GR5065 Assignment 3

Due by 4PM on March 5, 2020

1 Current Population Survey (CPS)

1.1 Getting CPS Data

You are going to need to download some CPS data, which is a monthly survey on individuals' and households' economic conditions. The easiest way to do that is to go to

http://ceprdata.org/cps-uniform-data-extracts/cps-outgoing-rotation-group/cps-org-data/

and download the compressed file for the year that you were born to the same directory as your RMarkdown file. Then, unzip that file in the same directory to produce a Stata formatted file, which will have a dta file extension. The Stata formatted file can be loaded into R properly with something like

```r
library(haven)
CPS <- as_factor(read_dta(dir(pattern = "^cepr_.*dta$")))
# as_factor changes the categorical variables in Stata to R factors
```

Finally, filter the `CPS` data.frame down to the month that you were born using the `month` variable so that it is not too big.

A brief description of the variables in `CPS` and the values they take (if categorical) can be obtained by

```r
defs <- sapply(CPS, FUN = attr, which = "label")
vals <- sapply(CPS, FUN = attr, which = "levels")
```

Additional documentation of these variables can be found at http://ceprdata.org/cps-uniform-data-extracts/cps-basic-programs/cps-basic-documentation/ or the from the links on that page but note that `CPS` does not include the household-level variables and recodes / combines / renames some of the individual-level variables. If you are familiar with Stata, it might be helpful to look at the dofiles that create the dataset which can be found at http://ceprdata.org/cps-uniform-data-extracts/cps-outgoing-rotation-group/cps-org-programs/ .

1.2 Prior Predictive Distribution

Draw a picture of a linear process that represents the prior predictive distribution for (the logarithm of) *hourly* wages using some of the other (possibly transformed) variables in the CPS as predictors. Use a normal prior for the intercept, assuming that the predictors will be centered. Otherwise, you should use inverse CDF transformations of standard uniform random variables to represent your beliefs about the other parameters marginally, rather than normal priors or a prior on the R^2.

You can use Tikz to draw the picture like I did in the RMarkdown files for Week06 or you can draw it on a piece of paper, take a picture of it with your phone, and include that in your Rmarkdown file.

1.3 Drawing from the Prior Predictive Distribution

Call

```r
rstan::expose_stan_functions("quantile_function.stan")
```

to access the relevant functions. Then draw $S = 1000$ times from the prior predictive distribution of log hourly wages over a sample of size N people that you described in the previous subproblem. Beforehand, you should drop any observations that are NA on any of the predictors you are using. Then, remember to center *all* the predictors as you draw from the prior predictive distribution.

For each of these S simulated outcome vectors of size N, calculate the prior

$$R_s^2 = \frac{\frac{1}{N-1} \sum_{n=1}^{N} \left(\widetilde{\mu}_n - \overline{\mu}\right)^2}{\widetilde{\sigma}^2 + \frac{1}{N-1} \sum_{n=1}^{N} \left(\widetilde{\mu}_n - \overline{\mu}\right)^2}$$

where $\overline{\mu}$ is the average of the conditional means over the N data points in the s-th simulation

1.4 Drawing from the Posterior Distribution

Use the `stan_lm` function in the rstanarm R package to draw from the posterior distribution of the parameters conditional on the logarithm of real wages (`rw` in the CPS dataset) and the predictors in the previous subproblem. Since rstanarm does not yet implement inverse CDF transformations, use the median of the R^2 values that you calculated over the S datasets in the previous subproblem to specify the `prior` argument. Pass a call to the `normal` function (with `autoscale = FALSE` in order to interpret it in raw log-wages rather than standardized terms) as your `prior_intercept` based on what you used above for the prior on the intercept.

1.5 Interpreting the Posterior Coefficients

Use the `as.matrix` function to obtain all the draws from the posterior. For which coefficients are all of the posterior draws positive?

