APSTA-GE 2123 Assignment 2

Due by 1:45 PM on May 4, 2022

1 On data science

Read this article. What is the relationship between data science and Bayesian analysis? What aspects of Bayesian analysis are or are not data science? What parts of data science are outside of Bayesian analysis? How does Bayesian analysis as it is commonly understood in the data science realm differ from the approach to Bayesian analysis that we have put forward in this course?

2 YouTube Views

Read https://osf.io/ux9et even though it is Frequentist.

The data on the number of times a video was watched on YouTube can be loaded with

```r
youtube <- readr::read_csv("https://osf.io/25sz9/download")
```

The main two predictors are `scol`, which is a measure of how accurate the video is, and `age2`, which is a measure of how long the video has been available. The outcome variable is called `views2`. You can include other predictors as well.

2.1 Stan Program

Write a Stan program that models `views2` with a negative binomial distribution and an antilog inverse link function. You will have to choose priors on all of the parameters. The **data** block of your Stan program should also include a line that says `int<lower = 0, upper = 1> prior_only;` that can be switched on to avoid conditioning on the data. The **generated quantities** block of your Stan program should draw from the predictive distribution of `views2` and store the results in a symbol called `y_rep`. In addition, the **generated quantities** block of your Stan program should calculate the contribution to the log-likelihood of each observation and store it in a symbol called `log_lik`.

2.2 Prior Predictive Distribution

Call the Stan program you wrote in the previous subproblem specifying `prior_only = TRUE` in the list of data that is passed to Stan. You can obtain the draws from the prior predictive distribution by calling `rstan::extract` on the object with `pars = "y_rep"`, which will output a list with one element that is a $4000 \times N$ matrix of non-negative integers. Make a scatter plot of the columnwise means of this object divided by `age2` on the vertical axis and `scol` on the horizontal axis. In what ways does this scatterplot seem like a reasonable data-generating process and in what, if any, aspects does it seem unreasonable?

2.3 Expected Log Predictive Density under the Prior

One way to quantify the contribution of the data is to compare the estimated ELPD under the prior to the ELPD under the posterior. Call `loo` on the object produced by Stan to compute the prior ELPD.

2.4 Posterior Distribution

Call the Stan program you wrote again but now with `prior_only = FALSE` in the list of data that is passed to Stan. Be sure to eliminate any warnings from Stan that may arise.

How would you describe your posterior beliefs on the effect of a video's accuracy on its views? In addition, does the posterior distribution suggest that a Poisson model would be adequate or is a negative binomial necessary?

2.5 Expected Log Predictive Density under the Posterior

Call `loo` on the object produced by Stan in the previous subproblem to estimate the posterior ELPD. How much bigger is it than the estimated prior ELPD? Which, if any, observations are inconsistent with the assumption of not having too major an influence on the posterior distribution?

3 Vaccination Rates

Revisit the data we used in Week3 on covid vaccinations at the county-level. You can load these data via

```
Gabba <- readr::read_csv(file.path("..", "..", "Week3", "Gabba.csv"),
                         col_types = c("ccccdddddddddd"), skip = 1, col_names =
                           c("FIPS", "ST", "State", "County", "Trump#", "Votes#", "Trump", "Pop",
                             "Vaccinated#", "Vaccinated", "Death1", "Death2", "Death3", "Death4"))
Gabba <- dplyr::filter(Gabba, Vaccinated < 100) # some data points were messed up
```

In Week3, we modeled the vaccinated percentage using a normal distribution. In this problem, we are going to utilize hierarchical GLMs.

3.1 Beta Likelihood

One concern when modeling a percentage with a normal distribution is that it puts some probability on the percentage being less than 0 or greater than 100. This problem can be avoided by modeling the proportion (not the percentage) with a beta distribution. The PDF of the Beta distribution is usually written as

$$f\left(x \mid \alpha, \beta\right) \propto x^{\alpha - 1} \left(1 - x\right)^{\beta - 1}$$

However, for the purposes of modeling, it is usually much better to make the substitutions

$$\alpha = \mu \kappa$$
$$\beta = \left(1 - \mu\right) \kappa$$

where $\kappa > 0$ is a parameter that governs the dispersion of the errors between x and its expectation, $\mu \in (0, 1)$. The PDF of the Beta distribution then becomes

$$f\left(x \mid \alpha, \beta\right) \propto x^{\mu \kappa - 1} \left(1 - x\right)^{\left(1 - \mu\right) \kappa - 1}$$

The expectation, μ, can then be modeled as a function of predictors using the usual GLM approach of specifying a linear predictor $\eta_n = \alpha + \sum_{k=1}^{K} \beta_k x_{nk}$ and an inverse link function $\mu_n = \frac{1}{1 + e^{-\eta_n}}$ that ensures $\mu_n \in (0, 1)$.

Write a Stan program that models the proportion of people in a county who are fully vaccinated as a function of the proportion of vote share received by Trump in that county and the state that the county is in. The effect of the state should have a hierarchical form, but you do not need to allow the effect of Trump support to vary by state. The `generated_quantities` block of your Stan program should draw from the predictive distribution and store it in a symbol called `y_rep`.

3.2 Beta Posterior

Call the Stan program you wrote in the previous subproblem, but make sure to eliminate any warnings that may be thrown during the sampling. How would you describe your posterior beliefs about the relationship between Trump support and vaccination prevalence?

3.3 Binomial Likelihood

Another approach would be to model the *number* of people who are fully vaccinated in a county but to condition on the population of that county. These variables are available in `Gabba` as `Vaccinated#` and `Pop` respectively but note that you have to enclose the former inside a pair of backticks in R to avoid it interpreting `#` as the start of a comment.

You can them introduce the log-odds of someone being vaccinated as $\eta_n = \alpha + \sum_{k=1}^{K} \beta_k x_{nk}$ and map it to the probability of vaccination with an inverse link function $\mu_n = \frac{1}{1+e^{-\eta_n}}$ that ensures $\mu_n \in (0, 1)$. Again, η_n should depend on the percentage of Trump support in that county as well as the state that the county is in. The effect of the state should have a hierarchical form, but you do not need to allow the effect of Trump support to vary by state.

Write a Stan program that models the number of people in a county who are vaccinated with a binomial log-likelihood. The `generated_quantities` block of your Stan program should draw from the predictive distribution — divided by the population of that county — and store the result in a symbol called `y_rep`.

3.4 Binomial Posterior

Call the Stan program you wrote in the previous subproblem, but make sure to eliminate any warnings that may be thrown during the sampling. Is this model preferable or not preferable to the previous model with the beta likelihood? You should, at a minimum, consider the fit of the posterior predictive distributions, which can be obtained by calling `rstan::extract` with `pars = "y_rep"`.