Visualization Strategies for Regression Estimates with Randomization Inference¹

Marshall A. Taylor²

*Department of Sociology
New Mexico State University*

Abstract

Coefficient plots are a popular tool for visualizing regression estimates. The appeal of these plots is that they visualize confidence intervals around the estimates and generally center the plot around zero, meaning that any estimate that crosses zero is statistically non-significant at at least the alpha-level around which the confidence intervals are constructed. For models with statistical significance levels determined via randomization models of inference and for which there is no standard error or confidence intervals for the estimate itself, these plots appear less useful. In this paper, I illustrate a variant of the coefficient plot for regression models with p -values constructed using permutation tests. These visualizations plot each estimates p -value and its associated confidence interval in relation to a specified alpha-level. These plots can help the analyst interpret and report both the statistical and substantive significance of their models. Illustrations are provided using a nonprobability sample of activists and participants at a 1962 anti-Communism school.

Keywords: Permutation Tests, Visualization, Coefficient Plots, P -Values

---

---

¹I would like to thank the editorial team at *Stata Journal* and an anonymous reviewer for their helpful feedback. I would also like to thank Omar Lizardo and Dustin S. Stoltz for their thoughts on earlier versions of this paper.

²orcid.org/0000-0002-7440-0723. Direct all correspondence to Marshall A. Taylor, Department of Sociology, New Mexico State University, Science Hall 286, 1290 Frenger Mall, Las Cruces, NM, 88003-0001; email: mtaylor2@nmsu.edu. A replication repository for this article can be found at: <https://github.com/Marshall-Soc/PPV>.

1. Introduction

Visual alternatives to traditional estimate tables have become increasingly popular in the social sciences, especially with the growing interest in data science and computational social science (Healy 2018; Wickham and Grolemund 2017). One visualization that is supplanting more traditional regression tables is the coefficient plot (Gelman et al. 2016; Jann 2014; Lander 2018). Coefficient plots can provide a more parsimonious method for reporting regression coefficients, especially when a larger number of independent variables and/or models make tables difficult to read or present. These plots have become quite popular in sociological research for this very reason, ranging in application from cultural sociology (e.g., Lizardo and Skiles 2016a, 2016b; Stewart, Edgell, and Delehanty 2018) and social networks (Bhr and Abraham 2016), to the sociology of education (Schhrer, Carbonaro, and Grodsky 2016) and the sociology of work and occupations (Friedman and Laurison 2017; Laurison and Friedman 2016).

Coefficient plots visualize the confidence intervals and their corresponding regression estimates. The graphs are generally centered around zero—meaning that any estimate that crosses zero is statistically non-significant at at least the alpha-level around which the confidence intervals are constructed. Consider, for example, the coefficient plot in Figure 1 below, which was generated from a basic linear-additive multiple OLS model regressing systolic blood pressure readings on a vector of covariates using the National Health and Nutrition Examination Survey data available from StataCorp. The plot is interpreted just like a regular regression table: black individuals, for example, have systolic blood pressure readings that, on average, are

about 2.333 mm/Hg. higher than white individuals' readings, net of their weight, height, sex, age, and self-reported health. The plot also shows that this difference is statistically significant with at least an $\alpha = .001$, since the thinnest band does not cross zero (the vertical line).

Figure 1: Coefficient Plot of OLS Estimated Effects on Systolic Blood Pressure
Note: Constant = 127.028 ($p < .001$). $n = 10,335$. Race reference category = White.
 Unstandardized coefficients reported. Confidence intervals are two-sided.

For all their utility, coefficient plots still come with assumptions about the inference statistics accompanying the estimates—namely, coefficient standard errors and their confidence intervals. This is, of course, a valid assumption for the majority of analysts using probability samples and relying on the asymptotic properties of random sampling distributions. However, for analysts using nonprobability samples, a different inference model must be adopted. Such an alternative—the

“randomization model” (Ernst 2004; Ludbrook and Dudley 1998) or what might be termed the “process inference” model (Darlington and Hayes 2017:513-14)—allows for sample-specific inferences by using Monte Carlo permutation tests to derive empirical p -values. These p -values indicate the proportion of test statistics—say, a specific coefficient—from the models computed from the randomly permuted samples that are greater than or equal to the observed test statistic (in the case of a right-tailed test).

The randomization model poses problems for the standard coefficient plot. Specifically, unlike the conceptually similar bootstrap technique, standard errors derived from Monte Carlo permutation tests are with respect to the p -value—not the coefficient itself. As such, there are no coefficient-specific confidence intervals to report and no way to visualize the effect’s proximity to zero without explicit reference to the effect’s p -value.

In this paper, I illustrate a variant of the coefficient plot for regression models with p -values constructed using permutation tests and discuss its utility in interpreting and presenting research. These visualizations, which I refer to as PPV plots, plot each estimate’s p -value and its associated confidence interval in relation to an alpha-level of the analyst’s choice. These plots can help the analyst interpret and report both the statistical and substantive significance of their models with permutation-based inference without recourse to tables. The paper unfolds in the following steps. First, I provide a very brief overview of the logic behind Monte Carlo permutation tests and how they can be used to construct sample-specific inferences, noting especially how they introduce complications for how regression estimates and their associated pre-

dicted values are visualized. Second, I introduce PPV plots, illustrate how they can be generated, and demonstrate their utility for visualizing regression estimates and predicted values. PPV plot illustrations are made using a series of models predicting heightened perceptions of communist threat among a nonprobability convenience sample of people who participated in the Christian Anti-Communism Crusade’s San Francisco Bay Region School of Anti-Communism in January and February of 1962 (Wolfinger et al. 1992). I then conclude with an outline of how PPV plots can be further expanded for data visualization.

2. Randomization Inference

2.1. Using Monte Carlo Permutation Tests to Generate P-Values

The traditional “population model” (Ernst 2004; Ludbrook and Dudley 1998) of statistical inference with which most social scientists are accustomed is not always appropriate when the sample is constructed using nonprobabilistic procedures or when the sample suffers from non-random systematic error that cannot be mitigated with bias-correcting statistical adjustments. When such “pre-statistical” methodological constraints are present, a different model must be adopted if the analyst wishes to use the language of inferential statistics. One option is the “randomization” (Ernst 2004; Ludbrook and Dudley 1998; Manly 2007:1-4) or “process inference” (Darlington and Hayes 2017:513-14) model—a strategy that has, unfortunately, been slow to gain ground in sociological research using observational data,³ save for work in

---

³The lack of a probability sample can often lead researchers to avoid the language of statistical inference altogether. In reviewing work using qualitative comparative analysis (QCA), for example,

social network analysis.⁴ The key distinction between the population model and the randomization model is the “thing” to which statistical inferences are being made. In both cases, the p -value and/or its associated confidence interval (for the coefficient in the population model and the p -value itself in the randomization model) are used to assess the confidence in the inference.⁵

Consider the following simple OLS estimation equation:

$$\hat{Y}_i = \hat{a} + \hat{\beta}X_i. \quad (1)$$

On the one hand, if p is the p -value attached to some estimated coefficient $\hat{\beta}$, then, under the population model and assuming a two-tailed test, p can be expressed as:

$$p_{\hat{\beta}} = p \left(|\beta| \geq |\hat{\beta}| \mid H_0 : \beta = 0 \right), \quad (2)$$

Thomann and Maggetti (2017:12) note that “[f]ew QCA applications resort to statistical inference, partly because its use on full or purposive samples yields nonsensical inference to a nonexistent population.”

⁴Permutation-derived p -values have been a common characteristic in statistical models for networks, since the interdependencies inherent in relational data necessarily violate the observational independence assumptions behind asymptotic test statistics (see, e.g., Butts 2007; Krackhardt 1988).

⁵Randomization inference is often (if not usually) discussed within the context of randomized experiments (e.g., Aronow 2012; Edgington and Onghena 2007; Ludbrook and Dudley 1998), where random assignment of participants into treatment and control groups is assumed but not necessarily random sampling. Nonetheless, as Manly (2007:180) notes, justification for using permutation tests for establishing statistical inference in multivariate cases can come in three forms: (1) in the case of random sampling, when the H_0 is that $Y \perp \mathbf{X}$; (2) in the case of random assignment to treatment and control conditions (but not necessarily in conjunction with random sampling); and (3) when, with observational data that is from neither a random sample nor following an experimental design, “suggest that the mechanism generating the data makes any of the Y values equally likely to occur with any of the sets of X values.” This paper adopts the “data-generating mechanism” justification for using permutation tests, given the prevalence of observational data relative to experimental data in sociology.

where β is the true but unobserved estimate of the effect of X on Y . This p -value expresses the probability that (if the coefficient was truly generated from a random sample from the target population and with no systematic error) an effect of at least the absolute size of $\hat{\beta}$ would be observed from a sample of that population if the true effect $\beta = 0$.

