Machine Learning and Poststratification[*]

Joseph T. Ornstein
Brown School of Social Work
Washington University in St. Louis
jornstein@wustl.edu

September 6, 2018

Abstract

I develop a procedure for estimating local-area public opinion called machine learning and poststratification (MLP), a generalization of classical multilevel regression and poststratification (MRP). This procedure incorporates an expanded set of predictive models – including random forest and k-nearest neighbors – to improve the cross-validated fit of the first-stage model. In a Monte Carlo simulation, MLP significantly outperforms MRP when there are deep interactions in the data generating process, without requiring the researcher to specify a complex parametric model in advance. In an empirical application, MLP produces county-level estimates of Trump support that correlate better with 2016 presidential vote share than classical MRP or disaggregated survey data.

[*] Thanks to Rob Franzese, Scott Page, Paul Kellstedt, and the participants of the Michigan Political Economy Workshop and the Statistical Learning Workshop for their helpful suggestions on earlier drafts.

1 Introduction

Subnational public opinion data is often difficult or costly to obtain. For political scientists who focus on lower-level units of government (e.g. legislative districts, counties, cities), this lack of local area public opinion data can be a significant impediment to empirical research. And so, over the past decade and a half, political methodologists have refined techniques for estimating subnational public opinion data from national-level surveys. A now standard approach is multilevel regression and poststratification (MRP), first introduced by Park, Gelman and Bafumi (2004).

MRP proceeds through a two-stage process. First, the researcher estimates a hierarchical linear model from individual-level survey data, using demographic and geographic variables to predict public opinion. Typically, this model takes the following form, where the outcome is a function of individual-level demographic variables (here, x_1 and x_2), and a region-specific intercept (α_n^{region}), itself a function of region-level characteristics (z_n):

$$\hat{y}_i = \beta^0 + \alpha_j[i]^{x_1} + \alpha_k[i]^{x_2} + \alpha_n^{region};$$

$$\alpha_j^{x_1} \sim N(0, \sigma_j^2);$$

$$\alpha_k^{x_2} \sim N(0, \sigma_k^2);$$

$$\alpha_n^{region} \sim N(\beta^z \times z_n, \sigma_{region}^2)$$

The predictions from this first stage model can then be used to estimate average opinion in each local-area of interest. To do so, the researcher takes each demographic group's predicted opinion and computes a weighted average using the observed demographic distribution. This second stage is called *poststratification*. If the predicted value for each demographic group is $\hat{y}_r$, and the frequency of that group in region s is N_{rs}, then the following equation yields the MRP estimate for region s:

$$Y_s^{MRP} = \frac{\sum_{r \in s} N_{rs} \hat{y}_r}{\sum_{r \in s} N_{rs}}$$

MRP has enabled a flowering of new research on political representation in states (Lax and Phillips, 2012), Congressional districts (Warshaw and Rodden, 2012), and cities (Tausanovitch and Warshaw, 2014). But the method is not without its critics. Buttice and Highton (2013) find that MRP performs poorly in a number of empirical applications, particularly when the first-stage model is a poor fit for the public opinion of interest. In particular, they find that MRP works best for predicting opinion on cultural issues (like support for gay marriage), where there is greater geographic heterogeneity in opinion. In these cases, public opinion is more strongly predicted by geographic-level variables, yielding better poststratified estimates. But for opinions on economic issues, MRP yields a poorer fit. The authors conclude by emphasizing the importance of model selection, noting that "predictors that work well for cultural issues probably will not work well for other issue domains and vice versa". This finding echoes Lax and Phillips (2009), who urge researchers to optimize their first-stage model for the issue of interest.

In this paper, I introduce a refinement of classical MRP, called Machine Learning and Poststratification (MLP). This technique improves first-stage model selection by expanding the set of candidate models to include machine learning techniques – including random forest and k-nearest neighbors. MLP then selects the model (or ensemble of models) that minimizes cross-validation prediction error at the individual level. I show, in both a Monte Carlo simulation and empirical application, that this technique produces superior estimates of subnational public opinion under certain conditions. I conclude with guidelines for best practice and some suggestions for future research.

2 The MLP Procedure

2.1 First-Stage Model Selection

Fundamentally, MRP is an exercise in out-of-sample prediction, using observed opinions from survey respondents to make inferences about the opinions of similar individuals who were not surveyed. As such, first-stage model should be selected on the basis of its out-of-sample predictive performance. Though classical MRP relies on hierarchical linear models, there is no reason ex ante to believe that such models will perform best at this task. Indeed, there is a large collection of models from the machine learning literature that may do better.

One potential downside of adopting machine learning techniques is that they tend to be "black box" approaches to prediction. A complicated model may produce better predictions than a simple linear model, but do a poor job explaining the outcome that it is modeling – at least in a manner that is interpretable by a human researcher. The most complex machine learning techniques (e.g. artificial neural networks, random forests) may be intuitive in theory, but in practice it becomes arduous to interrogate such models to determine *why* they reach the conclusions they do. For political science applications where the objective is explanation, such an approach falls short. But since subnational public opinion estimation is fundamentally a *prediction* problem, black box models are perfectly suitable, so long as they produce good predictions.[1]

In what follows, I will introduce two machine learning techniques that can be used for the first-stage model: K-Nearest Neighbors (KNN) and Random Forests (RF). I will give a brief overview of their properties, and will then demonstrate how to apply them to subnational public opinion estimation. I must stress at the start, however, that MLP need not be limited

[1] An important caveat to this point: although the purpose of MRP is to generate predictions, the uses to which these estimates are put are often explanatory (e.g. using MRP to study the responsiveness of government to public opinion). As such, political scientists using these estimates for explanatory research should always be aware of the details of their construction. If a predictor variable that was used to construct the MRP estimate is endogenous to your regression, then so is the MRP estimate.

to these two methods. There is a broad set of machine learning techniques – including support-vector machines, Bayesian additive regression trees, and artificial neural networks – that could serve as candidate first-stage models. I focus on KNN and RF here for illustrative purposes.

2.1.1 Random Forests

Random forests, first introduced by Breiman (2001*a*), are an ensemble approach to classification and regression. Rather than estimating a single model, the procedure constructs a large collection of models, then aggregates their predictions together. Each component model is a regression tree, a technique that generates predictions by successively partitioning the data on the X variables and taking the average outcome of observations at each terminal node. To ensure that these trees are not all identical, each tree is trained on a bootstrap sample of the dataset (thus the "random" in random forest). The forest prediction is then equal to the mean prediction of the constituent trees.

One advantage of this approach is that the researcher need not assume that public opinion obeys a pre-specified model in order for the poststratified predictions to make sense. Random forest is a popular technique among machine learning algorithms, because it requires few tuning parameters and data preprocessing. See Breiman (2001*b*) and Montgomery and Olivella (2018) for two excellent introductions to these types of models. The latter reference includes a section on using regression tree methods to improve MRP estimates, an insight which this paper builds on and formalizes, incorporating a broader set of machine learning techniques.

