

# Efficient scenario analysis in real-time Bayesian election forecasting via sequential meta-posterior sampling

Geonhee Han

Graduate School of Arts and Sciences, Columbia University\*

Graduate School of Public Policy, University of Tokyo

Andrew Gelman

Department of Statistics and Department of Political Science, Columbia University

Aki Vehtari

Department of Computer Science, Aalto University

20 Oct 2025

## Abstract

Bayesian aggregation lets election forecasters combine diverse sources of information, such as state polls and economic and political indicators: as in our collaboration with *The Economist* magazine. However, the demands of real-time posterior updating, model checking, and communication introduce practical methodological challenges. In particular, sensitivity and scenario analysis help trace forecast shifts to model assumptions and understand model behavior. Yet, under standard Markov chain Monte Carlo, even small tweaks to the model (e.g., in priors, data, hyperparameters) require full refitting, making such real-time analysis computationally expensive. To overcome the bottleneck, we introduce a meta-modeling strategy paired with a sequential sampling scheme; by traversing posterior meta-models, we enable real-time inference and structured scenario and sensitivity analysis without repeated refitting. In a back-test of the model, we show substantial computational gains and uncover non-trivial sensitivity patterns. For example, forecasts remain responsive to prior confidence in fundamentals-based forecasts, but less so to random walk scale; these help clarify the relative influence of polling data versus structural assumptions. Code is available at <https://github.com/geonhee619/SMC-Sense>.

**Keywords:** *Real-time election forecasting, Sequential Bayesian updating, Scenario analysis, Sensitivity analysis, Sequential Monte Carlo*

---

\*gh2610@columbia.edu. The collaboration and majority of this research was carried out while GH was a graduate student at Columbia University GSAS.

# 1 Introduction

## Bayesian aggregation for election forecasting

Bayesian aggregation provides a method for combining diverse sources of information that help explain election outcomes. These include the large number of publicly available state-level and national-level pre-election polls (Holbrook and DeSart, 1999; Linzer, 2013), fundamentals-based predictors such as national and state economic indicators (e.g., Campbell, 1992; Gelman and King, 1993; Wlezien and Erikson, 1996; Bartels and Zaller, 2001; Nadeau and Lewis-Beck, 2001; Erikson and Wlezien, 2008), and historical voting patterns (e.g., Lock and Gelman, 2010).

*The Economist* model is a recent example, developed through our collaboration with *The Economist* magazine (Morris, 2020; Heidemanns et al., 2020; Gelman et al., 2024, detailed in Section 2.1). The model integrates national and state-level polls with economic and political indicators summarized under fundamentals-based prior forecasts, within a hierarchical Bayesian framework that simultaneously accounts for correlation across states, temporal variation, and both sampling and non-sampling error. The model is estimated in real-time via Stan (Carpenter et al., 2017), and updated daily as opinion polls become newly available, therefore producing retrospective estimates and prospective forecasts that evolve over time: in line with the modeling framework first introduced by Linzer (2013) which aims to refine state-level election forecasts over time via Bayesian updating.

## The methodological challenge

This work is motivated by a unique methodological challenge in sequential Bayesian election forecasting, at the intersection of large-scale data fusion, daily updating, the need for thorough model checking, and communication. The model posterior is updated sequentially as new data and events arrive, which necessitates not only inference but also routine diagnostic checks of the model—as a mapping from observations to inferences—to ensure practical usability in real-time, as is the case in all statistical workflows (e.g., Cook, 1977, 1979; Belsley et al., 1980; Huber, 1981; Rosenbaum and Rubin, 1983; Eubank, 1984; Carnegie et al., 2016).

For example, consumers of the forecast are interested in (a) scenario analysis, in which we explore how the final predictions change under different hypothetical conditions (Gelman et al., 2020, Chap. 6), and producers are interested in (b) sensitivity analysis, in which we examine how robust and sensible the final inferences are to alternative configurations of model specifications or hyperparameters (e.g., Canavos, 1975; Ross, 1987; Gustafson, 1996; Clarke and Gustafson, 1998; Berger et al., 2000; Zhu et al., 2011; Roos et al., 2015; Masoero et al., 2018).

In back-testing the model, we fit the model to historical presidential elections and to the 2024 data to evaluate whether the resulting inferences aligned with current political expectations. *Forward* checks were performed in hypothetical polling scenarios to assess temporal responsiveness and to determine whether the model tracked meaningful shifts

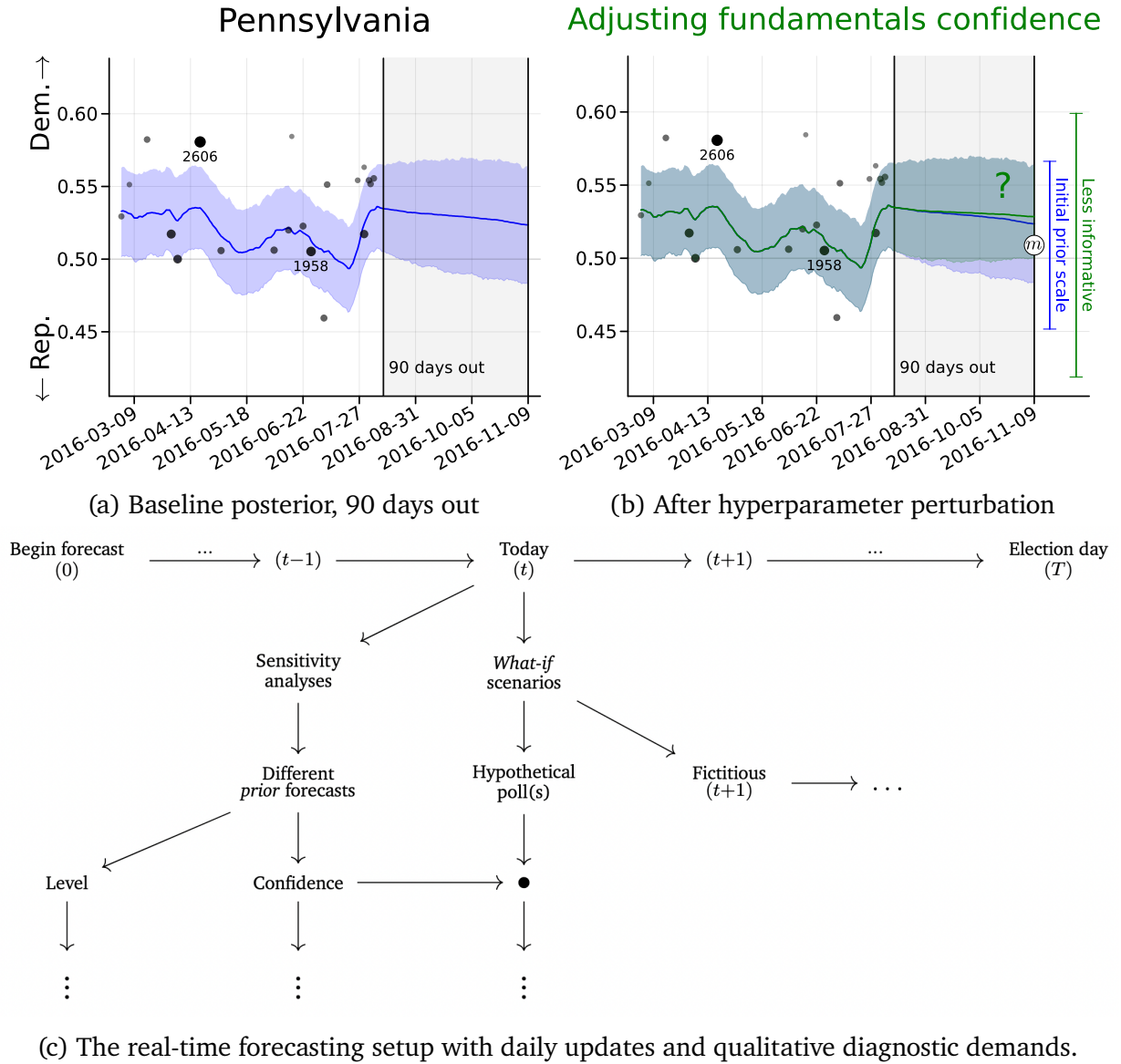


Figure 1: As new data and events arrive sequentially, the posterior is updated daily. Panels (a)–(c) illustrate the role of scenario/sensitivity analysis in the real-time forecasting setup. (a) exemplifies the baseline posterior before any perturbation. (b) demonstrates how modifying hyperparameters may affect inferences and forecasts. (c): This process is embedded in a live forecasting system; to support routine model checking and forecast revision, we seek computationally efficient methods for timely, time-specific scenario exploration and sensitivity analysis without repeated model refitting.

without overreacting to individual polls. Further examples are as follows.

- How does the model respond to new observations? (**data arrivals**)
- How do forecasts shift under alternative priors or structural assumptions (fundamentals-based forecasts: those from economic and political information)? How should relative trust in priors be calibrated/adjusted? What dependencies emerge when we perturb

the model? (**sensitivity analysis**)

- What if we varied the realized fundamentals-based projection to a different quantity? (**exploratory inference**)

In a large-scale sequential forecasting context, these analyses are especially valuable, as the model is never *final*; each new data point prompts revisions to both the forecasts and the model itself. As noted in [Gelman et al. \(2024\)](#), “we are constructing a sort of robot—a forecast that should be able to update itself over time as new polls and economic and political information arrive—so our checking is not just on the current forecast probabilities but also on how they develop over time.”

Accordingly, we embrace a continuous and iterative refinement process and emphasize motivating and explaining model updates in ways that are interpretable, transparent, and communicable. However, answering such time-specific queries in a real-time forecasting context is computationally demanding, particularly when relying on conventional Markov chain Monte Carlo (MCMC) methods. Each incoming batch of data points necessitates full posterior recomputation; any time-specific perturbation to quantities the model conditions upon (e.g., fundamentals-based forecasts; state-by-state correlation) requires model re-fitting. Consequently, performing additional diagnostic analyses by perturbing each data point and/or hyperparameter individually becomes impractical under standard MCMC workflows.

