Analysis and modeling plan

Simon Jackman

14 November 2019

Iowa

Iowa is our immediate focus.

We seek reasonable models of each candidate's share of the Iowa caucus vote.

Let $t > 0$ index days until the Iowa caucus. For instance, today, 14 Nov 2019, is 81 days before the Iowa caucuses on 3 Feb 2020.

Define the set of interesting candidates to be those who wound up with more than 10% of the vote in Iowa. **Please check how many viable candidates in a given year, say top four finishers – in either Iowa or nationally – are tossed out with this approach.**

Let c index this set, pooling over years and letting repeat candidacies enter separately (e.g., Biden 1988 is different from Biden 1992 or whatever, for instance).

EDA

Can we please plot, per candidate, poll results ahead of Iowa, as a time series from t days out on left of the graph to election day (right of panel) overlay loess trend (weighting each observation by square root of the sample size of the poll), and then `geom_hline` the actual result. Please also compute and report the error between the trend fitted by loess and the Iowa result. Perhaps report these in a table sorted from high to low in absolute value.

Predictive model

We want to fit a model of the form

$$y_c = f(\mathcal{P}_{ct}) + h(e_{ct}) + \epsilon_{ct}$$

where

- y_c is the share of the Iowa caucus vote won by candidate c
- $\mathcal{P}_{ct}$ are polls for candidate c available up through day t; more on how these enter the model, below.
- f is some function of those polls, again, see below.
- e_{ct} is an endorsement count or index measure, for candidate c up through day t
- h is some unknown function, possibly a spline
- ϵ_{ct} is random error

To keep things simple, let y_c be a proportion (not a percentage) and let any poll estimates also be expressed as proportions.

Rolling series of model fitting and predictions

We would want to fit this model every t, getting closer to the day of the Iowa caucus. For each c and t, we want to recover the forecast error, $y_c - \hat{y}_{ct}$, the error made in predicting y_c given information available on day t. At the end of the exercise we would want to plotting these errors as a function of t for each c.

Cross-valdiation

At some point – maybe not immediately – if might be a good idea to fit the models using cross-validation, holding out a random set of candidacies for testing model fit "out of sample".

How does a candidate's polling history enter the model?

Here are a number of ideas:

Most recent poll

Let $\mathcal{P}_{ct}$ be the most recent poll available on day t, X_{ct} and let f be a spline function, or a regression function. Weight the observation ct by the square root of the approximate variance of poll $\mathcal{P}_{ct}$, i.e.,

$$\sqrt{\frac{X_{ct}(1 - X_{ct})}{n_{ct}}}$$

where X_{ct} is the poll estimate and n_{ct} is the sample size of the poll. This step of weighting each observation could be dropped if the sample sizes of the polls are roughly similar.

Weighted sum of recent polls, with time decaying weights

Let $\mathcal{P}_{ct}$ be a set of polls for candidate c fielded on days $t + k, k \geq 0$, i.e., $\mathcal{P}_{ct} = \{X_{c,t+k}\}$. Then we would form a regressor

$$X_{ct} = \sum X_{c,t+k}\, \omega^k, \quad \forall\, X_{c,t+k} \in \mathcal{P}_{ct}$$

where $\omega \in [0, 1]$ is an unknown rate of decay parameter. Each observation would be weighted by the square root of the variance of the weighted sum of the polls, i.e., the variance of the weighted sum of the polls is

$$\begin{aligned} V(X_{ct}) &= V(\sum X_{c,t+k}\, \omega^k) \\ &= \sum \left[\omega^{2k}\, V(X_{c,t+k})\right] \\ &= \sum \left[\omega^{2k}\, \frac{X_{c,t+k}(1 - X_{c,t+k})}{n_{c,t+k}}\right] \end{aligned}$$

where the summation is over the set of polls $\mathcal{P}_{ct} = \{X_{c,t+k}\}$. Each observation would be weighted by the square root of this term. Again, this step of weighting each observation could be dropped if the sample sizes of the polls are roughly similar.

This model would be fit by non-linear least squares (see `nls` in R).