

Progress Documentation

- 10 / 08 / 2019:
 - Nimble crashes my Mac laptop every now and then, so I should do more simulation with that.
- 10 / 09 / 2019:
 - Meeting with Dr. Kahle. I want to look into getting the WINE path set globally instead of specifying it each time. I also have been dealing with the problem of getting STAN compile time included or not included in the benchmark file. Currently I'm having two function calls to "mark", one that uses saved rds file and has STAN already compiled, and one that renames the rds file to force recompile. I could instead try to split STAN up into the compile function and the run function.
- 10 / 09 / 2019:
 - I am interested not just in time it takes for each model to run, but also the accuracy of each model. For models that can be calculated analytically, I can look at the MSE of each method in their posterior means compared to the true posterior mean. For those where I can't calculate analytically, I can run STAN for a super long time and get the posterior mean, and maybe check it with a Gibbs sampler as well, and that will be my "true" posterior mean when calculating the MSEs for each method in that model setup.
- 10 / 10 / 2019:
 - How does each method do with specified number of iterations? 100? 1000? How close do they get to the truth?
- 10 / 30 / 2019:
 - I have successfully made the benchmarking process more clean by having a "benchmark-function.R" file that includes all the code to benchmark the methods against each other, with a function argument to re-compile STAN or not, which can be sourced in each benchmark file within each folder.
 - Nimble cannot save its compilation. It has to recompile each time, unlike STAN.
 - We also want to clean up the benchmark files by sourcing data setup code from each script file instead of copying and pasting it every time into the benchmark file.
 - I am also going to start working on incorporating GRETA, which is R-based and uses Tensorflow to run HMC.
 - We want to test how methods for models we already have will respond to data given that is weird or highly correlated (or like data on a circle from Jerry's PPP).