Existing computational strategies relevant to our modeling framework based on MCMC are summarized in Table 1. Beyond brute force MCMC, deletion-based perturbative methods based on importance reweighting have been proposed ([Gelfand and Dey, 1994](#); [Peruggia, 1997](#); [Epifani et al., 2008](#); [Vehtari et al., 2017](#); [Han and Gelman, 2025](#)) for case-influence analysis and out-of-sample predictive assessment. These approaches offer approximate case-deleted posterior diagnostics, but are limited in scope for our purposes involving very specific perturbations. Infinitesimal perturbation techniques provide local sensitivity analysis with respect to hyperparameters or likelihood contributions to the posterior mean ([Giordano, 2018](#); [Kallioinen et al., 2023](#); [Nguyen et al., 2024](#)). While fast and mathematically elegant, infinitesimal schemes can be difficult to interpret and communicate in applied and collaborative settings to collaborators also actively involved in the forecasting process.

## Scope

Our goal is to develop a strategy compatible with our applied forecasting context, for post-MCMC scenario and sensitivity analysis under real-time model sequential updating, enabling broader, user-adjustable perturbative schemes that remain interpretable. Notably, sequential Monte Carlo (SMC) samplers ([Del Moral et al., 2006](#)) provide a foundation for interpretable posterior diagnostics through user-controlled hyperparameter variation (e.g. [Bornn et al., 2010](#)). We extend this approach to further address the specific demands of our



	Use / pros	Mismatch	Literature (e.g.)
Brute force MCMC	Generic, interpretable, easy to implement	Inefficient re-computation	
Deletion reweighting	Case deletion (e.g., case influence; cross-validation for out-of-sample eval.)	Limited to case-influence	<a href="#">Gelfand and Dey (1994)</a> , <a href="#">Epifani et al. (2008)</a> , <a href="#">Vehtari et al. (2017)</a> .
Infinitesimal	Local and linear sensitivity to hyperparameter and/or prior/likelihood contribution	Can be hard to interpret and communicate	<a href="#">Giordano (2018)</a> , <a href="#">Kallioinen et al. (2023)</a> , <a href="#">Nguyen et al. (2024)</a> .
Sequential	Hyperparameter sensitivity; cross-validation	Want broader perturbation schemes (e.g., data arrival)	<a href="#">Bornn et al. (2010)</a> , <a href="#">Han and Gelman (2025)</a> .
<b>This work</b>	Sequentially re-cycle computation for efficiency. Applicable/applied to real-time scenario/sensitivity analysis. Interpretable; ease of communication. Modular; no code re-writing per scheme (BridgeStan accesses Stan internals).		

Table 1: *Summary of potentially applicable existing computational approaches. The listed literature is illustrative and not exhaustive. **Abbreviation:** MCMC = Markov chain Monte Carlo.*

setting, including sequential real and hypothetical data arrivals, time-specific sensitivity diagnostics, and time-specific *what-if* exploratory analysis.

At the core of our approach is a meta-modeling framework that operates over user-adjustable *knobs* that act on the joint distribution of a Bayesian model. These enable structured posterior traversal through targeted perturbations, facilitating operations such as exploration of hypothetical datasets, hyperparameter modulation, and sensitivity analysis. Then, the meta-model is paired with the SMC sampler to enable efficient posterior exploration along a continuum of structured perturbations. Taking advantage of the sequential proximity of *adjacent* models along this path, the approach avoids repeated model refitting to substantially reduce computational cost relative to brute-force MCMC, while retaining the interpretability comparable to full posterior reruns. The pairing also notably complements MCMC workflows; the analysis can be performed as a byproduct of a single MCMC output, which makes it well-suited to complement our forecasting framework.

For implementation we emphasize usability and modularity; we interface directly with Stan internals (via BridgeStan: [Roualdes et al. 2023](#)) so that users can specify perturbation schemes without re-writing code for every perturbative analysis. For example, varying the hyperparameter from one value to another requires only a declarative specification of the change (e.g., in the data block). Code is available at <https://github.com/geonhee619/SMC-Sense>.

**Structure.** The remainder of the paper is organized as follows. Section 2 presents the formal model specification and introduces the meta-model formulation. Section 3 details the implementation of scenario/sensitivity analysis through traversal of the meta-model. Section 4 demonstrates the framework through simulation studies. Section 4 also backtests the model using data from the 2016 U.S. presidential election. Section 5 concludes.

## 2 Bayesian aggregation for election forecasting

### 2.1 The model

We begin with a brief overview of the basic structure of *The Economist* model in 2024. Let  $y_i$  and  $n_i - y_i$  denote the number of respondents in poll  $i$  supporting the Democratic and Republican candidates, respectively. Both  $y_i$  and  $n_i$  are observed in the model with

$$y_i \stackrel{\text{ind}}{\sim} \text{binomial}(n_i, p_i). \quad (i = 1, \dots, N)$$

The likelihood does not explicitly account for design effects or non-sampling error; these are partly accounted for by additional error terms in the model (Gelman et al., 2024) as follows. The latent two-party support rate, represented by the binomial success probability  $p_i$ , is modeled on the logit scale with the additive structure,

$$\text{logit}(p_i) = \left\{ \begin{array}{ll} \mu_{t[i],s[i]} + u_{s[i]} & (i \text{ is state-level}) \\ \sum_{s=1}^S w_s (\mu_{t[i],s} + u_s) & (i \text{ is national}) \end{array} \right\} + \alpha_i + \sigma_i \varepsilon_i. \quad (i = 1, \dots, N)$$

Here,  $\mu_{t,s}$  denotes the latent support trend in state  $s$  at time  $t$ ,  $u_s$  is a state-level error term,  $w_s$  are weights used to aggregate state trends into national-level predictions,  $\alpha_i$  captures systematic bias,  $\sigma_i \varepsilon_i$  represents (non-centered) poll-specific error, and  $S$  is the number of states.

Now, let  $T$  denote the fixed time horizon of a given election year. We model the latent opinion time series  $\mu_{t,s}$  as a correlated random walk,

$$\boldsymbol{\mu}_t \sim \text{MVN}(\boldsymbol{\mu}_{t+1}, \boldsymbol{\Sigma}^{(\mu)}), \quad (t = T - 1, \dots, 1)$$

where  $\boldsymbol{\mu}_t = (\mu_{t,1}, \dots, \mu_{t,S})$  denotes the vector of latent state-level opinions at time  $t$ . To also incorporate information from economic and political indicators, we anchor the trajectory by placing an explicit prior on the final day,

$$\boldsymbol{\mu}_T \sim \text{MVN}(\mathbf{f}, \mathbf{C}),$$

where  $\mathbf{f}$  represents fundamentals-based forecasts (informed by economic and political indicators), and the covariance  $\mathbf{C}$  encodes both potential inter-state dependencies and prior uncertainty (i.e., relative *confidence* in some sense).

The bracketed term also includes the state-level errors  $u_s$  for  $s = 1, \dots, S$ , which are modeled jointly to capture inter-state correlation,

$$(u_1, \dots, u_S) \sim \text{MVN}(\mathbf{0}, \Sigma^{(u)}).$$

The poll-specific bias term  $\alpha_i$  is decomposed into a sum of components related to known sources of variation (e.g., [McDermott and Frankovic, 2003](#); [Wlezien and Erikson, 2006](#)): polling house effects, response mode effects, polling population effects, an adjustment pertaining to partisan lean, and dynamic errors via stationary autoregression; complete descriptions are omitted for brevity, as they are not central to this work. See [Heidemanns et al. \(2020\)](#) and [Gelman et al. \(2024\)](#) for descriptions. Finally, independent measurement errors are modeled as

$$\varepsilon_i \stackrel{\text{ind}}{\sim} \text{normal}(0, 1),$$

where the scale parameter  $\sigma_i = \sigma^{(\text{state})} > 0$  if  $i$  is a state poll and  $\sigma_i = \sigma^{(\text{national})} > 0$  if  $i$  is a national poll.

In essence, as described in [Heidemanns et al. \(2020\)](#), the model uses the currently available *noisy* poll data  $y_{1:N} = (y_1, \dots, y_N)$  to estimate the underlying public opinion over time, both retrospectively and prospectively, via the *denoised* latent multivariate random walk  $\boldsymbol{\mu}_{1:T} = (\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_T)$ . The state-level forecasts are then obtained from the marginal posterior distributions of  $(\mu_{1,s}, \dots, \mu_{T,s})$  for each state  $s$ ; national forecasts are calculated as the marginal posterior of the weighted average over time  $(\sum_{s=1}^S w_s \mu_{1,s}, \dots, \sum_{s=1}^S w_s \mu_{T,s})$ .

## 2.2 The meta-model

Both state-level and national forecasts evolve dynamically throughout the campaign—not in the sense that they are explicitly (day)  $t$  indexed, but because the number of data points  $N$  grows as the polling accumulates, and each day there is an increase  $t \leftarrow t + 1$ . Consequently, the generative model in Section 2.1 defines the joint distribution, specifically on day  $t$ ,

$$p_t(\boldsymbol{\Theta}, y_{1:N_t}) = p(a_1, \dots, a_J) p(u_1, \dots, u_S | \Sigma^{(u)}) p(\boldsymbol{\mu}_T | \mathbf{f}, \mathbf{C}) \left[ \prod_{t=1}^{T-1} p(\boldsymbol{\mu}_t | \boldsymbol{\mu}_{t+1}, \Sigma^{(\mu)}) \right] \left[ \prod_{i=1}^N p(\varepsilon_i) \right] \left[ \prod_{i=1}^{N_t} \mathbb{P}(y_i | n_i, p_i, \sigma_i) \right]. \quad (t = 1, \dots, T)$$

$N_t$  now explicitly depends on day  $t$  and accumulates throughout the campaign up to day  $T$ . We have compiled the parameters of interest shared throughout the campaign as  $\boldsymbol{\Theta} := ((a_j)_j, (u_s)_s, (\boldsymbol{\mu}_t)_t, (\varepsilon_i)_i)$ , where the relevant additive terms that appear in  $\alpha_i$  are enumerated as  $(a_j)_j$  for simplicity. Also, in practice, the poll-specific errors  $\varepsilon_i$  for  $i > N_t$  on each day  $t$  are marginalized out, but are nevertheless retained in the formulation; we operate over a consistent parameter space throughout.

