Prediction of the 2018 Midterm Elections

Jiafeng Chen* Joon Hyuk Yang†‡

December 2, 2018

1 Our model

In this section, we provide a quick review of our model. Let

$$Y_i = \frac{\text{Republican\%}}{\text{Republican\%} + \text{Democrat\%}} \tag{1}$$

be the outcome variable of interest, where i denote a district in a particular election. For Y being a 435-vector[1] representing elections in 2018, we form a linear probability prior

$$Y \sim \mathcal{N}(\mu_0, \Sigma_0) \tag{2}$$

where $\mu_0 = X\beta_0, \Sigma_0$ is estimated on the training data. To estimate β_0 and Σ_0 in (2), we use a cross-validated elastic net for β_0; this yields $\epsilon = Y_{\text{tr}} - \widehat{\mu}_0^{\text{tr}}$ on the training set. We consider two formats for Σ_0. With *diagonal* restriction, we simply let

$$\widehat{\Sigma_{0ii}} = \frac{1}{n_i} \sum_{j:i\in\text{state}(j)} = \sum_j \frac{\mathbb{1}\left(i \in \text{state}(j)\right)}{\sum_{j'} \mathbb{1}\left(i \in \text{state}(j')\right)} \epsilon_j^2, \qquad n_i = |\{j : i \in \text{state}(j)\}| \tag{3}$$

be a state-smoothed estimate of variance on the training data. With *unrestricted* Σ_0, consider the vectors $\epsilon_{(i)}$ being ϵ entries corresponding to districts with the same label (e.g. AL–01) as i,[2] indexed by time. We compute[3]

$$\widehat{\Sigma_{0ij}} = \kappa\widehat{\rho}_{ij} \left(\widehat{\Sigma_{0ii}}\widehat{\Sigma_{0jj}}\right)^{1/2}, \; i \neq j, \quad \widehat{\rho}_{ij} = \widehat{\text{Corr}}(\epsilon_{(i)}, \epsilon_{(j)}), \quad \kappa \in [0, 1] \tag{4}$$

*Harvard College, jiafengchen@college.harvard.edu

†Harvard College, joonhyukyang@college.harvard.edu

‡The authors thank Lucas Janson, Zhirui Hu, and Dongming Huang for helpful comments on an earlier draft. We shall address a number of comments in this report, which shall be denoted with the symbol [C].

[1] In practice, we exclude the uncompetitive races from Y.

[2] Due to redistricting, the entries in $\epsilon_{(i)}$ could be completely unrelated to district i in 2018.

[3] The corresponding procedure in the original report is incorrect due to a computational error in the following expression—we did not raise the variance terms to 1/2-power in computing the off-diagonal covariance entries. As a result, the off-diagonal entries are small in our original report, and the resulting predictions are extremely similar. This is no longer the case once we correctly implemented (4)—see an email to Zhirui Hu on election day regarding this issue.

where $\widehat{\text{Corr}}$ denotes the empirical correlation operator and κ is a shrinkage factor chosen so that the resulting estimate of Σ_0 is positive-definite. We assume that a poll outcome Z_j has a normal distribution conditional on Y: $Z_j \mid Y \sim \mathcal{N}(a_{Z_j}^T Y, \sigma_{Z_j}^2)$, where a_{Z_j} and $\sigma_{Z_j}^2$ are specified in our orginal report. This allows us to update the prior in (2) and arrive at a posterior

$$Y \mid Z \sim \mathcal{N}(\mu, \Sigma), \tag{5}$$

from which we generate our predictions by drawing from the posterior (5). The final predictions of the two models, diagonal and unconstrained, are plotted in Figure 1.[4]

