Prediction of the 2018 Midterm Elections

Jiafeng Chen* Joon Yang†

October 11, 2018

1 Introduction

2 Statistical model and prediction function

Let $Y_i = \mathbf{1}(\text{Democrats win in district } i)$ be the election outcome in district i. Let X_i be a vector of covariates for that district. Assume the simple GLM

$$Y_i \sim \text{Bern}(\sigma(X_i\beta)) \tag{Logistic Regression}$$

where σ is the sigmoid function. This yields $p = \sigma(X_i\beta)$ as a natural prior mean for the prediction in order to utilize polling data for the 2018 election. We add usual machine learning bells and whistles such as LASSO/ridge regularization, cross validation, etc.

A cooler alternative would be to use some sort of Beta model and write

$$\begin{aligned} p &\sim \text{Beta}(a, b) \\ Y_i &\sim \text{Bern}(p) \end{aligned}$$

where

$$\begin{bmatrix} a \\ b \end{bmatrix} = g(X\beta),$$

for some well-chosen function g—a nontrivial task. This allows for $\widehat{a}, \widehat{b}$ that directly play nicely with new polling data. However, since $\text{Beta}(a, b) \xrightarrow{d} \mathcal{N}(\mu, \sigma^2)$ as $a, b \to \infty$, we might be able to simply replace p as being drawn from $\mathcal{N}(X\beta, \sigma^2)$ and write a Bayesian model whose parameters are $\theta = (\beta, \sigma^2)$. Basically here the underlying $\text{Beta}(a, b)$ corresponds to the "fundamentals" in 538 model, and adjustments are done via the polling data we see.

*Harvard College, jiafengchen@college.harvard.edu
†Harvard College, joonhyukyang@college.harvard.edu

3 Fitting Method

4 Results

5 Uncertainty and robustness

6 Conclusion