2.1.2 K-Nearest Neighbors

K-Nearest Neighbors (KNN) is an intuitive nonparametric approach to regression. For each observation i, KNN predicts an outcome $\hat{y}_i$ by taking the k most similar observations in the training data (according to some predefined distance metric) and computing the mean of their observed outcomes. In classical KNN, this is an unweighted average of the k-nearest neighbors, but a more general approach uses a weighted average, with weights proportional to inverse distance. In the following exercise, I use the weighting scheme proposed by (Samworth, 2012).

As with random forests, the researcher need not assume a model of the DGP in order to produce estimates. Instead, KNN requires a more easily-accepted assumption: that similar people who live in similar places are likely to hold similar opinions.

Another advantage of this approach is that KNN can easily incorporate spatial predictors. For example, if each survey respondent provides their county of residence, then a prediction using KNN could incorporate the latitude and longitude of that county's centroid as predictors. Predictions would then be generated by a weighted average of nearest neighbors in *physical* space as well as some abstract variable-space. If black respondents in Tennessee have systematically different opinions than black respondents in Minnesota, then the KNN prediction would reflect that, without the researcher having to specify a battery of interaction terms in advance.

2.2 Cross-Validation

Now that we've introduced a number of possible models that one could use for the first-stage prediction, what is a principled way to go about model selection? If MRP is fundamentally a problem of out-of-sample prediction, then one should go about model selection with this criterion in mind. This naturally leads us to cross-validation.

Cross-validation is a common machine learning technique designed to guard against overfitting. A model is overfit if it produces good predictions for the dataset that was used to estimate it, but performs poorly out-of-sample. This is most likely to occur when a model is overly complex, picking up on chance patterns in the training data. Consider a common case of overfitting in political science research: models that include unit-specific or time period-specific fixed effects. Though these models may be useful for estimating causal effects, they are incapable of generating predicted values for observations outside the time periods or regions found in the training dataset.

To combat this, cross-validation partitions the data into two subsets: the training set, used to estimate the model's parameters, and the test set, against which the model's predictions are compared. By "hiding" part of the data from the model, this procedure allows the researcher to quantify how well a model performs at out-of-sample prediction. K-fold cross validation assigns $\frac{n}{k}$ observations to the test set and the remaining observations to the training set. The researcher then repeats this process k times, until each observation has been in the test set once. In the limit where $k = n - 1$, this procedure is known as "leave out one" cross-validation (LOO).

Because cross-validation error provides a measure of out-of-sample predictive accuracy, it is a principled way to select from among multiple predictive models (Stone, 1974). In the following sections, I will demonstrate that models with better cross-validated predictive accuracy typically produce better poststratified estimates than those that do not.

2.3 Poststratification

In addition to guiding the first-stage model selection process, cross-validation can help inform the researcher how best to generate the poststratification frame. Leemann and Wasserfallen (2017) introduce a promising refinement to MRP, which they call multilevel regression and *synthetic* poststratification (MrsP). Rather than creating poststratification estimates using the true joint distribution of the demographic variables in the individual-level model, this approach proceeds as if the demographic variables were statistically independent. Then, the poststratification weights can be derived from the product of the marginal distributions, a process they call synthetic poststratification. The authors conduct a Monte Carlo test of this procedure, demonstrating that, so long as the demographic variables are not too strongly correlated with one another, MrsP estimates do not significantly diverge from those of classical MRP.

The advantage of synthetic poststratification is that the first-stage model can include a larger set of individual-level predictors, for which we may not have joint distributions in the poststratification stage. This, however, presents a new problem. How does a researcher know if it's appropriate to use synthetic poststratification? In empirical applications where the joint distribution of interest is unavailable, then we cannot know how correlated the demographic variables are, so we don't know how badly MrsP would perform relative to MRP. In this paper, I propose a remedy for that problem.

In Appendix 1, I present a general proof that MrsP and classical MRP produce identical estimates if the first-stage model is additively separable. This suggests a straightforward decision rule for when to use synthetic poststratification. If a linear-additive model outperforms more complex machine learning techniques in the cross-validation stage, then the researcher should proceed with synthetic poststratification, because it allows for the inclusion of more individual-level predictors. If not, then one should use classical MRP or MLP.

2.4 Outline of MLP Procedure

Putting it all together, the MLP procedure is summarized in Table 1. This procedure varies from classical MRP in two places: (1) choosing a first-stage model based on cross-validated predictive accuracy, and (2) generating the poststratification frame synthetically if the best first-stage model is additively separable. How well does this procedure perform relative to classical MRP? To answer this question, I now turn to a Monte Carlo analysis.

Table 1: The MLP Procedure

Step	Procedure
1	Collect individual-level survey data on outcome of interest and predictors.
2	Select the model that minimizes cross-validated prediction error (or maximizes cross-validated R^2). Note: This could include HLM, or an ensemble average!
3	Fit the selected model to the entire dataset.
4	Generate predictions for each respondent type (demographics × geographic variables)
5	Poststratify by weighting these predictions against the known frequency of each type at the subnational level.
5a	If the best first-stage model is additively separable, then the poststratification frame may be generated synthetically.

3 Monte Carlo Simulation

For the following analysis, I simulate a data generating process where the outcome variable ($\mathbf{y}$) is a function of three demographic variables ($\mathbf{z_1}$, $\mathbf{z_2}$, $\mathbf{z_3}$), and geographic location. For simplicity, the DGP is linear-additive, except in two geographic regions, where the Z variables have a multiplicative effect. This produces a nonlinearity we might expect to observe in real data, where some demographic subgroups have very different opinions depending on their geography (e.g. white females in Vermont compared to white females in Georgia).

More formally, the data are generated through the following process. First, I create NM individuals, where M is the number of subnational units, and N is the number of observations per unit. Each individual has four latent (unobserved) characteristics, z_1 through z_4, drawn from a multivariate normal distribution with mean zero and variance-covariance matrix equal to

$$\begin{bmatrix} 1 & \rho & \rho & \rho \\ \rho & 1 & \rho & \rho \\ \rho & \rho & 1 & \rho \\ \rho & \rho & \rho & 1 \end{bmatrix}$$

The variable $\mathbf{z_4}$ is used to assign each observation to a subnational unit, which ensures that there is cross-unit variation on the latent characteristics. Each subnational unit, in turn, is assigned a random latitude and longitude, drawn from a bivariate uniform distribution between $(0,0)$ and $(1,1)$. Once I assign each observation a $\mathbf{z}$ vector and subnational unit, I generate the outcome variable, y, using the following equation:

$$y_i = z_{1i} + z_{2i} + z_{3i} + \left(\alpha D_i^0 z_{1i} z_{2i}\right) - \left(\alpha D_i^1 z_{1i} z_{3i}\right) + \varepsilon_i$$

D^0 is a function that is decreasing in distance from (0,0), and D^1 is decreasing in distance to (1,1), so that multiplicative effects are strongest near those points. ε_i is an iid normal error term with mean zero and variance σ^2. The parameter α governs the strength of the threeway interaction effect. When $\alpha = 0$, the DGP is simply a linear-additive combination of the demographic variables, but as α increases, the conditional effect of geography becomes stronger. Finally, I create discretized versions of the demographic variables $\mathbf{z_1}$ through $\mathbf{z_3}$, called $\mathbf{x_1}$ through $\mathbf{x_3}$. Although the outcome variable y is a function of the latent variables, Z, the researcher can only observe the discrete variables X.

