An Election Prediction Post-mortem

Jiafeng Chen Joon Hyuk Yang[*]

December 5, 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 denotes a district in a particular election. For Y being a 435-vector[1] representing elections in 2018, we form a linear probability prior[2]

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

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

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

be a state-smoothed estimate of variance on the training data. With *unrestricted* or *unconstrained* Σ_0, consider the vectors $\epsilon_{(i)}$ being ϵ entries corresponding to districts with the same label (e.g. AL–01) as i,[3] indexed by time. We compute[4]

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

[*] 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] [C] Zhirui has a comment that even though Y_i is approxiately Normal by the CLT, the underlying proportion p could still have a non-Normal distribution (p is random from a Bayesian perspective). The response is that we care about the distribution of Y_i, since it determines the winner, while the true proportion p does not.

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

[4] [C] 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.

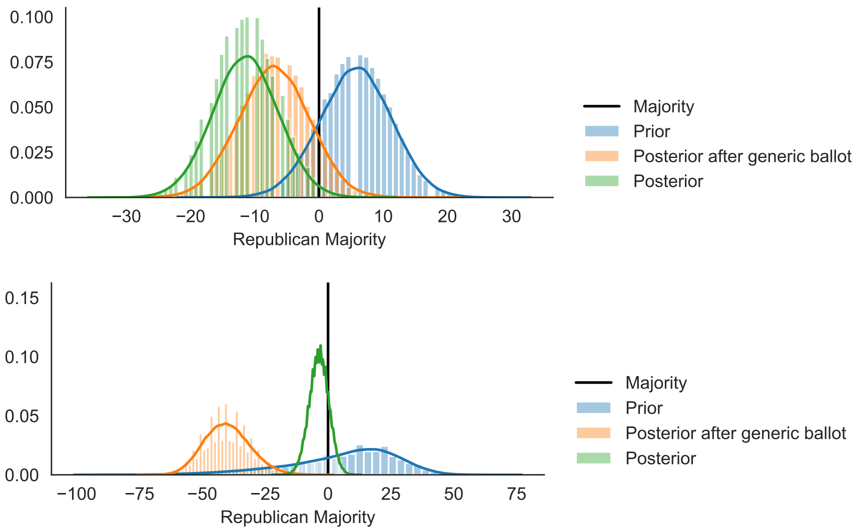

Figure 1: Model predictions. Top: diagonal. Bottom: unconstrained.

where $\widehat{\mathrm{Corr}}$ 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 it (5). The final predictions of the two models, diagonal and unconstrained, are plotted in Figure 1.[5]

The rest of this report is organized as follows. Section 2 gives an overview of our prediction quality compared to the FiveThirtyEight benchmark. Sections 3 to 5 discusses issues and methods in fitting μ_0, Σ_0, and the polling data, respectively. Section 6 concludes.

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.[6] The R^2 by regressing actual vote 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.[7] Table 1 shows the number of correctly called races by model and winning party. Both Figure 2 and Table 1 show

[5] [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.

[6] 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).

[7] Of course, the R^2 measure only considers correlation. In particular, a perfectly wrong prediction function can have $R^2 = 1$. But judging by Figure 2, R^2 does not look like a bad measure for goodness-of-fit.

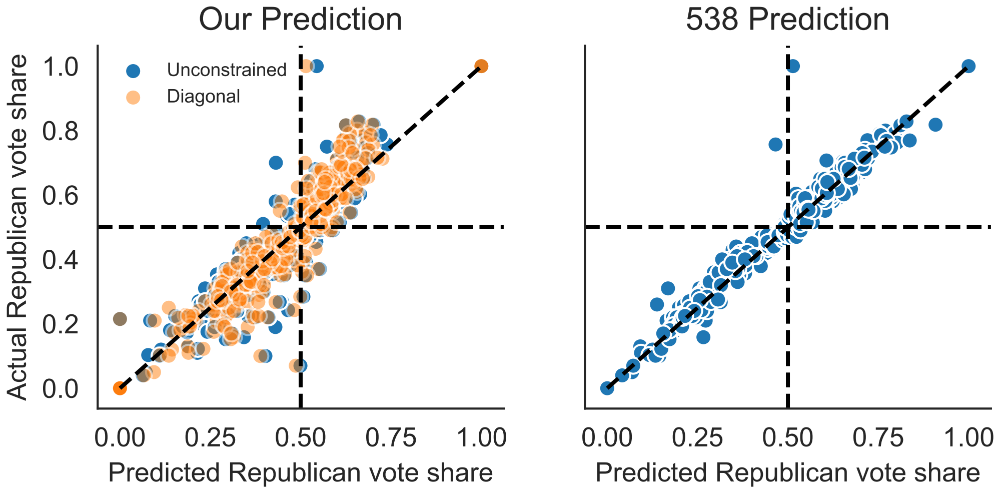

Figure 2: Quality of our prediction compared to that of FiveThirtyEight. The point that all predictions incurred large positive residuals is Alaska's district-at-large, where the Republican won a competitive race, yet the Republican's opponent is an Independent and not a Democrat. Thus (1) would define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

