text{Normal}\left(0, \sigma_\mu^2\right), \\ \delta_{t+1} &= \delta_t + \eta_{\delta,t}, & \eta_{\delta,t} &\sim \text{Normal}\left(0, \sigma_\delta^2\right), \end{aligned} \tag{18}$$

where μ_t and δ_t are the trend and slope at t, respectively. Brodersen et al. (2015) argue that the local linear trend components is preferable when doing short-term forecasting, while for long-term forecasting, adding this component might bring too much uncertainty.

The second time-series component is seasonality, defined by

$$\sum_{s=0}^{S-1} \gamma_{t-s} = \eta_{\gamma,t}, \tag{19}$$

where the number of seasons is denoted by S. By generalizing (19), one can add different components for seasonality with variant periods.

The static regression component $\sum_{i=1}^{N} \beta_i Y_{it}$ in (17) is preferred if the relationship between the response and covariates are stable prior to the intervention, whereas the dynamic regression takes into account the coefficients that are varying over time, and can be included in the state-space model if we set the regression component to $\mathbf{x}_t' \boldsymbol{\beta}_t = \sum_{i=1}^{N} x_{i,t}\beta_{i,t}$. Its corresponding transition equation is $\beta_{i,t+1} = \beta_{i,t} + \eta_{\beta,i,t}$, where $\eta_{\beta,i,t} \sim \text{Normal}\left(0, \sigma_{\beta_i}^2\right)$.

As is mentioned before, CIM also uses the Spike-and-Slab prior, but with different parameter specifications. The first deviation is that the probability of outcome i being included in the model, θ_i, is specifically specified by the fixed *expected model size* M. M denotes our prior expectation of how many regressors should be included in the model. And $\theta_i = \frac{M}{N}$. Another deviation is for the slab portion. The difference in (13) is because CIM sets $\nu = \nu_\varepsilon$ and $s = s_\varepsilon$, where ν_ε and s_ε are determined by the expected goodness of fit R^2 and the sample variance of the response s_y^2. R^2 is a fixed number between 0 and 1 which is specified by users, and $s_y^2 = \sum_{t=1}^{T_0}(Y_{0t} - \bar{Y}_0)/(T_0 - 1)$. Brodersen et al. (2015) fix R^2 to 0.8. And $s_\varepsilon = \nu_\varepsilon \left(1 - R^2\right) s_y^2$. And finally, define $\boldsymbol{\beta}_\gamma$ to be the set of coefficients β_i that are included in the model. The Spike-and-Slab prior for CIM is

$$\begin{aligned} \boldsymbol{\beta}_\gamma \mid \sigma_\varepsilon^2 &\sim \text{Normal}\left(\boldsymbol{b}_\gamma, \sigma_\varepsilon^2 \left(\boldsymbol{\Sigma}_\gamma^{-1}\right)^{-1}\right), \\ \frac{1}{\sigma_\varepsilon^2} &\sim \text{Gamma}\left(\frac{\nu_\varepsilon}{2}, \frac{s_\varepsilon}{2}\right), \\ \boldsymbol{\gamma} \mid \boldsymbol{\theta} &\sim \prod_{i=1}^{N} \theta_i^{\gamma_i}(1-\theta_i)^{1-\gamma_i}, \end{aligned} \tag{20}$$

where we choose $\boldsymbol{b}_\gamma$, the prior expectation for β_i that are included in the model, to be a vector of zeros. Σ_γ^{-1} is the matrix of rows and columns of Σ^{-1} that corresponds to units i where $\gamma_i = 1$, where $\Sigma^{-1} = \frac{1}{N}\left\{\frac{1}{2}Y'Y + \frac{1}{2}\operatorname{diag}\left(Y'Y\right)\right\}$, where $Y = [Y_1, \ldots, Y_N]$. See Brodersen et al. (2015)) and Chipman et al. (2001) for more details about this choice of Σ^{-1}. This type of the Spike-and-Slab prior directly set β_i to be 0, instead of using a normal mixture.

Additional tools are required to deal with the Bayesian structural time-series model, such as the Kalman filter (Kalman, 1960). A *fast mean Kalman smoother* (Durbin and Koopman, 2005) is applied in order to update previously computed means and variances. Another important tool is the data-augmentation step, which is used when we need to simulate the state from its posterior distribution. If we define $\boldsymbol{\Theta}$ as all the model parameters and $\boldsymbol{\alpha}$ as the set of all states, the applied Gibbs sampler alternates between the data-augmentation step, which simulates from $p(\boldsymbol{\alpha} \mid \boldsymbol{Y}_{0,1:t}, \boldsymbol{\Theta})$, and another step which simulates from $p(\boldsymbol{\Theta} \mid \boldsymbol{Y}_{0,1:t}, \boldsymbol{\alpha})$, where $\boldsymbol{Y}_{0,1:t}$ denotes the set of observed $Y_{0,t}$ until time T.

Note that the posterior sampling of CIM is done by the well-known Gibbs sampler, with explicit sampling steps, conditional and marginal posterior distributions provided by Brodersen et al. (2015) and Scott and Varian (2014). Derivation of the conditional posterior distributions for the Spike-and-Slab prior is given in Appendix B. Note that the derivation uses the Spike-and-Slab prior presented by (10) — (13). And one can follow the similar intuition to calculate the conditional posterior distributions for the priors employed by CIM. CIM as well as its statistical inferences are comprehensively implemented in the R package **CausalImpact**, which uses the Gibbs sampler. For BSCM-Spike-and-Slab and BSCM-Horseshoe, it is easy to make use of Stan, a probabilistic programming language for statistical inference written in C++. Stan's R interface, RStan, is used in this paper for their posterior inferences. The sampler utilized for these two BSCMs is a different one, called the No-U-Turn sampler (NUTS). It is beyond the scope of this paper to discuss about the difference between the Gibbs sampler and the NUTS, and those who are interested in the NUTS can consult Hoffman and Gelman (2014).

4 Monte Carlo simulation

This section describes the Monte Carlo simulation study used to compare the performances of our models. Several scenarios of the Data Generating Process (DGP) are considered here, including different levels of sparsity, lengths of total observation periods T, the constant effect sizes e, and the ratios between T and the number of control unit N. And this results in 56 distinct DGPs, and for each of them 50 datasets are generated. BSCM-Spike-and-Slab, BSCM-Horseshoe, and CIM are applied to each simulated datasets. Section 4.1 illustrates in detail how the DGP is designed, followed by the description of how the models decide whether the (cumulative) causal effects for a given dataset are different from 0 in Section 4.2. And the simulation results are shown and described in Section 4.3.

4.1 Simulation procedure

In the simulation study, the number of total observed periods T can either be large or small, where a large $T = 90$ and a small $T = 30$. For both large and small T, the treatment happens at $t = T_0 + 1$ where $T_0 = 2T/3$. Therefore, the pre-treatment periods are $t = 1, \ldots, 2T/3$, and the post-treatment periods are $t = 2T/3 + 1, \ldots, T$. Then we consider two different ratios between T and the number of control unit N, i.e., $T/N = 2$ and $T/N = 0.5$, where the second case can be seen as a high-dimensional problem. The level of sparsity, which is defined as the fraction of zero parameters, has also two scenarios. The high level of sparsity is equal to 0.8, which means 80% of the true regression parameters are 0, and the low level of sparsity is 0, meaning that all control variable are related to the response. Additionally, variant cases for the effect size, e, are also considered. Effect size e can be 0%, 1%, 5%, 10%, 50%, 100% and 200%. Given e as the true effect size of the treatment, the post-treatment target outcome is multiplied by $1 + e$. Moreover, since all the utilized models relax the convex constraints on the parameters, it is not necessary to distinguish between the cases where the convex constrains are met or not. And in this simulation study, the regression parameters still sum to 1 and the intercept is 0.

Additionally in the DGPs, supposing the number of non-zero parameters is p, then the p non-zero parameters are also split into two types, i.e. the "informative" parameters and the "weakly informative" parameters. The number of the informative parameters is $p/3$, while they take equal values and sum to 0.9. Therefore, the remaining $2p/3$ non-zero parameters are weakly informative, and they also take equal values and should sum to 0.1. By adopting this setup, a situation where control units are endowed different magnitudes of signal is mimicked. Moreover, following Kim et al. (2020) and Brodersen et al. (2015), the noise ε_t has small variance and zero mean.

Thus, the final DGPs can be summarized as

$$\begin{aligned} Y_{0t} &= (1 + eD_{0t})(\sum_{i=1}^{N} \beta_i Y_{it} + \varepsilon_t), \\ Y_{it} &\sim \text{Normal}(0, 1), \\ \varepsilon_t &\sim \text{Normal}(0, 0.02), \\ t &= 1, \dots, T, \end{aligned} \tag{21}$$

where

$$D_{0t} = \begin{cases} 0, \text{for } \ t = 1, \dots, T_0, \\ 1, \text{otherwise}, \end{cases}$$

$$\begin{aligned} e &= 0,\ 0.01,\ 0.05,\ 0.1,\ 1, \text{or } 2, \\ T &= 90 \text{ or } 30, \\ N &= T/2 \text{ or } 2T. \end{aligned}$$

And

$$\begin{pmatrix} \beta_1 = 0.9/\frac{N}{3} \\ \beta_2 = 0.9/\frac{N}{3} \\ \vdots \\ \beta_{N/3} = 0.9/\frac{N}{3} \\ \beta_{N/3+1} = 0.1/\frac{2N}{3} \\ \vdots \\ \beta_N = 0.1/\frac{2N}{3} \end{pmatrix},$$

in the cases that the level of sparsity equals 0, and

$$\begin{pmatrix} \beta_1 = 0.9/\frac{N}{15} \\ \beta_2 = 0.9/\frac{N}{15} \\ \vdots \\ \beta_{N/15} = 0.9/\frac{N}{15} \\ \beta_{N/15+1} = 0.1/\frac{2N}{15} \\ \vdots \\ \beta_{N/5} = 0.1/\frac{2N}{15} \\ \beta_{N/5+1} = 0 \\ \vdots \\ \beta_N = 0 \end{pmatrix},$$

when the level of sparsity is 0.8. For CIM which uses the Gibbs sampler, 4000 samples are carried out for each estimation, and the first 2000 samples are considered as the burn-in periods, which means that they are not used for the statistical inferences and only the remaining 2000 samples are employed for the credible intervals and posterior means. Additionally, for BSCM-Spike-and-Slab and BSCM-Horseshoe, which use the No-U-Turn sampler, each estimation utilizes 4 chains and each chain generates 4000 samples. And for each chain, the first 2000 samples are again removed, resulting in $(4000-2000)\times 4 = 8000$ samples for constructing the credible intervals and the posterior means. Convergence of the samplers for this setting is checked and confirmed by some trial runs.

4.2 Cumulative impacts

Once the counterfactual outcomes are predicted, the next question is how we can conclude whether there is any causal impact. In this paper to draw such conclusions, I use the so-called cumulative impact (Brodersen et al., 2015). Recall Equation (4) — the estimation of causal impact of the treated unit at time t, $t > T_0$. The estimated overall cumulative impact is simply the sum over all estimated post-treatment causal impacts, that is

$$\sum_{T_0+1}^{T} \hat{e}_{0t}. \tag{22}$$

And for each time point after the treatment, the estimated pointwise cumulative impact is defined as

$$\sum_{T_0+1}^{t} \hat{e}_{0t}, \quad \text{for} \quad t = T_0+1, ..., T. \tag{23}$$

Note that the overall cumulative impact is a special case of the pointwise cumulative impact where $t = T$. Then, we conclude that there is pointwise cumulative impact at t

or overall cumulative impact if the 95% credible interval of the corresponding cumulative impact did not include 0.

For each DGP of the 50 generated datasets, the percentages of each model correctly making overall cumulative causal decisions are shown. The main criteria to evaluate a model's performance — correctly making overall cumulative causal claims— means that the models conclude that at $t = T$, there is no cumulative causal impact when the true effect size e is 0, or conclude that there is cumulative causal impact when e is not 0. Additionally, the average percentages that each model makes correct pointwise cumulative causal claims are also compared for each DGP. That is, for all 50 simulated datasets of a DGP, the percentages of correct pointwise causal conclusions for $t = T_0 + 1, ..., T$ for each model is calculated, and averaged over the 50 datasets.

4.3 Simulation results

Table 1 — Table 7 give the results for the simulation study. We first look at the the proportions of correct conclusions regarding the overall and pointwise cumulative impacts. When $e = 0\%$ (Table 1), meaning there is no treatment effect in the DGPs, BSCM-Horseshoe gives the largest proportions of correct conclusions of the overall cumulative impact for 7 cases and of the pointwise cumulative impacts for all 8 cases. Sometimes, CIM and BSCM-Spike-and-Slab are too optimistic about making causal conclusions compared with BSCM-Horseshoe. For example, in the sparse case where $T = 30$, $N = 15$ and $e = 0\%$, CIM detects 28% overall cumulative impacts while the BSCM models only detect 10%; and in the cases where $T = 90$, $N = 180$ and $e = 0\%$, BSCM-Spike-and-Slab concludes that 32% and 38% of overall cumulative impacts are different from 0, whereas BSCM-Horseshoe only concludes that 10% of the overall cumulative impacts are not 0.

For Table 2 where $e = 1\%$, CIM outperforms BSCM-Spike-and-Slab and BSCM-Horseshoe in terms of both the correct conclusions of overall and pointwise cumulative causal impacts, except for the scenario that $T = 30$, $N = 15$ and the parameters are non-sparse, in which case BSCM-Spike-and-Slab is subject to the largest proportions of correct conclusions for overall cumulative impacts. And we find that CIM performs better than the two BSCMs for most of the cases in Table 2 to Table 7, in terms of correct conclusions about cumulative impacts. In short, CIM often draws more conclusions of having cumulative impacts, both overall and pointwise, than the two BSCMs, indicating that in general the performance of CIM is worse than those ofthe two BSCMs when there is in fact no treatment impacts in the DGPs, and is better in the cases where there are actual treatment impacts. Comparing BSCM-Spike-and-Slab and BSCM-Horseshoe, if we only look at the 48 cases where the effect sizes are not 0, regarding the overall cumulative impacts, BSCM-Spike-and-Slab exceeds BSCM-Horseshoe in 26 cases, whereas BSCM-Horseshoe does better in 13 cases, and they perform equally in the remaining 6

$e = 0\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.720	0.900	0.900	0.880	0.880	1.000
	Pointwise	0.708	0.977	0.979	0.874	0.952	0.993
	MSE (pre-treatment)	0.015	0.011	0.015	0.008	0.000	0.011
	MSE (post-treatment)	0.017	0.015	0.015	0.014	0.015	0.011
	Time (s)	0.722	49.891	9.795	1.201	259.863	21.737
Non-sparse	Overall	0.960	0.920	0.960	0.940	0.800	0.920
	Pointwise	0.948	0.987	0.992	0.872	0.994	0.978
	MSE (pre-treatment)	0.030	0.120	0.014	0.018	0.002	0.021
	MSE (post-treatment)	0.046	0.084	0.030	0.035	0.028	0.029
	Time (s)	0.789	36.756	12.781	1.093	168.269	21.517
$e = 0\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.860	0.840	0.920	0.880	0.680	0.900
	Pointwise	0.873	0.961	0.983	0.895	0.917	0.983
	MSE (pre-treatment)	0.021	0.008	0.018	0.019	0.000	0.015
	MSE (post-treatment)	0.012	0.017	0.012	0.011	0.024	0.012
	Time (s)	2.093	1146.075	21.565	3.034	673.553	76.661
Non-sparse	Overall	0.900	0.920	0.980	0.860	0.620	0.900
	Pointwise	0.905	0.967	0.977	0.859	0.897	0.972
	MSE (pre-treatment)	0.042	0.012	0.016	0.027	0.000	0.020
	MSE (post-treatment)	0.034	0.020	0.023	0.018	0.029	0.017
	Time (s)	2.057	139.992	27.451	2.937	685.231	70.447

Table 1: Simulation results for the cases where the effect size $e = 0$. In the table,'Overall' is the proportion of correctly detecting overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting pointwise cumulative impacts; the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; the average mean squared error of predictions for the post-treatment periods is denoted as 'MSE (post-treatment)';and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

cases. In terms of the pointwise cumulative impacts, BSCM-Spike-and-Slab gives larger proportions of correct causal conclusions in 31 out of the 48 cases where $e > 0$.