On the other hand, under the randomization model with 1,000 Monte Carlo permutation tests and again assuming a two-tailed test, $\hat{p}$ indicates the proportion of $\hat{\beta}^*$ estimates from the 1,000 permutations with an absolute value greater than or equal to the absolute value of $\hat{\beta}$ observed prior to randomly permuted the data (see also Baker and Hubert 1981:347).⁶ (The hat over the p will be explained in the next section.) In other words, this p -value can be written as:

$$\hat{p}_{\hat{\beta}} = \frac{\sum_{i=1} I \left(\left| \hat{\beta}_i^* \right| \geq \left| \hat{\beta} \right| \right)}{k}, \quad (3)$$

where k is the number of permutations, $\left| \hat{\beta}_i^* \right|$ denotes the i th estimate derived from one of the k permutations that is larger than or equal in absolute value to $\left| \hat{\beta} \right|$, and $I(\cdot)$ is an indicator function that tallies the total number of times $\left| \hat{\beta}_i^* \right| \geq \left| \hat{\beta} \right|$ (Ernst 2004:679; Stoltz and Taylor 2017:33). Making generalizations to some population is, of course, not possible under the randomization model (at least on statistical grounds) since the asymptotic properties that come with assuming a probability

---

⁶The variable that is permuted may vary. In experimental settings, the treatment variable, X , is often permuted while keeping the outcome vector Y the same. Others promote incorporating some form of the model residuals into the permutation procedure (e.g., Freedman and Lane 1983; Ogorman 2006). Still others suggest simply permuting the Y vector (Manly 2007; Stoltz and Taylor 2017). For an overview of different permutation techniques, see Anderson and Robinson (2001).

sample free of non-random error is a luxury not enjoyed for most analysts who adopt the randomization model over the population model. Instead, working from a null hypothesis that $\hat{\beta}$ can be accounted for by randomness, a $\hat{p}$ under some alpha-level would indicate, whatever the data-generating process that created $\hat{\beta}$ might be, it is likely not a purely random process. In other words, we reject the null hypothesis when randomness as a mechanism for generating the observed effects can be ruled out at some level of confidence. In this case, we test “randomness” by creating it artificially: i.e., by using the permutation procedure to randomly shuffle the observed Y_k s across the X_k s and therefore “forcing” the association between Y_k and X_k to be stochastic (Manly 2007:2). When the $\hat{p}$ associated with the observed $\hat{\beta}$ is under a specified α , then $\hat{\beta}$ was likely generated from a systematic process since its absolute size is larger than or equal to at least $1 - \alpha$ of the total $|\hat{\beta}^*|$, which *were* generated via an artificial random data-generating process. The emphasis on quantifying the likelihood of a random versus systematic data-generating process as opposed to estimating the probability of observing a given observed effect in some population under the condition that there isn’t one is the defining feature that differentiates the process model from the population model of inference.

2.2. *Complicating the Coefficient Plot*

Regression estimates generated under the randomization model pose unique problems for visualization. When it comes to the popular coefficient plot, the problem is quite fundamental: there are no standard errors or confidence intervals for the coefficient, $\hat{\beta}$, so a visualization such the one in Figure 1 above is not possible. Standard errors and confidence intervals are instead computed for the coefficient p -value,

$\hat{p}_{\hat{\beta}}$. Importantly, though $SE(\hat{\beta})$ and $CI(\beta)$ cannot be calculated without again assuming asymptotic distributional properties of $\hat{\beta}$, $SE(\hat{p}_{\hat{\beta}})$ and $CI(p_{\hat{\beta}})$ can be computed because they are with respect to k and not n (the observation sample size). Assuming that k is a random subset of the total number of possible permutations K , then $SE(\hat{p}_{\hat{\beta}})$ indicates the average deviation of any randomly sampled p -value $\hat{p}_{\hat{\beta}}$ from the true but unobserved $p_{\hat{\beta}}$ if every possible permutation were considered.⁷ $CI(p_{\hat{\beta}})$ then provides the range of p -values within which the true $p_{\hat{\beta}}$ would be located with a $1 - \alpha$ degree of confidence.⁸

The fact that standard errors and confidence intervals for p -values are still valid provides leverage for constructing a new type of coefficient plot for visualizing regression estimates and their subsequent predicted values under the randomization model. This new series of plots, which I refer to collectively as PPV plots, is outlined in the following section.

---

⁷Of course, $p_{\hat{\beta}}$ could be observed if every possible permutation can be accounted for.

⁸ $SE(\hat{p}_{\hat{\beta}})$ can be found with the standard error formula for a proportion, since $p_{\hat{\beta}}$ is necessarily bound between 0 and 1:

$$SE(\hat{p}_{\hat{\beta}}) = \sqrt{\frac{\hat{p}_{\hat{\beta}}(1 - \hat{p}_{\hat{\beta}})}{k}}, \quad (4)$$

and $CI(p_{\hat{\beta}})$ is then found with:

$$CI(p_{\hat{\beta}}) = \hat{p}_{\hat{\beta}} \pm (z \times SE(\hat{p}_{\hat{\beta}})). \quad (5)$$

3. The PPV Plot Toolkit

A PPV plot—short for “Permutation P -Value” plot—can be thought of as a coefficient plot for regression models with significance levels determined via permutation tests. Its structure is quite simple: instead of arraying model coefficients within a variable-by- Y -unit space, the p -values for each model coefficient are arrayed within a variable-by- $[0, 1]$ space. Unlike a traditional coefficient plot centered around 0 to indicate no statistically significant effect, the PPV plot is centered around an analyst-determined alpha-level. In this paper, I use a combination of Stata’s `permute` (StataCorp n.d.) command and a modified version of Jann’s `coefplot` (Jann 2014) command to construct PPV plots. The code to reproduce all tables and visualization is available as supplementary material.

I illustrate how PPV plots can be used to visualize regression coefficients and post-estimation predicted values. Each strategy is addressed in turn below. First, however, I provide a brief overview of the case that I use for illustration: perceptions of communist threat among a nonprobability convenience sample of far-right activists who participated in the 1962 San Francisco Bay Region School of Anti-Communism (Wolfinger et al. 1992).

3.1. *The Case: An Anti-Communism School*

The Christian Anti-Communism Crusade (CACC) was founded in 1953 by Fred Schwarz. Schwarz, an Australian originally trained in medicine, spearheaded the “second wave of religious anti-Communism” in the United States (Herzog 2011:207) and started CACC in an attempt to illuminate and teach on the insidious activities

of Communism—which was believed to be “the very mask of Satan” (The Schwarz Report 2018). CACC generated considerable funds by “promoting the belief that no bilateral negotiation could exist with communism” (Aiello 2005), largely through their tax-exempt and tuition-based “anti-Communism schools” that the organization would put together. In these schools, Schwarz and other organizers would teach on the un-American perils of Communism and put on fundraising activities (Wolfinger et al. 1964:22).

The fleeting nature of these meetings posed problems for academic researchers wishing to survey the attendees at these “schools.” No clearly definable sampling frame of participants existed—and, even if it did, it was unlikely that such a list would have been provided to researchers given the high skepticism of colleges and universities amid the McCarthyism of the time. Nonetheless, Wolfinger and colleagues wished to know just who these school attendees were and what they believed. To this end, they adopted purposive sampling techniques to gather data from attendees at the San Francisco Bay Region School of Ant-Communism, held from January 29 to February 2, 1962, at the Oakland Auditorium in Oakland, California. Wolfinger and colleagues describe their data collection process as follows:

We were able to conduct a study of the personal characteristics and political attitudes and activities of several hundred people who attended this Crusade school. Under our direction two dozen Stanford undergraduates interviewed members of the audience and distributed questionnaires (similar to the interview forms) which could be completed later and mailed back to Stanford. (Wolfinger et al. 1964:23)

As the excerpt suggests, respondents constituted a convenience (and clearly non-probabilistic) sample. Given that participants were not known beforehand and were there for activities completely unrelated to data collection efforts, data had to be gathered in a way that was less than optimally systematic. Using the population model to draw statistical inferences with this dataset would be inappropriate since the sample was not a random “draw” from a defined target population of American anti-Communists (or even the “population” of anti-Communism school attendees), meaning that asymptotic distributional properties of probability sampling cannot be assumed. The randomization model, however, can be used to draw sample-specific inferences on the likelihood that statistical associations and relationships could arise from a random data-generating process. These data (Wolfinger et al. 1992) are used in the remainder of this paper to illustrate the utilities of the PPV plot toolkit. More specific information on the variables used are discussed in the below sections.

3.2. PPV Plots and Regression Coefficients

Consider the following pseudo-research question: Among the participants at this school, what were some of the sociodemographic factors that predicted a heightened perception of impending Communist threat (given, of course, an already heightened baseline perception of Communist threat relative to the general population since they self-selected into an anti-Communism school)?⁹ To address this question, I con-

---

⁹This research question is “pseudo” in the sense that, from the tradition of quantitative sociology, it really only identifies one half of what would be expected from a proper question: the outcome, i.e., heightened perceptions of Communist threat. The “missing piece” here would be the specific independent variable (or variables) that we theorize will account for variation in that outcome. For illustrative purposes, however, this will suffice.

structed factor scores from a series of dummy variables related to different perceptions of Communist dangers (e.g., Communism in academia and Communism in the Democratic Party). Details on how these variables were constructed can be found in Appendix A, but suffice to say that higher scores indicate higher perceptions of Communist threat. This variable was then regressed on the following variables: degree of political engagement, party identification, level of educational attainment, union member status, and whether or not the respondent was strictly self-employed.¹⁰ All predictor variables were observed variables, save for the political engagement score, which was also constructed using factor analysis on a series of dummy variables. (Details on how these factor scores were constructed are in Appendix A.)

The results from this OLS model could be arranged within a table, as shown in Table 1. This table is very similar to a traditional regression table, with the only consequential difference being that the coefficient p -values rather than standard errors are reported in the parentheses.

Alternatively, the model could be represented graphically as a simple PPV plot, as illustrated in Figure 2. Like Figure 1, the y -axis lists the predictor variables. Unlike Figure 2, the x -axis can vary within the $[0, 1]$ interval and is used to plot the coefficient p -values and their associated confidence intervals. The markers are the p -values and the numbers above them are the coefficients themselves. The vertical line represents $\alpha = .05$ —meaning that any coefficient with a p -value that is to the left

---

¹⁰I qualify self-employment with “strictly” because, in addition to the “self-employed” category, the survey also included an option for those who were self-employed and employed by someone else. I collapsed this “dual employment” category with those who were employed but not self-employed, retired, unemployed, and who were students.