To simplify the exposition from here on, we rewrite the posterior on day  $t$  as

$$p_t(\boldsymbol{\Theta} | \mathbf{y}_{1:N_t}, \boldsymbol{\delta}_t) \propto \prod_{k=1}^K p_{\phi_k}(\boldsymbol{\theta}_k) \prod_{i=1}^{N_t} p_{\psi_i}(\mathbf{y}_i | \boldsymbol{\Theta}),$$

where the prior terms  $p_{\phi_k}(\boldsymbol{\theta}_k)$  are partitioned and enumerated such that they are conditionally independent across parameters  $\boldsymbol{\theta}_k$  indexed by  $k = 1, \dots, K$ , and the likelihood terms  $p_{\psi_i}(\mathbf{y}_i | \boldsymbol{\Theta})$  are conditionally independent across data units  $i = 1, \dots, N_t$  as before. Here,  $\phi_k$  denotes the hyperparameters for the  $k$ -th prior component, and  $\psi_i$  denotes the hyperparameters for the  $i$ -th likelihood.

$\delta_t$  encapsulates the set of fixed model quantities. Our task reduces to evaluating how varying  $\delta_t$ , affects the posterior distribution. That is, we work directly with the following meta-model, indexed by the configuration (further expanded beyond the tuple  $(\phi_{1:K}, \psi_{1:N_t}, \mathbf{y}_{1:N_t})$ ),

$$\begin{aligned} \delta_t &:= (\phi_{1:K}, \rho_{1:K}, \psi_{1:N_t}, \mathbf{f}_{1:N_t}, \gamma_{1:N_t}, \psi^*, \mathbf{y}^*, \gamma^*) \\ &\mapsto p_t(\boldsymbol{\Theta} | \mathbf{y}_{1:N_t}, \delta_t) \\ &\propto \prod_{k=1}^K p_{\phi_k}(\boldsymbol{\theta}_k)^{\rho_k} \prod_{i=1}^{N_t} p_{\psi_i}(\mathbf{f}_i(\mathbf{y}_i) | \boldsymbol{\Theta})^{\gamma_i} p_{\psi^*}(\mathbf{y}^* | \boldsymbol{\Theta})^{\gamma^*}. \end{aligned}$$

$\delta_t$  again encapsulates the full set of fixed model hyperparameters and perturbation controls at day  $t$ ;  $p_{\phi_k}(\cdot)$  is the prior on the  $k$ -th parameter  $\boldsymbol{\theta}_k$  governed by hyperparameter  $\phi_k$ ;  $\rho_k > 0$  is the power coefficient applied to the prior on  $\boldsymbol{\theta}_k$ ;  $p_{\psi_i}(\cdot)$  is the conditional likelihood for the  $i$ -th observation with hyperparameter  $\psi_i$ ,  $\mathbf{f}_i$  is the data perturbation function applied to the  $i$ -th unit  $\mathbf{y}_i$ ;  $p_{\psi^*}(\cdot)$  is the likelihood for hypothetically inserted data  $\mathbf{y}^*$  (which may consist of one or more observations) governed by hyperparameter  $\psi^*$ ; and  $\gamma_i, \gamma^* \geq 0$  are the power coefficients applied to the likelihood for observed and hypothetical data, respectively. The original posterior is recovered as a special case when  $\rho_k = \gamma_i = 1$ ,  $\gamma^* = 0$ , and the existing hyperparameters  $(\phi_{1:K}, \psi_{1:N_t})$  are kept as is. Operating over the knob  $\delta_t$  is therefore one way to retrieve hypothetical posteriors under varying configurations of hyperparameter settings, data-value perturbations, synthetic data insertions, and additionally prior and likelihood informativity.

### 3 Traversing the meta-model

Given the meta-model in Section 2.2, the computational problem now is to efficiently traverse a family of posteriors indexed by the knob  $\delta_t$ . We outline one such approach by noting that the formulation naturally lends itself to a sequential sampling strategy; we can take advantage of the sequential proximity of knob-turning to enable efficient posterior exploration across configurations without repeated per-configuration simulation.

#### 3.1 Turning the knob

We begin by formalizing how the knob  $\delta_t$  is modulated to induce variation in otherwise fixed model quantities. We first explicitly define some terminology.

**Definition 3.1.** Let the *baseline* distribution at day  $t$  be defined as the posterior under the unperturbed configuration  $\delta_{t,0}$  (the original one at time  $t$ ),

$$p_t(\Theta | \delta_{t,0}) := p(\Theta | \mathbf{y}_{1:N_t}, \delta_t = \delta_{t,0}).$$

We then introduce the notion of a perturbation (Weiss, 1996) as an action that modulates the configuration from  $\delta_{t,0}$  to a perturbed setting.

**Definition 3.2.** A *perturbation*  $h_{t,\ell}(\cdot)$  summarizes the modulation from  $\delta_{t,0}$  to  $\delta_{t,\ell}$  at day  $t$ , such that the perturbed posterior satisfies

$$p_t(\Theta | \delta_{t,\ell}) \propto p_t(\Theta | \delta_{t,0}) \cdot h_{t,\ell}(\Theta), \quad (\ell \in (0, 1])$$

where  $\ell$  indexes the perturbation level. At  $\ell = 0$ , define  $h_{t,0} := 1$  as idempotent.

The continuum  $(p_t(\cdot | \delta_{t,\ell}))_{\ell \in (0,1]}$  then defines the posterior trajectory at time  $t$  under controlled perturbation of fixed model quantities. In the following, we present concrete examples to illustrate some modulations relevant to our forecasting setup.

**Example 3.1** (Sequential updates). We want to capture the natural evolution of the posterior as new observations arrive to enable *chained* updates without full re-fitting. Therefore, given new polling data  $\mathbf{y}^* := (y_i)_{i=N+1}^{N+h}$  with corresponding survey sample sizes  $\psi^* := (n_i)_{i=N+1}^{N+h}$  upon transitioning from  $t$  to  $t+1$ , we first identify the target as

$$p_{t+1}(\Theta | \delta_{t+1,\ell=0}) = p_t(\Theta | \delta_{t,\ell=1}),$$

which we bridge from the baseline  $p_t(\Theta | \delta_{t,0})$  by, for example, the perturbation

$$h_{t,\ell}(\Theta) := p_{\psi^*}(\mathbf{y}^* | \Theta)^{\gamma^*(\ell)} \stackrel{\text{e.g.}}{=} \prod_{i=N+1}^{N+h} \text{binomial}(y_i | n_i, p_i(\Theta))^{\gamma^*(\ell)},$$

where  $\gamma^*$  monotonically increases from 0 to 1 for gradual data injection.

**Example 3.2** (Hypothetical data insertions). This is handled analogously to sequential updates, but with hypothetical data  $(\mathbf{y}^*, \psi^*)$ . A concrete example to highlight the specificity of this approach (also used subsequently in Section 4) is

$$\begin{aligned} \mathbf{y}^*(\ell) &= y_{N_t+1}(\ell) = \ell, \\ \frac{y_{N_t+1}(\ell)}{n_{N_t+1}(\ell)} &= \frac{1}{2}, \\ \psi^*(\ell) &= \begin{bmatrix} \text{pollster} \\ \text{mode} \\ \text{population} \end{bmatrix} \stackrel{\text{e.g.}}{=} \begin{bmatrix} \text{NBC} \\ \text{Live phone interview} \\ \text{Registered voters} \end{bmatrix}; \end{aligned}$$

the rationale here is to allow posterior scenario analysis on counterfactual polls.

**Example 3.3** (Prior *what-if* & sensitivity). We define the perturbation as

$$h_{t,\ell}(\Theta) := p_{\phi(\ell)}(\theta_k) \cdot p_{\phi_k}(\theta_k)^{-1},$$

where  $\phi(\ell)$  interpolates between baseline and alternative prior settings.

More concretely, the following is an example in which we modulate the fundamentals-based forecast  $\mathbf{f}$  (location hyperparameter) gradually up/downward,

$$h_{t,\ell}(\Theta) = \text{MVN}(\mu_T | \mathbf{f} + a_\ell \mathbf{e}_s, \mathbf{C}) \cdot \text{MVN}(\mu_T | \mathbf{f}, \mathbf{C})^{-1}. \quad (1)$$

$\mathbf{e}_s$  is the standard basis in  $\mathbb{R}^S$  with unity in the  $s$ -th entry, so  $a_\ell$  directly modulates the fundamentals-based forecast at a particular state  $s$ .

**Example 3.4** (Data-value *what-if* & sensitivity). Defining

$$h_\ell(\Theta) := p_{\psi_i}(f_\ell(\mathbf{y}_i) | \Theta) \cdot p_{\psi_i}(\mathbf{y}_i | \Theta)^{-1},$$

where  $f_\ell(\cdot)$  is a transformation of  $\mathbf{y}_i$  (with  $0 < f_\ell < n_i$  to avoid degeneracy), enables local sensitivity analysis to perturbations in the observed survey outcomes (beyond infinitesimal local linear approximation).

**Example 3.5** (Data informativity & prior diffuseness). Although not used in our setup, we can modulate the likelihood contribution via  $\gamma_i \in (0, 1]$  and similarly the prior contribution via  $\rho_k \in (0, 1]$  to control the influence of individual observations and tune the prior strength.

## 3.2 Sequential sampling

The above user-adjustable mechanism in Section 3.1 now enables both *horizontal* and *vertical* sequential updates as intended; with  $p_{t=0}(\cdot | \delta_{t=0,\ell=0})$  being the prior (i.e., model joint distribution with  $N_{t=0} = 0$ ), given a starting distribution  $p_t$  and its specified sequence of perturbation functions  $((h_{t,\ell})_\ell, \dots, (h_{T,\ell})_\ell)$ , we characterize a continuum path of posterior distributions over time (in the real and hypothetical sense), as illustrated directly below.

$$\begin{array}{ccccccc} p_0 & \longrightarrow & \dots & \longrightarrow & p_{t-1} & \xrightarrow{h_{t-1,\ell}} & p_t & \xrightarrow{h_{t,\ell}} & p_{t+1} & \xrightarrow{h_{t+1,\ell}} & \dots & \longrightarrow & p_T & \text{(Horizontal)} \\ & & & & & & \downarrow & \searrow & & & & & & \\ & & & & & & p_t^* & & p_t^{**} & \longrightarrow & \dots & & & \text{(Vertical)} \\ & & & & & & \downarrow & & & & & & & \\ & & & & & & \vdots & & & & & & & \end{array}$$

A given path defines our target, which we approximate via a sequential sampling scheme as follows. Suppose that we are on day  $t$  with samples available from the *baseline* distribution in the sense of Definition 3.1,

$$\Theta^{(r)} \sim p_t(\cdot | \delta_{t,0}), \quad (r = 1, \dots, R)$$



where  $R$  denotes the number of posterior draws. Using these samples, we aim to generate draws along the perturbation path by incrementally increasing  $\ell \in (0, 1]$ , eventually reaching the fully modulated  $p_t(\cdot | \delta_{t,\ell=1})$ .