I repeatedly simulate this data generating process, varying the parameters ρ and α.[2] For each simulated population, I then draw a random sample of size n, and generate three subnational estimates: disaggregation, classical MRP, and MLP. The first stage equation for the MRP estimation is a hierarchical linear model of the following form:

$$y_i = \beta^0 + \alpha_j[i]^{x_1} + \alpha_k[i]^{x_2} + \alpha_l[i]^{x_3} + \alpha_m^{unit} + \varepsilon_i;$$

$$\alpha_j^{x_1} \sim N(0, \sigma_j^2);$$

$$\alpha_k^{x_2} \sim N(0, \sigma_k^2);$$

$$\alpha_l^{x_3} \sim N(0, \sigma_l^2);$$

$$\alpha_m^{unit} \sim N(0, \sigma_{unit}^2)$$

[2] Appendix 2 provides a more detailed technical description of the simulation. Table 5 in that appendix lists the combinations of parameter values used.

For the first stage of the MLP, I train a KNN model using $\mathbf{x_1}$, $\mathbf{x_2}$, $\mathbf{x_3}$, latitude, and longitude as predictors, and LOO cross-validation to select the optimal value of k. I also train a random forest model using the same predictors. I then select the first-stage model or ensemble average that minimizes RMSE in 10-fold cross-validation.

Figure 1 illustrates the results of a representative run from the Monte Carlo simulation. Under certain conditions, MLP dramatically outperforms both disaggregation and classical MRP. When α is large, the machine learning models are better able to predict individual-level opinion than the hierarchical linear model, which in turn produces better poststratified estimates.

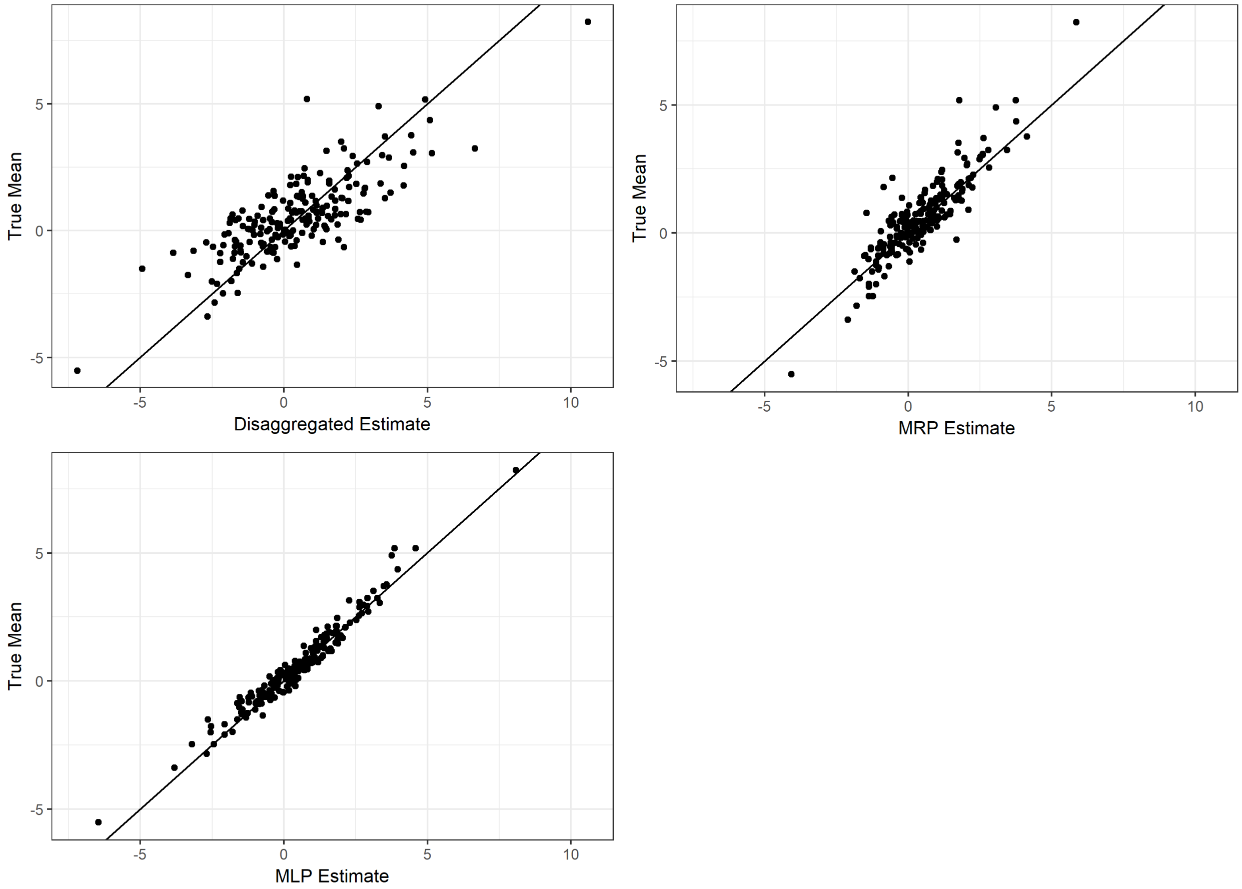

Figure 1: Representative simulation from Monte Carlo. Disaggregation, MRP, and MLP estimates are plotted against true subnational unit means. Parameter Values: $\alpha = 5$, $\rho = 0.4$, $N = 15000$, $M = 200$, $n = 5000$, $\sigma^2 = 5$.