Table 1: OLS Model Predicting Heightened Perceptions of Communist Threat (std) with Permutation-Derived P -Values

IV	Model
Political Engagement (std)	-0.236 (0.103)
Self-Employed	-0.175 (0.573)
Republican	-0.525 (0.393)
Other	-0.670 (0.336)
Union Member	0.268 (0.458)
Grammar School	0.155 (0.855)
College	-0.580 (0.159)
Constant	1.021 (0.137)
n	56
R^2 (adj.)	0.042
RMSE	0.979
# of Permutations of DV	1,000

Note: P -values in parentheses. Reference categories for the party and schooling variables are Democrat and High School, respectively. All schooling categories are “some/complete” for the corresponding level.

of the line is statistically significant at at least the .05 level. For instance, consider the self-employment coefficient, $\hat{\beta}_{SE} = 0.175$, which has a $\hat{p} = .573$. This p -value indicates that after randomly permuting the perception of Communist threat variable across observations 1,000 times and re-computing the difference in intercept between those who were strictly self-employed and those who were not each time, 57.3% of the permutation estimates had absolute sizes that were greater than or equal to the absolute size of the observed estimate of 0.175; as such, one could say that at an alpha-level of at least .05, it is highly possible that randomness can account for the association between self-employment and perceptions of Communist threat in this sample. Further, the p -value's 95% confidence interval is far to the right of the vertical line—suggesting the p -value itself is not statistically significant (i.e., it is likely larger than .05 if all permutations were accounted for and not just 1,000 of them).

Conceptually, a PPV plot such as Figure 2 is constructed in the following simple steps. First, the regression model itself estimated. Assume, for instance, the following linear-additive estimation model:

$$\hat{Y}_i = \hat{a} + \hat{\beta}_1 X_{1i} + \hat{\beta}_2 X_{2i} + \cdots + \hat{\beta}_m X_{mi}, \quad (6)$$

where the first entry is m —the number of predictors, including the reference category for categorical variables. This value is equal to 10 this case.

Second, the data (in the examples in this paper, the dependent variable) is randomly permuted k number of times, with the p -value for each $\hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_m$ determined following equation (2) above and the confidence intervals following the equa-

Figure 2: PPV Plot of OLS Estimated Effects on Perception of Communist Threat
Note: $n = 56$. Party reference category = Democrat; Schooling reference category = high school. All schooling categories are “some/complete” for the corresponding level. Confidence intervals are two-sided and based on 1,000 permutation tests. Vertical line = .05. Dependent variable is standardized.

tion in footnote 6. These statistics are then used to construct an m -by-3 matrix,

D:

$$\mathbf{D} = \begin{bmatrix} \hat{p}_{\hat{\beta}1} & LL(p_{\hat{\beta}1}) & UL(p_{\hat{\beta}1}) \\ \hat{p}_{\hat{\beta}2} & LL(p_{\hat{\beta}2}) & UL(p_{\hat{\beta}2}) \\ \vdots & \vdots & \vdots \\ \hat{p}_{\hat{\beta}m} & LL(p_{\hat{\beta}m}) & UL(p_{\hat{\beta}m}) \end{bmatrix}, \quad (7)$$

where $UL(p_{\hat{\beta}m})$ is the upper confidence interval limit $\left(\hat{p}_{\hat{\beta}} + \left(z \times SE(\hat{p}_{\hat{\beta}})\right)\right)$ for $p_{\hat{\beta}m}$ and $LL(p_{\hat{\beta}m})$ is the lower limit $\left(\hat{p}_{\hat{\beta}} - \left(z \times SE(\hat{p}_{\hat{\beta}})\right)\right)$.

Lastly, any coefficient plot function can be used to plot **D**—such as the `coefplot`

package in Stata (Jann 2014) or R (Lander 2018), or the arm package in R (Gelman et al. 2016). The goal is to simply map each p -value and its confidence interval into a variable-by- $[0, 1]$ space and with a clear demarcation of the alpha-level (on whichever axis represents the $[0, 1]$ interval) around which the confidence intervals were constructed.

The fact that $\mathbf{D}$ is treated just like any other coefficient matrix means that the analyst can take advantage of virtually any visualization strategies that are available with coefficient plots. Nested models, for instance, can also be reported in the same PPV plot. If we treat the regression in Figure 2 as nested within a larger model that also conditions perceptions of Communist threat on sex, age, whether or not the respondent was Protestant, income level, and industry of employment, then we can combine the plots as shown in Figure 3. This PPV plot illustrates that, after controlling on this new set of control variables, none of the effect p -values from Model 1 go to the left of the .05 line—meaning that none of the estimates reach statistical significance under the randomization model at $\alpha = .05$. However, the dashed vertical line—indicating $\alpha = .1$ —shows that, while a respondent’s level of political engagement is not associated with perceptions of Communist threat beyond what we would expect under the condition of a random data-generating process before conditioning on the variables added in Model 2, the observed estimate $\hat{\beta}_{PE_2} = -0.281$ in Model 2 is less likely to be due to random association when we relax our alpha-level to .1.

Similarly, the analyst can reconfigure the matrix of p -values and confidence intervals into other plot types. For example, Figure 3 can be displayed as a bar chart

Figure 3: Nested PPV Plot of OLS Estimated Effects on Perception of Communist Threat

Note: $n = 56$ for both models. Party reference category = Democrat; Schooling reference category = high school; Industry reference category = service; reference category for Protestant is “Other.” All schooling categories are “some/complete” for the corresponding level. Confidence intervals are two-sided and based on 1,000 permutation tests. Solid vertical line = .05; dashed vertical line = .1. Dependent variable is standardized.

(Figure 4), where a p -value below .05 (now a horizontal solid line) indicates statistical significance with at least $\alpha = .05$. In this plot, spikes above the .05 reference line indicate an observed coefficient ($\hat{\beta}$) that was equal to or smaller in absolute size than at least 5% of the coefficients ($\hat{\beta}^*$) derived from rerunning the model after randomly permuted the perceptions of Communist threat values across observations 1,000 times.

Figure 4: PPV Plot Reconfigured as a Bar Chart

Note: Solid horizontal line = .05; dashed horizontal line = .1. All other interpretation notes are the same as those for Figure 3.

The substantive significance of the estimates can also be incorporated into PPV plots using standard graphical options. For example, assume an analyst is interested in the gap in perceptions of Communist threat between those who were strictly self-employed and those who were not. They include an interaction between the political engagement scale and the dichotomous self-employment variable. They use 1,000 permutation tests and construct the PPV plot in the top panel of Figure 5, which shows a statistically significant interaction ($\hat{\beta}_{PE \times SE} = .862$; $\hat{p}_{PE \times SE} = .004$,

where $.001 \leq p_{PE \times SE} \leq .010$). This PPV plot can also be reconfigured so that the marker shape indicates whether the effect is positive or negative and the color intensity represents the effect size of the standardized coefficient. The bottom panel of Figure 5 introduces effect size visualizations in lieu of the estimates themselves, with squares indicating negative estimates, circles positive estimates, and the hue of the marker and confidence intervals corresponding to the size of the standardized effect size. As the plot suggests, the interaction between political engagement and self-employment status was not only the only statistically significant predictor of the extent to which an attendee perceived heightened Communist threats, but also one of the substantively strongest predictors (as indicated by the darker markers for the political engagement main effect and the engagement $\times$ employment cross-product term).

Figure 5: Multiplicative Model with and without Effect Size Visualization

Note: $n = 56$. Party reference category = Democrat; Schooling reference category = high school; Industry reference category = service; reference category for Protestant is “Other.” All schooling categories are “some/complete” for the corresponding level. Confidence intervals are two-sided and based on 1,000 permutation tests. Vertical line = .05. Bottom panel hues based on effect size of standardized estimates; however, estimates reported in top graph are unstandardized. Darker markers in bottom panel indicate larger absolute effect sizes.

3.3. PPV Plots and Predicted Values

P -values for post-estimation predicted values may be of little use under the population model of inference when zero is not a meaningful value in the units of Y . This is because the prediction p -values are typically derived from one-sample t - or z -tests of the null hypothesis that the “true” value for the hypothetical case in question is zero. The (two-tailed) prediction p -value, then, is just the probability of generating a predicted value of at least that absolute size if the true value for a case with that profile of independent variable characteristics were zero—again assuming that the sample used to derive that prediction is a random draw from the target population. This means that for dependent variables without a meaningful zero, most empirically plausible predictions with reasonable standard errors will generate very large test statistics and therefore very small p -values. Further, since a statistically non-significant predicted value under the population model simply indicates a $1 - \alpha$ or greater probability of observing that predicted value for that case when the “true” value is 0, a predicted value for a dependent variable with a meaningful zero (such as a standardized variable) could easily have $p < .05$ but still indicate a “non-zero” prediction.

With that said, it might be important for scholars basing statistical significance on permutation-based p -values to report the p -values for predictions. This is because permutation-based p -values are based on the differences between the observed prediction and the predictions from the permuted models and not on the difference of the prediction from zero. As such, a two-tailed permutation-based p -value for a single observed prediction following a regression model indicates the proportion of

predictions from the regressions on the permuted samples that are greater than or equal in absolute size to that observed prediction.¹¹ This p -value, then, provides an estimate of the extent to which the observed prediction deviates from the predictions generated from models where the association between the dependent and independent variables are random by design.