To this end, we explicitly prepare the mesh,

$$0 = u_{\ell=0} < u_{\ell=1} < \dots < u_{\ell=L} = 1,$$

and exploit the fact that the space  $\Theta$  is shared along the path; we can exploit the sequential proximity of neighboring configurations using SMC (Del Moral et al., 2006). Specifically, consider traversing the joint space

$$(\Theta_0, \Theta_1, \dots, \Theta_L) \equiv (\Theta | \delta_{t,u_0}, \Theta | \delta_{t,u_1}, \dots, \Theta | \delta_{t,u_L}),$$

where the  $t$ -dependence on the left-hand side is omitted for brevity. Introduce a backward Markov transition kernel  $\mathcal{L}_\ell$  over neighboring perturbed configuration-induced parameters  $(\Theta_\ell, \Theta_{\ell-1})$  for  $\ell = 1, \dots, L$ , such that

$$\tilde{p}_t(\Theta_0, \Theta_1, \dots, \Theta_L) := p_t(\Theta_L | \delta_{t,1}) \prod_{\ell=1}^L \mathcal{L}_\ell(\Theta_{\ell-1} | \Theta_\ell).$$

Then select the forward kernel  $\mathcal{M}_\ell$  to leave  $p_t(\cdot | \delta_{t,u_\ell})$  invariant (e.g., MCMC kernels); the *importance distribution* is

$$\tilde{q}_t(\Theta_0, \Theta_1, \dots, \Theta_L) := p_t(\Theta_0 | \delta_{t,0}) \prod_{\ell=1}^L \mathcal{M}_\ell(\Theta_\ell | \Theta_{\ell-1}).$$

Given  $\mathcal{M}_\ell$ , further specify the backward kernel  $\mathcal{L}_\ell$  as an  $\ell$ -reversal of  $\mathcal{M}_\ell$ , that is,

$$\mathcal{L}_\ell(\Theta_{\ell-1} | \Theta_\ell) := \mathcal{M}_\ell(\Theta_\ell | \Theta_{\ell-1}) \cdot \frac{p_t(\Theta_{\ell-1} | \delta_{t,u_\ell})}{p_t(\Theta_\ell | \delta_{t,u_\ell})} = \mathcal{M}_\ell(\Theta_\ell | \Theta_{\ell-1}) \cdot \frac{p_t(\Theta_{\ell-1}, \mathbf{y}_{1:N_t} | \delta_{t,u_\ell})}{p_t(\Theta_\ell, \mathbf{y}_{1:N_t} | \delta_{t,u_\ell})}.$$

With these choices, it follows that

$$\mathbb{E}_{\tilde{p}_t}(f(\Theta_\ell)) = \mathbb{E}_{\tilde{q}_t}\left(f(\Theta_\ell) \cdot \frac{\tilde{p}_t}{\tilde{q}_t}(\Theta_0, \Theta_1, \dots, \Theta_L)\right).$$

Here, the term  $\tilde{p}_t/\tilde{q}_t$  yields an expression for the *incremental weights* (Dai et al., 2022) within the multiplicand in terms of the perturbations (3.1),

$$\begin{aligned} \frac{\tilde{p}_t}{\tilde{q}_t}(\Theta_0, \Theta_1, \dots, \Theta_L) &\propto \prod_{\ell=1}^L \frac{p_t(\Theta_\ell, \mathbf{y}_{1:N_t} | \delta_{t,u_\ell})}{p_t(\Theta_{\ell-1}, \mathbf{y}_{1:N_t} | \delta_{t,u_{\ell-1}})} \cdot \frac{\mathcal{L}_\ell(\Theta_{\ell-1} | \Theta_\ell)}{\mathcal{M}_\ell(\Theta_\ell | \Theta_{\ell-1})} \\ &= \prod_{\ell=1}^L \frac{p_t(\Theta_{\ell-1}, \mathbf{y}_{1:N_t} | \delta_{t,u_\ell})}{p_t(\Theta_{\ell-1}, \mathbf{y}_{1:N_t} | \delta_{t,u_{\ell-1}})} = \prod_{\ell=1}^L \frac{h_{t,\ell}(\Theta_{\ell-1})}{h_{t,\ell-1}(\Theta_{\ell-1})}. \end{aligned} \quad (2)$$

The incremental weights at the  $\ell$ -th step  $(h_{t,\ell}/h_{t,\ell-1})(\Theta_{\ell-1})$  do not depend on  $\Theta_\ell$  but on  $\Theta_{\ell-1}$ ; it consequently becomes feasible to sample from  $\tilde{p}_t(\cdot)$  via the following sequential

scheme. At the  $\ell$ -th step, we first (essentially) perform importance resampling of  $\Theta_\ell$  using the previous step  $(\Theta_{\ell-1}^{(r)})_r$ , with normalized weights and empirical approximation,

$$W_\ell^{(r)} \propto \frac{h_{t,\ell}(\Theta_{\ell-1}^{(r)})}{h_{t,\ell-1}(\Theta_{\ell-1}^{(r)})}, \quad \widehat{\Theta}_\ell^{(r)} \stackrel{\text{ind}}{\sim} \hat{p}_t(\cdot | \delta_{t,u_\ell}) := \sum_{r=1}^R W_\ell^{(r)} \delta_{\Theta_{\ell-1}^{(r)}}(\cdot).$$

Then, we propagate forward (i.e., *rejuvenate*) via

$$\Theta_\ell^{(r)} \sim \mathcal{M}_\ell(\cdot | \widehat{\Theta}_\ell^{(r)}),$$

noting that  $\mathcal{M}_\ell$  targets  $p_t(\cdot | \delta_{t,u_\ell})$ . In our setup (and often in practice), this is executable because the ability to draw from  $(\Theta | \delta_{t,u_{\ell-1}})$  often directly implies the ability to draw from  $(\Theta | \delta_{t,u_\ell})$ , typically using the same sampler simply and manually varying the configurations to  $\delta_{t,u_\ell}$  as prescribed. In our setup, we use Hamiltonian Monte Carlo. However, importantly, the approach differs from brute-force MCMC in that the rejuvenation step is only triggered when the weights  $(W_\ell^{(1)}, \dots, W_\ell^{(R)})$  exhibit insufficient sample diversity, which we measure via the approximate effective sample size of [Kong et al. \(1994\)](#) and the generalized Pareto shape diagnostic ([Vehtari et al., 2024](#)) (see Section 3.3.3).

We thus arrive at a sampling-based approximation scheme along the perturbation path, which leverages the sequential proximity of adjacent perturbations along the path rather than re-approximating each via MCMC. We summarize the above in Algorithm 1.

---

**Algorithm 1:** SMC for scenario/sensitivity analysis

---

**Input:** MCMC draws  $\Theta^{(1)}, \dots, \Theta^{(R)} \sim p_t(\cdot | \delta_{t,0})$ , Perturbation schedule  $(h_{t,\ell}(\cdot))_{\ell=1}^L$ , ESS threshold  $\tau \in (0, N)$ .

Initialize particles  $(\Theta_0^{(r)}, W_0^{(r)}) \leftarrow (\Theta^{(r)}, 1/R)$  ;

**for**  $\ell = 1, \dots, L$  **do**

    Compute  $\widehat{W}_\ell^{(r)} \leftarrow h_{t,\ell}(\Theta_{\ell-1}^{(r)})/h_{t,\ell-1}(\Theta_{\ell-1}^{(r)})$  ;

    Compute  $W_\ell^{(r)} \propto W_{\ell-1}^{(r)} \widehat{W}_\ell^{(r)}$  ;

**if**  $\text{ESS}(W_\ell^{(1)}, \dots, W_\ell^{(R)}) < \tau$  **then**

$A_\ell^{(r)} \sim \text{MULTINOMIAL}(R; W_\ell^{(1)}, \dots, W_\ell^{(R)})$  ;

// Resampling

$\Theta_\ell^{(r)} \sim \mathcal{M}_\ell(\cdot | \Theta_{\ell-1}^{A_\ell^{(r)}})$  ;

// MCMC kernel

$W_\ell^{(r)} \propto 1$  ;

**else**

$\Theta_\ell^{(r)} \leftarrow \Theta_{\ell-1}^{(r)}$  ;

// Delta kernel

$\hat{k}_\ell \leftarrow \text{ParetoSmooth}(\widehat{W}_\ell^{(1)}, \dots, \widehat{W}_\ell^{(R)})$  ;

// Diagnostic

**if**  $\hat{k}_\ell > 0.7$  **then**

            Output warning

**end**

$\hat{f}_\ell \leftarrow \sum_{n=1}^N W_\ell^{(r)} f(\Theta_\ell^{(r)})$  ;

**end**

**return** Sequential sampling result  $(\hat{f}_\ell, (W_\ell^{(r)})_r, (\Theta_\ell^{(r)})_r)_{\ell=1}^L$  ;

---

## 3.3 Practical implementation notes

### 3.3.1 An example

Our implementation is carried out in Julia. In brief, the workflow proceeds as follows.

- (a) Specify the Bayesian model using Stan (Carpenter et al., 2017), as in the forecasting workflows of Heidemanns et al. (2020) and Gelman et al. (2024); see also Morris (2020).
- (b) Define a sequence of Stan meta-models.
- (c) Traverse the meta-models sequentially by accessing the model's log density and its gradient via BridgeStan (Roualdes et al., 2023).