However, the machine learning algorithms do not perform strictly better than HLM under all conditions. When α is small – and thus the true DGP is linear-additive – KNN and random forest provide no prediction advantage over HLM. Indeed, the flexibility of KNN is a detriment when the sample size of the survey is small, as KNN performs poorly when the number of predictors is large relative to the size of the training set (Beyer et al., 1999).[3]

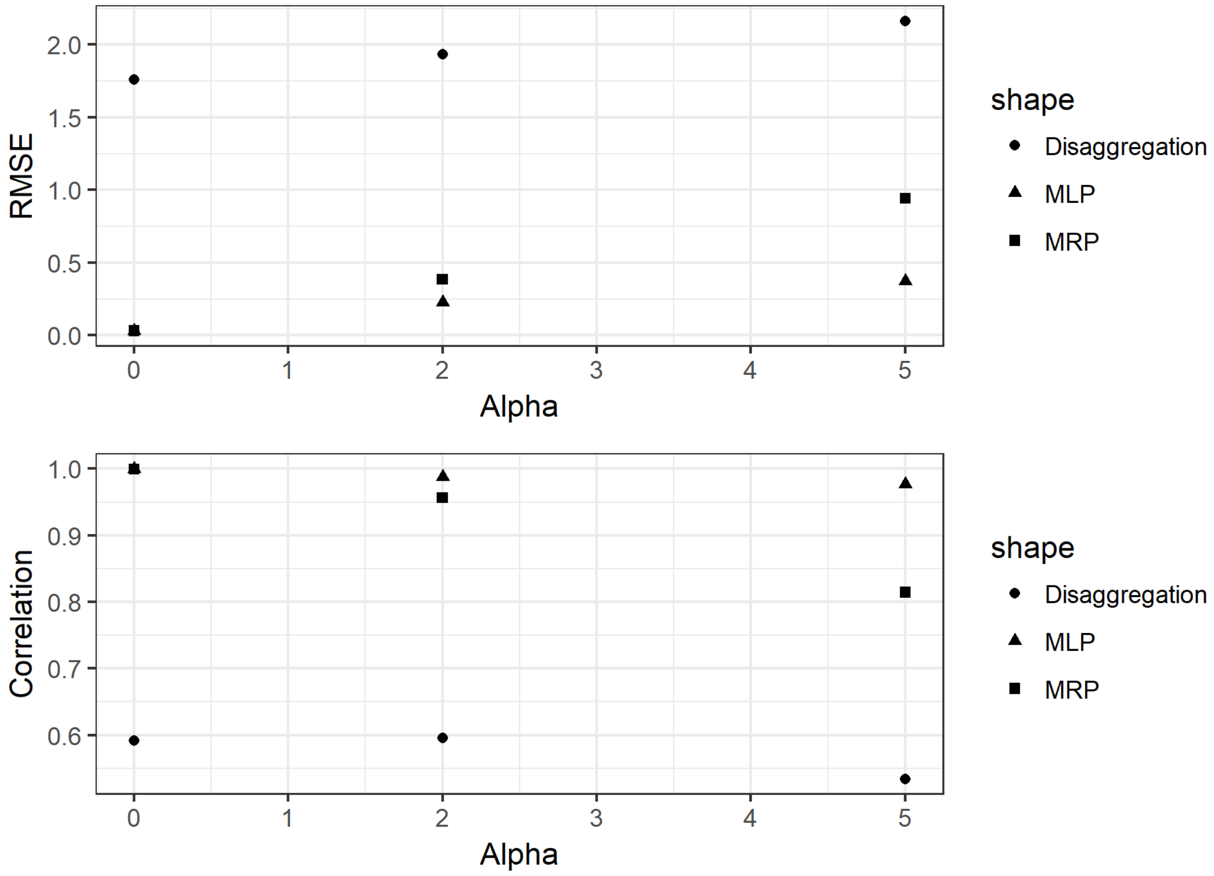

Figure 2: Relative performance of disaggregation, MRP, and MLP estimates, varying α. Parameters Used: $\rho = 0.4$, $n = 2000$, $M = 200$, $N = 15000$, $\sigma^2 = 5$.

Nevertheless, the benefits of MLP can be dramatic under some conditions. In cases where α and ρ are large, MRP performs only modestly better than disaggregation, while MLP produces estimates that are well-correlated with the true unit means. Figure 2 illustrates these relative performance gains for varying levels of α. Even when MLP underperforms MRP, it never performs *poorly*: the worst correlation produced across all simulations was

[3] More precisely, Beyer et al. (1999) show that KNN on high dimensional data will perform poorly regardless of the size of n, owing to the "curse of dimensionality". Euclidean distance does not meaningfully measure "closeness" in spaces with more than 10-15 dimensions.

