APSTA-GE 2123: Bayesian Inference

Analysis Project Guidelines

Instructor: eric.novik@nyu.edu

19 Aug 2024

At the beginning of the semester, we will randomly assign students to 3 or 4-person teams. Each team will perform the analysis, write a report, and present their methods and results to the class during finals week. Short project proposals are due at the end of the third week of class. You should use R Markdown or Quarto to prepare your reports. You can use whatever you want (e.g., PowerPoint) to prepare your presentation, but we recommend Quarto Presentation. You should use LaTeX to typeset your equations. See the LaTeX section for the guide.

1 Project Proposal

The proposal should not exceed one page in length. You should include the following information in the proposal:

1. Title of your project
2. What are you hoping to learn from the analysis
3. Short description of the data source
4. Optional: if you know how you plan to analyze data, please say so
5. Names of team members and the intended roles on the project. It's fine to have overlapping or non-overlapping roles, with one caveat below.

Ideally, discuss the project with your team and agree on what needs to be done. We recommend one person be the project manager and keep everyone on track. You can elect a PM or choose one randomly, or not at all – it's up to you. We recommend that *every person* performs the function of a final product tester – reads all the documents, runs all the code, and discusses the results with the teammates.

2 Project Report

For inspiration, you can take a look at the Introduction to multilevel modeling using rstanarm m.html) by Lee et al. Here is the complete list of Stan case studies.

2.1 Proposal

You should include the original project proposal as the first section of the report. You don't have to follow your proposal to the letter, but you should discuss significant deviations.

2.2 Attribution

In this section, state the teammates' names; each person should briefly describe their role with the understanding that not everyone can or should do everything. The grade will be assigned to the project team, not each individual.

2.3 Raw Data Summary

Perform an exploratory data analysis and highlight some interesting features in your data. Plot important relationships. Only summarize key features – please do not plot every variable, every two-way correlation, etc. Be selective and be nice to your readers.

2.4 Statistical Model

Describe the statistical model. Use statistical notation to specify your likelihood and all the priors. Briefly explain why you have selected a particular likelihood and prior(s). If you are not familiar with LaTeX, see the LaTeX section.

2.5 Prior Predictive Simulation

Perfome prior predictive simulation and compare it to the observed data. This should help you tune your prior distributions. Make sure to plot your prior predictive distribution. Does the scale seem reasonable?

2.6 Modeling Fitting, PPCs, and Model Selection

Fit your model in stages and perform model evaluation and selection as discussed in class. Summarize key convergence diagnostics, including R hats.

2.7 Discussion

Provide a short description of the limitations of your analysis, what worked and what didn't, and how the model can be improved.

3 Project Presentation

Prepare five (+/- 1) slides summarizing your work: 1) Title with the name of the project and team members; 2) Research questions; 3) Statistical model; 4) Summary of results/predictions; 5) Conclusions.

You will only have 10 minutes to present, followed by a couple of questions from the audience. Don't cram every piece of information on every slide. When it comes to slides, less is more.

4 Elements of Style

You should develop a consistent style for presenting your work, including graphs and tables. Here are a few guidelines.

- Label all your graphs and axes and specify units if applicable.
- The same goes for the tables if you choose to include them. In R, `knitr::kable` is a good and simple table generator, but many others exist.
- Don't use the default graph sizes, as they are generally too big (see the reference paper at the end of this section.) If you use R Markdown, see this. If you are using Quarto, see this and this.
- We recommend not using default point and line sizes, which are usually too large. To modify the defaults, you can use the `size` aesthetic in `geom_point` and `linewidth` aesthetic in `geom_line`.
- Use simple (ggplot) themes; you don't want the background of your plot to have too many high-contrast lines, as it is hard to see the data.
- Do not use box plots; instead, use `ggplot2::geom_linerange` or equivalent. Box plots waste space and are no longer necessary. Sorry, John Tukey, we still love you.
- If you display uncertainty intervals, you should indicate the interval width (i.e., 50%, 90%, etc.) and the type of interval (i.e., quantile, HDI, confidence). 50% posterior quantile intervals are nice to have when plotting over observations, as the reader can quickly assess calibration (i.e., about half the points should be inside and half outside the interval.)
- If you want to show several code-generated plots side-by-side, use `gridExtra::grid.arrange` or something like that.
- It is important to have a consistent coding style. See this, particularly if you use R's `tidyverse`. For Stan code, see this section of the Stan User's Guide.

See, this paper from Alex Kale, Matthew Kay, and Jessica Hullman for an example of some of the best practices in the field. Notice how each visualization is labeled and carefully designed.

5 Typesetting Equations with LaTeX

You should use LaTeX to typeset mathematical expressions. Markdown and Quarto documents support both inline and display equations. For inline equations, use single dollar signs like `$y`

`= f(x)$`, which will produce the following output: $y = f(x)$. For display equations, use double dollar signs like this:

```
$$ \begin{eqnarray} y &=& f(x) \\ g &=& f(z) \end{eqnarray} $$
```

The above will produce the following output:

$$\begin{aligned} y &= f(x) \\ g &= f(z) \end{aligned}$$

The double backslash is a new line character, and the ampersands align equations by the equal sign. The easiest way to get started is to use an online WYSIWYG equation editor and paste the formula into RStudio.