The workflow is modular. In (a), the user first obtains the baseline draw. In (b), for example, varying a hyperparameter (3.1) involves a user-specified change to the data block to encode how it should vary. The following is illustrative code.

```
1 model_0 = StanModel("model.stan", "data.json") # baseline model
2 data_0 = load_json("data.json") # baseline configuration
3
4 for (i, scalar) in enumerate([1.1, 1.2, 1.3])
5     data_1 = data_0 |> deepcopy # inherit original configurations
6     data_1["mu_b_T_scale"] *= scalar # new configuration
7     # more can be specified by user
8
9     save_as_json(data_1, "perturbed_data_${i}.json") # save
10    model_1 = StanModel("model.stan", "perturbed_data_${i}.json") # target
11 end
```

In (c), the ability to evaluate log density and its gradient satisfies the requirements of SMC without requiring code rewrites for each perturbation scheme. Sequential sampling is then executed as follows.

```
1 SMCS_Stan(
2     draws_0, # baseline draw
3     models_vec |> return_LogDensities # sequence of meta-models
4 );
```

```
[ Info: Delta Kernel (1/30; ESS=873/1000)
[ Info: Delta Kernel (2/30; ESS=609/1000)
[ Info: MCMC Kernel (3/30; ESS=378/1000)
Progress: 35%|██████████|
```

Figure 2: Illustrative intermediate output. Given a sequence of meta-models and access to log density, Markov kernel selection and diagnostics (e.g., ESS and  $\hat{k}$ ) are automatically performed as described in Section 3.2 to ensure sufficient sample diversity. In the above, rejuvenation was triggered due to  $ESS < 500 = 0.5(1000)$  at step #3 of 30.

### 3.3.2 Parallelization

On any given day, distinct and branching perturbative sequential updates (e.g., in **Horizontal**:  $p_t^*$  and  $p_t^{**}$  emerge as branches from  $p_t$  *between the arrows*), are inherently parallelizable, provided sufficient distributed computational resources. Another is *within the arrows*.

When triggered, the MCMC kernel is applied independently per particle for rejuvenation (i.e., short-run many-chain). In principle, assuming adequate resources, the rejuvenating computation can be fully distributed across particles, so that we can reduce the effective cost to that of a single-short-chain MCMC. We actively leverage the parallelizability in our implementation (and recommend doing so in other applicable contexts similarly).

### 3.3.3 Failure diagnostics

The *fail fast* principle (Gelman et al., 2020) extends naturally to the design of perturbation schemes here (beyond model fitting); it is critical to diagnose when a given approximation may be unreliable, so that computational effort is not wasted on perturbations that will ultimately be discarded.

For example, consider the illustrative but deliberately extreme one-step scheme

$$h_{t,\ell=1}(\Theta) = \text{binomial}(y^* = 89,999 \mid n = 90,000, p_i),$$

which corresponds to a hypothetical poll with sample size  $n = 90,000$  and two-party support  $y^* = 89,999$ . The influential data would most definitely induce substantial posterior shifts, and any finite weighted set of baseline draws would likely fail to adequately approximate the updated posterior.

A simple remedy is then to incrementally approach the target (e.g., with a power scaling coefficient for gradual data injection), alongside the safe strategy to preventively prepare a sufficiently fine mesh, since Algorithm 1 avoids the costly MCMC kernels wherever possible. However, determining whether the mesh is fine enough may not be obvious *a priori*. Given the incremental structure of importance weights in Equation (2) across mesh transitions, we implement the generalized Pareto shape diagnostic (Vehtari et al., 2024) on the incremental component. Diagnostic warnings indicated by  $\hat{k} > 0.7$  would signal the need to prepare finer mesh points to ensure a reliable approximation.

## 3.4 Existing approach and potential use cases

We finally acknowledge some connections to existing approaches. The perturbation view (in Section 3.1), and the choice of target function  $f$  (in Section 3.2) provide a unification to re-interpret and extend prior methods.

**Data insertion.** Insertion is classically used in particle-based state-space inference (e.g., bootstrap particle filters: Gordon et al., 1993; Kitagawa, 1996) for online latent state tracking but without retrospective refresh. Iterated batch importance sampling (Chopin,

2002) targets static parameters. More recently, [Fong and Holmes \(2021\)](#) applied mesh-based insertion for conformal uncertainty quantification in exchangeable Bayesian models, using add-one-in perturbation  $h(\Theta) = p(\mathbf{y}_{N+1} = \mathbf{y}^* | \Theta, \mathbf{y}_{1:N})$  and target as conformity score via posterior predictive  $p(\mathbf{y}_i | \mathbf{y}_{1:(N+1)})$ . This is approximated by weighted pre-perturbed posterior draws  $\Theta^{(r)} \sim p(\cdot | \mathbf{y}_{1:N})$  with  $W^{(r)} \propto h(\Theta^{(r)})$ . Although extreme  $\mathbf{y}^*$  may be avoided under a large miscoverage level  $\alpha$ , our framework can allow this, such as with sequential mesh-based approximation  $h_\ell(\Theta) = p(\mathbf{y}_{N+1} = \mathbf{y}_\ell^* | \Theta, \mathbf{y}_{1:N})$  by rejuvenating when diagnosed as necessary.

**Data deletion.** Although not the focus of our application, deletion also arises naturally in approximate leave-one-out cross-validation and variants (e.g., [Vehtari et al., 2017](#); [Han and Gelman, 2025](#)). These approaches ultimately remove data units by essentially modulating informativity  $\gamma_i$ . Typically, intermediate distributions are treated as computational scaffolding but not of direct inferential interest ([Han and Gelman, 2025](#)). In our application, we are also interested in these intermediates, as the variation in configurations is informed by the user’s qualitative query itself.

**Scenario analysis.** [Bornn et al. \(2010\)](#) examine the so-called (Bayesian) Lasso path, which can be viewed via perturbation functions of the form  $h_\ell(\Theta) = p_{\phi(\ell)}(\theta_k) \cdot p_{\phi_k}(\theta_k)^{-1}$  in [Example 3.1](#); the target and modulated quantity is the regression coefficient  $f(\Theta) = \beta$ , and the path is traced over different scale hyperparameters.

The meta-modeling lens would be relevant for scenario analysis outside of our specific application of real-time Bayesian election forecasting, such as in Bayesian macroeconomic forecasting. Conditional forecasts are often framed around counterfactuals (e.g., what would happen to the system if some variable follows a specified path?: [Waggoner and Zha, 1999](#)). [McCracken et al. \(2021\)](#) present a setting in which a low-frequency outcome is forecast conditional on a hypothetical path for a higher-frequency input; as new high-frequency data arrive, the forecast is routinely revised, which mirrors the sequential update we perform.

**Sensitivity.** From a procedural point of view, local sensitivity methods (e.g., [Giordano et al., 2018](#); [Giordano, 2018](#); [Kallioinen et al., 2023](#)) examine how posterior expectations shift under infinitesimal perturbations to hyperparameters  $(\phi_k)_k$  or power coefficients. These can be viewed as special cases of our perturbation framework, with  $h_\ell(\Theta) = \prod_k p_{\phi_k}(\theta_k)^{\rho_k(\ell)-1}$  encoding a one-step prior replacement, and the target function is a chosen sensitivity metric, such as cumulative Jensen–Shannon divergence or empirical covariance ([Giordano et al., 2018](#), Appendix B).

## 4 Backtesting the forecasting model

We now turn to evaluation and demonstration of the proposed approach in the context of real-time Bayesian election forecasting. We focus on qualitatively backtesting *The Economist* model in [Morris \(2020\)](#) by hypothetically placing ourselves in the 2016 cycle 90 days out; this mirrors the diagnostic workflow we employed in 2024 to better understand the model ([Gelman et al., 2024](#)).

### 4.1 Simulated validation

We begin with a simulated evaluation of the accuracy and computational efficiency. There is no shortage of choices for perturbations; we focus on the following examples.

- (a) Sensitivity of state-level forecasts to both the level and our confidence in fundamentals-based forecasts;
- (b) A what-if analysis on the variability of state-by-state random walks governed by shared hyperparameter `random_walk_scale`;
- (c) Sequential updating of the posterior as new and/or hypothetical data arrive, to assess forecast responsiveness to individual polls.

Across all scenarios, we evaluated the accuracy of the approach by comparing to MCMC draws as reference, each obtained independently from the corresponding perturbed posterior (i.e., brute force). Run time is also recorded for comparison. For each case, SMC is run with 30 discrete mesh points based on the absence of warning outputs (see Section 3.3.3). Moderate parallelization (4 threads) is used during particle rejuvenation (see Section 3.3.2). The MCMC kernel (chosen as Hamiltonian Monte Carlo) is applied 3 times, informed by baseline MCMC draw autocorrelation following [Han and Gelman \(2025\)](#). For the reference MCMC, we run 1200 iterations with a burn-in of 200. The 1000 posterior draws from  $p_t$  at time  $t$  (as shown in [Horizontal](#)) serve as input to SMC for all subsequent analyses. The ESS threshold is  $1/2$ , which is typical.

Figure 3 presents the posterior mean estimates for the 13829 dimensional parameter vector in the final perturbation step (i.e., the most distant posterior) in the fully unconstrained space. All parameters are in the non-centered and unconstrained parameterization of standard normal priors. The results show a minimal deviation between the two approaches in their respective perturbation schemes.

Figure 4 compares the run time per perturbation step (i.e., per mesh point) and cumulative (i.e., the total time to approximate the full modulation scheme). As for the former, sequential sampling is consistently faster, especially in cases where particle rejuvenation is not triggered, as indicated by the concentration of points near the horizontal axis; reweighting via density/likelihood evaluation is inexpensive relative to generating a new set of draws based on multiple gradient computations. The cumulative run time is also lower under equal and moderate parallelization (4 threads in our setup). For reference,