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

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

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 against FiveThirtyEight's predictions.

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,[8] 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

[8] As Zhirui's comment [C] pointed out.

the manner of (4) is not a good idea. From this point, we only consider the 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. Consider the the distribution of the likelihood $\log L(\tilde{Y})$ of data $\tilde{Y} \sim \mathcal{N}(\mu, \Sigma)$ generated under our model-fitted posterior. The 0.1%-percentile of the likelihood is 630.35. 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 the purported data-generating distribution: 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. Both estimates would strongly reject a null hypothesis of $\sigma^2 = 1$.

3 Fitting μ_0

3.1 Data and Preprocessing

Much of the data that we collected are of limited reliability. For instance, the gender of the candidiate, upon manual inspection, was wrong in many cases. We used a Python `gender-guesser` library to account for this while marking gender neutral names as 0.5.

Data reliability issues imply that (1) we could have improved our data cleaning process by looking up each candidate and filling in inconclusive entries, and (2) features such as presidential approval or education level may have suffered from inaccurate data reporting or exhibit high variance by being a point estimate as opposed to an aggregate over surveys from longer periods of time.

In Part I, we also discussed the effectiveness of Google Trends's relative popularity of search queries. However, there were several assumptions made that may have proved problematic. First, search frequency is a good proxy for interest, but it does not translate to support. We can imagine various scenarios such as an incumbent candidate not having as many search queries by virtue of already being well-known, or having a much higher search query due to a negative press release. As our model relies on a linear model with at most a second degree basis transform, tracing complex, non-linear relationships was limited without introducing overfitting. Second, for certain states (especially those with a low population), acquiring search frequency ratio of candidate names at a State-level granularity was impossible due to the lack of absolute number of queries. We imputed these ratios with the ratios of a nation-wide search, which might introduce bias, as there are more people living in blue states than in red states. Last and not least, even for states where there are enough queries to return a state-level ratio, such as in Alaska (Figure 3), the resulting ratio alone may not be a discerning indicator for candidate preference.

A better approach to utilizing Google Trends data would be to have a list of key issues

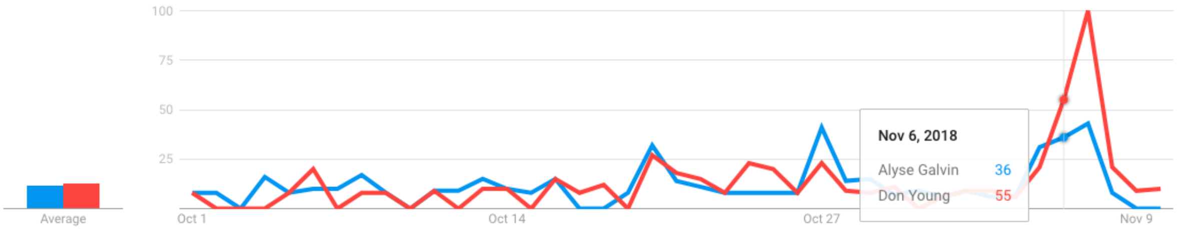

Figure 3: Google Trends search frequency ratio of Democratic vs. Republican candidate in Alaska at large.