PPV plots can also be used to visualize these permutation-based prediction p -values. This is accomplished by first writing a program that adds the calculation of predicted values as an extra step to each of the 1,000 permutation tests. Consider again, for instance, the multiplicative model used to construct the PPV plots in Figure 5, which showed a statistically significant interaction between an attendee's level of political engagement and their self-employment status when it comes to heightened perceptions of Communist threat. The analyst can now add one additional step to their permutation procedure for obtaining permutation-derived p -values for post-

---

¹¹Unless the predictions from the permuted samples take signs in the opposite direction of the observed prediction, then, for permutation-based p -values for predictions, then the two-tailed and one-tailed tests will have the following equivalencies. If the observed prediction is positive and all of its k predictions from the permuted samples are positive, then:

$$\left(\frac{I \left(\left| \hat{\beta}_t^* \right| \geq \left| \hat{\beta} \right| \right)}{k} \mid \left(\forall \hat{\beta}_t^* \in \beta \right) \right) \equiv \frac{I \left(\hat{\beta}_t^* \geq \hat{\beta} \right)}{k}. \quad (8)$$

And, if the observed prediction is negative and all of its k predictions from the permuted samples are negative, then:

$$\left(\frac{I \left(\left| -\hat{\beta}_t^* \right| \geq \left| -\hat{\beta} \right| \right)}{k} \mid \left(\forall -\hat{\beta}_t^* \in \beta \right) \right) \equiv 1 - \frac{I \left(-\hat{\beta}_t^* \leq -\hat{\beta} \right)}{k}. \quad (9)$$

In other words, the two-tailed test will be equivalent to the one-tailed test to the right for positively-valued predictions and the inverse of the one-tailed test to the left for negatively-valued predictions.

estimation predictions: after randomly permuting the dependent variable k times and estimating the regression model, a series of predictions can be generated for each permutation and compared to the predictions from the observed regression model. This analysis could result in something akin to Table 2, which presents predicted perceptions of Communist threat (on a standardized scale) for both the self-employed and those who are not, within a standard deviation of the sample mean political engagement scale.

Table 2: Predicted Standardized Perceptions of Communist Threat (std) from OLS Model

IV	Model
Not S-E's Political Engagement (std)	
-1 SD	0.771** (0.001)
Mean	0.051 (0.747)
+1 SD	-0.669* (0.023)
S-E's Political Engagement (std)	
-1 SD	-0.411 (0.214)
Mean	-0.269 (0.213)
+1 SD	-0.127 (0.636)
n	56
# of Permutations of DV	1,000

Note: P -values in parentheses. S-E is “self-employed.”

* $p < .05$; ** $p < .01$ (two-tailed tests)

Alternatively, these predictions could be displayed as a PPV plot that allows for a visual comparison of the relative statistical significance levels, as shown in Figure 6. These predicted value PPV plots can also be easily reconfigured into other plot

types—e.g., as a bar chart (Figure 7) similar to the coefficient PPV bar chart in Figure 4. Or, alternatively, the prediction p -values can be displayed as a line graph for continuous variables. Figure 8 provides an example, with separate political engagement p -value “slopes” for hypothetical respondents of each employment category.

Figures 6, 7, and 8 indicate that higher levels of political engagement are negatively associated with higher perceptions of Communist threat for those who aren’t strictly self-employed (according to the switch in signs for marker labels in Figure 6 and the x -axis labels in Figures 7 and 8 for those that aren’t self-employed) while the estimate remains negative for the self-employed but loses some of its effect size (as indicated by the smaller negative effect sizes for those that are self-employed). However, these figures suggest that this interaction may be asymmetric. Among these 56 school attendees, only those who aren’t strictly self-employed with lower than average and higher than average levels of political engagement have predicted levels of perceptions of Communist threat that deviate from what would be expected under the condition of a random data-generating process, with non-self-employed respondents with lower than average levels of political engagement exhibiting slightly above average perceptions of threat ($\hat{Y}_{NSE} = .771$; $.000 \leq p_{\hat{Y}_{NSE}} \leq .006$) and those with higher than average levels of political engagement exhibiting slightly below average perceptions ($\hat{Y}_{NSE} = -.669$; $.015 \leq p_{\hat{Y}_{NSE}} \leq .034$).

The non-self-employed with average levels of political engagement, and the self-employed at any level of political engagement, have predicted threat perceptions that never deviate from what we would expect if the predictions were computed from a model of random associations. If this hypothetical analyst were hoping to

Figure 6: PPV Plot of Predicted Standardized Perceptions of Communist Threat by Self-Employment and Level of Political Engagement

Figure 7: Predicted Value PPV Plot Reconfigured as a Bar Chart

Figure 8: Predicted Value PPV Plot Reconfigured as a Line Graph

draw substantive conclusions from these results—and assuming a properly specified regression model (which, since these models are for illustrative purposes, these may not be)—they may conclude something to the following effect: Among these 56 school attendees, those who weren’t strictly self-employed but who were less politically engaged than the average participant were more likely to have higher than average perceptions of a Communist threat—but, when they were more politically engaged than the average participant, they tended to have some of the weakest perceptions of Communist threat.

The asymmetry of this interaction would likely have been more difficult to pick up without adding prediction calculations to the permutation tests, and PPV plots certainly make this task of comparing effects across groups that much easier.

4. Summary and Conclusions

Recent interest in data analytics and computational social science has sparked a flurry of activity in data visualization among social scientists (Healy 2018; Wickham and Grolemund 2017). One popular graph type for displaying regression models is the coefficient plot. While coefficient plots are useful for simultaneously visualizing both the statistical and substantive significance of the estimates, their applicability assumes that statistical significance is determined under the population model of inference—i.e., that the coefficients have standard errors and confidence intervals, derived either with reference to theoretical sampling distributions or to some asymptotically-based resampling technique such as the bootstrap or jackknife (Shao and Tu 1995). The PPV plot “toolkit” introduced in this paper offers a way of lever-

aging the intuitiveness, parsimony, and aesthetic simplicity of the coefficient plot for regression models with statistical significance levels derived via the randomization model of inference, where standard errors and confidence intervals are reported for the coefficient p -values rather than the coefficients themselves and where a plot centered at zero is less useful.

In this paper, I illustrated the utility of PPV plots for visualizing permutation-based regression estimates and post-estimation predictions using the case of heightened perceptions of threat among 56 anti-Communists at an anti-Communism school. In both cases, PPV plots leverage the same benefits of coefficient plots in that they visualize the direction and size of the estimates while concomitantly incorporating estimate uncertainty into the graph. For coefficient plots, the uncertainty is captured with the confidence intervals around the estimates: the wider the interval, the more uncertainty there is about where the true population parameter may be. For PPV plots, the uncertainty is captured with the confidence interval around the p -value. Specifically, the wider the interval, the more uncertainty there is about where the true p -value would be had every possible permutation been accounted for. Accounting for this uncertainty is particularly important for boundary estimates: i.e., when a p -value indicates a statistically significant estimate with a confidence interval that spans across alpha-levels or a non-significant estimate that spans to the left of the alpha-level necessary to establish statistical significance. PPV plots can be particularly helpful in the case of a model with many predictors, since they can simultaneously display the number of boundary estimates. Importantly, since the size of the p -value confidence intervals is a function of the number of permutations

performed, PPV plots in this case can also provide a general sense of how much closer the analyst should approximate an exact test (i.e., account for every possible permutation) in order to shrink the confidence intervals so that the number of boundary estimates are minimized.

There will undoubtedly be the question of how much traction visualization strategies for randomization inference will get in sociology, where this type of statistical inference is not the disciplinary norm. Indeed, if the population model is far and away the more common mode of inference, then what real leverage do PPV plots provide? This concern is itself symptomatic of a fundamental question about quantitative inferential reasoning: namely, is it really inference if there is no population to which one can infer things (cf. Ludbrook and Dudley 1998:128)? Such a perspective is likely a partial function of the fact that the inferential statistics curricula for most statistics courses in sociology are predicated on the logic that inference is synonymous with generalizing to a population. Ironically, though, it is the other type of inference that seems to predominate the actual practice of sociological research, since, as network analysts have observed for some time, “most social science aims at studying mechanisms rather than describing specific populations” (Snijders 2011:135). Perhaps to no surprise, this implicit reliance on mechanistic reasoning seems particularly important in subfields where datasets constructed using rigorous probabilistic sampling procedures are by far the exception rather than the norm—e.g., cultural sociology, social movement studies, and organizational sociology. Following this, I conclude this paper with a more general call for quantitative sociologists to reflect critically on what sort of statistical inferences are appropriate given their sample design, and,

by extension, how they choose to “see” this significance.

5. References

- Aiello, Thomas. 2005. “Constructing ‘Godless Communism’: Religion, Politics, and Popular Culture, 1954-1960.” *Americana: The Journal of American Popular Culture, 1900 to Present* 4(1). Retrieved March 9, 2018 (<https://search.proquest.com/docview/15199651?origsite=gscholar>).
- Anderson, Marti J. and John Robinson. 2001. “Permutation Tests for Linear Models.” *Australian New Zealand Journal of Statistics* 43(1):75-88.
- Aronow, Peter M. 2012. “A General Method for Detecting Interference Between Units in Randomized Experiments.” *Sociological Methods Research* 41(1):3-16.
- Bhr, Sebastian and Martin Abraham. 2016. “The Role of Social Capital in the Job-Related Regional Mobility Decisions of Unemployed Individuals.” *Social Networks* 46:44-59.
- Baker, Frank B. and Lawrence J. Hubert. 1981. “The Analysis of Social Interaction Data: A Nonparametric Technique.” *Sociological Methods Research* 9(3):339-61.
- Butts, Carter T. 2007. “Permutation Models for Relational Data.” *Sociological Methodology* 37(1):257-81.
- Darlington, Richard B. and Andrew F. Hayes. 2017. *Regression Analysis and Linear Models: Concepts, Applications, and Implementation*. New York: The Guilford Press.
- Edgington, Eugene S. and Patrick Onghena. 2007. *Randomization Tests*. Boca Raton, FL: CRC Press.

