R Shiny Proposal

Aparajita Sur and Jack Wolf

4/2/2021

1. App name: *SMART Missingness Simulator*
2. App purpose: what is the main purpose of your app? Evaluate the performance of multiple imputation (MI) in a 2-stage SMART study depending on

   1. The degree of MAR vs. MNAR,
   2. the level of missingness (%), and
   3. the sample size.

   (There is currently no literature on how violations of the MAR assumption in SMART studies will affect analyses.)
3. What data source(s) will you be using, if any? Are they publicly available? If not, indicate why not, and if it will be possible to make your app publicly available after the end of the class.

   Source: an artificial SMART study dataset provided by Shortreed et al. (2014) in the paper "A multiple imputation strategy for sequential multiple assignment randomized trials." The dataset closely mimics first two stages of the Clinical Antipsychotic Trials of Intervention Effectiveness (CATIE) SMART study. This dataset is publicly available.

   We will simulate a dataset based on this artificial dataset, which will be publicly available on Github.
4. App functions: What are the main functions/actions that the app will allow the user to do? It is OK to be "ambitious" here and describe functions that you are not 100% sure you will be able to implement. I would rather that you try to push your R programming limits (even if, in the end, you can't complete everything you planned) than be overly safe and stick to things you already know how to do. The app will allow the user to adjust a slider for the level (%) of missingness in the SMART, the sample size of the SMART study, the number of simulations and the degree of non-random (MNAR) missingness. Additionally, the app will have a toggle to dictate if multiple imputation is used (vs complete case). The user can select which metrics the app should present (ex: coverage probability, MSE etc). The user can also compare several scenarios and choose which scenarios they are interested in.
5. Programming challenges: What are the main programming challenges that you anticipate facing in implementing your app?

   - Optimizing the computational cost of re-analyzing the dataset when the user inputs new settings.
   - Controling the missingness mechanism such that the missingness matches the user input (both the missingness percentage and whether MAR or MNAR).
6. Division of labor: How will your team collaborate to complete the project? Will each person be responsible for a particular function, or will team members work together on the entire codebase?

   We will collaborate through a GitHub repository. Aparajita has already written code to simulate and evaluate the performance of MI under varying conditions. We will expand on this code to create a user-friendly interface to control these conditions and more. We plan to work together on the entire codebase and code together over Zoom.