$e = 1\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.180	0.080	0.040	0.220	0.080	0.080
	Pointwise	0.160	0.015	0.012	0.246	0.026	0.023
	MSE (pre-treatment)	0.017	0.014	0.017	0.008	0.000	0.011
	Time (s)	0.770	50.153	9.972	1.109	228.131	19.208
Non-sparse	Overall	0.060	0.080	0.020	0.400	0.280	0.160
	Pointwise	0.076	0.017	0.010	0.480	0.115	0.077
	MSE (pre-treatment)	0.031	0.112	0.015	0.019	0.002	0.019
	Time	0.773	36.287	12.039	1.094	179.258	24.102
$e = 1\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.180	0.140	0.140	0.120	0.080	0.040
	Pointwise	0.137	0.032	0.022	0.099	0.025	0.017
	MSE (pre-treatment)	0.019	0.007	0.017	0.018	0.000	0.014
	Time	1.908	150.269	19.949	3.095	704.430	78.199
Non-sparse	Overall	0.180	0.100	0.080	0.160	0.080	0.000
	Pointwise	0.150	0.034	0.027	0.139	0.029	0.019
	MSE (pre-treatment)	0.042	0.011	0.015	0.028	0.000	0.019
	Time (s)	2.025	138.259	27.629	3.031	673.139	75.748

Table 2: Simulation results for the cases where the effect size $e = 1\%$. In the table, 'Overall' is the proportion of correctly detecting overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting pointwise cumulative impacts; the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

It is also of interest to compare the performances of each model under different dimensionality, namely, $N/T = 0.5$ and $N/T = 2$, indicating that we examine $4 \times 7 = 28$ cases for each model. And here we mainly look at the proportions of correctly detecting the overall cumulative impacts. In summary, there are 17 cases where the performance of CIM when $N/T = 0.5$ is equal to or better than the performance of CIM when when

$e = 5\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.260	0.060	0.100	0.240	0.060	0.060
	Pointwise	0.264	0.017	0.009	0.208	0.033	0.025
	MSE (pre-treatment)	0.014	0.010	0.013	0.008	0.000	0.011
	Time (s)	0.713	50.735	9.881	1.129	230.187	20.641
Non-sparse	Overall	0.100	0.100	0.020	0.240	0.120	0.040
	Pointwise	0.124	0.035	0.017	0.240	0.061	0.039
	MSE (pre-treatment)	0.035	0.119	0.014	0.018	0.001	0.020
	Time	0.736	34.550	11.936	1.158	179.841	24.882
$e = 5\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.160	0.120	0.080	0.100	0.100	0.080
	Pointwise	0.161	0.032	0.025	0.136	0.018	0.020
	MSE (pre-treatment)	0.020	0.008	0.018	0.019	0.000	0.015
	Time	1.904	140.782	19.806	3.095	712.958	75.777
Non-sparse	Overall	0.120	0.080	0.080	0.140	0.120	0.060
	Pointwise	0.157	0.031	0.028	0.118	0.035	0.018
	MSE (pre-treatment)	0.040	0.126	0.016	0.027	0.000	0.021
	Time (s)	2.078	130.848	27.706	3.171	757.275	73.380

Table 3: Simulation results for the cases where the effect size $e = 5\%$. In the table, 'Overall' is the proportion of correctly detecting overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting pointwise cumulative impacts; the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

$e = 10\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.360	0.080	0.040	0.200	0.080	0.060
	Pointwise	0.312	0.023	0.019	0.196	0.030	0.016
	MSE (pre-treatment)	0.014	0.011	0.014	0.008	0.000	0.011
	Time (s)	0.807	54.998	10.7044	1.148	228.652	20.963
Non-sparse	Overall	0.140	0.080	0.020	0.260	0.160	0.020
	Pointwise	0.158	0.023	0.017	0.262	0.055	0.023
	MSE (pre-treatment)	0.035	0.115	0.015	0.018	0.002	0.021
	Time	0.786	36.776	11.596	1.228	188.856	25.557
$e = 10\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.120	0.140	0.060	0.120	0.060	0.080
	Pointwise	0.127	0.037	0.016	0.135	0.021	0.024
	MSE (pre-treatment)	0.021	0.008	0.017	0.019	0.000	0.014
	Time	1.945	143.920	19.929	3.335	742.044	83.970
Non-sparse	Overall	0.140	0.120	0.120	0.200	0.020	0.080
	Pointwise	0.143	0.032	0.029	0.149	0.019	0.024
	MSE (pre-treatment)	0.038	0.011	0.015	0.026	0.000	0.020
	Time (s)	2.114	133.329	28.009	3.024	706.029	68.527

Table 4: Simulation results for the cases where the effect size $e = 10\%$. In the table, 'Overall' is the proportion of correctly detecting overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting pointwise cumulative impacts; the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

$N/T = 2$. And for BSCM-Spike-and-Slab and BSCM-Horseshoe, this number increases to 20 and 19, respectively. This is an indication that the estimation for high-dimensional datasets is more difficult than that for datasets where the numbers of control variables are smaller than the numbers of time points. Another important aspect that needs inspection is whether the proportions of correctly detecting overall cumulative impacts differ for each model when the true parameters are sparse and non-sparse, ceteris paribus. Table 8 tells us that CIM and BSCM-Horseshoe perform better for the sparse cases, whereas BSCM-Spike-and-Slab works better for the non-sparse cases.

There are two other metrics shown in Table 1 — Table 7, namely MSE and Time. MSE is the mean squared error between the observed target variable and the prediction of

$e = 50\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.360	0.160	0.180	0.300	0.160	0.120
	Pointwise	0.410	0.063	0.064	0.366	0.076	0.057
	MSE (pre-treatment)	0.014	0.010	0.014	0.008	0.000	0.010
	Time (s)	0.807	55.2576	10.879	1.060	231.297	19.598
Non-sparse	Overall	0.200	0.200	0.160	0.260	0.220	0.120
	Pointwise	0.190	0.051	0.047	0.328	0.105	0.060
	MSE (pre-treatment)	0.034	0.120	0.014	0.016	0.002	0.020
	Time	0.743	36.047	12.027	1.093	167.303	23.279
$e = 50\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.280	0.140	0.140	0.300	0.160	0.180
	Pointwise	0.283	0.055	0.063	0.293	0.056	0.068
	MSE (pre-treatment)	0.020	0.008	0.018	0.018	0.000	0.014
	Time	1.883	144.042	19.419	3.021	692.874	75.921
Non-sparse	Overall	0.200	0.220	0.200	0.180	0.120	0.100
	Pointwise	0.235	0.067	0.060	0.206	0.037	0.044
	MSE (pre-treatment)	0.043	0.012	0.017	0.028	0.000	0.020
	Time (s)	2.014	133.205	27.317	3.181	725.407	77.360

Table 5: Simulation results for the cases where the effect size $e = 50\%$. In the table, 'Overall' is the proportion of correctly detecting overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting pointwise cumulative impacts; the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

$e = 100\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.540	0.400	0.280	0.520	0.240	0.220
	Pointwise	0.482	0.105	0.091	0.462	0.088	0.090
	MSE (pre-treatment)	0.015	0.011	0.014	0.008	0.000	0.009
	Time (s)	0.759	53.017	10.451	1.129	274.767	21.927
Non-sparse	Overall	0.220	0.120	0.240	0.560	0.460	0.360
	Pointwise	0.358	0.081	0.101	0.520	0.145	0.092
	MSE (pre-treatment)	0.035	0.114	0.017	0.018	0.002	0.018
	Time	0.828	40.325	12.959	1.108	181.039	23.657
$e = 100\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.400	0.280	0.300	0.240	0.100	0.180
	Pointwise	0.473	0.106	0.118	0.357	0.062	0.091
	MSE (pre-treatment)	0.019	0.007	0.017	0.018	0.000	0.014
	Time	1.907	148.695	20.017	3.165	714.585	79.122
Non-sparse	Overall	0.360	0.420	0.360	0.340	0.220	0.220
	Pointwise	0.410	0.126	0.115	0.365	0.084	0.092
	MSE (pre-treatment)	0.044	0.012	0.016	0.027	0.000	0.021
	Time (s)	2.045	134.972	27.956	3.135	789.278	72.512

Table 6: Simulation results for the cases where the effect size $e = 100\%$. In the table, 'Overall' is the proportion of correctly detecting overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting pointwise cumulative impacts;the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

counterfactual. In Table 1, the MSEs for both the pre-treatment and the post-treatment periods are reported, averaged over the 50 generated datasets for each case; and for Table 2 to Table 7, only the pre-treatment MSEs are shown. The average MSE calculated for the pre-treatment periods can be regarded as a measure of the goodness of pre-treatment fit. A better pre-treatment fit results in a lower in-sample MSE, and vice versa. Similarly, the average MSE of the post-treatment periods is used to evaluate the goodness of post-treatment prediction accuracy. However, a smaller in-sample MSE does not necessarily result in a smaller prediction MSE. When the issue of overfitting is present, the pre-treatment MSE is generally small but the post-treatment MSE is large. Table 1 shows that BSCM-Horseshoe provides the smallest MSE (pre-treatment) for 6 out of 8 cases and

$e = 200\%$		$T = 30$ $N = 15$			$T = 30$ $N = 60$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.620	0.480	0.480	0.480	0.200	0.260
	Pointwise	0.644	0.170	0.169	0.518	0.111	0.123
	MSE (pre-treatment)	0.015	0.012	0.015	0.008	0.000	0.011
	Time (s)	0.723	49.393	9.775	1.068	246.192	19.258
Non-sparse	Overall	0.480	0.360	0.480	0.440	0.340	0.360
	Pointwise	0..544	0.143	0.178	0.552	0.162	0.135
	MSE (pre-treatment)	0.033	0.129	0.015	0.019	0.003	0.021
	Time	0.752	35.519	12.128	1.088	165.864	22.609
$e = 200\%$		$T = 90$ $N = 45$			$T = 90$ $N = 180$		
		CIM	BSCM-spike	BSCM-Horseshoe	CIM	BSCM-spike	BSCM-Horseshoe
Sparse	Overall	0.520	0.320	0.400	0.440	0.260	0.300
	Pointwise	0.570	0.143	0.163	0.484	0.090	0.130
	MSE (pre-treatment)	0.020	0.008	0.018	0.019	0.000	0.015
	Time	1.900	150.041	19.249	3.070	716.408	74.597
Non-sparse	Overall	0.480	0.480	0.480	0.560	0.400	0.500
	Pointwise	0.554	0.186	0.174	0.062	0.156	0.179
	MSE (pre-treatment)	0.039	0.011	0.015	0.027	0.000	0.019
	Time (s)	2.066	134.280	27.553	3.264	720.342	76.599

Table 7: Simulation results for the cases where the effect size $e = 200\%$. In the table, 'Overall' is the proportion of correctly detecting the overall cumulative impacts; 'Pointwise' is the proportion of correctly detecting the pointwise cumulative impacts; the average mean squared error of predictions for the pre-treatment periods is denoted as 'MSE (pre-treatment)'; and 'Time (s)' is the average computational time, measured in seconds. BSCM-spike is short for BSCM-Spike-and-Slab.

Number of cases	CIM	BSCM-Spike-and-Slab	BSCM-Horseshoe	Total
Sparse	14	8	13	35
Non-sparse	11	14	11	36
Equal	3	6	4	13
Total	28	28	28	84

Table 8: This table shows for each model, in terms of overall cumulative impacts how many times it performs better in the sparse case or the non-sparse case, or performs equally in the two cases, ceteris paribus.

the smallest MSE (post-treatment) for also 6 out of 8 cases, and this indicates the BSCM-Horseshoe performs well in terms of both in-sample fit and out-of-sample prediction. On the contrary, BSCM-Spike-and-Slab provides in 3 out of the 4 high-dimensional cases the smallest in-sample MSE and the largest out-of-sample MSE, which means that BSCM-Spike-and-Slab might be subject to overfitting issues in the high-dimensional cases.

The computational times of the three models differ dramatically, and it is clear that CIM is the fastest method and BSCM-Spike-and-Slab is subject to the largest computing times for all cases. Moreover, the computational time grows when the number of control variables is larger, and a sparse problem in average needs more time for estimation than a non-sparse problem.

In summary for the cases that are studied in this section, BSCM-Horseshoe is more reliable when the effect size is actually 0; CIM on the contrary is more optimistic to make causal claims and is computationally the fastest; BSCM-Spike-and-Slab has relatively poorer ability to deal with high-dimensional problems, due to its potential overfitting issues reflected by the pre-treatment the post-treatment MSEs shown in Table 1, and it needs more computational time. As is expected, the proportions of correctly detecting overall and pointwise cumulative impact increases as the true effect size grows in the DGP, ceteris paribus. Moreover, the accuracy of detecting causal impact is lower in high-dimensional problems for all three models. It is also important to mention that in most of the cases, CIM and BSCM-Horseshoe deal with sparse problems better than non-sparse problems, whereas the BSCM-Spie-and-Slab often works better in non-sparse cases.

5 German reunification

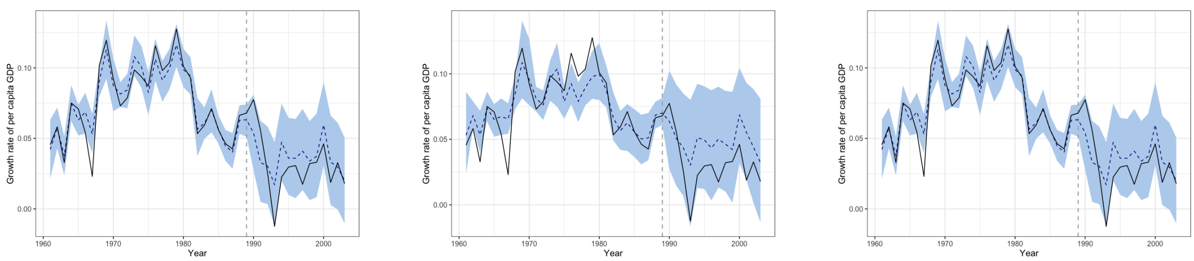
5.1 Background and data

This section studies the treatment impact on (the former) West Germany's economy of one significant postwar political integration — the German reunification in 1990. Being separated for 45 years, the German Democratic Republic (the former East Germany) and the Federal Republic of Germany reunified on 3 October, 1990. Abadie et al. (2015) were the first to estimate the reunification's impact on West Germany's per capita GDP using the standard synthetic control method. They constructed a synthetic West Germany in terms of its per capita GDP using the per capita GDPs and some other economic growth predictors of 16 developed countries. The synthetic West Germany serves as the counterfactual, namely what would have happened to West German per capita GDP had no reunification occurred. And the impact of the reunification is the the difference between the counterfactual and the observed per capita GDP of West Germany. The original dataset (Abadie et al.,2015) contains yearly economic indices of West Germany and other 16 OECD countries across 1960 - 2003. The 16 OECD countries are Australia, Austria, Belgium, Denmark, France, Greece, Italy, Japan, the Netherlands, New Zealand, Norway, Portugal, Spain, Switzerland, the U.K, and the U.S. And the post-reunification data for West Germany is only for the former West Germany's territory. The economic indices include the PPP-adjusted real per capita GDP in 2002 U.S dollars of the 17 countries and some other variables including the inflation rate, investment rate, and so on, which are not utilized in this paper.

One potential issue of the analysis about German reunification in Abadie et al. (2015) is that it ignores the time-series features of per capita GDPs such as non-stationarity, pointed out by Masini and Medeiros (2019). Thus, instead of per capita GDPs, in this paper the growth rates of per capita GDPs from 1991 to 2003 are utilized. The unification's impact on West Germany's growth rate of per capita GDP is studied using the three Bayesian methods, and the results are compared between each other. Another difference is that Abadie et al. (2015) use also other predictors for economic growth such as the inflation rate as control variables, while in this paper the three Bayesian methods only rely on the untreated countries' growth rates of per capita GDPs, following Kim et al. (2020) which stress the concerns about overfitting raised from using other control variables and following Doudchenko and Imbens (2016) which claim that the outcome variables have high predictive power.

Our target variable is the growth rates of per capita GDP of West Germany from 1991 to 2003, and the control variables are the growth rates of per capita GDPs for the other 16 OECD countries. For the Bayesian estimations, CIM generates 50000 MCMC samples and the first 25000 samples are regarded as the burn-in periods; BSCM-Spike-and-Slab and BSCM-Horseshoe each uses 4 No-U-Turn chains where each chain uses 4000 iterations with the first 2000 are again the burn-in periods. Because the CIM uses only 1 MCMC chain, the convergence is diagnosed and confirmed by the Geweke convergence diagnostic (Geweke, 1991). Additionally, because BSCM-Spike-and-Slab and BSCM-Horseshoe each generates more than 1 MCMC chain, in this paper the R-hat convergence diagnostic proposed by Vehtari et al. (2020) is employed to check their convergences. The convergence of BSCM-Horseshoe is confirmed by the R-hat convergence diagnostic, whereas BSCM-Spike-and-Slab does not eventually converge for even more iterations.

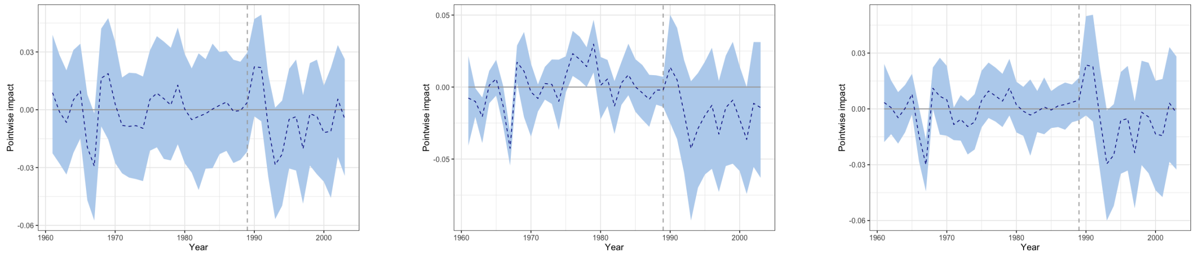
5.2 Estimation results



(a) CIM (b) BSCM-Spike-and-Slab (c) BSCM-Horseshoe

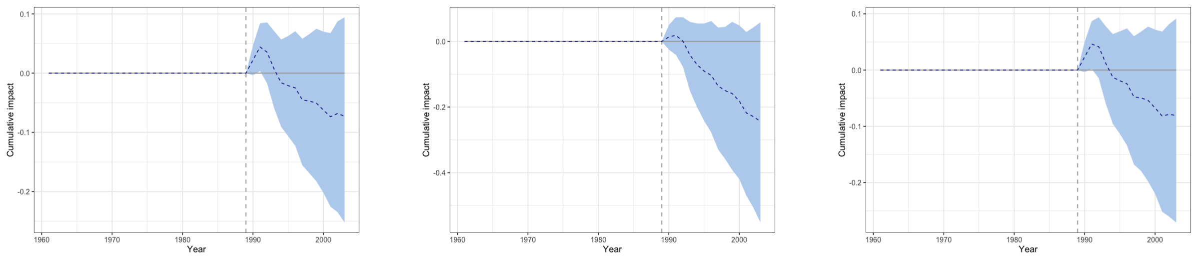
Figure 4: Figure (a), (b), and (c) show the plots of both the observed (black lines) and predicted (dashed lines) growth rate of per capita GDP of West Germany from 1961 to 2003, as well as the 95% credible intervals of the predictions, obtained from CIM, BSCM-Spike-and-Slab and BSCM-Horseshoe respectively,.