- Ernst, Michael D. 2004. "Permutation Methods: A Basis for Exact Inference." *Statistical Science* 19(4):676-85.
- Freedman, David and David Lane. 1983. "A Nonstochastic Interpretation of Reported Significance Levels." *Journal of Business Economic Statistics* 1(4):292-98.
- Friedman, Sam and Daniel Laurison. 2017. "Mind the Gap: Financial London and the Regional Class Pay Gap." *The British Journal of Sociology* 68(3):474-511.
- Gelman, Andrew, Yu-Sung Su, Masanao Yajima, Jennifer Hill, Maria Grazia Pittau, Jouni Kerman, Tian Zheng, and Vincent Dorie. "arm." R package version 1.9-3.
- Healy, Kieran. 2018. *Data Visualization: A Practical Introduction*. Princeton, NJ: Princeton University Press.
- Herzog, Jonathan P. 2011. *The Spiritual-Industrial Complex: America's Religious Battle against Communism in the Early Cold War*. New York: Oxford University Press.
- Jann, Ben. 2014. "Plotting Regression Coefficients and Other Estimates." *Stata Journal* 14(4):708-37.
- Krackhardt, David. 2007. "Predicting with Networks: Nonparametric Multiple Regression Analysis of Dyadic Data." *Social Networks* 10(4):359-81.
- Lander, Jared P. 2018. "coefplot." R package version 1.2.6. Retrieved March 1, 2018 (<https://cran.r-project.org/web/packages/coefplot/coefplot.pdf>).
- Laurison, Daniel and Sam Friedman. 2016. "The Class Pay Gap in Higher Professional and Managerial Occupations.' *American Sociological Review* 81(4):668-95.
- Lizardo, Omar and Sara Skiles. 2016a. "Cultural Objects as Prisms: Perceived Audience Composition of Musical Genres as a Resource for Symbolic Exclusion."

- Socius* 2:1-17.
- Lizardo, Omar and Sara Skiles. 2016b. "The End of Symbolic Exclusion? The Rise of 'Categorical Tolerance' in the Musical Tastes of Americans: 1993-2012." *Sociological Science* 3:85-108.
- Ludbrook, John and Hugh Dudley. 1998. "Why Permutation Tests are Superior to t and F Tests in Biomedical Research." *The American Statistician* 52(2):127-32.
- Manly, Bryan F. J. 2007. *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Boca Raton, FL: Chapman Hall/CRC.
- Ogorman, Thomas W. 2006. "An Adaptive Test for a Subset of Regression Coefficients Using Permutations of Residuals." *Journal of Statistical Computation and Simulation* 76(12):1095-105.
- Schhrer, Susanne, William Carbonaro, and Eric Grodksy. 2016. "Reproduction of Inequality in Educational Attainment through Curricular Differentiation in Secondary SchoolA Case Study." Pp. 249-67 in *Models of Secondary Education and Social Inequality: An International Comparison*, edited by H. Blossfeld, S. Bucholz, J. Skopek, and M. Triventi. Northampton, MA: Edward Elgar Publishing.
- Shao, Jun and Dongsheng Tu. 1995. *The Jackknife and the Bootstrap*. New York: Springer.
- Snijders, Tom A. B. 2011. "Statistical Models for Social Networks." *Annual Review of Sociology* 37:131-53.
- StataCorp. N.d. "**permute** – Monte Carlo Permutation Tests." College Station, TX: Stata Press. Retrieved August 21, 2019 (<https://www.stata.com/manuals13/rpermute.pdf>).

- Stewart, Evan, Penny Edgell, and Jack Delehanty. 2018. "The Politics of Religious Prejudice and Tolerance for Cultural Others." *The Sociological Quarterly* 59(1):17-39.
- Stoltz, Dustin S. and Marshall A. Taylor. 2017. "Paying with Change: The Purposeful Enunciation of Material Culture." *Poetics* 64:26-39.
- The Schwartz Report. 2018. "About Us." Manitou Springs, CO: CACC The Schwartz Report. Retrieved March 9, 2018 (<https://www.schwarzreport.org/about>).
- Thomann, Eva and Martino Maggetti. 2017. "Designing Research with Qualitative Comparative Analysis (QCA): Approaches, Challenges, and Tools." *Sociological Methods Research*.
- Thomson, Godfrey. 1951. *The Factorial Analysis of Human Ability*. London: University of London Press.
- Wickham, Hadley and Garrett Golemund. 2017. *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. Sebastopol, CA: OReilly Media.
- Wolfinger, Raymond E., Barbara Kaye Wolfinger, Kenneth Prewitt, and Sheilah Rosenhack. 1964. "Crusaders for the Right." *Trans-action* 1(5):22-27.
- Wolfinger, Raymond E. et al. 1992. *Americas Far-right, 1962*. ICPSR 07273. Ann Arbor, MI: Inter-University Consortium for Political and Social Research.

Appendix A. Principal Factor Analyses

Six variables were used to construct a “perception of communist threat” variable. The variables were all dichotomous; as such, tetrachoric correlations were decomposed to extract the principal factors (Gadermann and Zumbo 2012; Kubinger 2003). “Communist danger” indicates whether or not the respondent sees American communists as a danger to the country at that moment, where 0 = no/some/little danger and 1 = a great or very great deal of danger. “Communist professors” indicates whether or not the respondent believes communist professors have a lot of influence in U.S. higher education, where 0 = no and 1 = yes. “Communist Democrats” indicates whether or not the respondent sees communists as having a lot of influence in the U.S. Democratic Party, where 0 = no and 1 = yes. “Communist speech” indicates whether or not the respondent believes the U.S. Communist Party should be allowed to speak on radio or T.V., where 0 = no and 1 = yes. “Communist neighbor” indicates whether or not the respondent believes there are communists living in their neighborhood, where 0 = no and 1 = yes. Table A.3 provides the inter-item tetrachoric correlations.

Table A.3: Tetrachoric Correlations Between “Perceptions of Communist Threat” Variables

	C Danger	C Professors	C Democrats	C Speech	C Neighbor	Local C
C Danger	1					
C Professors	.560	1				
C Democrats	.580	.739	1			
C Speech	-.415	-.213	-.011	1		
C Neighbor	.235	.446	.256	-.056	1	
Local C	.752	.426	.497	-.079	.559	1

Table A.4 contains the loadings for the first principal factor after orthogonal varimax rotation. All variables load in the expected direction (positive) for a “perceptions of communist threat” latent variable. This factor explains as much variance as approximately 1.821 of the raw indicator variables (eigenvector, λ , = 1.821)—i.e., about 45.67% of the total variance. Factor weights were derived using the regression method (Thomson 1951). Higher values indicate heightened perceptions that communists pose a clear and imminent threat.

Table A.4: Factor Loadings for Table A.3

Variable	Loadings
Communist Danger	.367
Communist Professors	.600
Communist Democrats	.594
Communist Speech	.168
Communist Neighbor	.674
Local Communism	.707
λ	1.821
% Variance Explained	45.67%
n	116

Note: Principal factor loadings are with orthogonal varimax rotation.

Six dichotomous variables were also used to create a “political engagement” vari-

able, again using tetrachoric correlations and principal factor analysis. The inter-item correlations are presented in Table A.5. “Attend meetings” indicates whether or not the respondent goes to political rallies, meetings, and dinners, where 0 = no and 1 = yes. “Write political letters” indicates whether or not the respondent, in the past year before being surveyed, ever wrote letters or sent telegrams to the president, congressperson, or state politician, where 0 = no and 1 = yes. “Organizational member” indicates whether or not the respondent was a member of a political organization, where 0 = no and 1 = yes. “Influence voters” indicates whether or not the respondent ever tried to convince people to vote one way or another, where 0 = no and 1 = yes. “Give money/buy tickets” indicates whether or not the respondent ever gave money or bought tickets in an effort to help a candidate or their campaign or party. Finally, “work for the candidate” indicates whether or not the respondent did any other work for a candidate or party, where 0 = no and 1 = yes.

Table A.5: Tetrachoric Correlations Between “Political Engagement” Variables

	Attend	Letters	Org Member	Voters	Money/ Tickets	Work
Attend	1					
Letters	.466	1				
Org Member	.658	.120	1			
Voters	.318	.145	.442	1		
Money/Tickets	.698	.329	.630	.347	1	
Work	.554	.222	.802	.423	.582	1

Note: Correlations based on 190 valid cases after listwise deletion. All variables are dichotomous.

Table A.6 list the loadings for these variables on the first principal factor after orthogonal varimax rotation. All variables load in the direction consistent with a

“political engagement” latent variable (i.e., they are all positive). This “political engagement” factor explains as much variance as approximately 2.297 raw indicator variables (about 72.55% of the total variance). The regression method was used to compute the factor weights. Higher scores indicate that the respondent is more politically active.

Table A.6: Factor Loadings for Table A.5

Variable	Loadings
Attend Meetings	.541
Write Pol. Letters	.071
Org Member	.888
Influence Voters	.459
Give Money / Buy Tickets	.588
Work for Candidate	.808
λ	2.297
% Variance Explained	72.55%
n	190

Note: Principal factor loadings are with orthogonal varimax rotation.