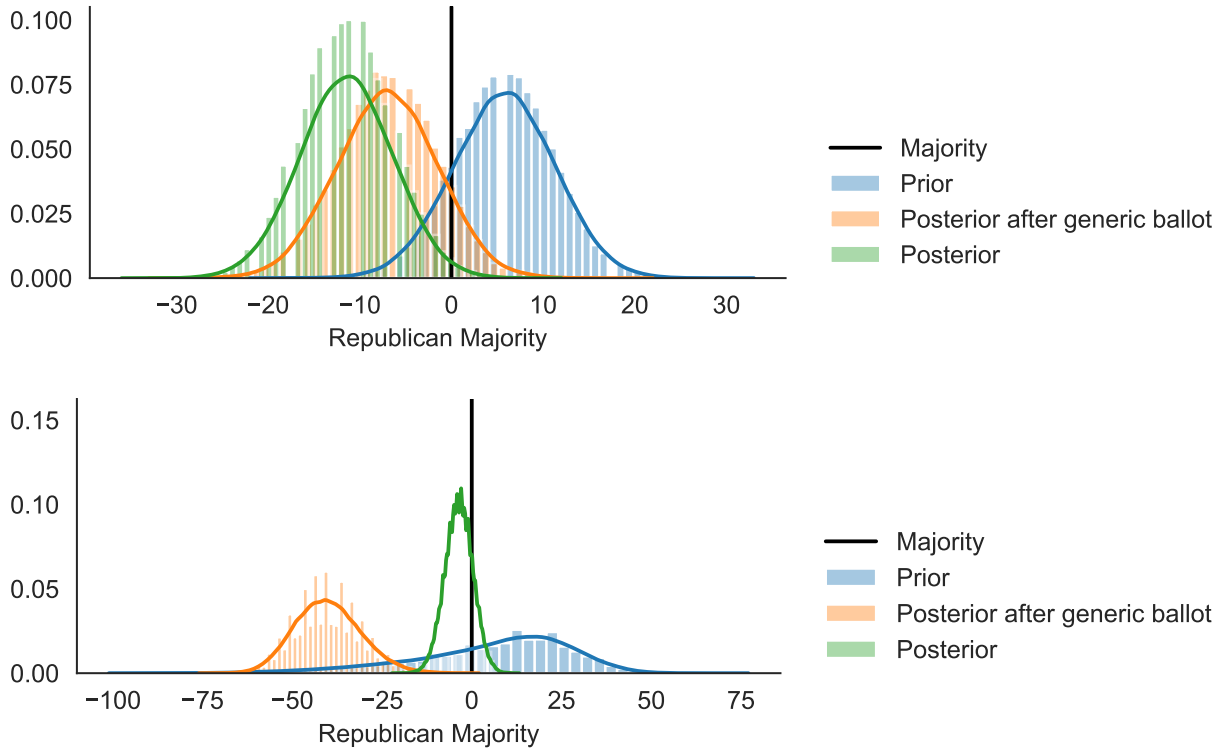

Figure 1: Model predictions. Top: diagonal. Bottom: unconstrained.

2 Overview of prediction quality

We plot a comparison of prediction quality between our prediction and that of FiveThirtyEight, broadly considered to be the state-of-the-art.[5] The R^2 by regressing actual vote

[4] [C] There was a comment regarding where the empty bars in the histograms in Figure 1 come from. We are calling the `seaborn.distplot` library function in Python. The data being plotted are integers, and so if the bin-size of the histogram is not integral, then we might see empty bins.

[5] We take the latest prediction for each district generated by FiveThirtyEight's house model (https://projects.fivethirtyeight.com/2018-midterm-election-forecast/house/, and transform the prediction into a statistic that corresponds to (1).

Table 1: Number of correctly called races for each model by winning party of each district, and number of expected seats won by Democrats compared to ground truth.

Winner	Total	Diagonal correct	Unconstrained correct	538 correct
Democrat	240	216	211	226
Republican	195	190	187	193
Expected Democratic Seats	240	230	223	233

share on predicted vote share and a constant for the diagonal model, unconstrained model, and FiveThirtyEight's model is 0.897, 0.893, and 0.967, respectively. Table 1 shows the number of correctly called races by model and winning party. Both Figure 2 and Table 1 show that the diagonal model performs better than the unconstrained model, and that both underperform relative to FiveThirtyEight's model. Moreover, Table 1 shows that all three models underestimated the Democrats' performance in the midterm elections, while the diagonal outperforms unconstrained, and both of our models underperform that of FiveThirtyEight's.