The dashed lines in Figure 4 are the predicted growth rates of per capita GDPs of West Germany, which are seen as the counterfactuals, obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe respectively. The observed growth rates of per capita GDP are plotted as black lines. We can see that the pre-treatment fits of the CIM and BSCM-Horseshoe are relatively good, whereas the pre-treatment fit for BSCM-Spike-and-Slab deviates more from the observed values. The counterfactuals are slightly smaller than the observed values at the beginning of the post-treatment periods, while they turn to exceed the observed growth rates of per capita GDPs later on. The gaps between the observed and the counterfactual growth rates are plotted in Figure 5, from where we see



(a) CIM (b) BSCM-Spike-and-Slab (c) BSCM-Horseshoe

Figure 5: Figure (a), (b), and (c) show the estimated treatment effect of the German reunification for each time point (calculated by Equation (4)) from 1961 to 2003, as well as the 95% credible intervals, obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe respectively.



(a) CIM (b) BSCM-Spike-and-Slab (c) BSCM-Horseshoe

Figure 6: Figure (a), (b), and (c) show the estimated cumulative treatment effect of the German reunification for each time point (post-treatment) from 1961 to 2003, as well as the 95% credible intervals, obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe respectively.

negative differences between the observed and the counterfactual for most of the post-treatment years. And this indicates that for all the three models, across 1990 to 2003 the cumulative effect of the German reunification on West Gernamy's growth rates of per capita GDP should be negative. And we confirm this by looking at Figure 6, which gives the cumulative effects of the reunification for each post-treatment year determined by the three models. Now we compare the estimated overall cumulative effect sizes computed by the three Bayesian models. BSCM-Spike-and-Slab gives the (negatively) largest estimated effect size, which is -22%, and CIM gives the smallest one which equals -15%, while a cumulative effect size of -16% is estimated by BSCM-Horseshoe. Although the effect size estimated by the models are all negative, Figure 6 tells us that the 95% credible interval of the overall cumulative effects for all the three model do not exclude 0. And this indicates that the overall cumulative impact of German reunification on West Germany's growth rates of per capita GDP should be be considered different from 0 according to the three models.

Model	Intercept	Australia	Austria	Belgium	Denmark	France
CIM	0 [0, 0]	0.003 [0, 0]	0.200 [0, 0.696]	0.217 [0, 0.727]	0.081 [0, 0.563]	0.138 [0, 0.849]
BSCM-Spike-and-Slab	0.014 [0.000, 0.026]	0.017 [-0.041, 0.072]	0.167 [-0.049, 0.820]	0.016 [-0.046, 0.081]	0.024 [-0.045, 0.096]	0.457 [-0.031, 1.183]
BSCM-Horseshoe	-0.010 [-0.031, -0.012]	0.020 [-0.083, 0.193]	0.151 [-0.292, 0.445]	0.083 [-0.064, 0.454]	0.170 [-0.021, 0.481]	0.057 [-0.115, 0.542]
Model	Greece	Italy	Japan	Netherlands	New Zealand	Norway
CIM	0.005 [0, 0]	0.001 [0, 0]	0.001 [0, 0]	0.051 [0, 0.543]	0.001 [0, 0]	0.001 [0, 0]
BSCM-Spike-and-Slab	0.009 [-0.037, 0.055]	0.009 [-0.052, 0.069]	0.004 [-0.059, 0.069]	0.019 [-0.044, 0.090]	0.017 [-0.034, 0.075]	0.009 [-0.051, 0.066]
BSCM-Horseshoe	0.027 [-0.021, 0.010]	0.012 [-0.100, 0.173]	-0.006 [-0.128, 0.093]	0.182 [-0.019, 0.500]	0.013 [-0.060, 0.119]	0.005 [-0.102, 0.127]
Model	Portugal	Spain	Switzerland	U.K.	U.S.	
CIM	0.000 [0, 0]	0.000 [0, 0]	0.007 [0, 0]	0.014 [0, 0.300]	0.368 [0, 0.637]	
BSCM-Spike-and-Slab	0.004 [-0.05, 0.065]	-0.002 [-0.063, 0.060]	0.019 [-0.045, 0.081]	0.015 [-0.045, 0.072]	0.152 [-0.044, 0.700]	
BSCM-Horseshoe	0.007 [-0.071, 0.103]	0.002 [-0.122, 0.132]	0.047 [-0.055, 0.265]	0.080 [-0.057, 0.416]	0.281 [-0.009, 0.576]	

Table 9: The table shows the posterior means and the 95% credible intervals of the intercept and the regression parameters β_i, $i = 1, ..., 16$, which are corresponding to the growth rates of per capita GDPs of the 16 OECD countries, estimated by CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe.

Now we investigate the regression parameters estimated by the three Bayesian models, by considering the posterior means and the 95% credible intervals shown in Table 9. For the intercept, the posterior mean obtained by CIM is 0, while it is not the case for BSCM-Horseshoe. For CIM, the largest posterior mean of the regression parameter is the U.S. (0.368), followed by Belgium (0.217) and Austria (0.200). For BSCM-Horseshoe, the U.S. again takes on the largest posterior mean of the regression parameter (0.281), followed by Denmark (0.170) and Austria (0.151). We do not discuss about the posterior results of BSCM-Spike-and-Slab because its MCMC chains do not converge. Additionally, none of the 95% credible intervals presented here exclude 0 in Table 9, meaning that using 95% credible intervals, none of the regression parameters is considered to be different from 0. This is an indication that the outcome variables of the control units, namely the growth rates of per capita GDPs of the 16 untreated OECD countries, cannot provide sufficient predictive power. As a result, the 95% credible intervals of the pre-treatment fits obtained from the Bayesian models sometimes exclude the true growth rate of West German per capita GDP (Figure 4). Therefore for this application, we should be concerned more about underfitting rather than overfitting, and one direction to the next step is employing other predictors. And the predictors for economic growth utilized by Abadie et al. (2015) such as inflation rates are sensible candidates.

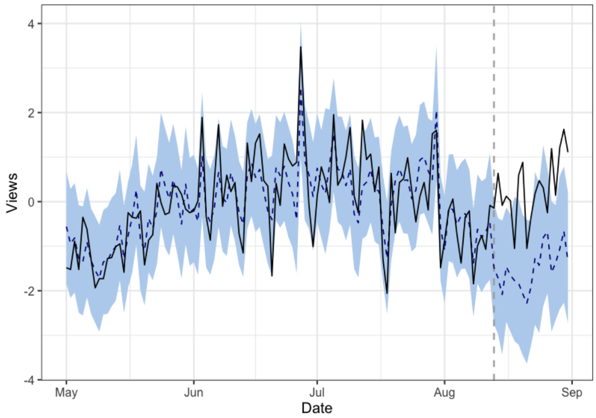
6 Online page views

6.1 Background and data

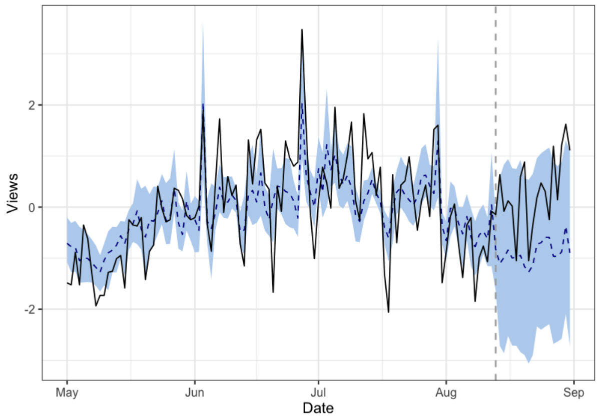
Bol.com is a Dutch webshop headquartered in Utrecht, the Netherlands, founded in 1999. It now has about 11 million active customers in the Netherlands and Belgium, offering more than 23 million articles from more than 20 product categories. The data used in this section is provided by bol.com, and the goal is to evaluate the causal impact of a treatment which is called "eagle" on average online page views of the treated products. Eagle is a method that helps to predict missing attributes of products on t he webshop. The missing attributes can be products' titles or descriptions. When a seller uploads to bol.com a product that is missing mandatory attributes of the product's description, normally the selling offer of this product cannot be shown on bol.com because of the missing mandatory components. But by using eagle, these missing attributes are predicted and filled in, and the product is then available for viewing and purchase. There are also cases where the mandatory attributes of the products are all filled in and so the product is available for purchase, but some other optional attributes are blank. Employing eagle, those blank optional attributes can also be predicted and filled, making the product' information that customers can see more comprehensive. It is expected that eagle provides a positive treatment impact on the average online page views of the treated product. The reason is that when a missing mandatory attribute is filled in by eagle, the product is available for purchase so that the consumers are able to view the product; and when some optional attributes are predicted by eagle, the information about the product provided to the customers are more sufficient and then we also expect that a customer tends to view a product that has more information than a product that provides less information.

The original dataset contains daily numbers of online page views respectively for 6465102 products on bol.com, from 1 May, 2020, to 31 August, 2020. The products are uniquely identified by "global_id". However, there are some products that were not uploaded to bol.com until some day between 1 May, 2020 and 31 August, 2020. For example, if a product was uploaded to bol.com on 1 June, 2020, then for this product we have missing values for the periods from 1 May to 31 May. Thus, such products are removed, resulting in 865745 products. Although this removal may cause a lose of information, it is still preferred since we assume that the relations between the control variables and the target variable is stable during the whole period, if the treatment did not occur. Thus, it makes more sense to only use the products for which the information is available for the whole period. Moreover, each product belongs to a "chunk", which is the label for the classification of this product. The 865745 products are distributed to 1433 different chunks, and the largest chunk, 'Hoesje voor mobiele telefoon' ('mobile phone cases', in English), has 53106 products, while the smallest chunk, 'Caravanhoes' ('caravan cover', in English) has only 1 product. The treatment, eagle, is performed on some products on 14 August, 2020, and some other products on 21 August, 2020. We want to solely analyze the treatment effect of eagle on the products that received eagle on 14 August. Therefore, the products that were treated on 21 August are removed. Moreover, to effectively reduce the computational burden, only the chunks that contain more than 1000 products are kept, resulting in 154 chunks and 651844 products.

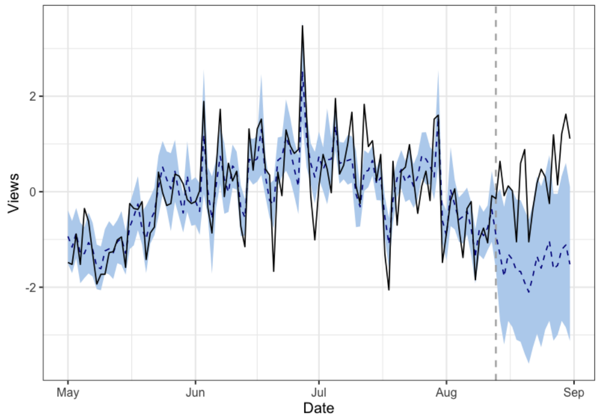
The treatment, eagle, is performed on 464 out of the 651844 products on 14 August, 2020. Note that the assignment of the treatment is not decided by the chunks, meaning that there might be some chunks that include no treated product. Those chunks containing no treated product and the products belonging to them are also removed. At the end we have 73 chunks that contain at least one treated product and the 73 chunks contain 467189 untreated products. The target time series is defined as the mean of the online page views of the 464 treated products, and is standardized so that it has 0 mean and its standard deviation is 1. The control time series utilized here are the means of online page views of the products within each chunk that did not receive the treatment. Therefore, we have 73 control time series. The pre-treatment period is from 1 May, 2020, to 13 August, 2020; and the post-treatment period is from 14 August, 2020, to 31 August, 2020.



(a) CIM



(b) BSCM-Spike-and-Slab

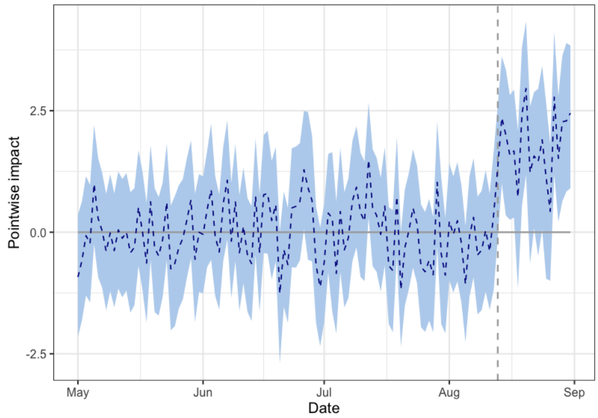


(c) BSCM-Horseshoe

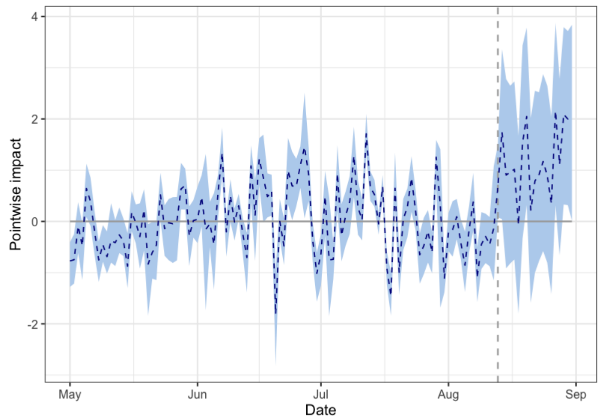
Figure 7: Figure (a), (b), and (c) show the plots of both the observed (black lines) and predicted (dashed lines) (average) online page views of the product that received eagle on 14 August, 2020, from 1 May, 2020, to 31 August, 2020, as well as the 95% credible intervals of the predictions, obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe respectively.

6.2 Estimation results

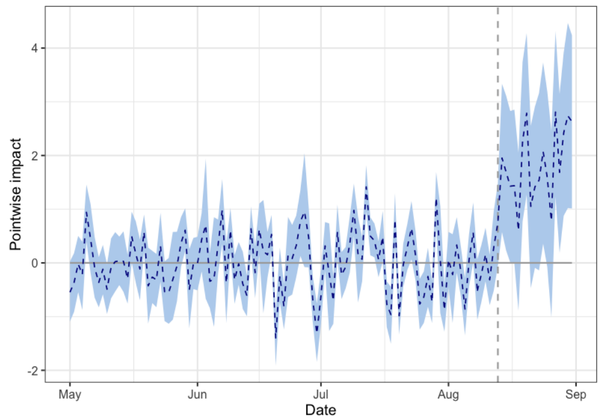
Figure 7 shows the (standardized) observed average online page views for the treated products, as well as the predictions obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe, respectively, from which we see clear differences between the post-treatment observed values and the predicted counterfactuals. And such differences are plotted in Figure 8. In the post-treatment periods, the pointwise impacts of the eagle



(a) CIM



(b) BSCM-Spike-and-Slab

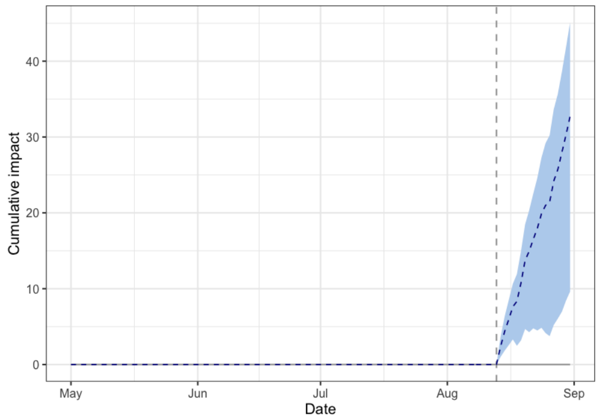


(c) BSCM-Horseshoe

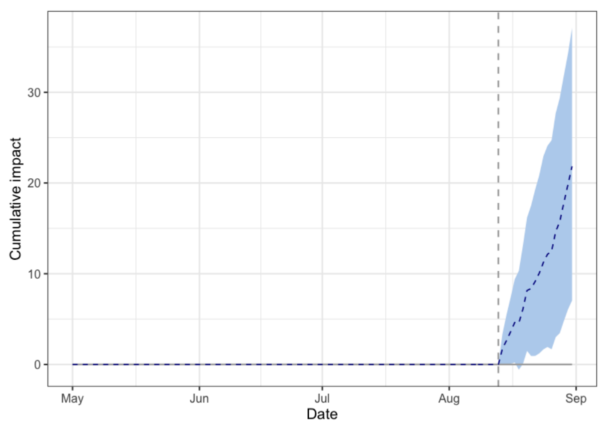
Figure 8: Figure (a), (b), and (c) show the estimated treatment effect of the eagle treatment for each time point (calculated as Equation (4)) from 1 May, 2020, to 31 August, 2020, as well as the 95% credible intervals, obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe respectively.

treatment on average online pages for the treated products are all larger than or equal to 0. The MSEs, which are the mean squared errors between the observed and the predicted values, for the pre-treatment predictions are 0.364, 0.444, and 0.291 respectively for CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe, meaning that BCM-Horseshoe has the best pre-treatment fit in terms of the MSE.