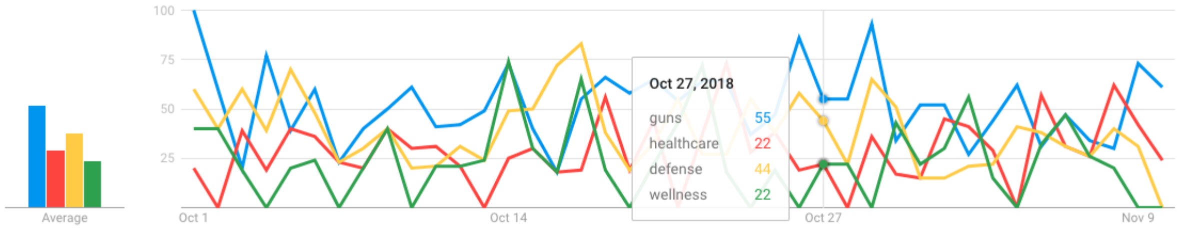

Figure 4: Google Trends search frequency ratio in Alaska on guns, healthcare, defense, and wellness. On average, guns are more frequently queried compared to healthcare, and defense more frequently than wellness.

by party from institutional knowledge or from analyses of party platforms, and compare the interest of Republican flagship issues against Democratic ones.[9] For instance, Figure 4 shows the Google Trends frequency results for four key issues: guns, defense, healthcare, and wellness. The first two are the flagship areas in which Alaska's Republican candidate emphasized, while the latter two are those prioritized by the Democratic candidate. The overall running average of these issues in Alaska suggests that there is ample room to augment our data by feeding in a more holistic picture of search query frequency ratios on topics that are brought forth by candidates. Don Young, the Republican, took Alaska's seat at the House.

3.2 Covariate and model selection in forming prior

We examine the effect of covariate selection by running the procedure while leaving one of the covariates out, somewhat mimicking a backward stepwise selection procedure. Since the design matrix X included both linear and quadratic terms of the covariates, leaving one covariate out leaves out all of 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

[9] This approach would require some heavy topic/language modeling techniques in machine learning just to collect the data.

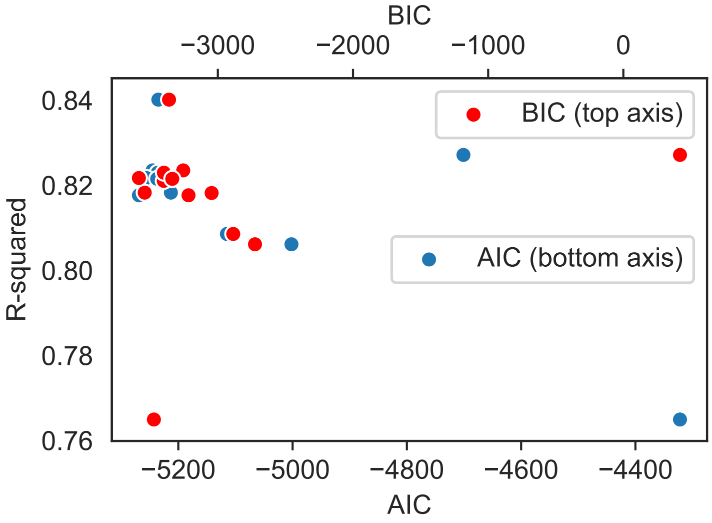

Figure 5: Comparison between the information criteria in Table 3 and R^2 in Table 2.

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.

We supplement the analysis in Table 2 with a calculation of Akaike and Bayesian information criteria for the elastic net fitting, as the R^2 on the 2018 data is not available at training time. In particular, we calculate the information criteria by computing[10]

$$\widehat{\mathsf{df}} = \operatorname{tr}\left(X_A(X_A^T X_A + \lambda_2 I)^{-1} X_A^T\right), \quad X_A = \text{active set of covariates } X \text{ under the LASSO part},$$

We assume the likelihood is generated by the homoskedastic model $Y_{\mathrm{tr}}^i \sim \mathcal{N}(X\beta, \sigma^2)$ and compute the information criteria for such a likelihood

$$\mathsf{AIC} = 2\left(\widehat{\mathsf{df}} - \sum_i \log L(Y_{\mathrm{tr}}^i)\right) \qquad \mathsf{BIC} = 2\left(\widehat{\mathsf{df}} \log N - \sum_i \log L(Y_{\mathrm{tr}}^i)\right).$$

We show in Figure 5 that the information criteria roughly correlates to realized goodness-of-fit on the new data,[11] suggesting that model selection via AIC and BIC at training time can reduce generalization error.

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

[10] The degree-of-freedom calculation follows this talk by Hui Zou and Trevor Hastie on the elastic net: https://web.stanford.edu/~hastie/TALKS/enet_talk.pdf

[11] Better, i.e. lower, AIC/BIC values generally translates to higher R^2.

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. `kernel_sigma` is the estimation procedure outlined in Section 4.