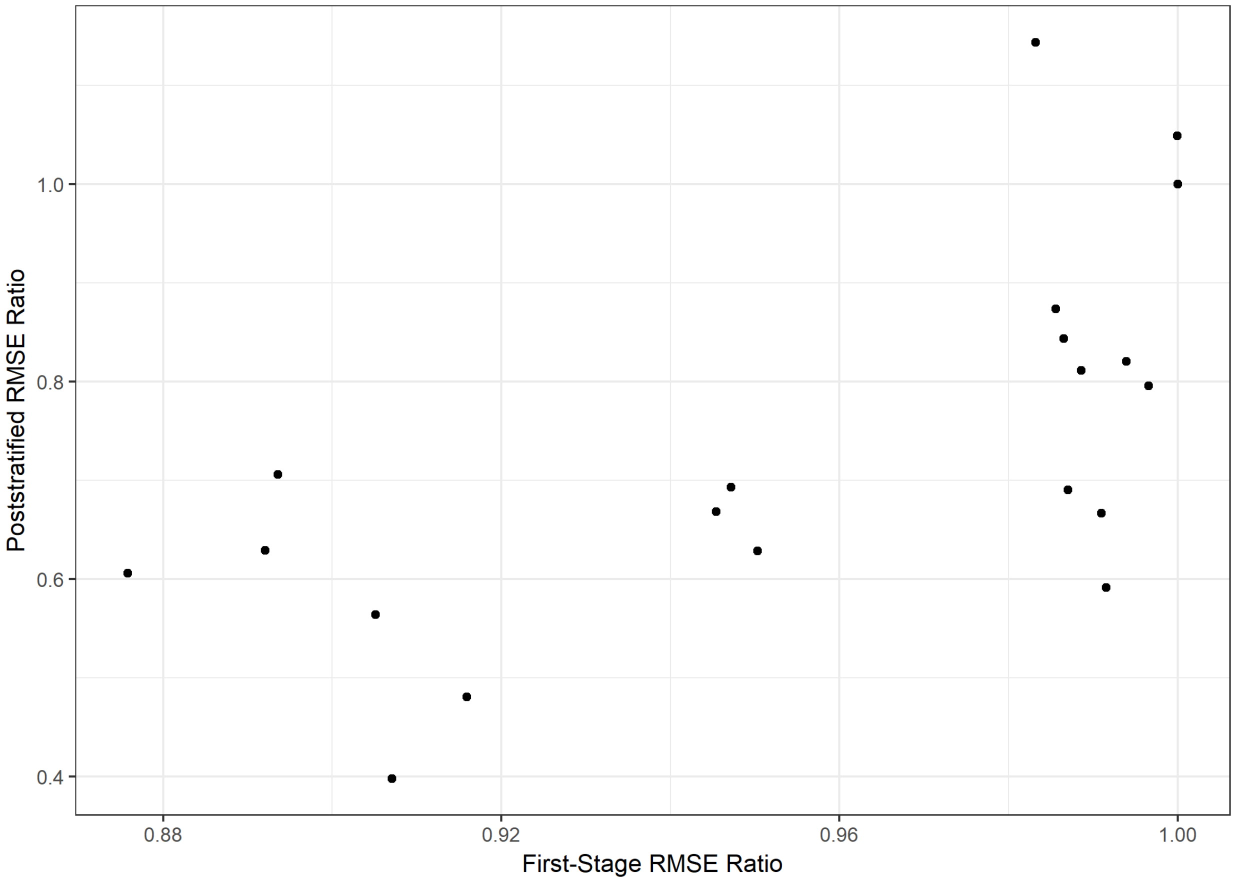

Figure 3: When machine learning outperforms HLM at individual-level prediction, MLP typically produces better poststratified estimates than MRP. Here, the ratio of root mean square error (RMSE) in the first stage is plotted against the RMSE ratio for the poststratified estimates.

a 0.92 for MLP, compared to 0.79 for MRP and 0.35 for disaggregation. The Monte Carlo also demonstrates the value of selecting a first-stage model through cross-validation. As Figure 3 shows, the model that provides better first-stage predictions typically produces better poststratified estimates as well.

4 Empirical Application: 2016 US Presidential Election

How does MLP perform in an empirical application? In this section, I demonstrate that US county-level MLP estimates of conservatism correlates very well with actual county-level

Table 2: Summary of variables included in first-stage models.

Variable	Level	Values	Source
Conservatism	Individual	See Apendix 3	CCES 2016
Race	Individual	{White, Black, Hispanic, Other}	CCES 2016
Age	Individual	{18-29, 30-44, 45-64, 65+}	CCES 2016
Female	Individual	{0,1}	CCES 2016
Education	Individual	{No HS, HS, Assoc's Degree, Bachelor's Degree, Postgraduate}	CCES 2016
Latitude/Longitude	County		Census US Gazetteer Files (2016)
Percent Veterans	County		American Community Survey (2012-2016)
Percent Urban	County		Decennial Census (2010)
Median Household Income	County		American Community Survey (2012-2016)
Percent Evangelical or Mormon	State		Pew Religious Landscape Survey (2014)

presidential vote share 2016, outperforming disaggregation and classical MRP.

For individual-level survey data, I draw on the 2016 Cooperative Congressional Election Survey (CCES), an extensive survey of over 64,000 Americans conducted prior to the 2016 presidential election (Ansolabehere and Schaffner, 2018). From that survey, I collect responses on vote choice, demographics, and geography, as listed in Table 2. Note that, even in such a large survey, estimating county-level public opinion through disaggregation alone is impractical. With over 3,000 counties in the United States, CCES contains roughly 20 observations per county on average. Since respondents are not drawn uniformly across counties, nearly half of the counties have five or fewer respondents in the CCES sample.

And so, if we want to estimate county-level public opinion, we will need a model-based technique. To begin, I first generate the outcome variable. One approach would be to simply use a dichotomous variable indicating whether the respondent planned to vote for Trump in 2016. For this exercise, however, I instead generate a continuous variable measuring conservative public opinion. In this way, I am not throwing out large amounts of useful information on preference intensity.

To generate this continuous variable, I first collect the responses to twenty questions on some of the most salient issues of the 2016 presidential campaign: immigration, gun control, criminal justice, trade, healthcare, and environmental regulation. These variables are catalogued in Appendix 3, Table 6. I then conduct a principal component analysis, taking the first component as my measure of conservatism. This measure is strongly correlated with self-reported intention to vote for Trump.[4]

With this measure in hand, I then use the cross-validation procedure to select the best-fitting individual-level model. The hierarchical linear model is of the following form:

$$y_i = \alpha_0 + \alpha_{j[i]}^{female} + \alpha_{k[i]}^{race} + \alpha_{l[i]}^{education} + \alpha_{m[i]}^{age} + \alpha_c^{county} + \varepsilon_i$$

$$\alpha_{j[i]}^{female} \sim N(0, \sigma_{female}^2)$$

$$\alpha_{k[i]}^{race} \sim N(0, \sigma_{race}^2)$$

$$\alpha_{l[i]}^{education} \sim N(0, \sigma_{education}^2)$$

$$\alpha_{m[i]}^{age} \sim N(0, \sigma_{age}^2)$$

$$\alpha_c^{county} \sim N(\alpha_s^{state} + \beta X_c, \sigma_{county}^2)$$

$$\alpha_s^{state} \sim N(\beta X_s, \sigma^2 state)$$

X_c and X_s are matrices of county-level and state-level variables, respectively, as reported in Table 2. I also train a KNN model (optimal cross-validated fit at $k = 23$) and a random forest, using the predictor variables in Table 2.

The cross-validated prediction error and correlations for each of these models are listed in Table 3. Of the three models, HLM performs the best. However, the best fitting predictions overall come not from a single model, but from an *ensemble model average*, taking the mean prediction of the hierarchical linear model and KNN. This reflects the advantages of combining diverse models into a single prediction (Page, 2008; Montgomery, Hollenbach and

[4] This procedure is similar to that used to generate the ideology measure in Tausanovitch and Warshaw (2014).

Table 3: First-stage 10-fold cross-validation results. An ensemble model average of the hierarchical linear model and KNN (italicized) performs best.

Model	**RMSE**	**Correlation**
Hierarchical Linear Model	1.022	0.332
K-Nearest Neighbors	1.043	0.302
Random Forest	1.058	0.289
Ensemble Model Average (HLM + KNN)	*1.019*	*0.338*
Ensemble Model Average (HLM + KNN + RF)	1.023	0.333

Ward, 2012).

Poststratifying the predictions from the HLM at the county-level yields my MRP estimates, and poststratifying the EMA predictions yields my MLP estimates. I also generate disaggregated estimates, taking the county-level mean of my outcome variable. Figure 4 compares these estimates against the true 2016 presidential vote shares by county. Clearly, disaggregation fares worst, particularly in small counties with few CCES respondents. MRP and MLP both perform significantly better, while MLP is the most strongly correlated of the three.

Although MLP's performance improvement seems modest when looking at the country as a whole, the difference is striking at the state-level. Figure 5 plots a few illustrative examples, while Figure 6 gives a more comprehensive overview. Within nearly every state, MLP correlates better with 2016 results than does MRP, and in some cases dramatically so. It is notable that MLP underperforms MRP in small Great Plains states like Wyoming, Montana, and the Dakotas. Because these states are geographically isolated and contain few CCES respondents, the KNN predictor must produce its estimates based on more distant observations.