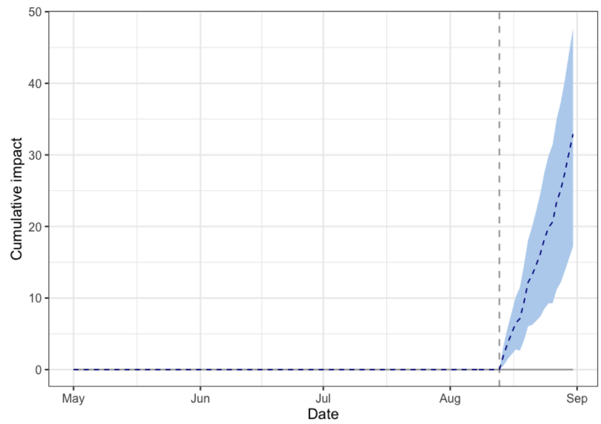
To further evaluate the treatment effect, we still need to investigate the cumulative impacts. The cumulative impacts and their 95% credible intervals estimated by CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe are plotted in Figure 9. Additionally, not only positive cumulative impacts are shown in the plots, but also we see the exclusions of 0 for the 95% credible intervals of the overall cumulative impacts. This is an evidence that the eagle treatment has imposed positive impact on the average online page views of the treated products. The estimated effect sizes obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe are respectively -121.19%, -135.48% and -121.04%, which are negative because the predicted counterfactuals are negative due to that the target outcome time-series is standardized, and therefore a negative estimated effect size indicates a positive overall cumulative impact. The three estimated effect sizes do not differ tremendously, and especially the two effect sizes estimated by CIM and BSCM-Horseshoe are very close to each other.



(a) CIM



(b) BSCM-Spike-and-Slab



(c) BSCM-Horseshoe

Figure 9: Figure (a), (b), and (c) show the estimated cumulative treatment effect of the eagle treatment for each time point (post-treatment), as well as the 95% credible intervals, obtained from CIM, BSCM-Spike-and-Slab, and BSCM-Horseshoe respectively.

In order to examine whether the estimations of regression parameters are different from 0, again we look at the exclusion of 0 of their 95% credible intervals. Because of the large number of control variables, we do not show all the 95% credible intervals obtained from the three models. In short, for CIM only the regression parameter that is corresponding to the 'Outdoor vest' chunk is different from 0; and for the two BSCM models, only the 95% credible intervals of their intercept parameters exclude 0, indicating no valid control variable is found by the models. This is again, opposite to the argument from Kim et al. (2020), which only use the outcome time-series of the untreated units as the control variables. Thus, for this application we can as well consider to employ other predictors, and opinions from experts who are experienced in online marketing about which control variables are sensible are needed.

The empirical study about online page views as well as the application in German reunification in Section 5 all correspond to the Monte Carlo study for which the DGPs are low-dimensional, where the numbers of control variables do not exceed the numbers of time periods. From Section 4 we know that BSCM-Horseshoe is conservative to make causal claims. This indicates that when BSCM-Horseshoe claims that the cumulative impact is different from 0, which is indeed the case for the two empirical studies, the conclusions of non-zero cumulative impact become more trustworthy.

7 Conclusion

In recent years, the evaluation of marketing and policy intervention has obtained more and more attention. Apart from doing randomized controlled experiments, quasi-experimental and observational methods are also widely applied. In this paper, I review the idea of one important observational method of evaluating treatment impact which is called the SCM (Abadie and Gardeazabal, 2003), and study three fully Bayesian models, namely CIM (Brodersen et al., 2015), BSCM-sike-and-slab, and BSCM-Horseshoe (Kim et al., 2020), which belong to the family of the SCM. A comparison between the three models about their performances of detecting cumulative impacts in different scenarios is conducted by a Monte Carlo simulation study, and they are also applied in two empirical cases, which to my best knowledge have not been done in the current literature.

In this research, I first illustrate how the Bayesian frameworks deal with the convex restrictions on weights, statistical inferences, and high-dimensional and sparse issues, as well as the theoretical background of the utilized Bayesian models. Then, the performances of correctly detecting treatment effects of the three Bayesian models under variant scenarios are compared by a Monte Carlo simulation study. In summary, CIM performs better in detecting existent cumulative impacts and is computationally the fastest, while BSCM-Spike-and-Slab and BSCM-Horseshoe are more conservative about making causal claims and require more running time. This also mean that CIM can be overly optimistic to conclude non-zero cumulative impact when there is in fact no treatment effect. It is also worth mentioning that even when the effect size in the DGPs is 200%, in the 8 scenarios, the models can at most correctly detect 62% of the overall cumulative impacts. Moreover, BSCM-Spike-and-Slab has relatively poor ability to deal with high-dimensional problems, which is reflected by the fact that it becomes very computationally demanding comparing with the other two methods, and that it may be subject to overfitting issue in high-dimensional cases.

To demonstrate how the three models are applied on real world data, I conduct two empirical studies. The first one is to evaluate the impact of the German reunification in 1990 on West Germany's growth rate of per capita GDP, from 1991 to 2003. Using the growth rates of per capita GDPs of 16 OECD countries, the estimated cumulative impacts form the three Bayesian models range from -15% to -22%. However, BSCM-

Spike-and-Slab does not converge, and none of the estimated overall cumulative impacts and regression parameters of the control variables are considered to be different from 0 by looking at their 95% credible intervals. The second empirical study aims to evaluate the impact of a technique called eagle, which is used to predict and complete the missing attributes of online products, on online page views. The estimated effect sizes of CIM, BSCM-Spike-and-Slab and BSCM-Horseshoe are respectively -121.19%, -135.48% and -121.04%, meaning that the 'eagle' treatment increases the (average) online page views. Moreover, the 95% credible intervals of the overall cumulative impacts obtained from the three Bayesian models exclude 0, which indicates that the three models all conclude that the overall cumulative impact is different from 0. Additionally, for this empirical study, again no regression parameters of the control variables are different from 0.

There are concerns that one has to consider when employing these methods. First, since they all evaluate the treatment impact by modelling the counterfactual, we would expect that the pre-treatment fit is good because without the treatment the counterfactual should by definition be the same as the observed outcome variable. And this requires that we have enough information about the control variables to obtain sensible predictions. The two empirical studies conducted in this thesis, following Kim et al. (2020) I intentionally only use the outcome time-series of the untreated units as the control variables, and it turns out that none of the corresponding regression parameter is considered different from 0. Thus, for further research of these two particular empirical studies, it is advised to employ more control variables other than only the outcome variables.

Additionally, although Bayesian variable selection frameworks are used by those models, a pre-selection of control variables according to experts' knowledge might be preferred. For example, Abadie et al. (2015) filtered out some other OECD countries beforehand because they were subject to far-reaching structural shocks during the post-treatment period, which means that the relation between those countries and the synthetic West Germany may change during the post-reunification period, resulting in the current data set used in this paper. Similarly, for the empirical study of average online page views, in the future one could further look into the control variables and remove those chunks that were exposed to any third factor that cause severe structural changes to the relation between the treated variable and the control variables.

Moreover, although in this paper I try to avoid the predictive uncertainty by not to use time-series components that are too flexible in CIM, whether or not to use the more dynamic time-series components needs to be determined on a case by case basis. When there is lack of information about the control variables, one might tend to explore more on the target variable's own time-series pattern. Finally, in this research, due to limited time and computational power of the machine, only 50 data sets are generated for each case of the simulation study. In future research, one can increase the number of simulated data sets, as well as employing more efficient and faster sampling mechanism.

References

Abadie, A. (2019). Using synthetic controls: Feasibility, data requirements, and methodological aspects. *Journal of Economic Literature (Forthcoming)*.

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

Abadie, A., Diamond, A., and Hainmueller, J. (2011). Synth: An r package for synthetic control methods in comparative case studies. *Journal of Statistical Software*, 42(13).

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

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

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

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

Arkhangelsky, D., Athey, S., Hirshberg, D. A., Imbens, G. W., and Wager, S. (2019). Synthetic difference in differences. Technical report, National Bureau of Economic Research.

Athey, S. and Imbens, G. W. (2017). The state of applied econometrics: Causality and policy evaluation. *Journal of Economic Perspectives*, 31(2):3–32.

Ben-Michael, E., Feller, A., and Rothstein, J. (2018). The augmented synthetic control method. *arXiv preprint arXiv:1811.04170*.

Bollinger, B. and Sexton, S. (2018). Local excise taxes, sticky prices, and spillovers: evidence from Berkeley's soda tax. *Sticky Prices, and Spillovers: Evidence from Berkeley's Soda Tax (January 12, 2018)*.

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

Bruhn, C. A., Hetterich, S., Schuck-Paim, C., Kürüm, E., Taylor, R. J., Lustig, R., Shapiro, E. D., Warren, J. L., Simonsen, L., and Weinberger, D. M. (2017). Estimating the population-level impact of vaccines using synthetic controls. *Proceedings of the National Academy of Sciences*, 114(7):1524–1529.

Cameron, D. B., Mishra, A., and Brown, A. N. (2016). The growth of impact evaluation for international development: how much have we learned? *Journal of Development Effectiveness*, 8(1):1–21.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2010). The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480.

Carvalhoab, C., Masinic, R., and Medeiros, M. C. (2018). ArCo: An artificial counterfactual approach for high-dimensional panel time-series data. *Journal of Econometrics*, 207(2):352–280.

Chipman, H., George, E. I., McCulloch, R. E., Clyde, M., Foster, D. P., and Stine, R. A. (2001). The practical implementation of Bayesian model selection. *Lecture Notes-Monograph Series*, pages 65–134.

de Vocht, F. (2016). Inferring the 1985–2014 impact of mobile phone use on selected brain cancer subtypes using Bayesian structural time series and synthetic controls. *Environment international*, 97:100–107.

de Vocht, F., Tilling, K., Pliakas, T., Angus, C., Egan, M., Brennan, A., Campbell, R., and Hickman, M. (2017). The intervention effect of local alcohol licensing policies on hospital admission and crime: a natural experiment using a novel Bayesian synthetictime-series method. *J Epidemiol Community Health*, 71(9):912–918.

Doudchenko, N. and Imbens, G. W. (2016). Balancing, regression, dierence-in- dierences and synthetic control methods: A synthesis. *National Bureau of Economic Research*, Working Paper 22791.

Durbin, J. and Koopman, S. J. (2002). A simple and efficient simulation smoother for state space time series analysis. *Biometrika*, 89(3):603–616.

Ferman, B. and Pinto, C. (2016). Revisiting the synthetic control estimator.

Fredheim, R. (2017). The loyal editor effect: Russian online journalism after independence. *Post-Soviet Affairs*, 33(1):34–48.

Gelman, A., Jakulin, A., Pittau, M. G., Su, Y.-S., et al. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, 2(4):1360–1383.

George, E. I. and McCulloch, R. E. (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889.

George, E. I. and McCulloch, R. E. (1997). Approaches for Bayesian variable selection. *Statistica Sinica*, 7:339–373.

Geweke, J. (1991). *Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments*, volume 196. Federal Reserve Bank of Minneapolis, Research Department Minneapolis, MN.

Gordon, B. R., Zettelmeyer, F., Bhargava, N., and Chapsky, D. (2019). A comparison of approaches to advertising measurement: Evidence from big field experiments at Facebook. *Marketing Science*, 38(2):193–225.

Hahn, J. and Shi, R. (2017). Synthetic control and inference. *Econometrics*, 5(52).

Hoeting, J. A., Madigan, D., Raftery, A. E., and Volinsky, C. T. (1999). Bayesian model averaging: a tutorial. *Statistical Science*, pages 382–401.

Hoffman, M. D. and Gelman, A. (2014). The no-u-turn sampler: adaptively setting path lengths in Hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623.

Hsiang, T. (1975). A Bayesian view on ridge regression. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 24:267–268.

Hsiao, C., Steve Ching, H., and Ki Wan, S. (2012). A panel data approach for program evaluation: measuring the benefits of political and economic integration of Hong Kong with mainland China. *Journal of Applied Econometrics*, 27(5):705–740.

Imbens, G. W. and Rubin, D. B. (2015). *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45.

Kaul, A., Klößner, S., Pfeifer, G., and Schieler, M. (2015). Synthetic control methods: Never use all pre-intervention outcomes together with covariates. *MPRA Paper 83790, University Library of Munich, Germany.*

Kim, S., Lee, C., and Gupta, S. (2020). Bayesian synthetic control methods. *Journal of Marketing Research*, Forthcoming.

Kreif, N., Grieve, R., Hangartner, D., Turner, A. J., Nikolova, S., and Sutton, M. (2016). Examination of the synthetic control method for evaluating health policies with multiple treated units. *Health Economics*, 25(12):1514–1528.

Kyung, M., Gill, J., Ghosh, M., and Casella, G. (2010). Penalized regression, standard errors, and Bayesian lassos. *Bayesian Analysis*, 5(2):369–411.

Li, K. T. (2019). Statistical inference for average treatment effects estimated by synthetic control methods. *Journal of the American Statistical Association*, 115(532):2068–2083.

Li, Q. and Lin, N. (2010). The Bayesian elastic net. *Bayesian Analysis*, 5(1):151–170.

Madhavan, M., Ali Sharafuddin, M., Piboonrungroj, P., and Yang, C.-C. (2020). Short-term forecasting for airline industry: The case of Indian air passenger and air cargo. *Global Business Review*, page 0972150920923316.

Mallick, H. and Yi, N. (2013). Bayesian methods for high dimensional linear models. *Journal of Biometrics And Biostatistics*, 1(005).

Märkle-Huß, J., Feuerriegel, S., and Neumann, D. (2018). Contract durations in the electricity market: Causal impact of 15 min trading on the epex spot market. *Energy Economics*, 69:367–378.

Masini, R. and Medeiros, M. C. (2019). Counterfactual analysis with artificial controls: Inference, high dimensions and nonstationarity. *Working Paper, SSRN no. 3303308.*

Menchetti, F. and Bojinov, I. (2020). Estimating causal effects in the presence of partial interference using multivariate Bayesian structural time series models. *arXiv preprint arXiv:2006.12269.*

Park, T. and Casella, G. (2008). The Bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686.

Powell, D. (2018). Imperfect synthetic controls: did the Massachusetts health care reform save lives? *Available at SSRN 3192710.*

Poyser, O. (2019). Exploring the dynamics of bitcoin's price: a Bayesian structural time series approach. *Eurasian Economic Review*, 9(1):29–60.

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

Rojas, C. and Wang, E. Y. (2017). Do taxes for soda and sugary drinks work? Scanner data evidence from berkeley and washington. *Scanner Data Evidence from Berkeley and Washington (September 23, 2017).*

Samartsidis, P., Seaman, S. R., Presanis, A. M., Hickman, M., and Angelis, D. D. (2019). Assessing the causal effect of binary interventions from observational panel data with few treated units. *Statistical Science*, 34(3):486–503.

Saunders, J., Lundberg, R., Braga, A. A., Ridgeway, G., and Miles, J. (2015). A synthetic control approach to evaluating place-based crime interventions. *Journal of Quantitative Criminology*, 31(3):413–434.

Scott, S. L. and Varian, H. R. (2014). Predicting the present with Bayesian structural time series. *International Journal of Mathematical Modelling and Numerical Optimisation*, 5(1-2):4–23.

Tirunillai, S. and Tellis, G. J. (2017). Does offline tv advertising affect online chatter? quasi-experimental analysis using synthetic control. *Marketing Science*, 36(6):862–878.

Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., and Bürkner, P.-C. (2020). Rank-normalization, folding, and localization: An improved R-hat for assessing convergence of MCMC. *Bayesian Analysis*.

Wing, C., Simon, K., and Bello-Gomez1, R. A. (2018). Designing difference in difference studies: Best practices for public health policy research. *Annual Review of Public Health*, 39:453–469.