Variable left out / Model name	R^2	Variable left out / Model name	R^2
`minority_percentage`	0.7649	`kernel_sigma_bw=1`	0.8229
`rep_incumbent`	0.8062	`kernel_sigma_bw=5`	0.8229
`percent_bachelor_or_above`	0.8086	`dem_percent`	0.8230
`gradient_boost`	0.8138	`rep_is_female`	0.8235
`pres_approval`	0.8177	`lasso_select`	0.8238
`log_med_income`	0.8182	`kernel_sigma_bw=.3`	0.8245
`rep_to_tot_oct`	0.8183	`kernel_sigma_bw=.1`	0.8253
`rep_to_tot_nov`	0.8211	`logit`	0.8259
`same_as_pres_party_rep`	0.8215	`dem_incumbent`	0.8272
`full`	0.8215	`quadratic`	0.8324
`dem_is_female`	0.8217	`same_as_pres_party_dem`	0.8402

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.[12] `gradient_boost` does not appear to have better fit than the elastic net. We suspect that nonparametric methods like gradient boosting trees do not utilize the rich probabilistic information in the input data,[13] 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

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

[13] the probabilistic model (2) fits fairly well by inspecting a Q–Q plot, despite the misspecification; see original report for details

Table 3: Effect on the information criteria of leaving one covariate out, along with all higher-power terms that involve the convariate. The table is sorted according to AIC.

	N	$\widehat{\mathsf{df}}$	AIC	BIC
`pres_approval`	1530	161	−5269	−3217
`dem_is_female`	1530	131	−5256	−3584
`rep_is_female`	1530	156	−5245	−3256
`same_as_pres_party_rep`	1530	149	−5237	−3337
`full`	1530	149	−5237	−3337
`rep_to_tot_nov`	1530	144	−5236	−3400
`dem_percent`	1530	144	−5236	−3399
`same_as_pres_party_dem`	1530	147	−5235	−3361
`rep_to_tot_oct`	1530	131	−5213	−3541
`log_med_income`	1530	170	−5212	−3046
`percent_bachelor_or_above`	1530	175	−5115	−2886
`rep_incumbent`	1530	179	−5002	−2725
`dem_incumbent`	1530	404	−4701	419
`minority_percentage`	1530	66	−4322	−3473

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.[14] 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 meaningful conclusions.

4 Fitting Σ_0

The procedure outlined in (3) would provide downward-biased estimates of Σ_0, since we are using the *fitted* residuals to estimate variance. A more "honest" (Athey and Imbens, 2016) approach for variance estimation would be to use a validation set. However, for the variance as in (3) to have any precision, each state in the validation set needs to include a substantial number of observations, which is difficult for states like Alaska or Montana. We consider an alternative approach in this section. Observe that (3) is a Nadaraya–Watson estimator where the kernel weighting is whether i, j are from the same state. We can thus generalize (3) into

$$\widehat{\Sigma_{0ii}} = \sum_j \frac{K_h(x_j - x_i)}{\sum_{j'} K_h(x_{j'} - x_i)} \epsilon_j^2 \tag{6}$$

for some kernel function $K_h(x - x_0)$. (6) codifies the intuition that districts with similar observable characteristics should have similar prediction error magnitudes, as in the intuition for FiveThirtyEight's CANTOR model,[15] which uses a kNN kernel to identify similar

[14] 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.

[15] `https://fivethirtyeight.com/features/2018-house-forecast-methodology/`

Table 4: Model evaluation of kernel-fitted Σ_0

	$h = .1$	$h = 1$
0.1%-percentile Log Likelihood	575.9772	560.6451
Data Log Likelihood (transformed)	185.7445	235.7885
Data Log Likelihood (untransformed)	458.2238	507.2840
Data-rationalizing σ^2 (transformed)	3.3033	2.9671
Data-rationalizing σ^2 (untransformed)	1.9167	1.5854

districts. In the implementation demonstrated, we let K_h be the Gaussian kernel

$$K_h(x - x_0) \propto \exp\left(-\frac{\|x - x_0\|^2}{2h}\right),$$

where x, x_0 are the *linear* terms comprising of all covariates used in `full`. We sum over j that are in a validation set, comprising of 10% of observations selected at random. The change in the prior variance Σ_0 have an impact on the posterior μ, since it changes the Bayesian updating. The impact of kernel weighting in estimation of Σ_0 is also shown in Table 2. We see that the kernel weighting methods seem to increase quality of fit by a minute amount, with relatively low sizes of bandwidth delivering better fits.