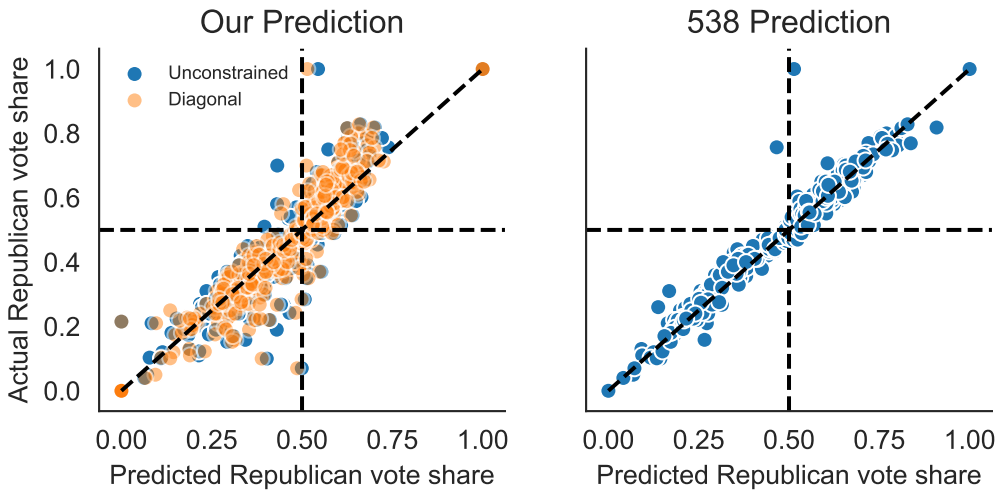

Figure 2: Quality of our prediction compared to that of FiveThirtyEight. The point that all predictions incurred large positive residuals is Alaska's district-at-large, where the Republican won a competitive race, yet the Republican's opponent is an Independent and not a Democrat. Thus (1) would define the response variable being 1, even though the race is fairly competitive.

It is clear from Table 1 and Figures 1 and 2 that the unconstrained model suffers from too little precision, as the correlation operator in (4) is extremely noisy, since the correlation is only taken over the four election years from 2010–2016. The unconstrained model was motivated by the fear that without modeling correlation of elections, the prediction model is going to be overly precise and would overlook systemic polling and modeling errors as was the

case with the 2016 presidential election. However, it does seem that modeling correlation in the manner of (4) is not a good idea. From this point, we only consider the diagonal model.

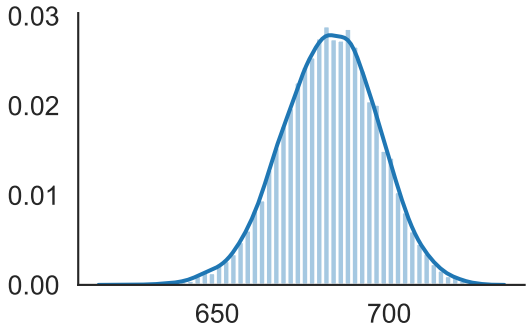

Figure 3: The distribution of log likelihood over data generated from the fitted diagonal model.

The benefit of explicit probabilistic modeling in the manner that we have done is that we can evaluate the likelihood of the outcome that materialized. Figure 3 displays the distribution of log-likelihood with simulated data generated according to the fitted model $\mathcal{N}(\mu, \Sigma)$. With the general intuition of hypothesis testing, we reject the model if the observed data has extremely low likelihood under the model. Evaluating the data transformed via (1) suffers from extraordinarily low likelihood of a few observations, due to the presence of Independents (as in the caption of Figure 2). However, even with untransformed data, the data-likelihood is still exceptionally low compared to Figure 3: The log-likelihood of transformed data (as in (1)) is 41.75, while the log-likelihood of untransformed Republican vote share is 442.25. The reason of the low likelihood seem to be that the variance of the model being too low, as μ is fairly close to the materialized outcome, by Figure 2. We can consider a variance inflation parameter σ^2 which maximizes the data likelihood under model $\mathcal{N}(\mu, \sigma^2\Sigma)$. The fitted σ^2 is 4.262 for transformed data and 2.224 for untransformed data.