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

Appendix

A Per capita GDP of 17 OECD countries

Year	West Germany	Australia	Austria	Belgium	Denmark	France	Greece	Italy	Japan
1960	2284	2373	1796	1782	2329	1858	707	1620	1010
1961	2388	2346	1899	1883	2485	1959	1028	1763	1143
1962	2527	2539	1977	2018	2665	2099	1049	1900	1262
1963	2610	2717	2074	2122	2700	2205	1188	2021	1374
1964	2806	2873	2224	2288	2980	2367	1316	2093	1546
1965	3005	2973	2336	2414	3177	2526	1492	2205	1647
1966	3168	3230	2537	2559	3377	2728	1634	2400	1889
1967	3241	3404	2668	2719	3580	2915	1754	2624	2133
1968	3571	3788	2908	2958	3892	3163	1967	2910	2496
1969	3998	4162	3239	3308	4352	3531	2298	3227	2917
1970	4367	4490	3732	3797	4541	3857	2631	3602	3293
1971	4686	4677	4100	4129	4858	4203	2988	3834	3575
1972	5055	5013	4513	4512	5262	4535	3383	4101	3988
1973	5553	5595	4969	5039	5719	5008	3844	4594	4491
1974	6074	6024	5622	5707	6121	5594	3907	5240	4778
1975	6603	6683	6147	6146	6565	6077	4505	5601	5300
1976	7367	7140	6810	6854	7371	6673	5024	6288	5768
1977	8090	7482	7588	7326	7901	7292	5417	6821	6344
1978	8928	8272	8117	8056	8585	8031	6138	7542	7084
1979	10067	9174	9288	8921	9566	8948	6781	8594	8027
1980	11083	10203	10312	10156	10362	9891	7374	9643	8931
1981	12115	11513	11242	11079	11106	10929	7870	10593	9986
1982	12761	11537	12148	11827	12115	11869	8204	11275	10813
1983	13519	12300	13048	12334	12822	12518	8388	11812	11346
1984	14481	13120	13533	13113	13777	13145	8834	12543	12064
1985	15291	14019	14296	13735	14698	13746	9297	13285	12978
1986	15998	14537	14921	14292	15608	14308	9525	13896	13590
1987	16679	15554	15549	15013	16024	14940	9548	14688	14425
1988	17786	16524	16595	16209	16766	16040	10277	15784	15862
1989	18994	17255	17768	17345	17418	17193	11036	16875	17269
1990	20465	17322	19070	18526	18237	18244	11405	17946	18815
1991	21602	17652	20172	19453	19070	19021	11971	18845	20055
1992	22154	18496	20960	20123	19829	19746	12198	19367	20648
1993	21878	19432	21220	20308	20199	19920	12166	19778	21122
1994	22371	20583	22139	21340	21691	20695	12566	20577	21757
1995	23035	21773	22976	22248	22693	21545	12991	21532	22551
1996	23742	22562	24008	22687	23781	22288	13418	22235	23714
1997	24156	23649	24472	23416	24878	23370	14127	22810	24478
1998	24931	24853	25314	24164	25669	24412	14665	23840	24429
1999	25755	26283	26558	24792	27113	25111	15213	24402	24709
2000	26943	27403	28359	26631	28798	26690	16272	25759	26015
2001	27449	28492	28855	28001	29837	28043	17332	26586	26619
2002	28348	29819	29942	29330	30318	28829	19119	27320	27196
2003	28855	31273	30796	30082	30853	29210	20479	27537	28071

Year	Netherlands	New Zealand	Norway	Portugal	Spain	Switzerland	UK	US
1960	2113	2545	1713	728	1074	3387	2158	2879
1961	2111	2659	1823	783	1201	3625	2216	2929
1962	2275	2713	1905	848	1335	3788	2276	3103
1963	2360	2866	1991	905	1467	3946	2386	3227
1964	2566	3001	2112	979	1554	4158	2534	3420
1965	2739	3224	2268	1084	1678	4360	2660	3667
1966	2874	3481	2417	1170	1841	4575	2794	3974

1967	3078	3360	2620	1293	1954	4790	2923	4154
1968	3399	3489	2784	1475	2160	5138	3173	4494
1969	3951	3969	3035	1592	2447	5645	3391	4805
1970	4342	4136	3213	1916	2715	6382	3615	4999
1971	4705	4438	3523	2153	2951	6938	3854	5362
1972	4984	4780	3814	2420	3293	7406	4149	5838
1973	5475	5335	4138	2845	3710	7998	4683	6464
1974	6165	6041	4728	3048	4227	8800	5036	6951
1975	6699	6382	5458	3045	4600	8934	5481	7519
1976	7345	6748	6116	3407	4965	9368	5953	8300
1977	7959	6867	6742	3784	5364	10240	6489	9146
1978	8676	7301	7412	4120	5761	11003	7176	10229
1979	9504	7915	8354	4663	6188	12189	7973	11306
1980	10458	8656	9557	5263	6856	13853	8502	12186
1981	11304	9736	10548	5812	7447	15338	9161	13533
1982	11784	10687	11205	6263	7957	15963	9917	13940
1983	12419	11262	12022	6479	8378	16611	10669	15008
1984	13234	12141	13220	6570	8812	17710	11336	16549
1985	13938	12556	14354	6959	9259	18812	12068	17600
1986	14613	13085	15184	7414	9744	19458	12795	18439
1987	15197	13402	15823	8126	10542	20120	13717	19407
1988	16082	13569	16299	9057	11434	21334	14864	20711
1989	17387	14124	17043	10042	12417	22965	15716	22047
1990	18665	14420	18004	10894	13365	24518	16397	23064
1991	19626	14252	19212	11783	14152	24840	16681	23507
1992	20224	14679	20185	12219	14564	25141	17069	24509
1993	20679	15605	21088	12236	14699	25427	17845	25409
1994	21588	16686	22534	12614	15325	26031	18975	26670
1995	22585	17402	23874	13413	16032	26485	19860	27574
1996	23531	17879	26263	13974	16720	26394	20923	28814
1997	24692	18510	27776	14804	17420	27850	22280	30262
1998	25811	18669	27294	15401	18479	28835	23206	31519
1999	26654	19937	30011	16363	19817	28887	23959	33028
2000	28467	20789	36273	17353	21074	30461	25583	34603
2001	30359	21825	37078	18071	22257	30806	27026	35341
2002	31284	22662	36617	18799	23756	32751	28969	36180
2003	31792	23728	37245	17603	24812	33516	29609	37548

Table 10: Yearly per capita GDPs (PPP-adjusted) in 2002 U.S. dollars from 1960 to 2003, for the 17 OECD countries.

B Conditional posterior distributions

In this section, derivations of the conditional posterior distributions of the spike-and-slab prior are provided.

B.1 Conditional posterior distributions for the spike-and-slab prior

For the spike-and-slab prior (Equation (10)—Equation (13)), the joint posterior distribution is

$$\begin{aligned} & p(\boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\tau}^2, \sigma_\varepsilon) \\ & = p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\tau}^2, \sigma_\varepsilon) p(\boldsymbol{\beta} \mid \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\tau}^2, \sigma_\varepsilon) p(\boldsymbol{\tau}^2 \mid \boldsymbol{\gamma}, \boldsymbol{\theta}, \sigma_\varepsilon) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}, \sigma_\varepsilon) p(\boldsymbol{\theta} \mid \sigma_\varepsilon) p(\sigma_\varepsilon) \end{aligned} \tag{24}$$

where $\boldsymbol{Y}_0 = (Y_{01}, \ldots, Y_{0T})'$, supposing we have observations up to time T. Due to conditional independence, the joint posterior distribution can be further simplified as

$$\begin{aligned} &p(\boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\tau}^2, \sigma_\varepsilon) \\ &= p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \sigma_\varepsilon) p(\boldsymbol{\beta} \mid \boldsymbol{\gamma}, \boldsymbol{\tau}^2, \sigma_\varepsilon) p(\boldsymbol{\tau}^2) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) p(\sigma_\varepsilon). \end{aligned} \tag{25}$$

To sample the parameters, we need the conditional posterior distributions for each parameter, which are derived in the subsections below.

B.1.1 Conditional posterior distribution of $\boldsymbol{\theta}$

The conditional posterior distribution of θ is

$$p(\boldsymbol{\theta} \mid \boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\tau}^2, \sigma_\varepsilon) = p(\boldsymbol{\theta} \mid \boldsymbol{\gamma}). \tag{26}$$

Given $\boldsymbol{\gamma}$, the conditional posterior distribution of each θ_i is independent.

$$\begin{aligned} &p(\theta_i \mid \gamma_i) \\ &= \frac{p(\gamma_i \mid \theta_i) p(\theta_i)}{p(\gamma_i)} \\ &= \frac{p(\gamma_i \mid \theta_i) p(\theta_i)}{\int p(\gamma_i \mid \theta_i) p(\theta_i) d\theta_i} \\ &\propto p(\gamma_i \mid \theta_i) p(\theta_i) \\ &\propto \gamma_i^{\theta_i} (1 - \gamma_i)^{1-\theta_i}, \end{aligned} \tag{27}$$

And this is the kernel of $\text{Beta}(\theta_i + 1, 1 + 1 - \theta_i)$. Because of independence, the joint conditional posterior distribution is

$$p(\boldsymbol{\theta} \mid \boldsymbol{\gamma}) = \prod_{i=1}^{n} \text{Beta}(\theta_i + 1, 1 + 1 - \theta_i). \tag{28}$$

B.1.2 Conditional posterior distribution of $\boldsymbol{\tau}^2$

First we investigate the conditional distribution $p(\boldsymbol{\beta} \mid \boldsymbol{\tau}^2, \boldsymbol{\gamma})$. Define $\boldsymbol{H} = (\boldsymbol{\tau}^2, \boldsymbol{1}\phi^2)'$, and $\boldsymbol{\Gamma} = (\boldsymbol{\gamma}, (\boldsymbol{1} - \boldsymbol{\gamma}))$, where $\boldsymbol{1}$ is a $N \times 1$ vector of ones. Also define $\boldsymbol{I}$ as the $N \times N$ identity matrix, and $\boldsymbol{\tau}^2_{-i}$ as the $(N-1) \times 1$ vector of $\boldsymbol{\tau}^2$ without τ_i^2. Since given $\boldsymbol{\gamma}$, all elements of $\boldsymbol{\beta}$ are independent from each other, we have

$$p(\boldsymbol{\beta} \mid \boldsymbol{\tau}^2, \boldsymbol{\gamma}) = \text{Normal}(0, \boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I}). \tag{29}$$

Therefore,

$$\begin{aligned} &p(\tau_i^2 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\tau}^2_{-i}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \sigma_\varepsilon) \\ &= \frac{p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \sigma_\varepsilon) p(\boldsymbol{\beta} \mid \boldsymbol{\tau}^2, \boldsymbol{\gamma}) p(\boldsymbol{\tau}^2) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) p(\sigma_\varepsilon)}{\int p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \sigma_\varepsilon) p(\boldsymbol{\beta} \mid \boldsymbol{\tau}^2, \boldsymbol{\gamma}) p(\boldsymbol{\tau}^2) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) p(\sigma_\varepsilon) d\boldsymbol{\tau}^2} \\ &\propto p(\boldsymbol{\beta} \mid \boldsymbol{\tau}^2, \boldsymbol{\gamma}) p(\tau_i^2) \\ &\propto ((2\pi)^N |\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I}|)^{-1/2} exp\Big(- \frac{1}{2} \boldsymbol{\beta}' (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1} \boldsymbol{\beta} \Big) \frac{1}{\Gamma(1/2)} (\frac{1}{2})^{1/2} (\tau_i^2)^{-\frac{1+2}{2}} exp\Big(- \frac{1}{2\tau_i^2} \Big) \\ &\propto |\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I}|^{-1/2} exp\Big(- \frac{1}{2} \boldsymbol{\beta}' (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1} \boldsymbol{\beta} \Big) (\frac{1}{2})^{1/2} (\tau_i^2)^{-\frac{1+2}{2}} exp\Big(- \frac{1}{2\tau_i^2} \Big) \\ &\propto \prod_{j=1}^{N} (\tau_j^2)^{-\frac{\gamma_j}{2}} (\tau_i^2)^{-\frac{1+2}{2}} exp\Big(- \frac{1}{2} \sum_{j=1}^{N} \frac{\gamma_j \beta_j^2}{\tau_j^2} \Big) exp\Big(- \frac{1}{2} \frac{1}{\tau_i^2} \Big) \\ &\propto (\tau_i^2)^{-(\frac{\gamma_i + 2 + 1}{2})} exp\Big(- \frac{1}{2} \frac{1}{\tau_i^2} (\gamma_i \beta_i^2 + 1) \Big) \end{aligned} \tag{30}$$

where in the first step $p(\boldsymbol{\tau}^2) = p(\tau_1^2)p(\tau_2^2)\dots p(\tau_N^2)$ is used, and where $\Gamma(\cdot)$ is the gamma function. We notice that the last step of expression (30) is the kernel of

$$\text{Inverted Gamma}\Big(\frac{\gamma_i+1}{2}, \frac{\gamma_i\beta_i^2+1}{2}\Big). \tag{31}$$

Thus, we can sample each τ_i^2 independently from all τ_j^2, $j \neq i$.

B.1.3 Conditional posterior distribution of $\boldsymbol{\beta}$

Define $\boldsymbol{Y} = [\boldsymbol{Y}_1 \quad \boldsymbol{Y}_2 \quad ... \quad \boldsymbol{Y}_N]$ as the matrix of observed outcomes of the control units until time T.

$$\begin{aligned}
&p(\boldsymbol{\beta} \mid \boldsymbol{Y}_0, \boldsymbol{\gamma}, \boldsymbol{\theta}, \sigma_\varepsilon, \tau^2)\\
&= p(\boldsymbol{\beta} \mid \boldsymbol{Y}_0, \boldsymbol{\gamma}, \sigma_\varepsilon, \tau^2)\\
&= \frac{p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \sigma_\varepsilon)p(\boldsymbol{\beta} \mid \boldsymbol{\gamma}, \tau^2)p(\tau^2 \mid \boldsymbol{\gamma})p(\boldsymbol{\gamma})p(\tau^2)}{\int p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \sigma_\varepsilon)p(\boldsymbol{\beta} \mid \boldsymbol{\gamma}, \tau^2)p(\tau^2 \mid \boldsymbol{\gamma})p(\boldsymbol{\gamma})p(\tau^2)d\boldsymbol{\beta}}\\
&\propto p(\boldsymbol{Y}_0 \mid \boldsymbol{\beta}, \sigma_\varepsilon)p(\boldsymbol{\beta} \mid \boldsymbol{\gamma}, \tau^2)\\
&\propto exp\Big(-\frac{1}{2\sigma_\varepsilon^2}(\boldsymbol{Y}_0 - \boldsymbol{Y}\boldsymbol{\beta})'(\boldsymbol{Y}_0 - \boldsymbol{Y}\boldsymbol{\beta})\Big)exp\Big(-\frac{1}{2}\boldsymbol{\beta}'(\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1}\boldsymbol{\beta}\Big)\\
&\propto exp\Big(-\frac{1}{2\sigma_\varepsilon^2}(\boldsymbol{Y}_0'\boldsymbol{Y}_0 - 2\boldsymbol{\beta}'\boldsymbol{Y}'\boldsymbol{Y}_0 + \boldsymbol{\beta}'\boldsymbol{Y}'\boldsymbol{Y}\boldsymbol{\beta}) - \frac{1}{2}\boldsymbol{\beta}'(\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1}\boldsymbol{\beta}\Big)\\
&\propto exp\Big(-\frac{1}{2\sigma_\varepsilon^2}(-2\boldsymbol{\beta}'\boldsymbol{Y}'\boldsymbol{Y}_0 + \boldsymbol{\beta}'\boldsymbol{Y}'\boldsymbol{Y}\boldsymbol{\beta} + \sigma_\varepsilon^2\boldsymbol{\beta}'(\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1}\boldsymbol{\beta})\Big)\\
&\propto exp\Big(-\frac{1}{2}(\boldsymbol{\beta}'(\boldsymbol{Y}'\boldsymbol{Y}\frac{1}{\sigma_\varepsilon^2} + (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1})\boldsymbol{\beta}) - 2\boldsymbol{\beta}'\boldsymbol{Y}'\boldsymbol{Y}_0\frac{1}{\sigma_\varepsilon^2}\Big)\\
&\propto exp\Big(-\frac{1}{2}(\boldsymbol{\beta} - (\boldsymbol{Y}'\boldsymbol{Y}\frac{1}{\sigma_\varepsilon^2} + (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1}\boldsymbol{Y}'\boldsymbol{Y}_0\frac{1}{\sigma_\varepsilon^2})'(\boldsymbol{Y}'\boldsymbol{Y}\frac{1}{\sigma_\varepsilon^2} + (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1})\\
&\quad \times (\boldsymbol{\beta} - (\boldsymbol{Y}'\boldsymbol{Y}\frac{1}{\sigma_\varepsilon^2} + (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1}\boldsymbol{Y}'\boldsymbol{Y}_0\frac{1}{\sigma_\varepsilon^2})\Big).
\end{aligned} \tag{32}$$

This is the kernel of Normal$\Big((\boldsymbol{Y}'\boldsymbol{Y}\frac{1}{\sigma_\varepsilon^2} + (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1})\boldsymbol{Y}'\boldsymbol{Y}_0\frac{1}{\sigma_\varepsilon^2}, (\boldsymbol{Y}'\boldsymbol{Y}\frac{1}{\sigma_\varepsilon^2} + (\boldsymbol{H}\boldsymbol{\Gamma}'\boldsymbol{I})^{-1})^{-1}\Big)$.

