# Recreating a Presidential Election Predictor

Sarah Hensley

***Abstract*—In this project, we recreate the certain results from *Dynamic Bayesian Forecasting of Presidential Elections in the States*. This model presents an easily interpretable example of a high-dimensional Bayesian model. We recreate two figures to demonstrate the match between our model and the model presented. Overall, the figures appear very similar, though slight differences appear, either due to randomness or missing poll information.**

## I. Introduction

Predicting the results of US presidential elections with polling data has long been a topic of interest. However, there are several problems. Data comes in the form of polls, which have inconsistent numbers of participants and are noisy, irregularly spaced in time, and irregularly spaced across states. Furthermore, the polls and final election results are correlated across states. In *Dynamic Bayesian Forecasting of Presidential Elections in the States*, Linzer proposes a model that uses polling results from all states to capture both state-level and national-level trends to predict the percent of the vote that Barack Obama would win in the 2008 presidential election. The parameter of interest estimates the percent of people in each state who would vote Democrat in each state on each day leading up to the election. This model is well-suited for studying Bayesian models and inference, as it is high-dimensional with several variables, requiring sophisticated techniques, yet has an understandable interpretation.

This project recreates certain results from this paper. Specifically, we focus on two figures. The first shows the mean percentage and Bayesian credible interval as a function of time for Florida and Indiana, the former of which has several polls and the latter of which has relatively few. This demonstrates the influence of the national trend parameter; it also demonstrates the match of the mean and variance between our implementation and the implementation presented in the paper. Second, we generate a figure that shows the predicted winner of each state at various times before the election, conveying the high-dimensional information in an understandable way. Both figures selected compare our implementation of the model with the one presented in the paper, with the first giving a detailed look at the posterior of a few dimensions and the second giving a general summary of many dimensions.

## II. Model

The model can be summarized as follows. The $k^{th}$ poll takes place in state $i$ on day $j$ and has results $y_k$, the number of people who would vote Democrat. This is distributed as

$$y_k \sim \text{Binomial}(\pi_{ij}, n_k)$$

where $n_k$ is the number of people surveyed and $\pi_{ij}$ is the parameter of interest, the proportion of people who would vote Democrat in state $i$ on day $j$. In turn, $\pi_{ij}$ is a deterministic function given by

$$\text{logit}(\pi_{ij}) = \beta_{ij} + \delta_j$$

where the state-level trend in $i$ on $j$ is denoted as $\beta_{ij}$ and the national-level trend on $j$ is denoted as $\delta_j$. These are modeled as

$$\begin{aligned} \beta_{i,\text{election day}} &\sim \mathcal{N}(\text{logit}(h_i, \tau^{-1})) \\ \beta_{i,j} &\sim \mathcal{N}(\beta_{i,j+1}, \sigma_\beta^2) \\ \delta_{\text{election day}} &= 0 \\ \delta_j &\sim \mathcal{N}(\delta_{j+1}, \sigma_\delta^2) \end{aligned}$$

where $h_i$ is the state prediction from an accepted political science estimation tool, $\tau$ is the precision set by the user, and $\sigma_\beta$ and $\sigma_\delta$ are the standard deviations for the state and national trend random walks, respectively. The traditional estimate $h_i$ is given by the 2004 presidential election results in each state, plus 6 points for Hawaii and Texas and minus 6 points for Massachusetts and Arizona, to correct for home state advantage; to this, add 8 points for estimates done between six months to ten weeks before the election, or 5.5 points for estimates done less than ten weeks before the election. The precision $\tau$ is set at 10 for the six month to ten week range, or 20 for the less than ten week range. Finally, these standard deviations are given priors, as

$$\begin{aligned} \sigma_\beta &\sim \text{Uniform}(0, 10) \\ \sigma_\delta &\sim \text{Uniform}(0, 10) \end{aligned}$$

These parameters are mentioned as having priors in the paper, although they are not stated. We determined what they were via email with the author.