## Last step posterior means (unconstrained)

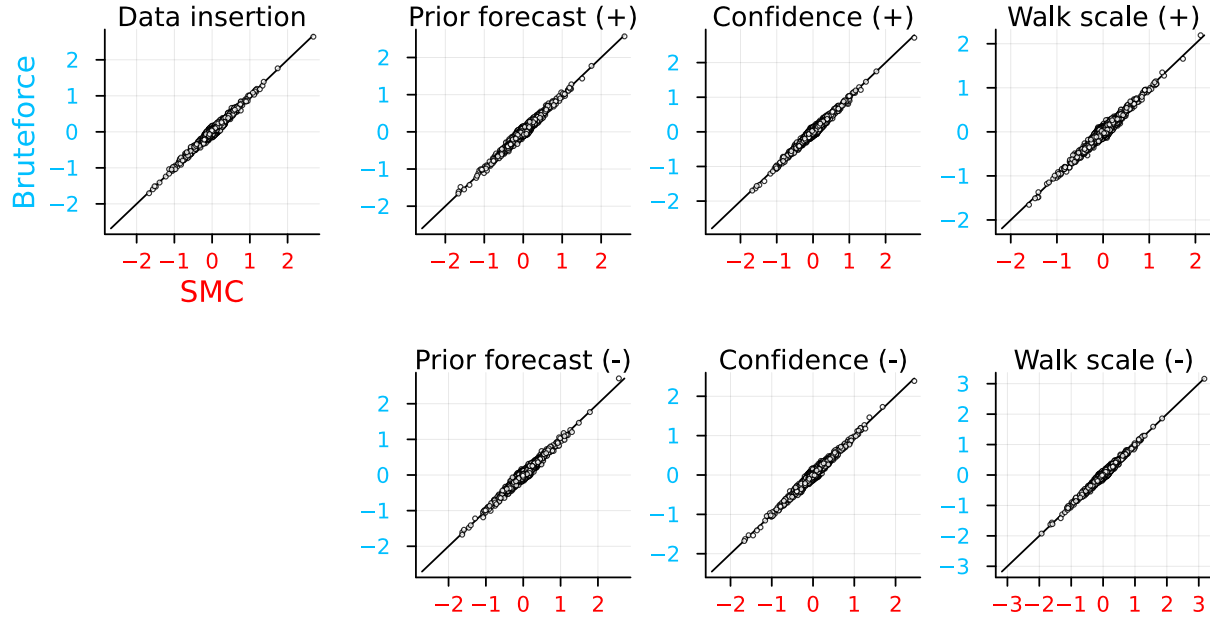


Figure 3: Comparison between brute force MCMC and sequential sampling (SMC). Each point represents a posterior mean for one of the 13,892 non-centered parameters in the unconstrained space. From left to right, perturbations are: (i) data insertion (4.4); (ii) prior forecast (modulation of fundamentals-based inputs, with  $+/-$  indicating upward/downward shifts: 4.2); (iii) relative confidence adjustment (via scale parameter modulation where  $+/-$  indicates scale up/down: 4.2); and (iv) trend variability ( $+/-$  indicates scale up/down of  $random\_walk\_scale$ : 4.3).

we also report the hypothetical minimum number of parallel machines required (up to the mesh count of 30) using brute force MCMC to match those of sequential sampling. Depending on the perturbation scheme, achieving comparable performance would require much greater computational resources.

## 4.2 Vertical chaining: what-if posterior analyses

The extent to which the economic and political indicators, encoded in the fundamentals-based forecast  $f$ , shape the final forecast is not immediately clear. In states with abundant polling, such as California, one might expect the prior  $\mu_T \sim \text{MVN}(f, C)$  to exert limited influence. However, this deserves scrutiny, given the uneven distribution of polling across states, the dense covariance structure  $C$ , and the model’s hierarchical structure, which allows information to propagate in nontrivial ways.

Figure 5a shows a perturbation scheme in which the fundamentals-based forecast for  $s = \text{California}$  is gradually shifted up and down, as defined in Equation 1 in Example 3.1. In California, the posterior responds visibly to the perturbation, which is expected. In contrast, Arkansas and Ohio (two selected for visibility) show weaker propagation. This

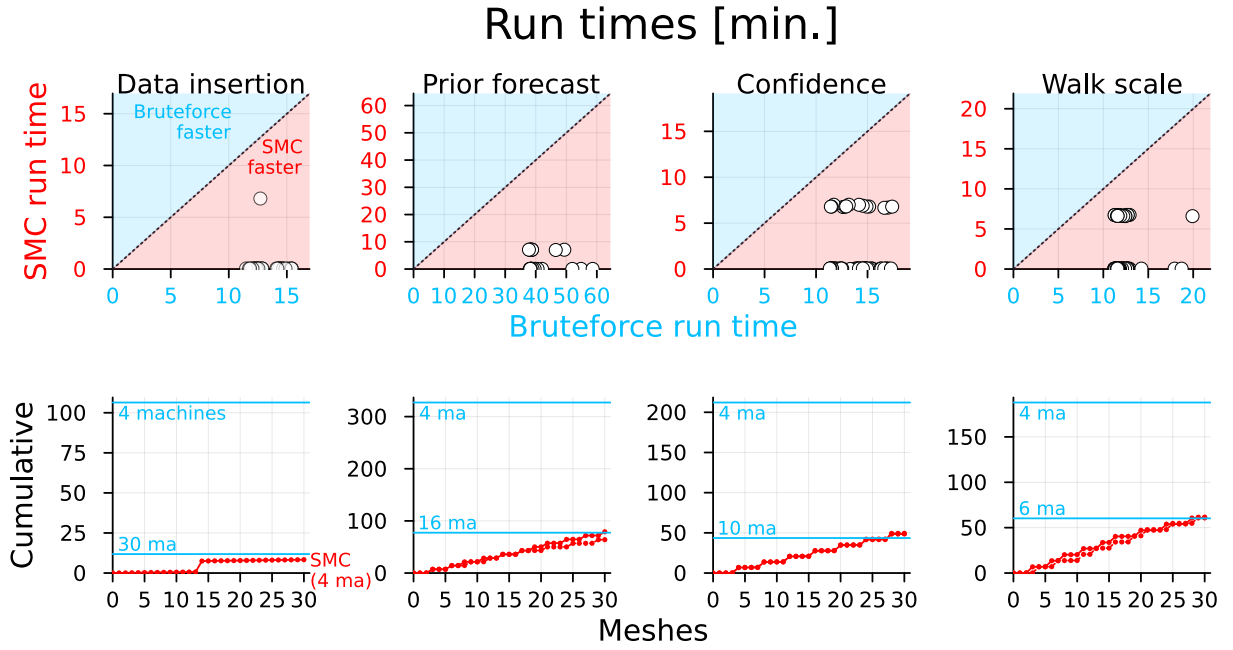


Figure 4: Run time comparison between brute force MCMC and sequential sampling (SMC). **Top:** Comparison per perturbation step. **Bottom:** Cumulative run time across all 30 steps. For the sequential approach, two curves are shown for bidirectional perturbations (forward and reverse); see Figure 3 caption for succinct descriptions. Moderate parallelization (4 threads) is used. The maximum number of parallel chains required for brute force MCMC to outperform the sequential approach is also shown.

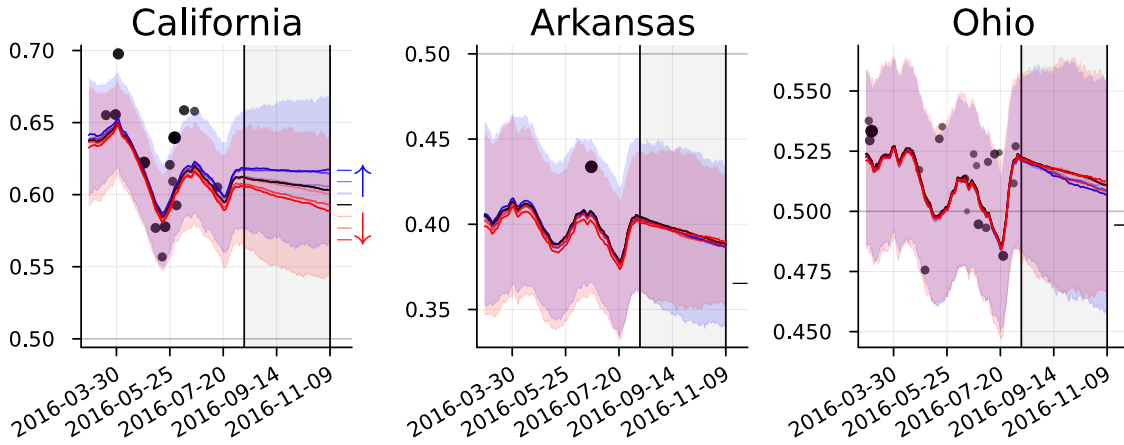
is consistent with Ohio, where polling is more frequent, but in some ways unexpected in Arkansas, which has only a single state-level poll at 90 days out, yet the sensitivity remains limited. The qualitative analysis suggests that other factors (e.g., low correlation in  $C$  for these states) likely contribute to the muted effect.

Next we modulate the marginal scale of the prior using

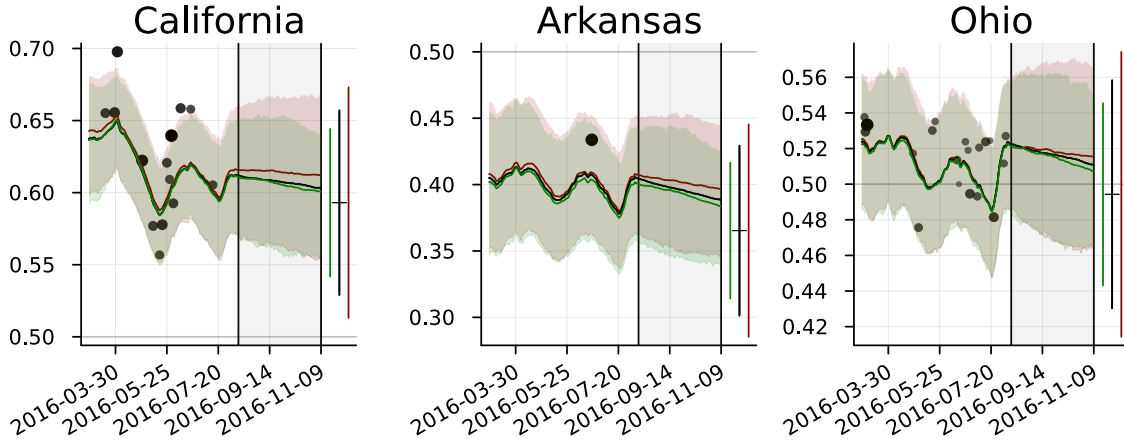
$$h_{t,\ell}(\Theta) = \text{MVN}(\mu_T | \mathbf{m}, a_\ell \mathbf{V}) \cdot \text{MVN}(\mu_T | \mathbf{m}, \mathbf{V})^{-1},$$

where  $a_\ell$  controls the scale, shared across states; we vary  $a_\ell$  from 1 to 1.25 (and in the other direction via the reciprocal).

The results for the three states are shown in Figure 5b, and for all states in Figure 6a. In Arkansas, where polling is sparse, the posterior mean shifts upward as the prior becomes more diffuse. In contrast, states such as California and Ohio show more localized shifts, mostly in the days after *today* at 90 days out, which is consistent with the abundant polls. The interpretability is worth noting; the perturbation targets a specific model quantity and the resulting posterior shifts are directly observable. This transparency is valuable for communication.



