

PLSC 40502: Problem Set 3

February 9, 2024

This problem set is due at **11:59 pm on Wednesday, February 21st**.

Please upload your solutions as a .pdf file saved as “Yourlastname_Yourfirstinitial_pset3.pdf”). In addition, an electronic copy of your .Rmd file (saved as “Yourlastname_Yourfirstinitial_pset3.Rmd”) must be submitted to the course website at the same time. We should be able to run your code without error messages. In addition to your solutions, please submit an annotated version of this .rmd file saved as “Yourlastname_Yourfirstinitial_pset3_feedback.rmd”, noting the problems where you needed to consult the solutions and why along with any remaining questions or concerns about the material. In order to receive credit, homework submissions must be substantially started and all work must be shown.

Problem 1

We’ll continue with the same dataset from last problem set, the **Congressional Election Study**, an annual, large, nationally representative survey of the American population. We’ll focus on the same outcome - predicting the level of support for the Section 232 Steel and Aluminum tariffs.

Be very careful in reading the variable names and definitions (feel free to use your code from last problem if it differed from the code in this problem).

The code below will load in the Common Content from the 2020 CES. Please download the file directly from <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi%3A10.7910/DVN/E9N6PH> and place it into your data folder:

```
ces <- read_csv("data/CES20_Common_OUTPUT_vv.csv")
```

The next few code fragments will carry out the relevant pre-processing to generate bins for all four of the covariates of interest.

```
# State FIPS
ces$statefips <- str_pad(ces$inputstate, 2, pad="0")

# Outcome
ces <- ces %>% mutate(tariff = case_when(CC20_338b == "1" ~ 1,
                                         CC20_338b == "2" ~ 0,
                                         TRUE ~ NA_real_))

# Covariates
# Age
ces <- ces %>% mutate(age = 2020 - birthyr)
ces <- ces %>% mutate(age_bin = case_when(age>=18&age<=29 ~ "18-29",
                                         age>=30&age<=44 ~ "30-44",
                                         age>=45&age<=64 ~ "45-64",
                                         age>=65 ~ "65+"))

# Gender
ces <- ces %>% mutate(gender_bin = case_when(gender == 1 ~ "Male",
                                             gender == 2 ~ "Female"))
```

```
# Race
ces <- ces %>% mutate(race_bin = case_when(race==3|hispanic==1 ~ "Hispanic",
                                           race==1 ~ "White",
                                           race ==2 ~ "Black",
                                           race == 4 ~ "Asian",
                                           TRUE ~ "Other"))

# Education
ces <- ces %>% mutate(educ_bin = case_when(educ>=1&educ<=2 ~ "H.S. or less",
                                           educ>=3&educ<=4 ~ "Some college",
                                           educ==5 ~ "College degree",
                                           educ==6 ~ "Postgraduate"))
```

Make the variables factors with the correct baselines

```
ces$gender_bin <- relevel(as.factor(ces$gender_bin), "Male")
ces$age_bin <- relevel(as.factor(ces$age_bin), "18-29")
ces$educ_bin <- relevel(as.factor(ces$educ_bin), "H.S. or less")
ces$race_bin <- relevel(as.factor(ces$race_bin), "White")
```

Subset down to the non-missing data

```
ces_full <- ces %>% filter(!is.na(tariff)&!is.na(age_bin)&!is.na(race_bin)&
                          !is.na(gender_bin)&!is.na(educ_bin)&!is.na(statefips))
```

Part A

Using just the data from the CES survey and none of the covariates, estimate the share of residents in Rhode Island (FIPS code = 44) who support the tariff. Use the survey weights `commonweight`. Provide a 95% confidence interval for your estimate.

Part B

Using `brms`, fit a multilevel regression model that models the probability of supporting the tariff as a function of the four discrete covariates and a varying state-level intercept that pools to a common “us-wide” mean.

Warning: On my laptop, running 4 cores for 1000 burnin and 2000 total iterations took approximately **30 minutes** on this model so be aware of the likely run-time and prep accordingly. It may help to run fewer iterations to get a sense of the likely overall runtime on your computer.

You can and should **save** the output of a finished model in an `.Rdata` or `.rds` file to load into your workspace so that you don’t have to repeatedly run the model after your first

Part C

Now fit a model using `brms` that incorporates a state-level intercept that is pooled to a “census region-wide” mean instead (`region` in `ces`)

Hint: Consider a model that includes both a state and a region random intercept - in this model, the state-level intercepts can be interpreted as deviations from the “region” intercept.

Part D

Load in the state-level post-stratification frame from the 2020 ACS as well as the state-region codebook (you’ll want to merge this in to get the region codes for each frame).

```
acs_2020 <- read_csv("data/state_frame_2020_IPUMS.csv")
state_to_region <- read_csv("data/state_to_region.csv")
```

Obtain the post-stratified posterior mean estimate for the proportion supporting the tariff in each state using your model from Part B and your model for Part C. For each of the models, report the posterior mean estimate for Rhode Island and construct a 95% credible interval. Compare the results from the two models with your result from Part A.

Hint: Use `fitted` on your `brmsfit` objects to obtain predicted means for the post-stratification frame. Using the argument `summary = T` will generate the posterior means, but you may want to turn `summary = F` to obtain predictions for each MCMC draw in order to generate a credible interval for your Rhode Island estimate.

Part E

Using the `usmap` package, generate two maps of your posterior mean MRP estimates - one for the state-varying intercepts model and another for the state-/region- intercepts model. Compare the results. What do you notice about the output of the two models?