The interpretation of the model is as follows. Polls represent noisy estimates of the proportion of Democrat voters on that day in that state. This proportion is a function of a national trend and a state trend. The national trend captures correlations across states in the polls. The state trend captures correlations across days in the same state in the polls. These trends are also noisy, which is represented as reverse random walks. On election day, the national trend is zero, as this is simply captured by examining the winner, and the state trends are noisy estimates of the traditional election estimator's results. Each day $j$ before the election, the trends take a random step away from the trends on $j + 1$, as we know the trends on the final day. We put a prior on the size of the random step, as we do not know it *a priori*. When estimating the results of the election before election day, only the polls published to that point are included in the model. Because of this, the model produces different results when estimating the election at, for example, ten weeks prior to the election versus four weeks prior.

The parameter of interest is $\pi_{ij}$ for all states and for all days, as this gives our estimates of the winner. However, this is very high-dimensional, so we rely on conveying a few dimensions of interest through the selected figures.

## III. Data

The data consists of all polls on or after May 1, 2008, across all fifty states. Each poll had four parameters: state, date, number of respondents, and percent who would vote Democrat. This data could be harvested from `pollster.com/polls`. Notably, the original paper stated that it had roughly 1700 polls; however, we collected just around 1370. We expect that the missing polls do not significantly affect the model near election day, because they make up a relatively small proportion of total polls. If these polls were taken early, though, they could have a noticeable affect on estimated results form early in the campaign, as there are much fewer polls at those points.

Importantly, the number of respondents and percent voting Democrat had to be corrected to exclude undecided voters. This is briefly mentioned and easy to overlook, but has disastrous consequences. Figure 1 shows $\pi_{ij}$ in Florida fitted to the uncorrected polls, with the scatter points first drawn as the uncorrected polls and then as the corrected polls. The model fits to uncorrected polls well, but evidently failing to correct for undecided voters underestimates Democrat voters.

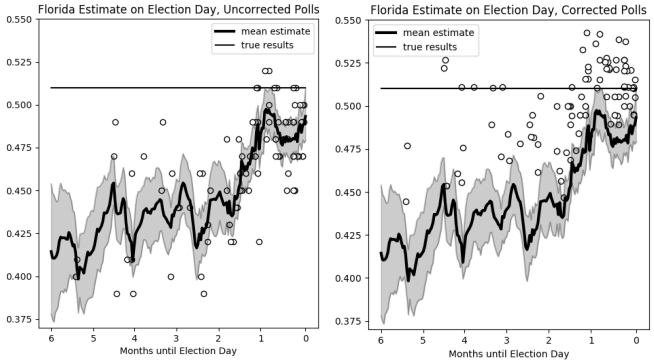


Fig. 1. $\pi_{ij}$ for $i =$ Florida and $j$ as the x-axis. When the polls are not corrected, the model fits to them well, but consistently underestimates the proportion of votes Obama should receive. Correcting the polls clearly shows the model's failure.

## IV. Implementation

The original implementation was done in WinBUGS, a probabilistic programming language that implements Gibbs sampling. Our initial goal was to develop our own Gibbs sampler in Python, for the purpose of learning more about Gibbs sampling. However, the normalization constants for the distributions conditioned on all other variables cannot be calculated in closed form. The high dimensionality of this model implied that methods designed for such situations, such as rejection sampling, importance sampling, and slice sampling, would be very slow. In high dimensions, rejection sampling rejects the vast majority of samples, importance sampling has difficulty sampling tails of the distribution, and slice sampling performs very slow random walks. Performing one of these on each step of Gibbs sampling would be impractical. Numerically integrating to obtain the normalization constant would be similarly slow, and approximating the distributions with a finite set of discrete points fails, as the range of parameters $\beta_{ij}$ and $\delta_j$ is over all real numbers.

Because of these issues, we ultimately chose to do our implementation in WinBUGS. This allows the distributions to be defined as in Section II, instead of as conditioned on all other variables. However, the WinBUGS implementation was not without its issues. WinBUGS expects clearly defined models and does not have the high-level power of Python. Polls were conducted irregularly, with some days having multiple polls and most days having zero polls; this made indexing the poll variables difficult. To overcome this, we wrote a Python script to write a vector of the 1000+ poll variables with the parameter of each set to the correct variable.