2 Stock Car Racing

This question is about stock car racing, which is very simple: The first car to complete a specified number of laps wins. You can grasp the essence of it from this short commercial for one of the most important races each year:

https://youtu.be/aI4home7J8c?t=10

You need to download a dataset on the results of such car races via

```r
nascar <- read.table("http://ww2.amstat.org/publications/jse/datasets/nascard.dat.txt",
                     stringsAsFactors = FALSE)
colnames(nascar) <- c("RaceID", "Year", "RaceOfYear",
                      "Finish", "Start", "Laps", "PrizeMoneyWon",
                      "Cars", "Make", "Driver")
nascar$Winner <- nascar$Finish == 1
```

There are 36 races per year. Contact Ben on CampusWire to see which `Year` of data you should analyze for the rest of this problem. The variables are described in more detail in Appendix 1A of

http://jse.amstat.org/v14n3/datasets.winner.html

We are only looking at the dataset of drivers.

In the week or two leading up to each race (which takes place on a Saturday or Sunday), each driver has several opportunities to drive by themselves on the track in an attempt to complete one lap in the shortest time. Whoever has the shortest time to complete one practice lap gets to start the actual race at the front left of the line of cars. The driver with the second fastest practice lap gets to start the race at the front right of the line of cars. And so on, such that the driver with the slowest practice lap has to start at the very back of the line of cars.

The starting position of the car is reflected in the `Start` variable. Having a lower `Start` variable increases the probability of winning the race in three ways:

1. It means your (heavily customized) car is working well.
2. It means there are fewer cars that you have to pass in order to win.
3. The pitboxes where cars stop to fill up on gasoline and get new tires (see the above video starting at 0:47) are also arranged in the same order that the cars start. If multiple cars are trying to exit the pit area at the same time, they can get in each other's way and cause slowdowns. So, it is slightly advantageous to be able to park toward the front of the pit area.

2.1 Inverse CDF

Suppose U is distributed standard uniform and

$$Y = \mu + \ln\left(-\ln U\right)$$

where $\mu \in \mathbb{R}$ is a location parameter but not actually $\mathbb{E}Y$.

Write an R function to draw from the implied distribution of Y using this inverse CDF function.

2.2 Prior Predictive Distribution

Draw $S = 1000$ times from the prior predictive distribution of the *time* it would take each car to finish the first race in your dataset, even though such a variable is not among those in the `nascar` dataset. Use `Start` as the primary predictor but also including a dummy variable for each driver. You should use normal priors for all of the coefficients and make μ_n a linear function of the starting position and driver of the n-th car in the first race. Use the function you wrote in the previous problem to draw the time it takes each car to finish, which should be about 3.5 hours on average.

The prior predictive distribution of the winner of the first race is then defined as the car with the smallest finishing time. Under your prior, what is the probability that the car who starts in the first position wins the race?

2.3 Posterior Distribution

Use the `stan_clogit` function in the rstanarm package to draw from the posterior distribution of the slope parameters conditional on the `Winner`, `Start`, and `Driver` over all the races in a year.

The `stan_clogit` function uses a version of a standard logistic likelihood. However, unlike a regular logit model, in this case the outcomes are *not* independent across cars within a race. The fact that someone wins necessarily implies that everyone else in the race does not win. The winners are conditionally independent across races. For this reason, the `stan_clogit` function has a mandatory `strata` argument, which you should set equal to `RaceID`, in order to indicate how the cars are grouped into races. You should use the same normal priors as in the previous subproblem. However, there is no intercept, even though you might have included one as part of the process of drawing from the prior predictive distribution. Since the intercept would be a constant across all cars in the race, there is no information in these data with which to estimate it.

2.4 Which Driver Is Best?

We can interpret the posterior distribution of the coefficients on the driver-specific dummy variables as a measure of driver skill, where higher numbers are better. Use the `as.matrix` function to obtain a matrix of posterior draws of all the parameters. What is the posterior probability that each driver in your dataset is the most skilled driver that year?