(a) Modulating the prior forecast



(b) Modulating the relative confidence

Figure 5: **Left:** Modulating the fundamentals-based prior forecast for California; upward shifts correspond to increased support for the Democratic candidate. **Middle and right:** Resulting marginal posterior forecast adjustments in two other states selected for visibility. **Black dots:** Individual polls with size proportional to sample size. **Time-varying black line:** Posterior mean. **Shaded region:** 90% credible interval. **Vertical black line:** The hypothetical today (at 90 days out).

### 4.3 Sensitivity analyses

It is helpful to understand how the model balances polling data and fundamentals-based forecasts. Given existing concerns surrounding polling bias (e.g., [Gelman and Azari, 2017](#)), it is useful to be able to identify which components of the model can effectively adjust relative confidence between these sources and concretely quantify their influence on the final inference.

Figure 6a shows the effect of varying the scale of the fundamentals-based prior (as in the previous Section 4.2), and Figure 6b presents results from modulating the random

walk variability,

$$h_\ell(\Theta) = \prod_{t=1}^{T-1} \text{MVN}(\mu_t | \mu_{t+1}, a_\ell \Sigma^{(\mu)}) \cdot \text{MVN}(\mu_T | \mu_{T+1}, \Sigma^{(\mu)})^{-1}.$$

In general, the model tends to overshoot (i.e., favor the Democratic candidate) relative to the fundamentals-based forecast. Increasing prior confidence pulls the forecast toward the fundamentals (e.g., Utah showing the clearest such shift). Effects are visible across states but less pronounced in poll-rich states such as California and Ohio. Modulating the random walk scale yields more mixed results.

#### 4.4 Horizontal chaining: sequential updating

We are working with sequential Bayesian posteriors that should be updated both retrospectively and prospectively as new data arrive. One useful qualitative diagnostic is to assess how sensitive the forecasts are to individual polls.

Figure 7 illustrates a hypothetical data insertion (see Example 3.1); consider a new poll conducted in Pennsylvania (state) at  $t = 80 = 90 - 10$  days out, administered by NBC (pollster) using live phone interviews (mode) with registered voters (population). These are freely specifiable by the user. To examine the effect of poll sample size, we gradually shift the observed two-party Democratic support from 10/20 to 300/600 ( $y_t$ ).

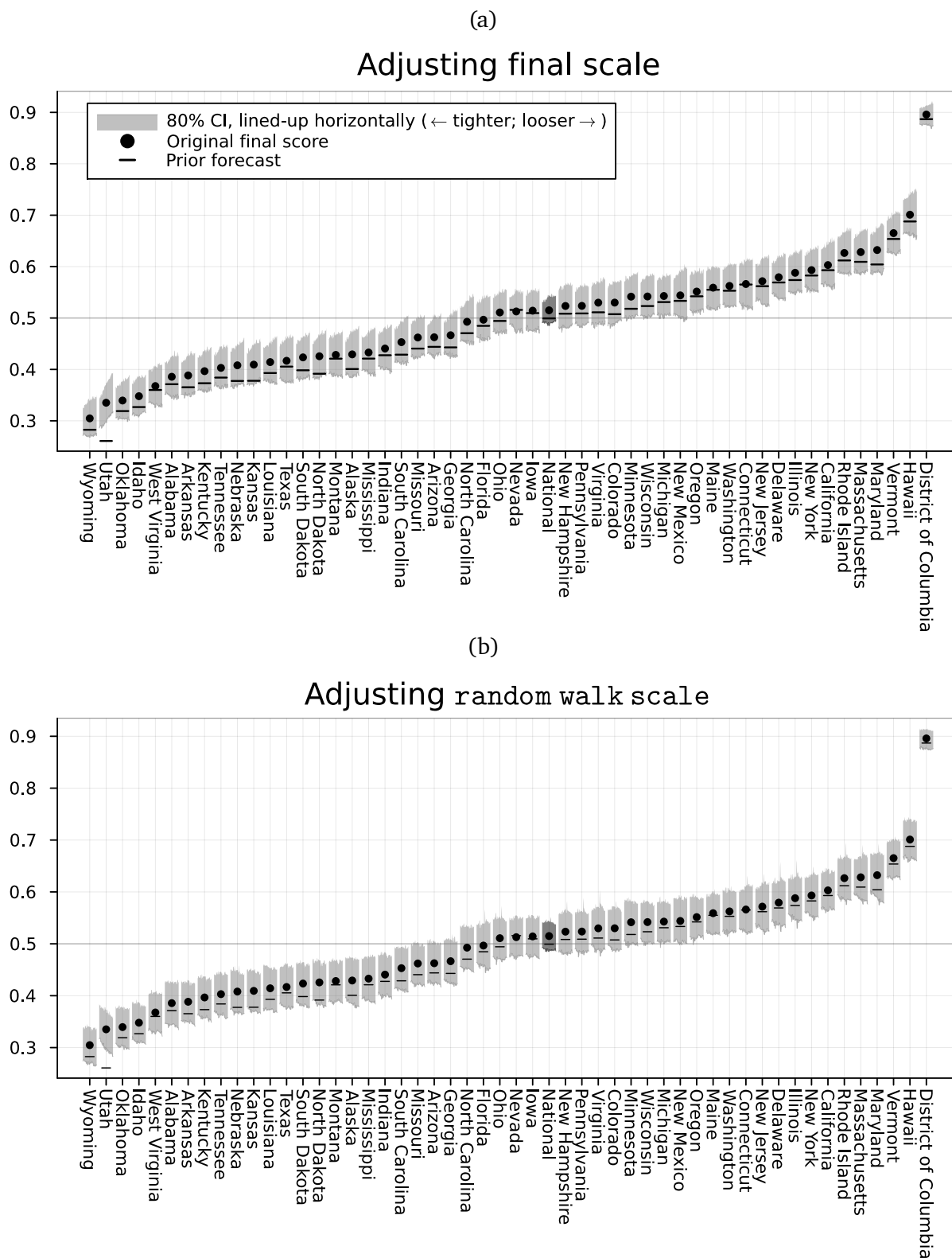
On the one hand, the resulting behavior is intuitive; larger sample sizes exert greater influence on the projected latent trend. On the other hand, the effect is already notable at 100/200, which is modest compared to other polls conducted in Pennsylvania. This highlights how the model can be highly responsive to individual polls, even in poll-abundant states. The scheme enables us to quantify how changes occur as a function of poll characteristics.

## 5 Discussion

Real-time Bayesian dynamic election forecasting presents a unique methodological and practical challenge at the intersection of daily posterior updates, time-specific scenario and sensitivity analysis, and collaborative communication. The core issue is that the model is never truly complete; rather, it is a living object, revised routinely as new data and events arrive. To move toward a model that is reliable, transparent, interpretable, and useful in real-time, we sought a framework that accommodates timely updates and qualitative diagnostic queries.

We developed a meta-modeling framework, paired with a sequential sampling scheme, to enable efficient model and forecast revision in real-time, dynamic forecasting environments where data and assumptions evolve continuously. The former meta-modeling component first generalizes the original forecasting model by explicitly operating over fixed model quantities (e.g., hyperparameters) to enable interpretable scenario and sensitivity

Figure 6: Results of sensitivity analysis. Within each state, perturbations range from tighter (left) to looser (right). Original final score (posterior mean) and prior forecast are also shown for interpretation in reference to these quantities. Upper levels indicate stronger support for the Democratic candidate.



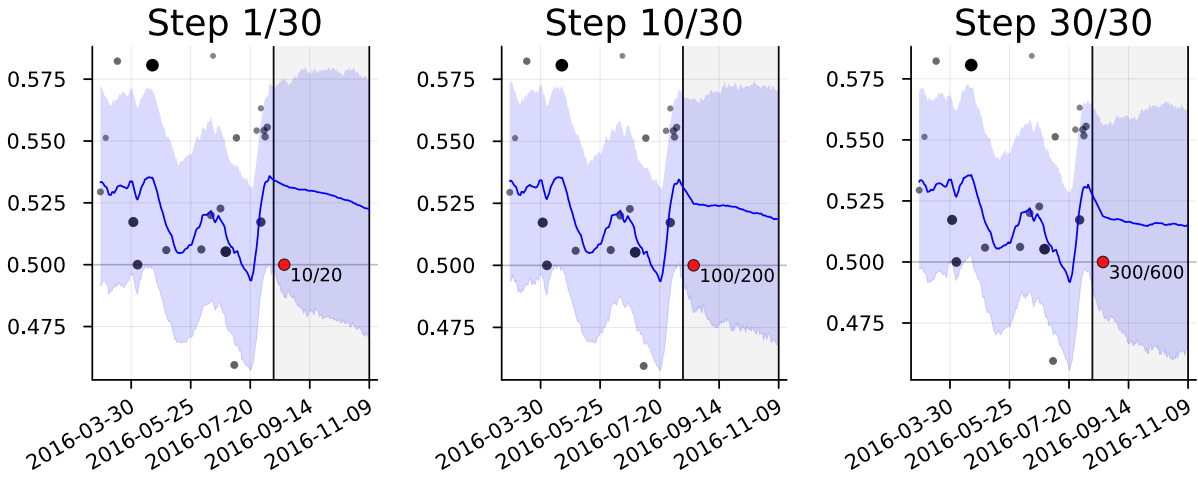


Figure 7: Results from gradual hypothetical data insertion (**red**): in Pennsylvania (*state*) at  $80 = 90 - 10$  days out ( $t$ ), shifting from 10/20 to 300/600 two-party Democratic support ( $y_i/n_i$ ), conducted by NBC (pollster) using live phone interviews (*mode*) with registered voters (*population*). Upper levels indicate stronger support for the Democratic candidate.

analyses. The latter sequential sampling scheme addresses the computational inefficiencies of brute force MCMC (i.e., the need to re-fit the model for each perturbed posterior) by taking advantage of the sequential proximity of posterior distributions along a perturbation path.

Our case study revealed several nontrivial insights. For example, the model exhibits sensitivity to the prior specification involving fundamentals-based forecasts, by design, but still relies heavily on individual polling data in its final projections. The latent trend appears to be predominantly poll-driven and tends to overshoot relative to fundamentals-based projections. Given the documented prevalence of polling biases, we demonstrate that modulating the prior scale is an effective mechanism for adjusting confidence on fundamentals-based forecasts than alternative strategies such as regularizing trend variability. Moreover, we show that the sequential sampling scheme enables such analyses at a fraction of the computational cost compared to brute force MCMC, while yielding posterior approximations that are nearly indistinguishable. Finally, we emphasize how the approach is interpretable and well-suited for communication.