Another key implementation point is that the state-level trend is not updated every day; instead, it is updated once every three days. The paper presents this primarily as an approximation to improve speed of convergence. However, it also reduces how quickly the state-level trend varies. When the state-level trend is updated every day, the estimate of $\pi_{ij}$ varies too quickly and thus overfits to the poll results.

Our sampling parameters do not exactly follow the original ones. In particular, we chose to run just one MCMC chain, as opposed to three. We also use 100,000 samples with 50,000 eliminated for burn-in and thinning to one every 150, instead of 200,000 samples with 100,000 burn-in and thinning of 300. All of these choices risk making our implementation inaccurate, in return for dramatically reducing the computational cost. However, we chose to do this after implementing the model exactly according to their specifications and comparing it to one with these simplifications; visual inspection showed no significant difference between the two. Thus, we finished all other computations with these simplifications.

## V. Results and Discussion

The figures generated by our implementation and the ones in the original paper are shown in Figures 2, 3, 4, and 5.

Figures 2 and 3 show the mean of $\pi_{ij}$ plotted as the thick black line, with the gray shaded region as the 90% Bayesian credible interval. The individual points show the polls conducted in that state. The dashed horizontal line is the traditional estimator's prediction, which we do not plot, while the solid line illustrates the true results.

Our results are very closely aligned with those from the original paper. The most noticeable difference appears near the fifth and sixth month before the election. This suggests that the missing polls are disproportionately near the beginning of the campaign, affecting our early results but becoming less significant for late results.

To obtain the results in Figure 5, we had to run the WinBUGS sampler nine separate times, to account for not knowing future poll results. Each column corresponds to the model's prediction of the state's Democrat proportion on election day,

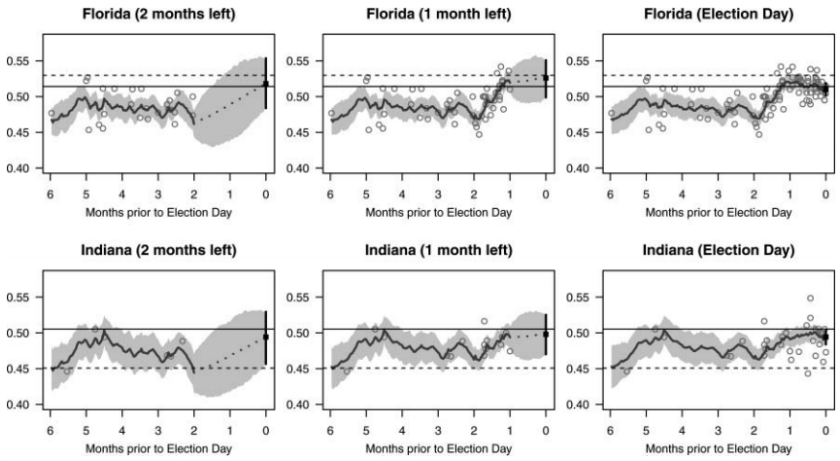


Fig. 2. The close look at Florida and Indiana presented in the original paper.

$\pi_{i,\text{election day}}$, given the polls available at that many weeks before election day. A black box indicates a swing state, i.e. that the proportion 0.5 fell within the 10% to 90% credible interval. Incorrectly predicted states are crossed out.

For the most part, our results closely match those from the original paper. Even strange behavior, such as the model confidently but incorrectly predicting Missouri's behavior two weeks before the election, is replicated. Notably, our credible intervals were much wider for the 14 week estimate, so we instead restarted a chain and collected 200000 samples, thinning to one every 300 and discarding the first 100000 as burn-in for these estimates. However, this did not strongly affect results; less than five states changed designation, and the net number of swing states stayed the same. This strongly indicates that the 330 missing polls played a large roll determining the posterior estimates for these states. More generally, aside from our reduced number of samples, there are a few possible explanations. First, the missing polls have a noticeable influence. Alternatively, since we use just one chain, it is possible our initialization was poor. Finally, pure randomness could be at fault; for many states, the 10% and 90% points of the credible interval were very close to the border that determined whether the state was a swing state, suggesting that just one sample could have changed the results. Restarting the chain, having a longer burn-in, and collecting further spaced samples should address most of these points, hence the conclusion that the missing polls are the source of systematic error in week 14.