We perform the same exercise at the end of Section 2 to check how far the model under our posterior deviates from the data, and results are shown in Table 4. We see that the kernel fitting alleviates, but does not eliminate, the problem of overly sharp posterior variance discussed in Section 2: The posterior model is still sharply rejected by the data, but the inflation factor that the posterior needs to rationalize the data is smaller with kernel smoothing of variance than with our original model. Moreover, it appears that a higher degree of smoothing generates more agnosticism in the posterior, and therefore makes the posterior distribution contradict the data less.

5 Updating with polls

We have seen from Section 4 that our final estimates are still overly precise, despite considerable efforts to more conservatively infer the prior variance. Another reason for the rejection of our posterior model is generalization error—the information contained in the training data are 2010–2016, which might not generalize to 2018, leaving bias in the prior estimates (though Figure 2 seems to suggest that this is not the case). One way to reduce the generalization gap is to use polling information, which encode the political environment in 2018 that we cannot have learned from the 2010–2016 data. The Bayesian update (5) also affects posterior variance. In this section we consider alternative methods to deal with polling data; however, due to time and resource constraints, we do not implement models considered in this section.

First, it has come to our attention that FiveThirtyEight does publish past polling data, and thus the simplest way to incorporate polling data is to construct covariates that take into account both recency and accuracy of polls. Second, one large issue we faced is that the variance of polling results cannot be explained by a simple Binomial likelihood. Our remedy in the previous report is to approximate the Binomial likelihood with a Normal likelihood, with variance inferred from the empirical variance of data. An alternative approach, inspired by the problem of overdispersion in GLMs and quasi-likelihood estimation (Agresti, 2015, Chapter 8), is to consider the Binomial model as an exponential dispersion family model $p(y) = \exp\left(\frac{y\theta - b(\theta)}{\phi} + c(y, \phi)\right)$ with unknown dispersion parameter ϕ. For a given district, we can consider the *maximum quasi-likelihood estimate* (QMLE) (Agresti, 2015) for the Binomial parameter p and its variance τ^2 as a summary statistic of the data, and use the QMLE to update the prior. Lastly, we are assuming that conditional on Y_i, polls are independently distributed, which drastically increase the influence of additional polls. However, one probably expects polls to be correlated [C]. However, it is difficult to define or estimate correlation among polls, since the polling data is not of a panel format. One approach might be to consider some hyperparameter ρ and correlation matrix of the form $\text{Corr}(Z_i, Z_j) = \rho$.

As a further direction, due to the lack of polling in certain districts, we could consider certain methods for imputing polling responses in unpolled districts. For instance, suppose we observe individual-level data on, say, browing history in a large number of districts, where some are polled and others are not. We can consider the individual as either a Democrat or a Republican, a latent variable that we do not observe. However, polling gives the proportions of Democrats and Republicans in some districts. One could build a model based on the EM algorithm to infer the latent political leanings variable as a function of observable individual-level data, as Comarela et al. (2018) have done.[16] This model would then be able to predict polling outcomes in unpolled districts by imputing the political leanings of individuals observed in those districts. Of course, such a method requires access to vast amounts of granular data, which we do not have.

6 Discussion and conclusion

We have considered modifications to all parts of our estimation procedure, outlined in Section 1. We find in Section 2 that our procedure generally creates good point estimates, which are a bit less precise than the state-of-the-art; however, we also find that the posterior is too precise for the data to not reject the model, requiring an inflation factor of about two to four times to the variance in order for the model to not be rejected. We consider preprocessing, model selection, and feature selection in Section 3; we consider a generalization of (3) and kernel methods to estimate variance in Section 4; and we sketch a few possible directions to improve the Bayesian updating.

[16] For instance, if many individuals in a heavily Republican district visit the website of Fox News regularly, we would learn that browsing Fox News is correlated with being a Republican, even if we have no knowledge of individual political leanings. We can then infer that an unpolled district in which many people browse Fox News must lean Republican.

References

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

Athey, Susan and Guido Imbens. 2016. "Recursive partitioning for heterogeneous causal effects." *Proceedings of the National Academy of Sciences* 113 (27):7353–7360.

Comarela, Giovanni, Ramakrishnan Durairajan, Paul Barford, Dino Christenson, and Mark Crovella. 2018. "Assessing Candidate Preference Through Web Browsing History." In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery &#38; Data Mining*, KDD '18. New York, NY, USA: ACM, 158–167. URL http://doi.acm.org/10.1145/3219819.3219884.