B.1.4 Conditional posterior distribution of $\boldsymbol{\gamma}$

Note that when $\boldsymbol{\beta}$ is given, $\boldsymbol{\gamma}$ should be independent with $\boldsymbol{Y}_0$. Also, given $\boldsymbol{\theta}$, each of γ_i's does not depend on the other γ_i's. Define γ_{-i} as the vector $\boldsymbol{\gamma}$ excluding γ_i, $\boldsymbol{\beta}_{-i}$ as the vector of $\boldsymbol{\beta}$ excluding β_i, matrix $\boldsymbol{\Gamma}_{-i}$ as $\boldsymbol{\Gamma}$ excluding the i-th row, $\boldsymbol{H}_{-i}$ as $\boldsymbol{H}$ excluding the i-th row, and $\boldsymbol{I}_{-i}$ as the $(N-1)-$dimensional identity matrix. Thus,

$$\begin{aligned}
&p(\gamma_i \mid Y_0, \boldsymbol{\beta}, \sigma_\varepsilon, \tau^2, \boldsymbol{\gamma}_{-i}, \boldsymbol{\theta})\\
&= p(\gamma_i \mid \boldsymbol{\beta}, \boldsymbol{\gamma}_{-i}, \tau^2, \boldsymbol{\theta})\\
&= \frac{p(\boldsymbol{\beta} \mid \boldsymbol{\theta}, \tau^2, \gamma_i, \boldsymbol{\gamma}_{-i})p(\gamma_i, \boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta})}{\underbrace{p(\boldsymbol{\beta} \mid \boldsymbol{\theta}, \tau^2, \gamma_i = 1, \boldsymbol{\gamma}_{-i})p(\gamma_i = 1, \boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta})}_{Z_1} + \underbrace{p(\boldsymbol{\beta} \mid \boldsymbol{\theta}, \tau^2, \gamma_i = 0, \boldsymbol{\gamma}_{-i})p(\gamma_i = 0, \boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta})}_{Z_0}}\\
&= \frac{1}{Z_1 + Z_0}p(\boldsymbol{\beta} \mid \boldsymbol{\theta}, \tau^2, \gamma_i, \boldsymbol{\gamma}_{-i})p(\gamma_i, \boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta}).
\end{aligned} \tag{33}$$

And

$$\begin{aligned}
&Z_1\\
&= p(\boldsymbol{\beta} \mid \boldsymbol{\theta}, \boldsymbol{\tau}^2, \gamma_i, \boldsymbol{\gamma}_{-i})p(\gamma_i, \boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta})\\
&= (2\pi|\boldsymbol{H}_{-i}\boldsymbol{\Gamma}'_{-i}\boldsymbol{I}_{-i}|)^{-\frac{1}{2}}exp\Big(-\frac{1}{2}\boldsymbol{\beta}'_{-i}(\boldsymbol{H}_{-i}\boldsymbol{\Gamma}'_{-i}\boldsymbol{I}_{-i})^{-1}\boldsymbol{\beta}_{-i}\Big)(2\pi\tau_i^2)^{-\frac{1}{2}}exp\Big(-\frac{1}{2\tau_i^2}\beta_i^2\Big)\\
&\quad \times \theta_i \prod_{j=1, j\neq i}^{N} \theta_j^{\gamma_j}(1-\theta_j)^{1-\gamma_j}.
\end{aligned} \tag{34}$$

Similarly,

$$
\begin{aligned}
&Z_0 \\
&= p(\boldsymbol{\beta} \mid \boldsymbol{\theta}, \boldsymbol{\tau}^2, \gamma_i, \boldsymbol{\gamma}_{-i}) p(\gamma_i, \boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta}) \\
&= (2\pi |\boldsymbol{H}_{-i} \boldsymbol{\Gamma}'_{-i} \boldsymbol{I}_{-i}|)^{-\frac{1}{2}} exp\Big(-\frac{1}{2} \boldsymbol{\beta}'_{-i} (\boldsymbol{H}_{-i} \boldsymbol{\Gamma}'_{-i} \boldsymbol{I}_{-i})^{-1} \boldsymbol{\beta}_{-i} \Big) (2\pi\phi^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\phi^2} \beta_i^2 \Big) \\
&\quad \times (1-\theta_i) \prod_{j=1, j\neq i}^{N} \theta_j^{\gamma_j} (1-\theta_j)^{1-\gamma_j}.
\end{aligned} \tag{35}
$$

Therefore,

$$
\begin{aligned}
&p(\gamma_i = 1 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\tau}^2, \boldsymbol{\gamma}_{-i}, \boldsymbol{\theta}) \\
&= \frac{(\tau_i^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) \theta_i}{(\tau_i^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) \theta_i + (\phi^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\phi^2} \beta_i^2 \Big)(1-\theta_i)},
\end{aligned} \tag{36}
$$

and

$$
\begin{aligned}
&p(\gamma_i = 0 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\tau}^2, \boldsymbol{\gamma}_{-i}, \boldsymbol{\theta}) \\
&= \frac{(\phi^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\sigma_\varepsilon^2} \beta_i^2 \Big)(1-\theta_i)}{(\tau_i^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) \theta_i + (\phi^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\phi^2} \beta_i^2 \Big)(1-\theta_i)}.
\end{aligned} \tag{37}
$$

Therefore,

$$
\gamma_i \mid \boldsymbol{Y}_0, \boldsymbol{\beta}, \boldsymbol{\tau}^2, \boldsymbol{\gamma}_{-i} \sim \text{Bern}\left(\frac{(\tau_i^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) \theta_i}{(\tau_i^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) \theta_i + (\phi^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\phi^2} \beta_i^2 \Big)(1-\theta_i)} \right). \tag{38}
$$

However, if we are using the Gibbs sampler, there is a potential problem for this conditional posterior distribution. We know that for the Gibbs sampler, Equation (38) implies that the current γ_i is sampled using previously sampled β_i. If the previous β_i is zero, the current γ_i will also be sampled to be 0, which makes the next sampled β_i also most probably be 0. Therefore, the issue is that β_i is likely to be stuck to being 0 with a limited number of sampling steps. In other words, the sampler converges to the joint distribution extremely slowly. To resolve this problem, we can try to sample γ_i without using β_i.

$$
\begin{aligned}
&p(\gamma_i = 1 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2) \\
&= \frac{p(\boldsymbol{Y}_0 \mid \gamma_i = 1, \boldsymbol{\gamma}_{-i}, \boldsymbol{\beta}_{-i}, \sigma_\varepsilon, \boldsymbol{\theta}, \boldsymbol{\tau}^2) p(\beta_{-i} \mid \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\tau}^2) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) p(\boldsymbol{\tau}^2) p(\sigma_\varepsilon^2)}{p(\boldsymbol{Y}_0 \mid, \boldsymbol{\gamma}_{-i}, \boldsymbol{\beta}_{-j}, \sigma_\varepsilon, \boldsymbol{\theta}, \boldsymbol{\tau}) p(\boldsymbol{\beta}_{-i} \mid \boldsymbol{\gamma}_{-i}, \boldsymbol{\theta}, \boldsymbol{\tau}^2) p(\boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta},) p(\boldsymbol{\theta}) p(\boldsymbol{\tau}^2) p(\sigma_\varepsilon^2)} \\
&= \frac{p(\boldsymbol{Y}_0 \mid \gamma_i = 1, \boldsymbol{\gamma}_{-i}, \boldsymbol{\beta}_{-i}, \sigma_\varepsilon, \boldsymbol{\theta}, \boldsymbol{\tau}^2) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta})}{\underbrace{p(\boldsymbol{Y}_0 \mid, \gamma_{-i}, \beta_{-j}, \sigma_\varepsilon, \boldsymbol{\theta})) p(\boldsymbol{\gamma}_{-i} \mid \boldsymbol{\theta})}_{K_1}} \\
&= \frac{1}{K_1} \int p(\boldsymbol{Y}_0, \beta_i \mid \gamma_i = 1, \boldsymbol{\gamma}_{-i}, \boldsymbol{\beta}_{-i}, \sigma_\varepsilon, \boldsymbol{\theta}, \boldsymbol{\tau}^2) p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}) d\beta_i \\
&= \frac{1}{K_1} p(\boldsymbol{\gamma} \mid \boldsymbol{\theta}) \int p(\boldsymbol{Y}_0 \mid \gamma_i = 1, \boldsymbol{\gamma}_{-i}, \boldsymbol{\beta}_{-i}, \sigma_\varepsilon^2, \boldsymbol{\theta}) p(\beta_i \mid \gamma_i = 1, \tau_i^2) \\
&= \frac{1}{K_1} \prod_{j=1}^{N} \gamma_j^{\theta_j} (1-\gamma_j)^{1-\theta_j} \\
&\quad \times \int (2\pi\sigma_\varepsilon^2)^{-\frac{T}{2}} exp\Big(-\frac{1}{2\sigma_\varepsilon^2} (\boldsymbol{Y}_0 - \boldsymbol{Y\beta})'(\boldsymbol{Y}_0 - \boldsymbol{Y\beta}) \Big) (2\pi\tau_i^2)^{-\frac{1}{2}} exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) d\beta_i \\
&= \frac{1}{K_1} \underbrace{\prod_{j=1, j\neq i}^{N} \Big(\theta_j^{\gamma_j} (1-\theta_j)^{1-\gamma_j} \Big) (2\pi\sigma_\varepsilon^2)^{-\frac{T}{2}}}_{K_2} (2\pi\tau_i^2)^{-\frac{1}{2}} \theta_i \int exp\Big(-\frac{1}{2\sigma_\varepsilon^2} (\boldsymbol{Y}_0 - \boldsymbol{Y\beta})'(\boldsymbol{Y}_0 - \boldsymbol{Y\beta}) \Big) \\
&\quad \times exp\Big(-\frac{1}{2\tau_i^2} \beta_i^2 \Big) d\beta_i \\
&= \frac{K_2}{K_1} (2\pi\tau_i^2)^{-\frac{1}{2}} \theta_i \int exp\Big(-\frac{1}{2\sigma_\varepsilon^2} (\boldsymbol{Y}_0 - \boldsymbol{Y\beta})'(\boldsymbol{Y}_0 - \boldsymbol{Y\beta}) - \frac{1}{2\tau_i^2} \beta_i^2 \Big) d\beta_i.
\end{aligned} \tag{39}
$$

Letting $\boldsymbol{Y}_{-i}$ be the matrix of $\boldsymbol{Y}$ excluding the i-th column, we have $\boldsymbol{Y}_0 - \boldsymbol{Y\beta} = \boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i} - \boldsymbol{Y}_i\boldsymbol{\beta}_i$. Setting

$\boldsymbol{d} = \boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i}$, we have that $(\boldsymbol{Y}_0 - \boldsymbol{Y}\boldsymbol{\beta})'(\boldsymbol{Y}_0 - \boldsymbol{Y}\boldsymbol{\beta}) = \sum_{t=i}^{T}(d_j - \beta_i Y_{it})^2$. Thus,

$$\begin{aligned}
&p(\gamma_i = 1 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2) \\
&= \frac{K_2}{K_1}(2\pi\tau_i^2)^{-\frac{1}{2}}\theta_i \int exp\Big(- \frac{1}{2\sigma_\varepsilon^2}\sum_{t=i}^{T}(d_j - \beta_i Y_{it})^2 - \frac{1}{2\tau_i^2}\beta_i^2\Big)d\beta_i \\
&= \frac{K_2}{K_1}(2\pi\tau_i^2)^{-\frac{1}{2}}\theta_i \int exp\Big(- \frac{1}{2\sigma_\varepsilon^2}\Big[\sum_{t=i}^{T}d_i^2 - 2\beta_i\sum_{t=1}^{T}d_iY_{it} + \beta_i^2\sum_{t=1}^{T}Y_{it}^2\Big] - \frac{1}{2\tau_i^2}\beta_i^2\Big)d\beta_i \\
&= \frac{K_2}{K_1}(2\pi\tau_i^2)^{-\frac{1}{2}}\theta_i exp\Big(- \frac{1}{2\sigma_\varepsilon^2}\boldsymbol{d}'\boldsymbol{d}\Big)\int exp\Big(- \frac{1}{2\sigma_\varepsilon^2}\Big[- 2\beta_i\sum_{t=1}^{T}d_iY_{it} + \beta_i^2\sum_{t=1}^{T}Y_{it}^2\Big] - \frac{1}{2\tau_i^2}\beta_i^2\Big)d\beta_i \\
&= \frac{K_2}{K_1}(2\pi\tau_i^2)^{-\frac{1}{2}}\theta_i exp\Big(- \frac{1}{2\sigma_\varepsilon^2}(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})'(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})\Big) \\
&\quad\times \underbrace{\int exp\Big(- \frac{1}{2\sigma_\varepsilon^2}\Big[- 2\beta_i\sum_{t=1}^{T}d_iY_{it} + \beta_i^2\sum_{t=1}^{T}Y_{it}^2\Big] - \frac{1}{2\sigma_\varepsilon^2\frac{\tau_i^2}{\sigma_\varepsilon^2}}\beta_i^2\Big)d\beta_i.}_{K3}
\end{aligned} \tag{40}$$

$$\begin{aligned}
&K3 \\
&= \int exp\Big(- \frac{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}{2\sigma_\varepsilon^2}\Big[\Big(\beta_i - \frac{\sum_{t=1}^{T}d_iY_{it}}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^2 - \frac{(\sum_{t=1}^{T}d_iY_{it})^2}{(\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2})^2}\Big]\Big)d\beta_i \\
&= exp\Big(- \frac{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}{2\sigma_\varepsilon^2}\frac{(\sum_{t=1}^{T}d_iY_{it})^2}{(\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2})^2}\Big) \\
&\quad\times \int exp\Big(- \frac{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}{2\sigma_\varepsilon^2}\Big(\beta_i - \frac{\sum_{t=1}^{T}d_iY_{it}}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^2\Big)d\beta_i \\
&= exp\Big(- \frac{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}{2\sigma_\varepsilon^2}\frac{(\sum_{t=1}^{T}d_iY_{it})^2}{(\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2})^2}\Big)\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^{-\frac{1}{2}} \\
&\quad\times \underbrace{\int\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^{\frac{1}{2}} exp\Big(- \frac{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}{2\sigma_\varepsilon^2}\Big(\beta_i - \frac{\sum_{t=1}^{T}d_iY_{it}}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^2\Big)d\beta_i}_{=1} \\
&= exp\Big(- \frac{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}{2\sigma_\varepsilon^2}\frac{(\sum_{t=1}^{T}d_iY_{it})^2}{(\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2})^2}\Big)\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^{-\frac{1}{2}} \\
&= exp\Big(- \frac{(\sum_{t=1}^{T}d_iY_{it})^2}{(\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2})}\Big)\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^{-\frac{1}{2}}.
\end{aligned} \tag{41}$$

Therefore,

$$\begin{aligned}
&p(\gamma_i = 1 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2) \\
&= \frac{K_2}{K1}(2\pi\tau_i^2)^{-\frac{1}{2}}\theta_i exp\Big(- \frac{1}{2\sigma_\varepsilon^2}(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})'(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})\Big) \\
&\quad\times exp\Big(- \frac{(\sum_{t=1}^{T}d_iY_{it})^2}{(\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2})}\Big)\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^{T}Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\tau_i^2}}\Big)^{-\frac{1}{2}}.
\end{aligned} \tag{42}$$

To derive $p(\gamma_i = 0 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \tau_i^2)$, we use the similar procedure.

$$\begin{aligned}
&p(\gamma_i = 0 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2)\\
&= \frac{K_2}{K_1}(2\pi\phi^2)^{-\frac{1}{2}}(1-\theta_i)\int exp\Big(-\frac{1}{2\sigma_\varepsilon^2}(Y_0 - \boldsymbol{Y}\boldsymbol{\beta})'(\boldsymbol{Y}_0 - \boldsymbol{Y}\boldsymbol{\beta}) - \frac{1}{2\sigma_\varepsilon^2\frac{\phi^2}{\sigma_\varepsilon^2}}\beta_i^2\Big)d\beta_i\\
&= \frac{K_2}{K_1}(2\pi\phi^2)^{-\frac{1}{2}}(1-\theta_i)exp\Big(-\frac{1}{2\sigma_\varepsilon^2}(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})'(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})\Big)\\
&\quad\times \underbrace{\int exp\Big(-\frac{1}{2\sigma_\varepsilon^2}\Big[-2\beta_i\sum_{t=1}^T d_iY_{it} + \beta_i^2\sum_{t=1}^T Y_{it}^2\Big] - \frac{1}{2\sigma_\varepsilon^2\frac{\phi^2}{\sigma_\varepsilon^2}}\beta_i^2\Big)d\beta_i}_{K_4}.
\end{aligned} \tag{43}$$

And,

$$\begin{aligned}
&K_4\\
&= \int exp\Big(-\frac{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}{2\sigma_\varepsilon^2}\Big[\Big(\beta_i - \frac{\sum_{t=1}^T d_iY_{it}}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^2 - \frac{(\sum_{t=1}^T d_iY_{it})^2}{(\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2})^2}\Big]\Big)d\beta_i\\
&= exp\Big(-\frac{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}{2\sigma_\varepsilon^2}\frac{(\sum_{t=1}^T d_iY_{it})^2}{(\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2})^2}\Big)\int exp\Big(-\frac{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}{2\sigma_\varepsilon^2}\Big(\beta_i - \frac{\sum_{t=1}^T d_iY_{it}}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^2\Big)d\beta_i\\
&= exp\Big(-\frac{(\sum_{t=1}^T d_iY_{it})^2}{2\sigma_\varepsilon^2(\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2})}\Big)\int exp\Big(-\frac{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}{2\sigma_\varepsilon^2}\Big(\beta_i - \frac{\sum_{t=1}^T d_iY_{it}}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^2\Big)d\beta_i\\
&= exp\Big(-\frac{(\sum_{t=1}^T d_iY_{it})^2}{2\sigma_\varepsilon^2(\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2})}\Big)\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^{\frac{1}{2}}\\
&\quad\times \underbrace{\int\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^{-\frac{1}{2}} exp\Big(-\frac{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}{2\sigma_\varepsilon^2}\Big(\beta_i - \frac{\sum_{t=1}^T d_iY_{it}}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^2\Big)d\beta_i}_{=1}.
\end{aligned} \tag{44}$$

Therefore,

$$\begin{aligned}
&p(\gamma_i = 0 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2)\\
&= \frac{K_2}{K_1}(2\pi\phi^2)^{-\frac{1}{2}}(1-\theta_i)exp\Big(-\frac{1}{2\sigma_\varepsilon^2}(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})'(\boldsymbol{Y}_0 - \boldsymbol{Y}_{-i}\boldsymbol{\beta}_{-i})\Big)\\
&\quad\times exp\Big(-\frac{(\sum_{t=1}^T d_iY_{it})^2}{2\sigma_\varepsilon^2(\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2})}\Big)\Big(2\pi\frac{\sigma_\varepsilon^2}{\sum_{t=1}^T Y_{it}^2 + \frac{\sigma_\varepsilon^2}{\phi^2}}\Big)^{\frac{1}{2}}.
\end{aligned} \tag{45}$$

Thus, we can sample γ_i with the Bernoulli distribution with the parameter

$$\frac{p(\gamma_i = 1 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2)}{p(\gamma_i = 1 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2) + p(\gamma_i = 0 \mid \boldsymbol{Y}_0, \boldsymbol{\beta}_{-i}, \boldsymbol{\theta}, \sigma_\varepsilon, \boldsymbol{\gamma}_{-i}, \boldsymbol{\tau}^2)}, \tag{46}$$

in which K_1 and K_2 are cancelled out, and the remaining part does not depend on β_i.