3 Model selection in fitting μ_0

We examine the effect of covariate selection by running the procedure while leaving one of the covariates out, somewhat mimicking a backward stepwise model selection procedure. Since the design matrix X included both linear and quadratic terms of the covariates, leaving one covariate out leaves out all its quadratic terms. We present the R^2 of the regression of the 2018 data on the predictions (as in Figure 2) in Table 2. We see that leaving out most covariates have little effect on the prediction quality, while racial makeup, incumbency, and educational background are particularly important for prediction. Moreover, for quite a few covariates, leaving them out actually *improves* fit, suggesting that the elastic net regularizer is not a panacea for overfitting—in particular, if the covariates have little predictive power, then in order for the regularizer to adequately control for overfitting, it must have a high level of shrinkage, which may result in underfitting, as the regularizer would discount variables that are highly predictive.

Table 2: Effect on fit (among competitive races) of leaving one covariate (along with all higher-power terms that involve the covariate) out; we also show the performance of certain alternative prediction functions for μ_0. `full` means full model. `quadratic` means leaving out all quadratic terms. `lasso_select` means using a cross-validated LASSO to select covariates by discarding all covariates with zero fitted coefficent and running an elastic net on the rest of the covariates. `gradient_boost` is a gradient boosting regression tree with validation-guided early stopping. `logit` is a model where $\mu_0 = X\beta_0$ is replaced with $\mu_0 = \text{logit}^{-1}(X\beta_0)$—we fit an elastic net on logit-transformed training data to obtain β_0.

Variable left out / Model name	R^2
`minority_percentage`	0.7649
`rep_incumbent`	0.8062
`percent_bachelor_or_above`	0.8086
`gradient_boost`	0.8138
`pres_approval`	0.8177
`log_med_income`	0.8182
`rep_to_tot_oct`	0.8183
`rep_to_tot_nov`	0.8211
`same_as_pres_party_rep`	0.8215
`full`	0.8215
`dem_is_female`	0.8217
`dem_percent`	0.8230
`rep_is_female`	0.8235
`lasso_select`	0.8238
`logit`	0.8259
`dem_incumbent`	0.8272
`quadratic`	0.8324
`same_as_pres_party_dem`	0.8402

We also experiment with a some methods to improve model selection in Table 2, treating the data from 2018 as a validation set. In particular, the `lasso_select` entry in Table 2 represents a two-step procedure where a first-step LASSO regression is used to select covariates and a second-step elastic net is used to further control for overfitting and shrinkage. We see that this method marginally increases quality of fit. In `quadratic`, we simply leave out all quadratic terms in X, and rather discouragingly, this much sparser set of covariates perform better than both the `full` and the `lasso_select` models.

We also investigate whether alternative methods for fitting $\mu_0 = f_\beta(X)$, for some functional form f and parameters β, would have done better. We fit a gradient boosting regression tree (`gradient_boost`) on the same input space as the original model (`full`). We hold out a validation set and use an early stopping rule—stopping when the validation error fails to improve for a number of iterations.[6] `gradient_boost` does not appear to have better fit

[6]We tried multiple hyperparameters for the early stopping; all of them failed to generate better fit than

than the elastic net. We suspect that nonparametric methods like gradient boosting trees do not utilize the rich probabilistic information in the input data (the probabilistic model (2) fits fairly well by inspecting a Q-Q plot; see original report for details), which results in a worse fit. This leads us to suspect that better specified probabilistic models should do better. We correct for the misspecification in (2) by fitting β on the logit-transformed space of the original data:

$$\text{logit}(Y_{\text{tr}}) \sim \mathcal{N}(X_{\text{tr}}\beta_0, \Sigma_{\text{tr}}),$$

mimicking the *logit-normal model* (Agresti, 2015, page 283)—so that the Normal distribution is properly specified on data that can take values in $\mathbb{R}$. However, modeling in the logit space does not lend well to the simple Bayesian updating in (5) without using expensive MCMC simulations, and for convenience's sake, in prediction, we modify (2) to $\mathcal{N}(\text{logit}^{-1}(X\beta_0), \Sigma_0)$, which is again a slight misspecification.[7] This model, `logit`, does appear to perform slightly better than the original model `full`, but the degree of improvement is probably too small to show anything meaningful.

References

Agresti, Alan. 2015. *Foundations of linear and generalized linear models.* John Wiley & Sons.

`full`.

[7] Therefore, the `logit` model in Table 2 is not strictly-speaking a logit-normal model as in Agresti (2015)—rather an ad hoc alternative inspired by the logit-normal model in the literature.