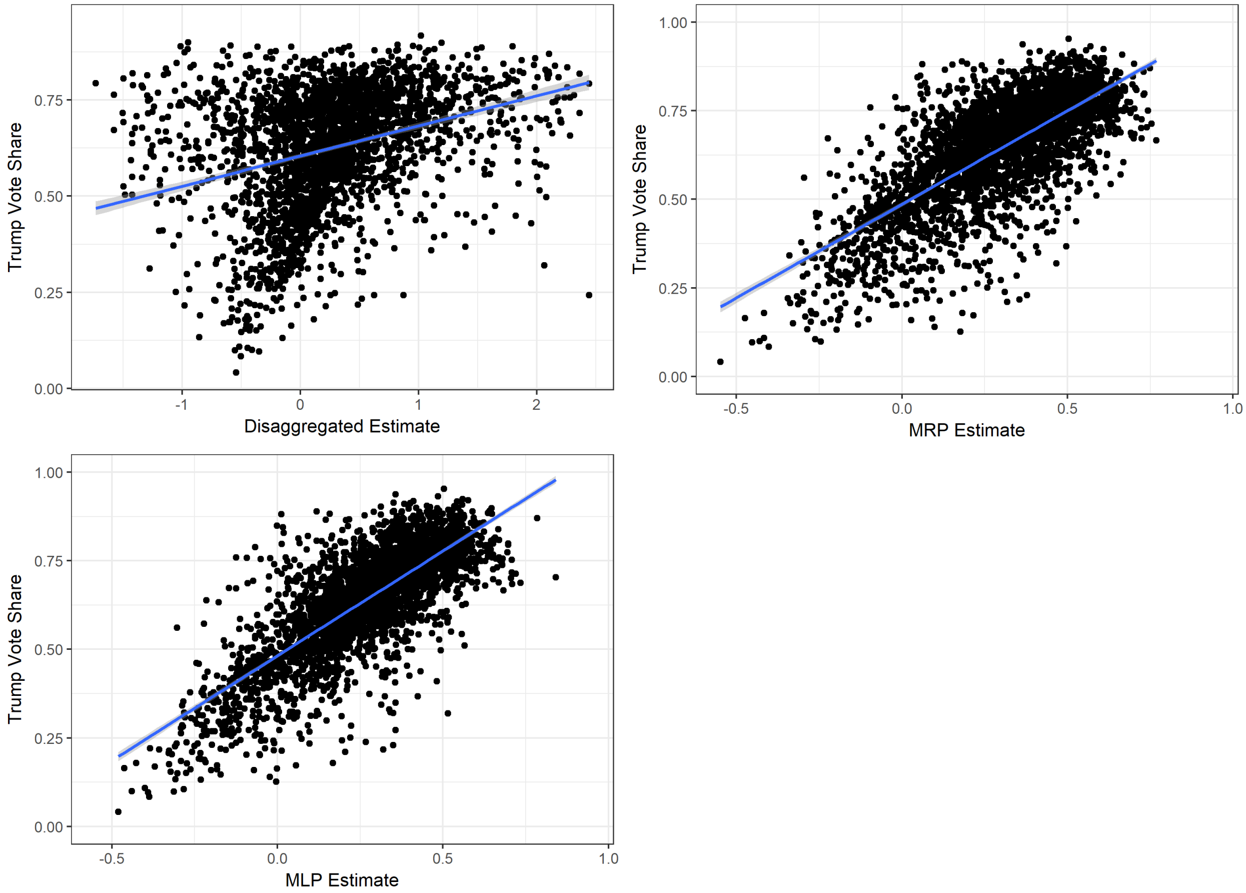

Figure 4: Trump 2016 vote share plotted against disaggregated, MRP, and MLP estimates. Correlations are 0.32, 0.72, and 0.77 respectively.

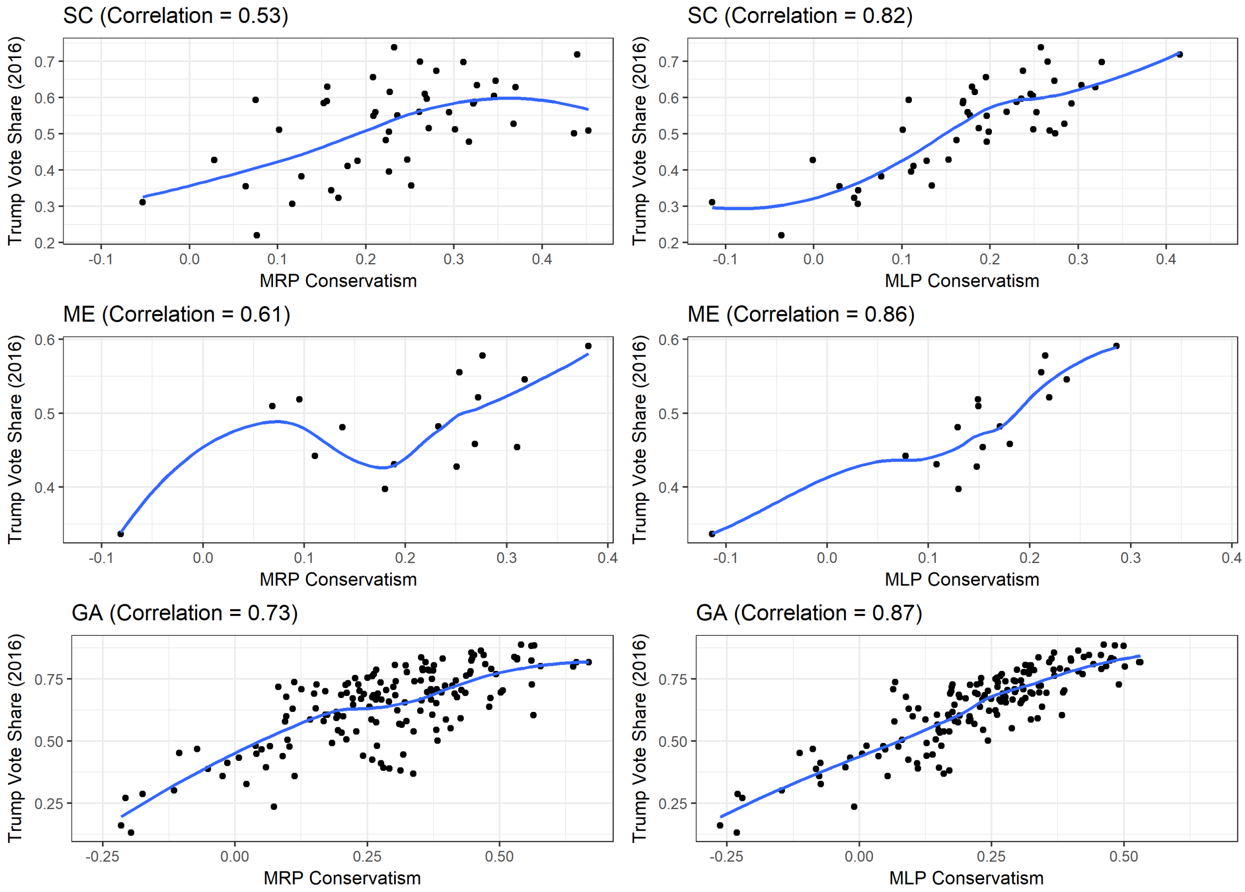

Figure 5: MLP and MRP estimates in select states, plotted against 2016 presidential vote shares.