C Programming code

C.1 R code for the simulation study

```
library(rstan)
library(assertthat)
library(ggplot2)
install.packages("/Users/rzhi/Documents/CausalBsts",repos=NULL, type="source")
library(CausalBsts)
## simulate data
simulation <- function(T,N,sparsity,SNR,alpha){
  #define the variable of noises according to the SNR
```

```
  sigma2 = 1/SNR

  #simulate the control outcomes
  Y = matrix( rnorm(T*N,mean=0,sd=1), T, N)

  #define the parameters beta
  true.beta = matrix(0,N,1)
  if (sparsity == 0)
  {
    true.beta[1:N/3,1] = 0.9/(N/3)
    true.beta[(N/3+1):N,1] = 0.1/(2*N/3)
  }else
  {
    true.beta[1:N/15,1] = 0.9/(N/15)
    true.beta[(N/15+1):N/5,1] = 0.1/(2*N/15)
    true.beta[(N/5+1):N,1] = 0
  }
  #simulate the noises
  epsilon = rnorm(T, mean = 0, sd = sqrt(sigma2))



  #define the treated unit Y0
  Y0 = Y%*%true.beta+epsilon
  Y0[(2*T/3+1):T] = Y0[(2*T/3+1):T] * (1 + alpha)


  sim = list("Y"=Y,"Y0"=Y0,"true.beta"=true.beta,"epsilon"=epsilon,
             "N"=N,"T"=T,"sparsity"=sparsity,"SNR"=SNR, "alpha" = alpha)
  return(sim)
}


cumsum.na.rm <- function(x) {
  # Cumulative sum

  if (is.null(x)) {
    return(x)
  }
  nas <- is.na(x)
  s <- cumsum(ifelse(nas, 0, x))
  s[nas] <- NA
  return(s)
}

ComputeCumulativeEffects <- function(y.samples, point.pred, y,
                                     post.period.begin, alpha = 0.05) {
  # Computes summary statistics for the cumulative posterior predictions over
  # the unobserved data points in the post-intervention period.
  #
  # Args:
  #   y.samples:         Matrix of simulated response trajectories, as returned
  #                      by \code{ComputeResponseTrajectories()}.[number of post-burn-in MCMC samples] x [time points]
  #   point.pred:        Data frame of point predictions, as returned by
  #                      \code{ComputePointPredictions()}.
  #   y:                 Actual observed response, from the beginning of the
  #                      pre-period to the end of the observed period.
  #   post.period.begin: Index of the first data point of the post-period.
  #   alpha:             The resulting coverage of the posterior intervals will
  #                      be \code{1 - alpha}.
  #
  # Returns:
  #   data frame with 3 columns:
  #     cum.effect:        posterior prediction of the means of cumulative effects
  #     cum.effect.lower: lower limit of a \code{(1 - alpha)*100}% interval
  #     cum.effect.upper: upper limit

  # After pre-inference standardization of the response variable has been
  # undone, we can form cumulative time series of counterfactual predictions.
  # Note that we only explicitly compute these for the post-period. The
  # cumulative prediction for the pre-period and the gap between pre- and post-
  # period (if any) is forced to equal the (cumulative) observed response in the
  # pre-period. Thus, the posterior intervals of the cumulative predictions do
  # not inherit variance from the pre-period, which would be misleading when
  # subtracting cumulative predictions from the cumulative observed response to
  # obtain the cumulative impact, which is the main use case. Thus, the
  # cumulative impact will be zero by construction before the beginning of the
  # post-period.
```

```
  # Compute posterior mean
  is.post.period <- seq_along(y) >= post.period.begin
  cum.pred.mean.pre <- cumsum.na.rm(as.vector(y)[!is.post.period])
  non.na.indices <- which(!is.na(cum.pred.mean.pre))
  assert_that(length(non.na.indices) > 0)
  last.non.na.index <- max(non.na.indices)
  cum.pred.mean.post <- cumsum(point.pred$mean[is.post.period]) +
    cum.pred.mean.pre[last.non.na.index]
  cum.pred.mean <- c(cum.pred.mean.pre, cum.pred.mean.post)

  # Check for overflow
  assert_that(identical(which(is.na(cum.pred.mean)),
                        which(is.na(y[!is.post.period]))),
              msg = "unexpected NA found in cum.pred.mean")

  # Compute posterior interval
  cum.pred.lower.pre <- cum.pred.mean.pre
  cum.pred.upper.pre <- cum.pred.mean.pre
  y.samples.cum.post <- t(apply(y.samples[, is.post.period, drop = FALSE], 1,
                                cumsum)) +
    cum.pred.mean.pre[last.non.na.index]
  if (sum(is.post.period) == 1) {
    y.samples.cum.post <- t(y.samples.cum.post)
  }
  assert_that(is.scalar(alpha), alpha > 0, alpha < 1)
  prob.lower <- alpha / 2      # e.g., 0.025 when alpha = 0.05
  prob.upper <- 1 - alpha / 2  # e.g., 0.975 when alpha = 0.05
  cum.pred.lower.post <- as.numeric(t(apply(y.samples.cum.post, 2, quantile,
                                            prob.lower)))
  cum.pred.upper.post <- as.numeric(t(apply(y.samples.cum.post, 2, quantile,
                                            prob.upper)))
  cum.pred.lower <- c(cum.pred.lower.pre, cum.pred.lower.post)
  cum.pred.upper <- c(cum.pred.upper.pre, cum.pred.upper.post)


  cum.y.model <- cumsum.na.rm(y)
  cum.effect.upper = cum.y.model-cum.pred.lower
  cum.effect.lower = cum.y.model-cum.pred.upper
  cum.effect.mean =  cum.y.model-cum.pred.mean
  # Put cumulative prediction together
  cum.effect<- data.frame(cum.effect = cum.effect.mean,
                          cum.effect.lower, cum.effect.upper)
  return(cum.effect)
}



check_interval_pointwise = function(lower,upper){
  #this function checks for each post-intervention time point whether 0 is excluded in the credible interval
  #   of the impact effects, and returns the proportion that 0 is excluded.
  length = length(lower)
  exclude.count = 0
  for (i in 1:length){
    if(!(lower[i]<=0 && upper[i]>=0)){
      exclude.count = exclude.count+1
    }
  }
  return(exclude.count/length)
}
check_interval_overall= function(lower,upper){
  #this function checks whether 0 is excluded in the credible interval of the last post-intervention period
  # and 1 is returned if it is, 0 otherwise

  length = length(lower)
  exclude.zero = 0
  if(!(lower[length]<=0 && upper[length]>=0)){

    exclude.zero = 1

  }


  return(exclude.zero)
}




SimulationAnalysis = function(T, N.T.ratio, sparsity, SNR, alpha,
credible.interval, sampling.total, number.simulations=50){
```

```
result.CIM = matrix(0, number.simulations, 4)# initialize the matrix for storing the results
result.spike = matrix(0, number.simulations, 4)
result.horseshoe = matrix(0, number.simulations, 4)

T0 = 2*T/3
N = T*N.T.ratio
sampling.warmup = 0.5*sampling.total

for (i in 1:number.simulations){
  print(i)
  print(T)
  print(N.T.ratio)
  print(sparsity)
  sim.data = simulation(T,N, sparsity ,SNR, alpha)

  #CIM
  start.time <- Sys.time()
  impact =CausalBsts:: CausalImpact(cbind(sim.data$Y0,sim.data$Y),pre.period = c(1,T0),
  post.period = c(T0+1,T),model.args=list(niter=sample.total))
  end.time <- Sys.time()


  result.CIM[i,4] = difftime(end.time, start.time,units = "secs")

  interval.CIM = as.matrix(cbind(impact$series$cum.effect.lower,impact$series$cum.effect.upper))
  interval.CIM=interval.CIM[(T0+1):T,]
  exclusion.CIM.pointwise=check_interval_pointwise(interval.CIM[,1],interval.CIM[,2])
  exclusion.CIM.overall = check_interval_overall(interval.CIM[,1],interval.CIM[,2])
  result.CIM[i,1] = exclusion.CIM.overall
  result.CIM[i,2] = exclusion.CIM.pointwise

  mse.CIM = sum((impact$series[1:T0,'response'] -impact$series[1:T0,'point.pred'])^2)/T0
  result.CIM[i,3] = mse.CIM


  #BSCM-spike-and-slab
  data.bscm = list(N_train = T0, N_test = T-T0, p = N, y_train = sim.data$Y0[1:T0],
                   X_train = sim.data$Y[1:T0,], X_test = matrix(sim.data$Y[(T0+1):T,],(T-T0),N) )

  start.time <- Sys.time()
  fit.spike <- stan(file = 'spike-and-slab.stan', data = data.bscm,iter=sampling.total,warmup = sampling.total/2,
                    control = list(adapt_delta = 0.80,max_treedepth = 15),chains=4)
  end.time <- Sys.time()

  result.spike[i,4] = difftime(end.time, start.time,units = "secs")

  y.pred.spike= summary(fit.spike, pars = c("y_fit","y_test"), probs = credible.interval)$summary
  y.pred.spike = as.data.frame(y.pred.spike[,c("mean","2.5%","97.5%")])

  samples.spike= cbind(extract(fit.spike)$y_fit,extract(fit.spike)$y_test) #extract the sampled target variable
  samples.spike = samples.spike[(sampling.warmup+1):sampling.total,] #only use those sampled after warm-up
  interval.spike = ComputeCumulativeEffects(y.samples=samples.spike, point.pred= y.pred.spike, y=sim.data$Y0,
                                            post.period.begin=T0+1, alpha = 0.05)
  exclusion.spike.pointwise=check_interval_pointwise(interval.spike[,2],interval.spike[,3])
  exclusion.spike.overall=check_interval_overall(interval.spike[,2],interval.spike[,3])

  result.spike[i,1] = exclusion.spike.overall
  result.spike[i,2] = exclusion.spike.pointwise



  mse.spike = sum((sim.data$Y0[1:T0]-y.pred.spike$mean[1:T0])^2)/T0
  result.spike[i,3] = mse.spike



  #BSCM-horseshoe

  start.time <- Sys.time()
  fit.horseshoe <- stan(file = 'horseshoe.stan', data = data.bscm,iter=sampling.total,warmup = sampling.total/2,
                        control = list(adapt_delta = 0.80,max_treedepth = 15),chains=4)
  end.time <- Sys.time()

  result.horseshoe[i,4] = difftime(end.time, start.time,units = "secs")

  y.pred.horseshoe= summary(fit.horseshoe, pars = c("y_fit","y_test"), probs = credible.interval)$summary
  y.pred.horseshoe = as.data.frame(y.pred.horseshoe[,c("mean","2.5%","97.5%")])

  samples.horseshoe= cbind(extract(fit.horseshoe)$y_fit,
```

```
    extract(fit.horseshoe)$y_test) #extract the sampled target variable
    samples.horseshoe = samples.horseshoe[(sampling.warmup+1):sampling.total,] #only use those sampled after warm-up
    interval.horseshoe = ComputeCumulativeEffects(y.samples=samples.horseshoe,

    point.pred= y.pred.horseshoe, y=sim.data$Y0,
                                            post.period.begin=T0+1, alpha = 0.05)
    exclusion.horseshoe.pointwise=check_interval_pointwise(interval.horseshoe[,2],interval.horseshoe[,3])
    exclusion.horseshoe.overall=check_interval_overall(interval.horseshoe[,2],interval.horseshoe[,3])

    result.horseshoe[i,1] = exclusion.horseshoe.overall
    result.horseshoe[i,2] = exclusion.horseshoe.pointwise


    mse.horseshoe = sum((sim.data$Y0[1:T0]-y.pred.horseshoe$mean[1:T0])^2)/T0
    result.horseshoe[i,3] = mse.horseshoe
  }

  result.CIM = as.data.frame(result.CIM, colnames=names("OverallImpact","PointwiseEffectProportion",
  "MSE","RunningTime"))
  result.spike = as.data.frame(result.spike, colnames=names("OverallImpact","PointwiseEffectProportion",
  "MSE","RunningTime"))
  result.horseshoe = as.data.frame(result.horseshoe, colnames=names("OverallImpact","PointwiseEffectProportion",
  "MSE","RunningTime"))

  result = list("CIM"=result.CIM,"spike-and-slab"= result.spike, 'horseshoe'=result.horseshoe)
  return(result)
}
```