Appendix B. Stata Code to Reproduce Tables and Figures

```
*****
version 13.1
clear all
macro drop _all
log using ppvplot_code.log, replace text
set more off

*****
***COMMANDS BEGIN HERE ***
*****
use rright.dta, clear //You will need to download the aforementioned
//dataset from ICPSR. I re-named it "rright," which you can
//obviously change.

//Give variables some descriptive labels
//Communist danger
gen cdanger = .
replace cdanger = 0 if v7==5 | v7==4 | v7==3
replace cdanger = 1 if v7==1 | v7==2
label def cdanger 0 "some/little danger" 1 "very great/great danger"
label var cdanger "Communist danger at present time?"
label values cdanger cdanger

//Letter-writing
gen letter = .
replace letter = 0 if v13==4
replace letter = 1 if v13==2
label def noyes 0 "no" 1 "yes"
label var letter "Written to politician in last year?"
label values letter noyes

//Partisan affiliation
gen party = .
replace party = 0 if v9==4
replace party = 1 if v9==2
replace party = 2 if v9==6 | party==8
label def party 0 "dem" 1 "rep" 2 "other"
label var party "partisan affiliation"
label values party party

//Fight Russia?
gen russia = .
replace russia = 0 if v8==1 | v8==2 | v8==4 | v8==8
replace russia = 1 if v8==6
label var russia "Fight Russia?"
label values russia noyes

//Attend political meetings?
gen meeting = .
replace meeting = 0 if v16==4
replace meeting = 1 if v16==2
label var meeting "Attend political meetings?"
label values meeting noyes

//Belong to political organization?
gen org = .
replace org = 0 if v18==4
replace org = 1 if v18==2
label var org "Belong to political org?"
label values org noyes

//Public officials care about your opinion?
gen popinion = .
replace popinion = 0 if v36==2
replace popinion = 1 if v36==4
label def popinion 0 "don't care" 1 "care"
label var popinion "Public officials care about your op?"
label values popinion popinion

//Should Socialist P publish in US during peacetime?
gen snws = .
replace snws = 0 if v37==2
replace snws = 1 if v37==4
```



```

label var snews "Should Socialist P publish in US during peacetime?"
label values snews noyes

//Do Communist professors in US have much influence?
gen profess = .
replace profess = 0 if v38==2
replace profess = 1 if v38==4
label var profess "Do Communist professors in US have much influence?"
label values profess noyes

//Do Communists have a lot of influence in the D Party?
gen cdems = .
replace cdems = 0 if v40==2
replace cdems = 1 if v40==4
label var cdems "Do Communists have a lot of influence in the D Party?"
label values cdems noyes

//Free speech for Communists on US radio/TV?
gen cspeech = .
replace cspeech = 0 if v42==4
replace cspeech = 1 if v42==2
label var cspeech "Free speech for Communists on US radio/TV?"
label values cspeech noyes

//Communists in my your neighborhood?
gen cneighbor = .
replace cneighbor = 0 if v43==4
replace cneighbor = 1 if v43==2
label var cneighbor "Communists in my your neighborhood?"
label values cneighbor noyes

//Local communists more dangerous than Russian or China?
gen clocal = .
replace clocal = 0 if v50==2
replace clocal = 1 if v50==4
label var clocal "Local communists more dangerous than Russian or China?"
label values clocal noyes

//Sex
gen sex = .
replace sex = 0 if v54==2
replace sex = 1 if v54==4
label def sex 0 "male" 1 "female"
label var sex "sex of R"
label values sex sex

//Age
gen age=v55
recode age (00=10)(10=11)(99=.)
label def age 01 "20-24" 11 "70+"
label var age "age of R"
label values age age

//Schooling
gen school=v56
recode school (1=1)(2=1)(3=2)(4=2)(5=3)(6=3)(else=.)
label def school 1 "some/complete grammar school" 2 "some/complete HS" ///
3 "some/complete college"
label var school "How much schooling did you complete?"
label values school school

//Union member?
gen union = .
replace union = 0 if v61==4
replace union = 1 if v61==2
label var union "Union member?"
label values union noyes

//Religion
gen relig = .
replace relig = 0 if v68==0
replace relig = 1 if v68==2
replace relig = 2 if v68==4
replace relig = 3 if v68==6
replace relig = 4 if v68==8
label def relig 0 "none" 1 "protestant" 2 "catholic" 3 "jewish" 4 "other"
label var relig "R's religion"
label values relig relig

```

```

//Protestant
gen relig2 = .
replace relig2 = 0 if v68==0
replace relig2 = 1 if v68==2
replace relig2 = 0 if v68==4
replace relig2 = 0 if v68==6
replace relig2 = 0 if v68==8
label def relig2 0 "other" 1 "protestant"
label var relig2 "Is R Protestant"
label values relig2 relig2

//Income
gen income=v72
recode income (2=1)(4=2)(6=3)(8=4)(0=5)(else=.)
label def income 1 "under $4,000" 5 "$15,000 and over"
label var income "Total fam income in previous year bef taxes"
label values income income

//Industry
gen industry=v58
recode industry (1=1)(2=2)(3=2)(4=2)(5=2)(6=2)(7=3)(8=3)(9=3)(0=4)(else=.)
label def industry 1 "agriculture" 2 "business" 3 "servivice" 4 "government"
label var industry "head of house industry"
label values industry industry

//Influence vote?
gen influence = .
replace influence = 0 if v14==4
replace influence = 1 if v14==2
label var influence "Do you try to influence vote?"
label values influence noyes

//Give money/buy tickets?
gen money = .
replace money = 0 if v15==4
replace money = 1 if v15==2
label var money "Do you give money/buy tickets for campaigns?"
label values money noyes

//Do work for party/candidate?
gen work = .
replace work = 0 if v17==4
replace work = 1 if v17==2
label var work "Do you do any other work for party/cand?"
label values work noyes

//Employer
gen selfemp = .
replace selfemp = 0 if v57==4 | v57==6 | v57==8
replace selfemp = 1 if v57==2
label def selfemp 0 "not self-employed" 1 "self-employed"
label var selfemp "Is R's head of house self-employed?"
label values selfemp selfemp

//Create scales
//Communist threat
qui reg cdanger profess cdems cspeech cneighbor clocal
gen flag = e(sample)
univar cdanger profess cdems cspeech cneighbor clocal if flag==1

mat define mean = (0.92, 0.88, 0.67, 0.18, 0.51, 0.71)
mat define sd = (0.27, 0.33, 0.47, 0.39, 0.50, 0.46)
mat colnames mean = cdanger profess cdems cspeech cneighbor clocal
mat colnames sd = cdanger profess cdems cspeech cneighbor clocal

polychoric cdanger profess cdems cspeech cneighbor clocal if flag==1
global N = r(sum_w)
matrix r = r(R)
factormat r, n({N}) factors(2) means(mean) sds(sd)
rotate
predict threat

//Political engagement
qui reg meeting letter org influence money work
gen flag2 = e(sample)
univar meeting letter org influence money work if flag2==1

```

```

mat define mean2 = (0.61, 0.61, 0.28, 0.92, 0.63, 0.39)
mat define sd2 = (0.49, 0.49, 0.45, 0.28, 0.48, 0.49)
mat colnames mean2 = meeting letter org influence money work
mat colnames sd2 = meeting letter org influence money work

polychoric meeting letter org influence money work if flag2==1
global N = r(sum_w)
matrix r = r(R)
factormat r, n({N}) factors(2) means(mean2) sds(sd2)
rotate
predict engage

//The models and their figures
//Plot 1
save "rright.dta", replace
clear
webuse nhanes2f
reg bpsystol weight height i.sex i.race age health
est store m1

coefplot m1, ///
mlabcolor(black) ///
xline(0, lcolor(black)) mcol(black) msize(small) mlabsize(2) ///
graphregion(fcolor(white) lcolor(white) lwidth(thick)) drop(_cons) ///
coeflabels(weight="Weight (kg.)" height="Height (cm.)" 2.sex="Female" ///
2.race="Black" 3.race="Other" age="Age" health="Health") ///
groups(?.race ?.race = "{bf:Race}") mlabels(weight=12 "0.463" height=12 "-0.315" ///
2.sex=12 "-2.945" 2.race=12 "2.333" 3.race=12 "2.294" age=12 "0.580" ///
health=12 "-0.865") levels(99.9 99 95) legend(order(1 "99.9% CI" 2 ///
"99% CI" 3 "95% CI") rows(1)) ciopts(lcolor(black black black)) ///
saving(exiplot.gph, replace)

clear
use "rright.dta"

//Plot 2
qui reg threat engage i.party i.sex age ib2.school i.union i.relig income ///
i.industry i.selfemp
gen flag3 = e(sample)

egen threat2 = std(threat) if flag3==1
egen engage2 = std(engage) if flag3==1

mat p_model = J(12, 3, .)

qui reg threat2 engage2 i.party i.union ib2.school i.selfemp if flag3==1
mat def beta = e(b)
forvalues k = 1/12 {
    global k'k' = round(beta[1,'k'], .001)
}
permute threat2 _b, reps(1000) seed(5): ///
qui reg threat2 engage2 i.party i.union ib2.school i.selfemp if flag3==1
mat def m1 = r(p)
mat def m12 = r(ci)
forvalues k = 1/12 {
    mat p_model['k',1] = m1[1,'k']
    mat p_model['k',2] = m12[1,'k']
    mat p_model['k',3] = m12[2,'k']
}

mat rownames p_model = engage2 0.party 1.party 2.party 0.union 1.union ///
1.school 2.school 3.school 0.selfemp 1.selfemp constant

coefplot (matrix(p_model[,1]), ci((p_model[,2] p_model[,3]))), ///
mlabels(engage2=12 ${k1} 1.party=12 ${k3} 2.party=12 ${k4} 1.union=12 ${k6} ///
1.school=12 ${k7} 3.school=12 ${k9} 1.selfemp=12 ${k11} constant=12 ${k12}) mlabcolor(black) ///
xline(0.05, lcolor(black)) mcol(black) msize(small) mlabsize(2) ///
graphregion(fcolor(white) lcolor(white) lwidth(thick)) drop(0.party 0.union 2.school 0.selfemp) ///
coeflabels(engage2="Pol Engage (std)" 1.party="Republican" ///
2.party="Other" 1.union="Union Member" 1.school="Grammar School" ///
3.school="College" 1.selfemp="Self-Employed" constant="Constant") legend(order(1 "95% Confidence Interval")) ///
ciopts(lcolor(black)) groups(1.party 2.party = "{bf:Party}" ///
1.school 3.school="{bf:Schooling}") xtitle("{it:P}-Values") ///
order(engage2 ?.selfemp ?.party ?.union ?.school constant) ///
saving(ex2plot.gph, replace)

//Plot 3
mat p_model12 = J(22, 3, .)