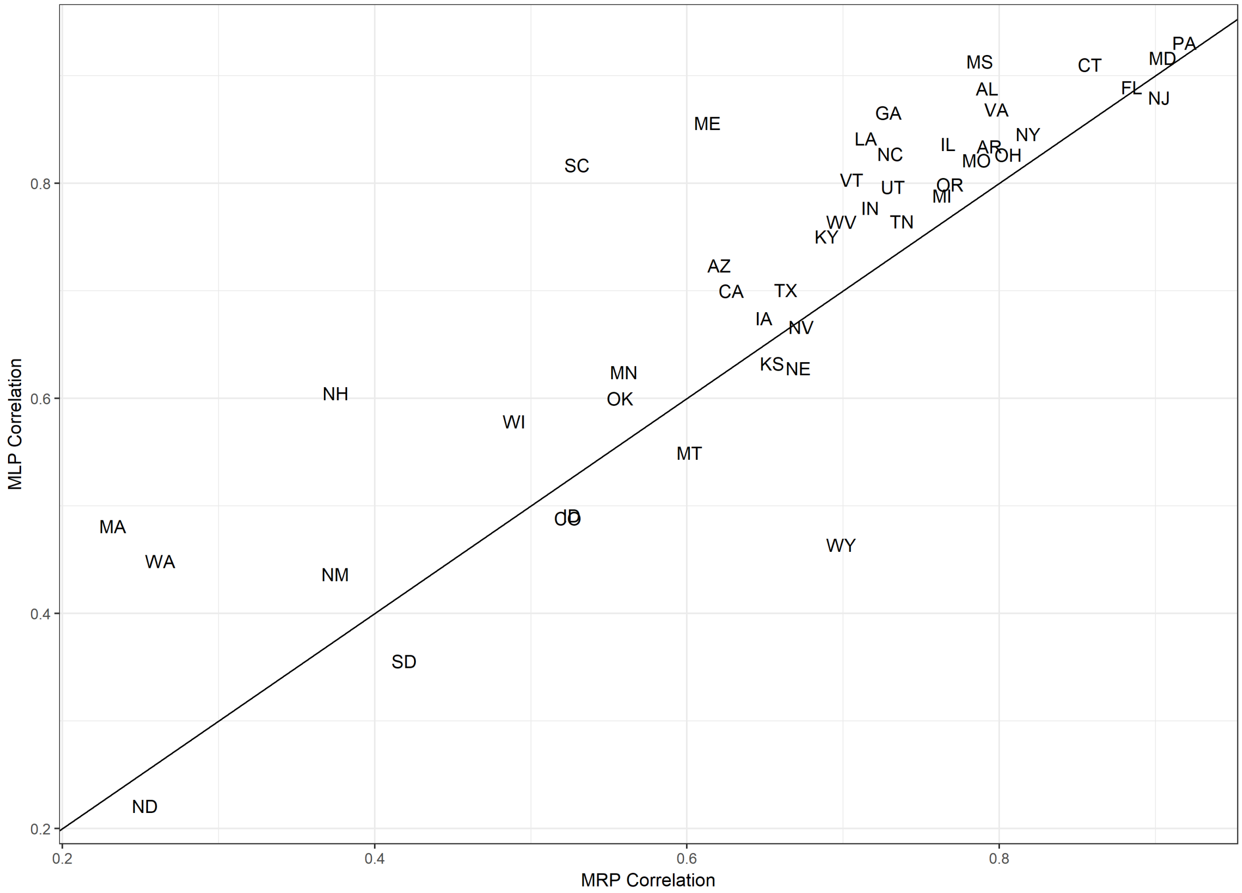

Figure 6: MLP and MRP correlations with 2016 presidential vote share by state. In nearly all cases, MLP outperforms MRP, in some cases considerably.

5 Conclusion

In this paper, I have developed a generalization of MRP, which expands the set of candidate first-stage models. Machine learning algorithms can produce significant improvements in local area public opinion estimates, particularly when the relationship between opinion and demographic variables is nonlinear. It is important to to note, however, that MLP does not always produce better estimates than classical MRP. As the Monte Carlo analysis demonstrates, MLP will only outperform MRP when the data generating process is complex, with nonlinear interactions that are unlikely to be specified in advance by the researcher's model. Fortunately, cross-validation provides a principled method to determine whether MLP is likely to outperform MRP, and to select from among this new menagerie of first-stage models.

In future work, I hope to further expand the set of MLP first-stage models. Although I focus in this paper on RF and KNN, there may perhaps be other techniques better-suited to modeling public opinion. New methodological research could test the technique on a broader range of issue areas, and observe the set of public opinion topics where it over- and under-performs relative to MRP. Ultimately, I hope that MLP proves to be a useful addition to the empirical social scientist's toolbox, spurring further research into subnational politics.

References

Ansolabehere, Stephen and Brian F Schaffner. 2018. "CCES Common Content, 2016.".
URL: *https://doi.org/10.7910/DVN/GDF6Z0*

Beyer, Kevin, Jonathan Goldstein, Raghu Ramakrishnan and Uri Shaft. 1999. "When Is "Nearest Neighbor" Meaningful?" *International conference on database theory* pp. 217–235.

Breiman, Leo. 2001*a*. "Random forests." *Machine Learning* 45(1):5–32.

Breiman, Leo. 2001*b*. "Statistical Modeling: The Two Cultures." *Statistical Science* 16(3):199–231.

Buttice, Matthew K. and Benjamin Highton. 2013. "How does multilevel regression and poststratification perform with conventional national surveys?" *Political Analysis* 21(4):449–467.

Lax, Jeffrey R. and Justin H. Phillips. 2009. "How Should We Estimate Public Opinion in The States?" *American Journal of Political Science* 53(1):107–121.

Lax, Jeffrey R. and Justin H. Phillips. 2012. "The Democratic Deficit in the States." *American Journal of Political Science* 56(1):148–166.