C.2 R code for the empirical study

```
library("dplyr")
library("xts")
library("zoo")
library("foreign")
library("rstan")
library("corpcor")
library("robustHD")
library("ggplot2")
library("assertthat")

install.packages("/Users/rzhi/Documents/CausalBsts",repos=NULL, type="source")
library(CausalBsts)


####################
CreatePointwiseImpactPlot = function(data, T, T0){
  # this function plots the cumulative impacts and the corresponding credible interval
  # data: dataframe of 3 columns: impact, lower, upper
  # T: total number of time points
  # T0: the last time period of pre-treatment periods

  # Initialize plot
  q <- ggplot(as.data.frame(data), aes(x = time)) + theme_bw(base_size = 15)
  q <- q + xlab("Year") + ylab("Pointwise_impact")

  # Add prediction intervals
  q <- q + geom_ribbon(aes(ymin = lower, ymax = upper),
                       as.data.frame(data), fill = "slategray2")

  # Add pre-period markers
  xintercept <- CreatePeriodMarkers(as.integer(c(1,T0)),
                                    as.integer(c(T0+1,T)),
                                    time(1:T))
  q <- q + geom_vline(xintercept = xintercept,
                      colour = "darkgrey", size = 0.8, linetype = "dashed")

  # Add zero line to pointwise and cumulative plot
  q <- q + geom_line(aes(y = 0),
                     colour = "darkgrey", size = 0.8, linetype = "solid",
                     na.rm = TRUE)

  # Add cumulative impacts
  q <- q + geom_line(aes(y = impact), data,
                     size = 0.6, colour = "darkblue", linetype = "dashed",
                     na.rm = TRUE)
  q<-q+scale_y_continuous(labels = function(x) format(x, scientific = FALSE))
```

```
  plot(q)
}
CreateCumulativeImpactPlot = function(data, T, T0){
  # this function plots the cumulative impacts and the corresponding credible interval
  # data: dataframe of 3 columns: impact, lower, upper
  # T: total number of time points
  # T0: the last time period of pre-treatment periods

  # Initialize plot
  q <- ggplot(as.data.frame(data), aes(x = time)) + theme_bw(base_size = 15)
  q <- q + xlab("Year") + ylab("Cumulative impact")

  # Add prediction intervals
  q <- q + geom_ribbon(aes(ymin = lower, ymax = upper),
                       as.data.frame(data), fill = "slategray2")

  # Add pre-period markers
  xintercept <- CreatePeriodMarkers(as.integer(c(1,T0)),
                                    as.integer(c(T0+1,T)),
                                    time(1:T))
  q <- q + geom_vline(xintercept = xintercept,
                      colour = "darkgrey", size = 0.8, linetype = "dashed")

  # Add zero line to pointwise and cumulative plot
  q <- q + geom_line(aes(y = 0),
                     colour = "darkgrey", size = 0.8, linetype = "solid",
                     na.rm = TRUE)

  # Add cumulative impacts
  q <- q + geom_line(aes(y = impact), as.data.frame(data),
                     size = 0.6, colour = "darkblue", linetype = "dashed",
                     na.rm = TRUE)
  q<-q+scale_y_continuous(labels = function(x) format(x, scientific = FALSE))

  plot(q)
}
CreatePredictionPlot = function(data, T, T0){
  # this function plots the results of posterior prediction and the corresponding credible interval,
  #as well as the true response
  # data: dataframe of 4 columns: lower, upper, prediction and response
  # T: total number of time points
  # T0: the last time period of pre-treatment periods

  # Initialize plot
  q <- ggplot(as.data.frame(data), aes(x = time)) + theme_bw(base_size = 15)
  q <- q + xlab("Year") + ylab("per capita GDP (PPP-adjusted)")


  # Add prediction intervals
  q <- q + geom_ribbon(aes(ymin = lower, ymax = upper),
                       as.data.frame(data), fill = "slategray2")

  # Add pre-period markers
  xintercept <- CreatePeriodMarkers(as.integer(c(1,T0)),
                                    as.integer(c(T0+1,T)),
                                    time(1:T))
  q <- q + geom_vline(xintercept = xintercept,
                      colour = "darkgrey", size = 0.8, linetype = "dashed")


  # Add point predictions
  q <- q + geom_line(aes(y = mean), as.data.frame(data),
                     size = 0.6, colour = "darkblue", linetype = "dashed",
                     na.rm = TRUE)

  # Add observed data
  data$response = unlist(data$response)
  q <- q + geom_line(aes(y = response), size = 0.6,  na.rm = TRUE)
  q<-q+scale_y_continuous(labels = function(x) format(x, scientific = FALSE))

  plot(q)
}
GetPeriodIndices <- function(period, times) {
  # Computes indices belonging to a period in data. This function is used for plotting.
  #
  # Args:
  #   period:  two-element vector specifying start and end of a period, having
  #            the same data type as `times. The range from `period[1]` to
  #            `period[2]` must have an intersect with `times`.
```

```
#   times:    vector of time points; can be of integer or of POSIXct type.
#
# Returns:
#   A two-element vector with the indices of the period start and end within
#   `times`.

  # Check input
  assert_that(length(period) == 2)
  assert_that(!anyNA(times))
  assert_that(identical(class(period), class(times)) ||
                (is.numeric(period) && is.numeric(times)))
  # Check if period boundaries are in the right order, and if `period` has an
  # overlap with `times`.
  assert_that(period[1] <= period[2])
  assert_that(period[1] <= tail(times, 1), period[2] >= times[1])

  # Look up values of start and end of period in `times`; also works if the
  # period start and end time are not exactly present in the time series.
  indices <- seq_along(times)
  is.period <- (period[1] <= times) & (times <= period[2])
  # Make sure the period does match any time points.
  assert_that(any(is.period),
              msg = "The period must cover at least one data point")
  period.indices <- range(indices[is.period])
  return(period.indices)
}
CreatePeriodMarkers <- function(pre.period, post.period, times) {
  # Creates a vector of period markers to display. This function is used for plotting.
  #
  # Args:
  #   pre.period:  vector of 2 time points that define the pre-period.
  #   post.period: vector of 2 time points that define the post-period.
  #   times:       vector of time points.
  #
  # Returns:
  #   Vector of period markers that should be displayed, generally depicting the
  #   first and last time points of pre- and post-period. The start of the pre-
  #   period is not shown if it coincides with the first time point of the time
  #   series; similarly, the last time point of the post-period is not shown if
  #   it coincides with the last time point of the series. If there is no gap
  #   between pre- and post-period, the start marker of the post-period is
  #   omitted.

  pre.period.indices <- GetPeriodIndices(pre.period, times)
  post.period.indices <- GetPeriodIndices(post.period, times)
  markers <- NULL
  if (pre.period.indices[1] > 1) {
    markers <- c(markers, times[pre.period.indices[1]])
  }
  markers <- c(markers, times[pre.period.indices[2]])
  if (pre.period.indices[2] < post.period.indices[1] - 1) {
    markers <- c(markers, times[post.period.indices[1]])
  }
  if (post.period.indices[2] < length(times)) {
    markers <- c(markers, times[post.period.indices[2]])
  }
  markers <- as.numeric(markers)
  return(markers)
}
cumsum.na.rm <- function(x) {
  # Cumulative sum

  if (is.null(x)) {
    return(x)
  }
  nas <- is.na(x)
  s <- cumsum(ifelse(nas, 0, x))
  s[nas] <- NA
  return(s)
}
ComputeCumulativeEffects <- function(y.samples, point.pred, y,
                                     post.period.begin, alpha = 0.05) {
  #THIS FUNCTION IS MODIFIED FROM GITHUB OF CAUSALIMPACT, AND SHOULD BE ONLY USED FOR BSCMs.

  # Computes summary statistics for the cumulative posterior predictions over
  # the unobserved data points in the post-intervention period.
  #
  # Args:
  #   y.samples:          Matrix of simulated response trajectories, as returned
  #                       by \code{ComputeResponseTrajectories()}.[number of post-burn-in MCMC samples] x [time points]
```

```
  #   point.pred:         Data frame of point predictions, as returned by
  #                       \code{ComputePointPredictions()}.
  #   y:                  Actual observed response, from the beginning of the
  #                       pre-period to the end of the observed period.
  #   post.period.begin: Index of the first data point of the post-period.
  #   alpha:              The resulting coverage of the posterior intervals will
  #                       be \code{1 - alpha}.
  #
  # Returns:
  #   data frame with 3 columns:
  #     cum.effect:        posterior prediction of the means of cumulative effects
  #     cum.effect.lower: lower limit of a \code{(1 - alpha)*100}% interval
  #     cum.effect.upper: upper limit



  # Compute posterior mean
  is.post.period <- seq_along(y) >= post.period.begin
  cum.pred.mean.pre <- cumsum.na.rm(as.vector(y)[!is.post.period])
  non.na.indices <- which(!is.na(cum.pred.mean.pre))
  assert_that(length(non.na.indices) > 0)
  last.non.na.index <- max(non.na.indices)
  cum.pred.mean.post <- cumsum(point.pred$mean[is.post.period]) +
    cum.pred.mean.pre[last.non.na.index]
  cum.pred.mean <- c(cum.pred.mean.pre, cum.pred.mean.post)

  # Check for overflow
  assert_that(identical(which(is.na(cum.pred.mean)),
                        which(is.na(y[!is.post.period]))),
              msg = "unexpected NA found in cum.pred.mean")

  # Compute posterior interval
  cum.pred.lower.pre <- cum.pred.mean.pre
  cum.pred.upper.pre <- cum.pred.mean.pre
  y.samples.cum.post <- t(apply(y.samples[, is.post.period, drop = FALSE], 1,
                                cumsum)) +
    cum.pred.mean.pre[last.non.na.index]
  if (sum(is.post.period) == 1) {
    y.samples.cum.post <- t(y.samples.cum.post)
  }
  assert_that(is.scalar(alpha), alpha > 0, alpha < 1)
  prob.lower <- alpha / 2      # e.g., 0.025 when alpha = 0.05
  prob.upper <- 1 - alpha / 2  # e.g., 0.975 when alpha = 0.05
  cum.pred.lower.post <- as.numeric(t(apply(y.samples.cum.post, 2, quantile,
                                            prob.lower)))
  cum.pred.upper.post <- as.numeric(t(apply(y.samples.cum.post, 2, quantile,
                                            prob.upper)))
  cum.pred.lower <- c(cum.pred.lower.pre, cum.pred.lower.post)
  cum.pred.upper <- c(cum.pred.upper.pre, cum.pred.upper.post)


  cum.y.model <- cumsum.na.rm(y)
  cum.effect.upper = cum.y.model-cum.pred.lower
  cum.effect.lower = cum.y.model-cum.pred.upper
  cum.effect.mean =  cum.y.model-cum.pred.mean
  # Put cumulative prediction together
  cum.effect<- data.frame(cum.effect = cum.effect.mean,
                          cum.effect.lower, cum.effect.upper)
  return(cum.effect)
}

check_interval_pointwise = function(lower,upper){
  #this function checks whether 0 is included in the credible interval
  # and 1 is returned if it is,  0 otherwise
  length = length(lower)
  include.zero = 1
  for (i in 1:length){
    if(!(lower[i]<=0 && upper[i]>=0)){
      include.zero = 0
    }
  }
  return(include.zero)
}
check_interval_overall = function(lower,upper){
  #this function checks whether 0 is included in the credible interval
  # and 1 is returned if it is,  0 otherwise
  length = length(lower)
  include.zero = 1
  if(!(lower[length]<=0 && upper[length]>=0)){
```

```
    include.zero = 0

  }
  return(include.zero)
}
##################################################


data = x[,c("country","year","gdp")]

gdp.westgermany = data[(data$country =='West_Germany'), ]
gdp.westgermany = aggregate(data = gdp.westgermany, gdp ~ year, FUN=mean) ##### specify the target variable here
#rownames(gdp.westgermany) = 1960:2003
gdp.westgermany=gdp.westgermany%>%
  dplyr::select(everything(),-year)

other = x[x$country !='West_Germany', ]
other = aggregate(data = other, gdp ~ year+country, FUN=mean) ##### specify the target variable here
other = reshape(other, direction="wide",timevar = "country",idvar = "year")
#rownames(other) = 1960:2003
other=other%>%
  dplyr::select(everything(),-year)


####### CIM
data.cim=cbind(gdp.westgermany,other)
rownames(data.cim) = time
start.time.CIM <- Sys.time()
impact.cim= CausalBsts::CausalImpact(data.cim,pre.period = c(1,30),post.period = c(31,44),
                                    model.args=list(niter=20000))
end.time.CIM <- Sys.time()
time.CIM = difftime(end.time.CIM, start.time.CIM,units = "secs")

time = 1960:2003

#plot both the true response and predicted response, as well as the credible interval of the predictions

y.pred.CIM = data.frame(mean = impact.cim$series$point.pred,lower = impact.cim$series$point.pred.lower,
                        upper = impact.cim$series$point.pred.upper,response = impact.cim$series$response)
colnames(y.pred.CIM) = c("mean","lower","upper","response")
rownames(y.pred.CIM) = time
CreatePredictionPlot(y.pred.CIM,T=2003,T0=1990)

#compute and plot the pointwise impact and credible interval
pointwise.impct.CIM = data.frame(impact = impact.cim$series$response-impact.cim$series$point.pred,
                                 lower = impact.cim$series$response -impact.cim$series$point.pred.upper,
                                 upper = impact.cim$series$response-impact.cim$series$point.pred.lower)
colnames(pointwise.impct.CIM) = c('impact','lower','upper')
CreatePointwiseImpactPlot(pointwise.impct.CIM,T=2003,T0=1990)

# plot the cumulative impacts and credible interval

cumulative.interval.CIM = data.frame(impact = impact.cim$series$cum.effect,
                                     lower = impact.cim$series$cum.effect.lower,
                                     upper = impact.cim$series$cum.effect.upper)
colnames(cumulative.interval.CIM)=c("impact","lower","upper")
rownames(cumulative.interval.CIM) = time
CreateCumulativeImpactPlot(cumulative.interval.CIM,T=2003,T0=1990)





summary(impact.cim)


######### horseshoe##########
time = 1960:2003
data.bscm =list(N_train = 30, N_test = 14, p = 16, y_train = gdp.westgermany[1:30,],
                X_train = as.matrix(other[1:30,]), X_test = as.matrix(other[31:44,]))
start.time.horseshoe <- Sys.time()
fit.horseshoe <- stan(file = 'horseshoe.stan', data = data.bscm,iter=20000,warmup = 10000,
                      control = list(adapt_delta = 0.8,max_treedepth = 15))
end.time.horseshoe <- Sys.time()
time.horseshoe = difftime(end.time.horseshoe, start.time.horseshoe,units = "secs")
y.pred.horseshoe= summary(fit.horseshoe , pars = c("y_fit","y_test"), probs = c(0.025, 0.975))$summary
y.pred.horseshoe = as.data.frame(y.pred.horseshoe[,c("mean","2.5%","97.5%")])
y.pred.horseshoe$response = as.matrix(gdp.westgermany)
colnames(y.pred.horseshoe) = c("mean","lower","upper","response")
rownames(y.pred.horseshoe) = time
```

```
#plot both the true response and predicted response, as well as the credible interval of the predictions
CreatePredictionPlot(y.pred.horseshoe,T=2003,T0=1990)


#compute the pointwise impact and credible interval
pointwise.impact.horseshoe = data.frame(mean= y.pred.horseshoe$response-y.pred.horseshoe$mean,
                                        upper=y.pred.horseshoe$response-y.pred.horseshoe$lower
                                        ,lower = y.pred.horseshoe$response-y.pred.horseshoe$upper)
colnames(pointwise.impact.horseshoe) = c('impact','lower','upper')
CreatePointwiseImpactPlot(pointwise.impact.horseshoe,T=2003,T0=1990)


#compute the credible interval of cumulative effects
samples.horseshoe= cbind(extract(fit.horseshoe)$y_fit,
extract(fit.horseshoe)$y_test) #extract the sampled target variable
#samples.horseshoe = samples.horseshoe[10001:20000,] #only use those sampled after warm-up

cumulative.interval.horseshoe = ComputeCumulativeEffects(y.samples=samples.horseshoe,
                                                          point.pred= y.pred.horseshoe,
                                                          y=as.matrix(gdp.westgermany),
                                                          post.period.begin=31, alpha = 0.05)


# plot the cumulative impacts and credible interval
cumulative.interval.horseshoe = as.data.frame(cumulative.interval.horseshoe)
colnames(cumulative.interval.horseshoe)=c("impact","lower","upper")
rownames(cumulative.interval.horseshoe) = time
CreateCumulativeImpactPlot(cumulative.interval.horseshoe,T=2003,T0=1990)




###############spike-and-slab########
start.time.spike <- Sys.time()
fit.spike <- stan(file = 'spike-and-slab.stan', data = data.bscm, iter=100000,warmup = 80000,
                  control = list(adapt_delta = 0.8,max_treedepth = 15))
end.time.spike <- Sys.time()
time.spike = difftime(end.time.spike, start.time.spike, units = "secs")
y.pred.spike= summary(fit.spike , pars = c("y_fit","y_test"), probs = c(0.025, 0.975))$summary
y.pred.spike = as.data.frame(y.pred.spike[,c("mean","2.5%","97.5%")])
y.pred.spike$response = as.matrix(gdp.westgermany)
colnames(y.pred.spike) = c("mean","lower","upper","response")
rownames(y.pred.spike) = time
#plot both the true response and predicted response, as well as the credible interval of the predictions
CreatePredictionPlot(y.pred.spike,T=2003,T0=1990)

#compute the pointwise impact and credible interval
pointwise.impact.spike = data.frame(mean= y.pred.spike$response-y.pred.spike$mean,
                                    upper=y.pred.spike$response-y.pred.spike$lower
                                    ,lower = y.pred.spike$response-y.pred.spike$upper)
colnames(pointwise.impact.spike) = c('impact','lower','upper')
CreatePointwiseImpactPlot(pointwise.impact.spike,T=2003,T0=1990)

#compute the credible interval of cumulative effects
samples.spike= cbind(extract(fit.spike)$y_fit,extract(fit.spike)$y_test) #extract the sampled target variable
#samples.spike = samples.spike[10001:20000,] #only use those sampled after warm-up

cumulative.interval.spike = ComputeCumulativeEffects(y.samples=samples.spike, point.pred= y.pred.spike,
                                                      y=as.matrix(gdp.westgermany),
                                                      post.period.begin=31, alpha = 0.05)


# plot the cumulative impacts and credible interval
cumulative.interval.spike = as.data.frame(cumulative.interval.spike)
colnames(cumulative.interval.spike)=c("impact","lower","upper")
rownames(cumulative.interval.spike) = time
CreateCumulativeImpactPlot(cumulative.interval.spike,T=2003,T0=1990)
```

C.3 RStan code for the Spike-and-Slab prior (Kim et al. (2020))

```
data{
int N_train; //Number of observations in the pre-treatment periods
int N_test; //Number of observations in the post-treatment periods
int p; //Number of control units
real y_train[N_train]; //Treated unit in the pre-treatment periods
matrix[N_train, p] X_train; //Control unit matrix in the pre-treatment
```

```
matrix[N_test, p] X_test; //Control unit matrix in the post-treatment
}
parameters{
real beta_0; //Intercept
real<lower=0> sigma2; //Error term variance
vector[p] beta; //Control unit weights
//Hyperparameters prior
vector<lower=0>[p] tau2; //Variance slab
vector<lower=0,upper=1>[p] gamma; //Inclusion probability
}
transformed parameters{
real<lower=0> sigma; //Error sd
vector[N_train] X_beta; //Synthetic control unit prediction in the pre-treatment period
sigma = sqrt(sigma2);
X_beta = beta_0 + X_train*beta;
}
model{
//Pre-treatment estimation
tau2 ~ inv_gamma(0.5, 0.5);
gamma ~ uniform(0, 1);
beta_0~cauchy(0,10);
sigma ~ cauchy(0,10);
for(j in 1:p){
target += log_sum_exp(1-log(gamma[j]) + normal_lpdf(beta[j] | 0, sqrt(0.001)),log(gamma[j]) + normal_lpdf(beta[j] | 0, s
}
target += -2 * log(sigma);
y_train ~ normal(X_beta, sigma);
}
generated quantities{
//Post-treatment prediction & Log-likelihood
vector[N_train] y_fit; //Fitted synthetic control unit in the pre-treatment
vector[N_test] y_test; //Predicted synthetic control unit in the post-treatment
vector[N_train] log_lik; //Log-likelihood
y_fit = beta_0 + X_train * beta;
for(i in 1:N_test){
y_test[i] = normal_rng(beta_0 + X_test[i,] * beta, sigma);
}
for (t in 1:N_train) {
log_lik[t] = normal_lpdf(y_train[t] | y_fit[t], sigma);
}
}
```

C.4 RStan code for the Horseshoe prior (Kim et al. (2020))

```
data {
  int N_train; //Number of observations in the pre-treatment periods
  int N_test; //Number of observations in the post-treatment periods
  int p; //Number of control units
  real y_train[N_train]; //Treated unit in the pre-treatment periods
  matrix[N_train, p] X_train; //Control unit matrix in the pre-treatment
  matrix[N_test, p] X_test; //Control unit matrix in the post-treatment
}

parameters {
  real beta_0; //Intercept
  real<lower=0> sigma2; //Error term variance
  vector[p] beta_raw; //Control unit weights (will be transformed)
  //Hyperparameters prior
  vector<lower=0, upper=pi()/2>[p] lambda_unif;
  real<lower=0> tau; //Global shrinkage
}

transformed parameters{
  vector[p] beta; //Control unit weights
  real<lower=0> sigma; //Error term sd
  vector<lower=0>[p] lambda; //Local shrinkage
  vector[N_train] X_beta; //Synthetic control unit prediction in the pre-treatment period
  lambda = tau * tan(lambda_unif); // => lambda ~ cauchy(0, tau)
  for(j in 1:p){
    beta[j] = lambda[j] * beta_raw[j];
  }
  sigma = sqrt(sigma2);
  X_beta = beta_0 + X_train*beta;
  }

model {
```

```
  //Pre-treatment estimation
  beta_raw ~ normal(0, 1); //=> beta ~ normal(0, lambda^2)
  tau ~ cauchy(0, sigma);
  sigma ~ cauchy(0,10);
  beta_0 ~ cauchy(0,10);
  y_train ~ normal(X_beta, sigma);
}

generated quantities{
  //Post-treatment prediction & Log-likelihood
  vector[N_train] y_fit; //Fitted synthetic control unit in the pre-treatment
  vector[N_test] y_test; //Predicted synthetic control unit in the post-treatment
  vector[N_train] log_lik; //Log-likelihood
  y_fit = beta_0 + X_train * beta;
  for(i in 1:N_test){
    y_test[i] = normal_rng(beta_0 + X_test[i,] * beta, sigma);
  }
  for (t in 1:N_train) {
    log_lik[t] = normal_lpdf(y_train[t] | y_fit[t], sigma);
  }
}
```