```

```

qui reg threat2 engage2 i.party i.union ib2.school i.sex age i.relig2 income ///
    ib3.industry i.selfemp if flag3==1
mat def beta2 = e(b)
forvalues i = 1/22 {
    global i`i' = round(beta2[1,`i'], .001)
}
permute threat2 _b, reps(1000) seed(5): ///
qui reg threat2 engage2 i.party i.union ib2.school i.sex age i.relig2 income ///
    ib3.industry i.selfemp if flag3==1
mat def mm1 = r(p)
mat def mm12 = r(ci)
forvalues i = 1/22 {
    mat p_model2[`i',1] = mm1[1,`i']
    mat p_model2[`i',2] = mm12[1,`i']
    mat p_model2[`i',3] = mm12[2,`i']
}

mat rownames p_model2 = engage2 0.party 1.party 2.party 0.union 1.union ///
    1.school 2.school 3.school 0.sex 1.sex age 0.relig2 1.relig2 income 1.industry ///
    2.industry 3.industry 4.industry 0.selfemp 1.selfemp constant

coefplot (matrix(p_model[1,]), mlabels(engage2=11 ${k1} 1.party=11 ${k3} ///
    2.party=11 ${k4} 1.union=11 ${k6} 1.school=11 ${k7} 3.school=11 ${k9} ///
    constant=11 ${k12} 1.selfemp=11 ${k11}) mlabcolor(black) xline(0.05, lcolor(black)) ///
    xline(0.1, lcolor(black) lpattern(dash)) mcol(black) ///
    msize(small) mlabsize(2) ci((p_model[2,] p_model[3,]) ciopts(lcolor(black))) ///
    (matrix(p_model2[1,]), mlabels(engage2=1 ${i1} 1.party=1 ${i3} ///
    2.party=1 ${i4} 1.union=1 ${i6} 1.school=1 ${i7} 3.school=1 ${i9} ///
    1.sex=1 ${i11} age=1 ${i12} 1.relig2=1 ${i14} income=1 ${i15} ///
    1.industry=1 ${i16} 2.industry=1 ${i17} 4.industry=1 ${i19} constant=1 ${i22} ///
    1.selfemp=1 ${i21}) mlabcolor(gray) mcol(gray) msize(small) mlabsize(2) ///
    ci((p_model2[2,] p_model2[3,]) ciopts(lcolor(gray))), graphregion(fcolor(white) lcolor(white) ///
    lwidth(thick)) drop(0.party 0.union 0.sex 0.relig2 2.school 3.industry 0.selfemp) ///
    coeflabels(engage2="Pol Engage (std)" 1.party="Republican" 2.party="Other" ///
    1.union="Union Member" 1.school="Grammar School" 3.school="College" ///
    1.sex="Female" age="Age" 1.relig2="Protestant" income="Family Income" ///
    1.industry="Agriculture" 2.industry="Business" 4.industry="Government" constant="Constant" ///
    1.selfemp="Self-Employed") groups(1.party 2.party = "{bf:Party}" 1.school 3.school = "{bf:Schooling}" ///
    1.industry 2.industry 4.industry = "{bf:Industry}") ///
    legend(order(2 "Model 1" 3 "Model 2")) xtitle("{it:P}-Values") ///
    order(engage2 ?.selfemp ?.party ?.union ?.school ?.sex age ?.relig2 income ?.industry ///
    constant) saving(ex3plot.gph, replace)

//Plot 4
#delimit ;

coefplot (matrix(p_model[1,]), ci((p_model[2,] p_model[3,]) bcolor(black))
    (matrix(p_model2[1,]), ci((p_model2[2,] p_model2[3,]) bcolor(gray)),
    graphregion(fcolor(white) lcolor(white) lwidth(thick)) drop(0.party 0.union
    0.sex 0.relig2 2.school 3.industry 0.selfemp) coeflabels(
    engage2="""{bf:Pol Engage (std)}"" "{&beta}1 = ${k1}" "{&beta}2 = ${i1}""',
    1.party="""{bf:Republican}" "{&beta}1 = ${k3}" "{&beta}2 = ${i3}""',
    2.party="""{bf:Other}" "{&beta}1 = ${k4}" "{&beta}2 = ${i4}""',
    1.union="""{bf:Union Member}" "{&beta}1 = ${k6}" "{&beta}2 = ${i6}""',
    1.school="""{bf:Grammar School}" "{&beta}1 = ${k7}" "{&beta}2 = ${i7}""',
    3.school="""{bf:College}" "{&beta}1 = ${k9}" "{&beta}2 = ${i9}""',
    1.sex="""{bf:Female}" "{&beta}2 = ${i11}""',
    age="""{bf:Age}" "{&beta}2 = ${i12}""',
    1.relig2="""{bf:Protestant}" "{&beta}2 = ${i14}""',
    income="""{bf:Family Income}" "{&beta}2 = ${i15}""',
    1.industry="""{bf:Agriculture}" "{&beta}2 = ${i16}""',
    2.industry="""{bf:Business}" "{&beta}2 = ${i17}""',
    4.industry="""{bf:Government}" "{&beta}2 = ${i19}""',
    1.selfemp="""{bf:Self-Employed}" "{&beta}1 = ${k11}" "{&beta}2 = ${i21}""',
    constant="""{bf:Constant}" "{&beta}1 = ${k12}" "{&beta}2 = ${i22}""', labsize(vsmall) angle(90))
    groups(1.party 2.party = "{bf:Party}" 1.school 3.school = "{bf:Schooling}"
    1.industry 2.industry 4.industry = "{bf:Industry}") order(engage2 ?.selfemp ?.party
    ?.union ?.school ?.sex age ?.relig2 income ?.industry constant)
    ciopts(lcolor(black) recast(rcap rcap))
    recast(bar) vertical yline(0.05, lcolor(black)) yline(0.1, lcolor(black) lpattern(dash))
    ytitle("{it:P}-Value") barwidth(.3) fcolor(*.8) citop xlabel(, labsize(vsmall)
    angle(90)) legend(order(1 "Model 1" 3 "Model 2")) saving(ex4plot.gph, replace) ;

#delimit cr

//Plot 5, Top Panel
mat p_model3 = J(24, 3, .)

```

```

qui reg threat2 c.engage2##i.selfemp i.party i.union ib2.school i.sex age i.relig2 c.income ///
    ib3.industry if flag3==1
mat def beta2 = e(b)
forvalues i = 1/24 {
    global n`i' = round(beta2[1,`i'], .001)
}
permute threat2 _b, reps(1000) seed(5): ///
qui reg threat2 c.engage2##i.selfemp i.party i.union ib2.school i.sex age i.relig2 c.income ///
    ib3.industry if flag3==1
mat def mm1 = r(p)
mat def mm12 = r(ci)
forvalues i = 1/24 {
    mat p_model3[`i',1] = mm1[1,`i']
    mat p_model3[`i',2] = mm12[1,`i']
    mat p_model3[`i',3] = mm12[2,`i']
}

mat rownames p_model3 = engage2 0.selfemp 1.selfemp 0.selfemp#c.engage2 ///
    1.selfemp#c.engage2 0.party 1.party 2.party 0.union 1.union ///
    1.school 2.school 3.school 0.sex 1.sex age 0.relig2 1.relig2 income ///
    1.industry 2.industry 3.industry 4.industry constant

coefplot (matrix(p_model3[1,]), mlabels(engage2=12 ${n1} 1.selfemp=12 ${n3} ///
    1.selfemp#c.engage2=12 ${n5} 1.party=12 ${n7} ///
    2.party=12 ${n8} 1.union=12 ${n10} 1.school=12 ${n11} 3.school=12 ${n13} ///
    1.sex=12 ${n15} age=12 ${n16} 1.relig2=12 ${n18} income=12 ${n19} ///
    1.industry=12 ${n20} 2.industry=12 ${n21} 4.industry=12 ${n23} constant=12 ${n24}) ///
    mlabcolor(black) mcol(black) msize(small) mlabsize(2) xline(0.05, lcolor(black)) ///
    ci((p_model3[2,] p_model3[3,])) ciopts(lcolor(black))), graphregion(fcolor(white)) ///
    lcolor(white) lwidth(thick)) drop(0.selfemp 0.party 0.union 0.relig2 2.school 0.sex ///
    0.selfemp#c.engage2 3.industry) ///
    coeflabels(engage2="Pol Engage (std)" 1.selfemp="Self-Employed" ///
    1.selfemp#c.engage2="Self-Emp*Pol-Eng" 1.party="Republican" 2.party="Other" ///
    1.union="Union Member" 1.school="Grammar School" 3.school="College" ///
    1.sex="Female" age="Age" 1.relig2="Protestant" income="Family Income" ///
    1.industry="Agriculture" 2.industry="Business" 4.industry="Government" ///
    constant="Constant") groups(1.party 2.party = "{bf:Party}" 1.school ///
    3.school = "{bf:Schooling}" 1.industry 2.industry 4.industry = "{bf:Industry}") ///
    legend(order(1 "95% Confidence Interval")) xtitle("{it:P}-Values") ///
    order(engage2 ?.selfemp 1.selfemp#c.engage2 ?.party ?.union ?.school ?.sex ///
    age ?.relig2 income ?.industry constant) saving(ex51plot.gph, replace)

//Plot 5, Bottom Panel
mat p_model3 = J(24, 3, .)

permute threat2 _b, reps(1000) seed(5): ///
qui reg threat2 c.engage2##i.selfemp i.party i.union ib2.school i.sex age i.relig2 c.income ///
    ib3.industry if flag3==1
mat def mm1 = r(p)
mat def mm12 = r(ci)
forvalues i = 1/24 {
    mat p_model3[`i',1] = mm1[1,`i']
    mat p_model3[`i',2] = mm12[1,`i']
    mat p_model3[`i',3] = mm12[2,`i']
}

mat rownames p_model3 = engage2 0.selfemp 1.selfemp 0.selfemp#c.engage2 ///
    1.selfemp#c.engage2 0.party 1.party 2.party 0.union 1.union ///
    1.school 2.school 3.school 0.sex 1.sex age 0.relig2 1.relig2 income ///
    1.industry 2.industry 3.industry 4.industry constant

qui reg threat2 c.engage2##i.selfemp i.party i.union ib2.school i.sex age i.relig2 c.income ///
    ib3.industry if flag3==1, b
qui listcoef
mat list r(table)

#delimit ;

coefplot

(matrix(p_model3[1,]), keep(engage2)
mcol(black*.72) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[2,] p_model3[3,])) ciopts(lcolor(black*.72)))
(matrix(p_model3[1,]), keep(1.selfemp)
mcol(black*.16) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[2,] p_model3[3,])) ciopts(lcolor(black*.16)))
(matrix(p_model3[1,]), keep(1.selfemp#c.engage2)
mcol(black*.61) xline(0.05, lcolor(black))