We conclude by pointing to several directions for future work. Although our case study focused on real-time Bayesian election forecasting, the framework is readily extensible to other domains involving sequential revision. For example, macroeconomic monitoring often features asynchronous data arrivals and frequent updates to select indicators (e.g. [McCracken et al., 2021](#)), which makes the data-value perturbation and insertion strategies in Section 3.1 directly relevant. More broadly, in general statistical analyses, data-value perturbation may be used to encode aspects of human decision-making in the *analytical pipelines*, such as data preparation, to gauge the stability of (posterior) inferences ([Rewolin-ski and Yu, 2025](#)). On the methodological side, while our use of vertical branching updates



(see Section 3.2) was applied to model quantities specific to our forecasting task, the same approach could be useful for exploring broader model changes in Bayesian settings: e.g., inclusion of interaction terms in hierarchical regression. Such a step-by-step exploration of model structure, combined with fast sequential simulation-based inference, holds promise for building flexible, efficient, and interpretable Bayesian workflows.

## Acknowledgments

We thank Ben Goodrich and Dan Rosenheck for collaboration and the Office of Naval Research for partial support of this work.

## Data Availability Statement

The data and accompanying code are available at <https://github.com/geonhee619/SMC-Sense>. These data were derived from <https://github.com/TheEconomist/us-potus-model>.

## References

- Bartels, L. and Zaller, J. (2001). Presidential vote models: A recount. *PS: Political Science and Politics*, 34:9–20. 2
- Belsley, D. A., Kuh, E., and Welsh, R. E. (1980). *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. Wiley, New York. 2
- Berger, J. O., Insua, D. R., and Ruggeri, F. (2000). Bayesian robustness. In Insua, D. R. and Ruggeri, F., editors, *Robust Bayesian Analysis*, pages 1–32. Springer, New York. 2
- Bornn, L., Doucet, A., and Gottardo, R. (2010). An efficient computational approach for prior sensitivity analysis and cross-validation. *Canadian Journal of Statistics*, 38(1):47–64. 4, 5, 15
- Campbell, J. E. (1992). Forecasting the presidential vote in the states. *American Journal of Political Science*, 36(2):386–407. 2
- Canavos, G. C. (1975). Bayesian estimation: A sensitivity analysis. *Naval Research Logistics Quarterly*, 22(3):543–552. 2
- Carnegie, N., Harada, M., and Hill, J. L. (2016). Assessing sensitivity to unmeasured confounding using a simulated potential confounder. *Journal of Research on Educational Effectiveness*, 9:395–420. 2

- Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32. 2, 13
- Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, 89(3):539–551. 14
- Clarke, B. and Gustafson, P. (1998). On the overall sensitivity of the posterior distribution to its inputs. *Journal of Statistical Planning and Inference*, 71(1):137–150. 2
- Cook, R. D. (1977). Detection of influential observations in linear regression. *Technometrics*, 19:15–18. 2
- Cook, R. D. (1979). Influential observations in linear regression. *Journal of the American Statistical Association*, 74:169–174. 2
- Dai, C., Heng, J., Jacob, P. E., and Whiteley, N. (2022). An invitation to sequential Monte Carlo samplers. *Journal of the American Statistical Association*, 117(539):1587–1600. 11
- Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society, Series B*, 68(3):411–436. 4, 11
- Epifani, I., MacEachern, S. N., and Peruggia, M. (2008). Case-deletion importance sampling estimators: Central limit theorems and related results. *Electronic Journal of Statistics*, 2. 4, 5
- Erikson, R. S. and Wlezien, C. (2008). Leading economic indicators, the polls, and the presidential vote. *PS: Political Science and Politics*, 41(4):703–707. 2
- Eubank, R. L. (1984). The hat matrix for smoothing splines. *Statistics and Probability Letters*, 2:9–14. 2
- Fong, E. and Holmes, C. C. (2021). Conformal Bayesian computation. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*. 15
- Gelfand, A. E. and Dey, D. K. (1994). Bayesian model choice: Asymptotics and exact ealculations. *Journal of the Royal Statistical Society. Series B*, 56(3):501–514. 4, 5
- Gelman, A. and Azari, J. (2017). 19 things we learned from the 2016 election. *Statistics and Public Policy*, 4(1):1–10. 19
- Gelman, A., Goodrich, B., and Han, G. (2024). Grappling with uncertainty in forecasting the 2024 U.S. presidential election. *Harvard Data Science Review*, 6(4). <https://hdsr.mitpress.mit.edu/pub/yoa73r1m>. 2, 4, 6, 7, 13, 16

- Gelman, A. and King, G. (1993). Why are American presidential election campaign polls so variable when votes are so predictable? *British Journal of Political Science*, 23(4):409—451. 2
- Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., Kennedy, L., Gabry, J., Bürkner, P.-C., and Modrák, M. (2020). Bayesian workflow. <https://arxiv.org/abs/2011.01808>. 2, 14
- Giordano, R. (2018). StanSensitivity. <https://www.github.com/rgiordan/StanSensitivity>. 4, 5, 15
- Giordano, R., Broderick, T., and Jordan, M. I. (2018). Covariances, robustness, and variational Bayes. *Journal of Machine Learning Research*, 19(51):1–49. 15
- Gordon, N., Salmond, D., and Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F (Radar and Signal Processing)*, 140:107–113. 14
- Gustafson, P. (1996). Local sensitivity of inferences to prior marginals. *Journal of the American Statistical Association*, 91(434):774–781. 2
- Han, G. and Gelman, A. (2025). Adaptive sequential Monte Carlo for structured cross validation in Bayesian hierarchical models. 4, 5, 15, 16
- Heidemanns, M., Gelman, A., and Morris, G. E. (2020). An updated dynamic Bayesian forecasting model for the U.S. presidential election. *Harvard Data Science Review*, 2(4). <https://hdsr.mitpress.mit.edu/pub/nw1dzd02>. 2, 7, 13
- Holbrook, T. M. and DeSart, J. A. (1999). Using state polls to forecast presidential election outcomes in the American states. *International Journal of Forecasting*, 15(2):137–142. 2
- Huber, P. J. (1981). *Robust Statistics*. Wiley, New York. 2
- Kallioinen, N., Paananen, T., Bürkner, P.-C., and Vehtari, A. (2023). Detecting and diagnosing prior and likelihood sensitivity with power-scaling. *Statistics and Computing*, 34(1). 4, 5, 15
- Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25. 14
- Kong, A., Liu, J. S., and Wong, W. H. (1994). Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288. 12
- Linzer, D. A. (2013). Dynamic Bayesian forecasting of presidential elections in the states. *Journal of the American Statistical Association*, 108(501):124–134. 2
- Lock, K. and Gelman, A. (2010). Bayesian combination of state polls and election forecasts. *Political Analysis*, 18(3):337–348. 2

- Masoero, L., Stephenson, W. T., and Broderick, T. (2018). Sensitivity of Bayesian inference to data perturbations. In *1st Symposium on Advances in Approximate Bayesian Inference*. 2
- McCracken, M. W., Owyang, M. T., and Sekhposyan, T. (2021). Real-time forecasting and scenario analysis using a large mixed-frequency Bayesian VAR. *International Journal of Central Banking*, 17(71):1–41. 15, 22
- McDermott, M. L. and Frankovic, K. A. (2003). Review: Horserace polling and survey method effects: An analysis of the 2000 campaign. *Public Opinion Quarterly*, 67(2):244–264. 7
- Morris, G. E. (2020). Economist election model. <https://github.com/TheEconomist/us-potus-model/tree/master>. 2, 13, 16
- Nadeau, R. and Lewis-Beck, M. (2001). National economic voting in U.S. presidential elections. *Journal of Politics*, 63:159–181. 2
- Nguyen, T. D., Giordano, R., Meager, R., and Broderick, T. (2024). Sensitivity of MCMC-based analyses to small-data removal. <https://arxiv.org/abs/2408.07240>. 4, 5
- Peruggia, M. (1997). On the variability of case-deletion importance sampling weights in the Bayesian linear model. *Journal of the American Statistical Association*, 92(437):199–207. 4
- Rewolinski, Z. T. and Yu, B. (2025). PCS workflow for Veridical Data Science in the age of AI. 22
- Roos, M., Martins, T. G., Held, L., and Rue, H. (2015). Sensitivity analysis for Bayesian hierarchical models. *Bayesian Analysis*, 10(2):321–349. 2
- Rosenbaum, P. R. and Rubin, D. B. (1983). Assessing sensitivity to an unobserved binary covariate in an observational study with binary outcome. *Journal of the Royal Statistical Society, Series B*, 45:212–218. 2
- Ross, W. H. (1987). The geometry of case deletion and the assessment of influence in nonlinear regression. *Canadian Journal of Statistics*, 15:91–103. 2
- Roualdes, E. A., Ward, B., Carpenter, B., Seyboldt, A., and Axen, S. D. (2023). Bridgestan: Efficient in-memory access to the methods of a stan model. *Journal of Open Source Software*, 8(87):5236. 5, 13
- Vehtari, A., Gelman, A., and Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27:1413–1432. 4, 5, 15

- Vehtari, A., Simpson, D., Gelman, A., Yao, Y., and Gabry, J. (2024). Pareto smoothed importance sampling. *Journal of Machine Learning Research*, 25(72):1–58. 12, 14
- Waggoner, D. F. and Zha, T. (1999). Conditional forecasts in dynamic multivariate models. *The Review of Economics and Statistics*, 81(4):639–651. 15
- Weiss, R. (1996). An approach to Bayesian sensitivity analysis. *Journal of the Royal Statistical Society, Series B*, 58(4):739–750. 9
- Wlezien, C. and Erikson, R. S. (1996). Temporal horizons and presidential election forecasts. *American Politics Quarterly*, 24(4):492–505. 2
- Wlezien, C. and Erikson, R. S. (2006). The horse race: What polls reveal as the election campaign unfolds. *International Journal of Public Opinion Research*, 19(1):74–88. 7
- Zhu, H., Ibrahim, J. G., and Tang, N. (2011). Bayesian influence analysis: A geometric approach. *Biometrika*, 98(2):307–323. 2