## VI. Conclusion

The results of our implementation of the model and Gibbs sampler match those in the original fairly closely. The differences in the results likely come from missing data, our reduction in the burn-in and thinning, or randomness. The model presents an easily understandable example, with interpretations of high-dimensional data as states' voter estimates over many days, and with credible intervals determining whether a state is a swing state. Further work could expand this to other presidential elections, as this simply represents a change in dataset and traditional state predictions.

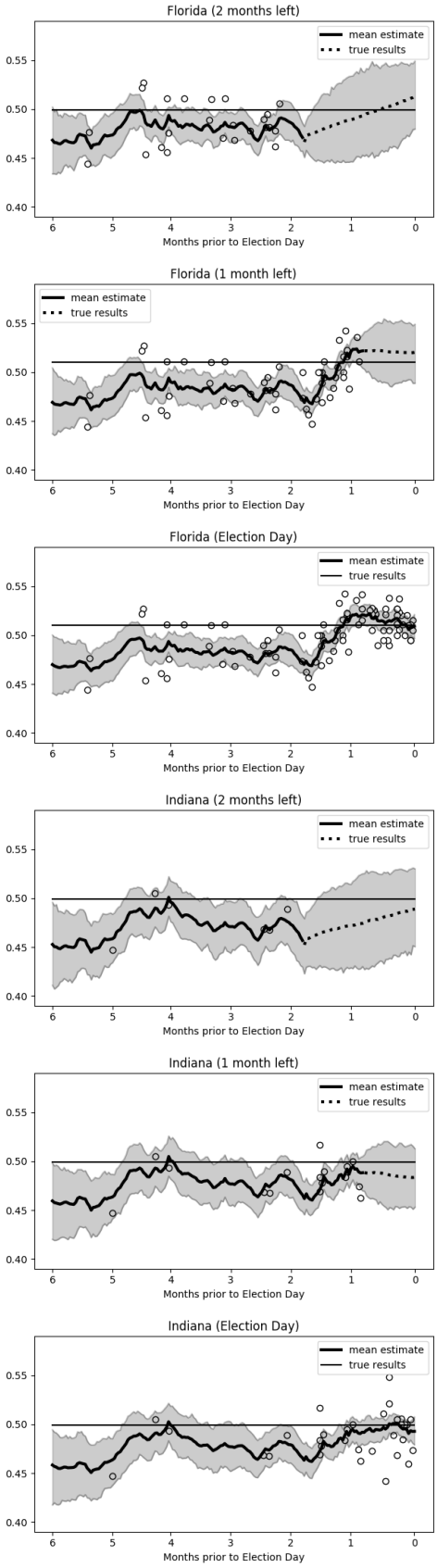


Fig. 3. Our recreation of the close look at Florida and Indiana

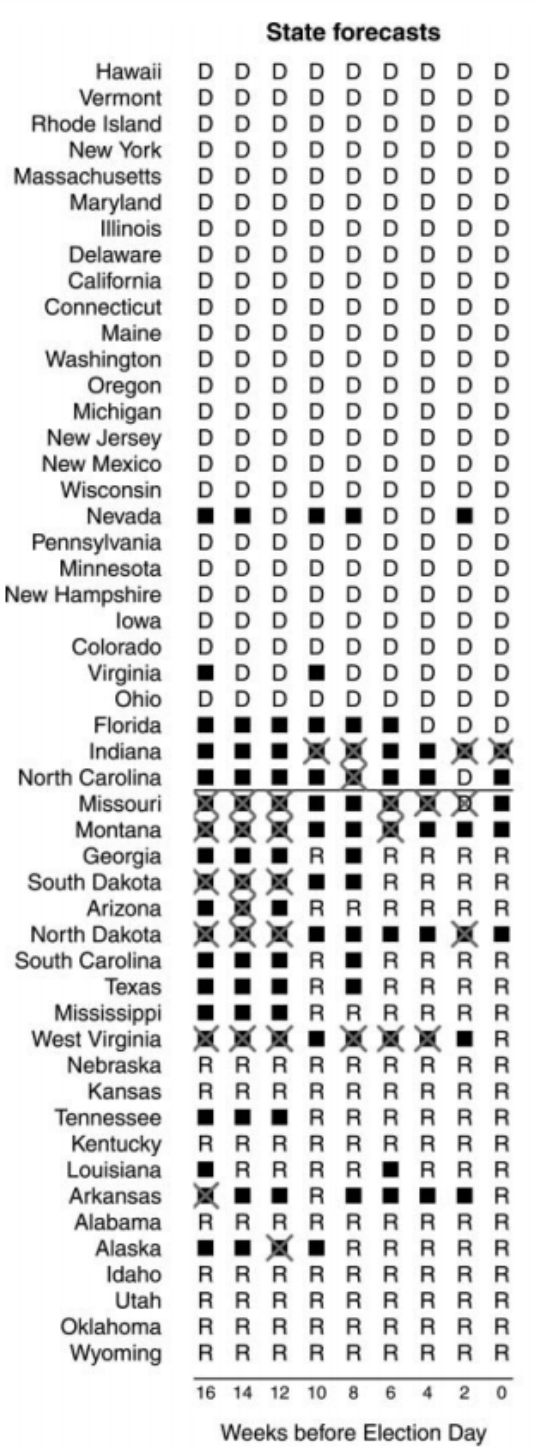


Fig. 4. The general behavior across all states presented in the original paper.

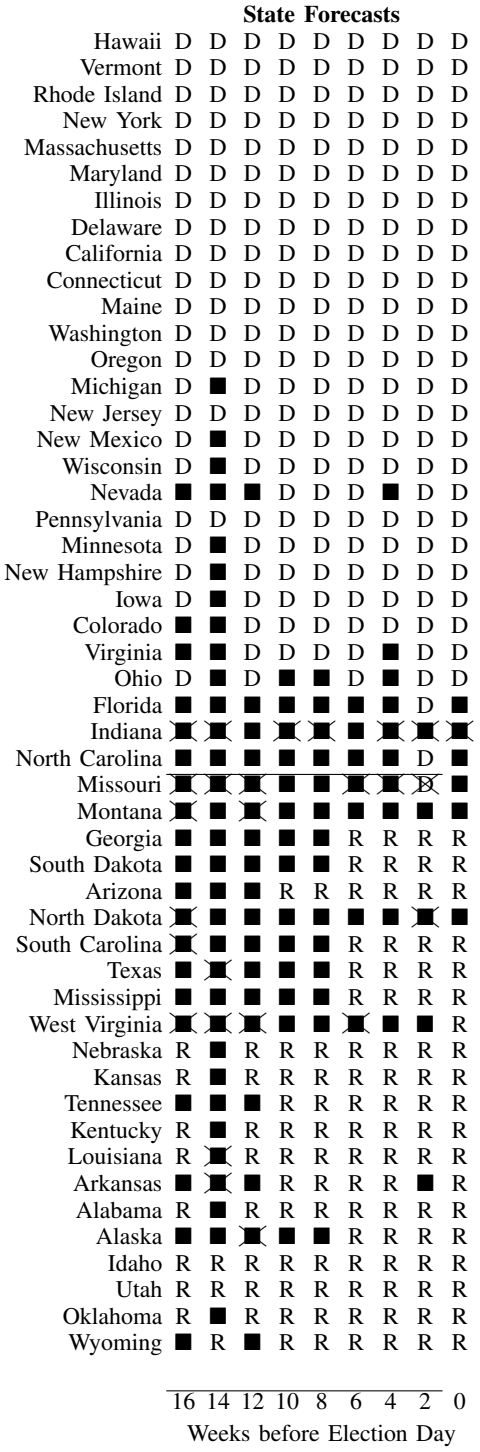


Fig. 5. Our recreation of the general behavior across all states.