```

```

ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.61)))
(matrix(p_model3[,1]), keep(1.party))
mcol(black*.60) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.60)))
(matrix(p_model3[,1]), keep(2.party))
mcol(black*.54) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.54)))
(matrix(p_model3[,1]), keep(1.union))
mcol(black*.003) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.003)))
(matrix(p_model3[,1]), keep(1.school))
mcol(black*.08) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.08)))
(matrix(p_model3[,1]), keep(3.school))
mcol(black*.23) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.23)))
(matrix(p_model3[,1]), keep(1.sex))
mcol(black*.12) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.12)))
(matrix(p_model3[,1]), keep(age))
mcol(black*.01) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.01)))
(matrix(p_model3[,1]), keep(1.relig2))
mcol(black*.10) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.10)))
(matrix(p_model3[,1]), keep(income))
mcol(black*.08) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.08)))
(matrix(p_model3[,1]), keep(1.industry))
mcol(black*.06) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.06)))
(matrix(p_model3[,1]), keep(2.industry))
mcol(black*.06) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.06)))
(matrix(p_model3[,1]), keep(4.industry))
mcol(black*.10) msymbol(square) xline(0.05, lcolor(black))
ci((p_model3[,2] p_model3[,3])) ciopts(lcolor(black*.10))),

graphregion(fcolor(white) lcolor(white) lwidth(thick))
coeflabels(engage2="Pol Engage (std)" 1.selfemp="Self-Employed"
1.selfemp#c.engage2="Self-Emp*Pol-Eng" 1.party="Republican" 2.party="Other"
1.union="Union Member" 1.school="Grammar School" 3.school="College"
1.sex="Female" age="Age" 1.relig2="Protestant" income="Family Income"
1.industry="Agriculture" 2.industry="Business" 4.industry="Government")
groups(1.party 2.party = "{bf:Party}" 1.school
3.school = "{bf:Schooling}" 1.industry 2.industry 4.industry = "{bf:Industry}")
legend(order(2 "Negative Standardized Estimate" 6 "Positive Standardized Estimate")
cols(1))
xtitle("{it:P}-Values")
order(engage2 ?.selfemp 1.selfemp#c.engage2 ?.party ?.union ?.school ?.sex
age ?.relig2 income ?.industry) saving(ex52plot.gph, replace) ;

#delimit cr

//Plot 6
qui reg threat2 c.engage2##i.selfemp i.party i.union ib2.school i.sex age i.relig2 c.income ///
ib3.industry if flag3==1

program define permtest
version 13.1
qui reg threat2 i.selfemp##c.engage2 i.party i.union ib2.school i.sex age i.relig2 c.income ///
ib3.industry if flag3==1
margins, at(engage2=(-1 0 1) selfemp=(0 1)) asobserved post
end

permute threat2 _b, reps(1000) seed(5): permtest if flag3==1

mat predm = J(3, 3, .)
mat predf = J(3, 3, .)

qui permtest
mat def pred_beta = e(b)
mat pred_betam = pred_beta[1,1..3]
mat defin pred_betaf = pred_beta[1,4..6]
forvalues m = 1/3 {
    global m'm' = round(pred_betam[1,'m'], .001)
}
forvalues f = 1/3 {

```

```

        global f'f' = round(pred_betaf[1,'f'], .001)
    }

    permute threat2 _b, reps(1000) seed(5): permtest if flag3==1
    mat def pm1 = r(p)
    mat def pm12 = r(ci)
    mat pm1m = pm1[1,1..3]
    mat pm12m = pm12[1..2,1..3]
    mat pm1f = pm1[1,4..6]
    mat pm12f = pm12[1..2,4..6]
    forvalues m = 1/3 {
        mat predm['m',1] = pm1m[1,'m']
        mat predm['m',2] = pm12m[1,'m']
        mat predm['m',3] = pm12m[2,'m']
    }
    forvalues f = 1/3 {
        mat predf['f',1] = pm1f[1,'f']
        mat predf['f',2] = pm12f[1,'f']
        mat predf['f',3] = pm12f[2,'f']
    }

    coefplot (matrix(predm[,1]), mlabels(r1=12 ${m1} r2=12 ${m2} r3=12 ${m3})) ///
    mlabcolor(black) xline(0.05, lcolor(black)) mcol(black) msize(small) ///
    mlabsize(2) ci((predm[,2] predm[,3])) ciopts(lcolor(black))) ///
    (matrix(predf[,1]), mlabels(r1=12 ${f1} r2=12 ${f2} r3=12 ${f3})) ///
    mlabcolor(gray) mcol(gray) msize(small) mlabsize(2) ///
    ci((predf[,2] predf[,3])) ciopts(lcolor(gray)), graphregion(fcolor(white) ///
    lcolor(white) lwidth(thick)) coeflabels(r1="-1 SD" r2="Mean" ///
    r3="+1 SD") legend(order(2 "Not S-E's Perception of Communist Threat (Standardized)" ///
    4 "S-E's Perception of Communist Threat (Standardized)") cols(1) size(vsmall)) ///
    xtitle("{it:P}-Values") xsize(1) ysize(1) ///
    ytitle("Political Engagement (Standardized)") saving(ex6plot.gph, replace)

    //Plot 7
#delimit ;

coefplot (matrix(predm[,1]), ci((predm[,2] predm[,3])) bcolor(black))
(matrix(predf[,1]), ci((predf[,2] predf[,3])) bcolor(gray)),
graphregion(fcolor(white) lcolor(white) lwidth(thick)) coeflabels(
r1="{bf:-1 SD}" "Y-hat{subscript:NS-E} = ${m1}" "Y-hat{subscript:S-E} = ${f1}"
r2="{bf:Mean}" "Y-hat{subscript:NS-E} = ${m2}" "Y-hat{subscript:S-E} = ${f2}"
r3="{bf:+1 SD}" "Y-hat{subscript:NS-E} = ${m3}" "Y-hat{subscript:S-E} = ${f3}"
ciopts(lcolor(black) recast(rcap rcap))
recast(bar) vertical
ytitle("{it:P}-Value") barwidth(.3) fcolor(*.8) citop xlabel(,
labsize(vsmall)) yline (0.05, lcolor(black)) legend(order(1 "Not S-E's PCT {it:P}-Value"
3 "S-E's PCT {it:P}-Value"))
xtitle("Political Engagement (Standardized)") saving(ex7plot.gph, replace) ;

#delimit cr

//Plot 8
program define permtest2
version 13.1
qui reg threat2 i.selfemp##c.engage2 i.party i.union ib2.school i.sex age i.relig2 c.income ///
ib3.industry if flag3==1
margins, at(engage2=(-1(.1)1) selfemp=(0 1)) asobserved post
end

mat predm2 = J(21, 3, .)
mat predf2 = J(21, 3, .)

qui permtest2
mat def pred_beta2 = e(b)
mat pred_betam2 = pred_beta2[1,1..21]
mat defin pred_betaf2 = pred_beta2[1,22..42]
forvalues m = 1/21 {
    global m'm' = round(pred_betam2[1,'m'], .001)
}
forvalues f = 1/21 {
    global f'f' = round(pred_betaf2[1,'f'], .001)
}

permute threat2 _b, reps(1000) seed(5): permtest2 if flag3==1
mat def pm1b = r(p)
mat def pm12b = r(ci)
mat pm1m2 = pm1b[1,1..21]
mat pm12m2 = pm12b[1..2,1..21]

```

```

mat pm1f2 = pm1b[1,22..42]
mat pm12f2 = pm12b[1..2,22..42]
forvalues m = 1/21 {
    mat predm2['m',1] = pm1m2[1,'m']
    mat predm2['m',2] = pm12m2[1,'m']
    mat predm2['m',3] = pm12m2[2,'m']
}
forvalues f = 1/21 {
    mat predf2['f',1] = pm1f2[1,'f']
    mat predf2['f',2] = pm12f2[1,'f']
    mat predf2['f',3] = pm12f2[2,'f']
}

#delimit ;

coefplot (matrix(predm2[,1]), ci((predm2[,2] predm2[,3])) lcolor(black)
    ciopts(lcolor(black) recast(rcap)))
    (matrix(predf2[,1]), ci((predf2[,2] predf2[,3])) lcolor(gray)
    ciopts(lcolor(gray) recast(rcap))),
    graphregion(fcolor(white) lcolor(white) lwidth(thick))
    recast(line) vertical coelabels(
    r1="""{bf:-1 SD}" "Y-hat{subscript:NS-E} = ${m1}" "Y-hat{subscript:S-E} = ${f1}""",
    r11="""{bf:Mean}" "Y-hat{subscript:NS-E} = ${m11}" "Y-hat{subscript:S-E} = ${f11}""",
    r21="""{bf:+1 SD}" "Y-hat{subscript:NS-E} = ${m21}" "Y-hat{subscript:S-E} = ${f21}""",
    r2=" " r3=" " r4=" " r5=" " r6=" " r7=" " r8=" " r9=" " r10=" " r12=" "
    r13=" " r14=" " r15=" " r16=" " r17=" " r18=" " r19=" " r20=" ")
    ytitle("{it:P}-Value") xtitle("Political Engagement (Standardized)")
    yline(0.05, lcolor(black)) xlabel(,labsize(tiny) notick)
    legend(order(1 "Not S-E's PCT {it:P}-Value" 3 "S-E's PCT {it:P}-Value"))
    saving(ex8plot.gph, replace) ;

#delimit cr

*****
*** COMMANDS END HERE ***
*****
log close
exit

```