Leemann, Lucas and Fabio Wasserfallen. 2017. "Extending the Use and Prediction Precision of Subnational Public Opinion Estimation." *American Journal of Political Science* 61(4):1003–1022.

Montgomery, Jacob M, Florian Hollenbach and Michael D Ward. 2012. "Improving Predictions Using Ensemble Bayesian Model Averaging." *Political Analysis* 20(3):271–291.

Montgomery, Jacob M and Santiago Olivella. 2018. "Tree-Based Models for Political Science Data." *American Journal of Political Science* 62(3):729–744.

Page, Scott E. 2008. "Uncertainty, Difficulty, and Complexity." *Journal of Theoretical Politics* 20(2):115–149.

Park, David K., Andrew Gelman and Joseph Bafumi. 2004. "Bayesian multilevel estimation with poststratification: State-level estimates from national polls." *Political Analysis* 12(4):375–385.

Samworth, Richard J. 2012. "Optimal weighted nearest neighbour classifiers." *Annals of Statistics* 40(5):2733–2763.

Stone, M. 1974. “Cross-Validatory Choice and Assessment of Statistical Predictions.” *Journal of the Royal Statistical Society, Series B (Methodological)* 36(2):111–147.

Tausanovitch, Chris and Christopher Warshaw. 2014. “Representation in Municipal Government.” *The American Political Science Review* 108(03):605–641.

Warshaw, Christopher and Jonathan Rodden. 2012. “How Should We Measure District-Level Public Opinion on Individual Issues?” *The Journal of Politics* 74(1):203–219.

Appendix 1: Synthetic Poststratification Proof

In this appendix, I demonstrate that synthetic poststratification and classical MRP produce identical estimates if the first-stage model is additively-separable. Let $\hat{\mathbf{y}}$ be the vector of predictions for each type of respondent, and $\mathbf{p}$ be the true empirical pmf for each type. The classical MRP poststratified estimate is the dot-product $\hat{\mathbf{y}} \cdot \mathbf{p}$. MrsP uses the same vector of predictions $\hat{\mathbf{y}}$, but uses a synthetic joint probability distribution, where each entry is the product of marginal probabilities. I will denote this synthetic poststratification vector as $\boldsymbol{\pi}$. Therefore, the poststratified MrsP estimates will be $\hat{\mathbf{y}} \cdot \boldsymbol{\pi}$.

Let X_1 through X_m be discrete random variables, and the $c \times m$ matrix X be a matrix in which the each row is one of the c possible combinations of values that X_1 through X_m can take. Crucially, we are not assuming that X_1 through X_m are independent, so $P(X_1 = x_{1i}, ..., X_m = x_{mk})$ need not equal $P(X_1 = x_{1i})...P(X_m = x_{mk})$.

Suppose the model is additively separable, such that $\hat{y} = X\hat{\beta}$. The vector of MrsP predictions for each unit is therefore $\pi'\hat{y}$, where π is the synthetic distribution vector. To complete the proof, we must show that $p'X\hat{\beta} = \pi'X\hat{\beta}$. Because β is a vector, this is equivalent to showing that $p'X = \pi'X$:

$$p'X = \begin{bmatrix} \sum_i ... \sum_k P(X_1 = x_{1i}, ..., X_m = x_{mk}) x_{1i} \\ \vdots \\ \sum_i ... \sum_k P(X_1 = x_{1i}, ..., X_m = x_{mk}) x_{mk} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_i P(X_1 = x_{1i}) x_{1i} \\ \vdots \\ \sum_k P(X_m = x_{mk}) x_{mk} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_i ... \sum_k P(X_1 = x_{1i}) ... P(X_m = x_{mk}) x_{1i} \\ \vdots \\ \sum_i ... \sum_k P(X_1 = x_{1i}) ... P(X_m = x_{mk}) x_{mk} \end{bmatrix} = \pi' X$$

This completes the proof. If our underlying first-stage model is additively separable, then our poststratified estimates will be identical whether we use MrsP or classical MRP.

Appendix 2: Monte Carlo Technical Summary

The X variables are generated by discretizing each Z variable, according to procedure in Table 4. Subnational units are assigned using the Z_4 variable. The N observations with the smallest value of Z_4 are assigned to Unit 1, the next smallest N observations assigned to Unit 2, and so on.

Table 4: Assignment procedure for X variables

Z	X
Less than 1 SD below mean	1
1 SD below mean to mean	2
Mean to 1 SD above mean	3
More than 1 SD above mean	4

The functions D^0 and D^1 in the data-generating process are defined as follows, so that the former is increasing as it approaches (0,0), while the latter is decreasing.

$$D_i^0 = \sqrt{2} - \sqrt{lat_i^2 + lon_i^2}$$

$$D_i^1 = \frac{\sqrt{lat_i^2 + lon_i^2}}{2}$$

Table 5 lists the parameter values swept in the Monte Carlo. All simulation code will be made available at the author's website.

Table 5: List of parameter values used in the Monte Carlo Simulation

Parameter	**Values**	**Description**
ρ	{0.2, 0.4, 0.6}	Correlation between Z variables
α	{0, 2, 5}	Strength of the threeway interaction effect
n	{2000, 5000, 10000}	Sample size drawn for disaggregation, MRP, and MLP estimates
N	15000	Observations per unit
M	200	Number of units
σ^2	5	Error term variance in DGP

Appendix 3: Generating the Outcome Variable (CCES)

The outcome variable in the empirical application is generated from the twenty variables reported in Table 6. Each variable has a binary outcome (with 1 representing the conservative opinion), producing a vector of length 20 for each respondent. Taking the first component from a principal component analysis maps each individual onto a unidimensional measure of conservatism.

Table 6: The CCES public opinion questions used to generate the outcome variable in the empirical application.

Opinion	**CCES Code**	**Category**
Background Checks	CC16_330a	Gun Control
Publishing Gun-Owners Names	CC16_330b	Gun Control
Assault Rifle Ban	CC16_330d	Gun Control
Concealed Carry	CC16_330e	Gun Control
Legal Status for Employed Immigrants	CC16_331_1	Immigration
More Border Patrol	CC16_331_2	Immigration
DACA	CC16_331_3	Immigration
Deportation	CC16_331_7	Immigration
CO2 Regulation	CC16_333a	Environment
Fuel Efficiency Standards	CC16_333b	Environment
Renewable Energy	CC16_333c	Environment
Clean Air Act	CC16_333d	Environment
Mandatory Minimums	CC16_334a	Criminal Justice
Police Body Cameras	CC16_334b	Criminal Justice
Increase Police Force	CC16_334c	Criminal Justice
Three Strike Laws	CC16_334d	Criminal Justice
TPP	CC16_351B	Trade
Iran Sanctions	CC16_351G	Foreign Policy
Repeal ACA	CC16_351I	Healthcare
Minimum Wage	CC16_351K	Economy