GR5065 HW5

1 Auto Insurance Claims

Skim this paper on the recent use of "telematic" apps that auto insurance companies incentivize drivers to use in order to collect data on how good or bad they are driving in order to determine how high their auto insurance premiums are. Such datasets are never publicly available, so the authors describe how they created a synthetic dataset that mimics the characteristics of a dataset that was given to them privately by a Canadian auto insurance company.

You can load the synthetic data via

```
FILE <- "telematics_syn-032021.csv"
if (!file.exists(FILE)) {
  ROOT <- "http://www2.math.uconn.edu/~valdez/"
  download.file(paste0(ROOT, FILE), destfile = FILE)
}
UBI <- readr::read_csv(FILE, show_col_types = FALSE)
```

The variables are described in the paper. In the interest of making the models run faster, you can take a subset of the observations for training and testing

```
library(dplyr)
incl <- sample(1:nrow(UBI), size = 10000, replace = FALSE)
training <- slice(UBI, incl)
UBI <- slice(UBI, -incl)
incl <- sample(1:nrow(UBI), size = 10000, replace = FALSE)
testing <- slice(UBI, incl)
```

1.1 Poisson Model

Use the `brm` function to estimate a model where `NB_Claim` — the number of auto insurance claims — is the outcome using `data = training` and `family = poisson` with the default log link. You can use any predictors that you want except `AMT_Claim` (which is the monetary amount of those claims), but you must use at least one continuous or integer predictor inside `s()` in order to estimate a spline function.

Also, be sure to specify `save_pars = save_pars(all = TRUE)` when calling `brm` because you may need it when estimating the ELPD with moment matching.

1.2 Zero Inflated Poisson Model

We have not discussed zero inflated models in lecture but they were covered in Chapter 12 of the McElreath reading. These combine a Bernoulli model for whether the outcome is zero with — given that the Bernoulli random variable is not zero — a count model for the number of events.

- First, explain why a zero-inflated model might be appropriate in the context of auto insurance claims.
- Second, estimate a zero-inflated Poisson model that generalizes the model in the previous subproblem using `data = training` and `family = zero_inflated_poisson` . In order to put predictors into the Bernoulli part — where the outcome is labeled `zi` — and the count part, you have to use the custom `brmsformula` function `brmsformula(NB_Claim ~ ..., zi ~ ...)` .

1.3 Model Comparison

- Use the `loo_compare` function to evaluate which of the models in the previous two subproblems is expected to predict better for N future observations. Specify `moment_match = TRUE` if there are any observations where the Pareto k values are too high.
- Call the `log_lik` function for the preferred model with `newdata = testing` to obtain the log probabilities for the N observations in the testing data, which will be a matrix with R rows and N columns. Call `apply` on this matrix with arguments `MARGIN = 2` and `FUN = loo::logMeanExp` to obtain a vector of size N with the average (over the posterior distribution) log predictive probability for each testing observation and then call `sum` on that vector to estimate the ELPD for the entire testing data. How much does this estimate differ from what you estimated above?

2 Pandemic Election

Data were recently released that allow researchers to join the dataset from the post-2020 American National Election Survey (ANES) with the General Social Survey (GSS) panel survey where the same people were first asked questions in either 2016 or 2018 and then asked a largely identical set of questions a few months before the 2020 election. However, these surveys were conducted over the internet in 2020 due to the covid pandemic, whereas they were historically conducted with face-to-face interview methods in previous years.

To load these data, it is best to start with the Stata version of the GSS panel

```
suppressPackageStartupMessages(library(haven))
if (!file.exists("GSS_2020_panel_stata_1a.zip")) {
  download.file("https://gss.norc.org/Documents/stata/GSS_2020_panel_stata_1a.zip",
                destfile = "GSS_2020_panel_stata_1a.zip")
  unzip("GSS_2020_panel_stata_1a.zip")
}

GSS <- as_factor(read_dta("gss2020panel_r1a.dta"))
```

The `GSS` data.frame is organized in "wide" form with the fields that have a `_1a` suffix being collected in 2016 the fields that have a `_1b` suffix were collected in 2018, and the fields with a `_2` suffix were collected before the 2020 election.

To obtain the 2020 ANES data, you have to register with your Columbia email at

https://electionstudies.org/data-center/anes-gss-2020-joint-study/

and then download the Stata version of the dataset from the top left to your working directory. At which point, you can execute

```
unzip("anes_timeseries_2020_gss_stata_20220408.zip") # creates a few files
ANES <- as_factor(read_dta("anes_timeseries_2020_gss_stata_20220408.dta"))
```

These variables were collected in late 2020, after the November election.

To merge the two datasets, use

```
ANES_GSS <- inner_join(ANES, GSS, by = c(YEARID = "yearid"))
```

Refer to the documentation of the ANES and GSS for more information on the 5031 variables.

2.1 Proportional Odds Model

Choose one ordinal outcome variable in `ANES_GSS` that can be modeled using `brm` with `family = cumulative` and the default logit link function. You can use any plausible predictors but must include at least one ordered factor as a predictor where it makes sense to restrict the coefficients on the generated dummy variables to be monotonically related to the outcome probabilities. In addition, chose at least one "grouping variable" — which does not have to be geography — and allow the intercept and slope of some predictor to deviate from the population parameters by level of the grouping variable (`g`) by including something like `(1 + x | g)` in your formula.

2.2 Visualization

Use the `conditional_effects` function to plot the relationship between the outcome probabilities and the monotonic predictor. What do you conclude from this plot?

2.3 Frequentism

Briefly explain why the hierarchical model that you estimated above would not be appropriate from a Frequentist perspective, even if you used maximum likelihood rather than MCMC.