Methodology:

Introducing Stan and the Bayesian Framework:

To approach the research question at hand, this paper employs a Bayesian framework. As Bayesian analysis is still in the minority to frequentist statistics, this paper will spend a portion of the methodology section explaining the framework, and why it is an improvement on other approaches.

To implement the Bayesian approach we will take advantage of the Stan programing language. Stan is a probabilistic high level programming language that is used to specify statistical models. A Stan program defines a log probability function over parameters conditioned on specified data and constraints. Stan provides the user with the ability to conduct full Bayesian inference through Markov chain Monte Carlo methods such as No-U-Turn sampling. Stan is set up such that densities, gradients, and Hessians, along with intermediate quantities are easily accessible. Prior to Stan it was difficult to accurately compute Bayes rule and apply it to research settings.

Stan utilizes markov-chain monte carlo simulation to get draws from the posterior predicitive distribution. A Markov process is a sequence of random variables with a particular dependence structure where the future is conditionally independent of the past given the present, but noting is marginally independent of anything else. We can construct a Markov process such that the marginal distribution of a random variable is a given target distribution as the number of simulations moves to infinity. As a result, we can get a random draw, or a set of dependent draws, from the target distribution by letting the Markov process to run for many interations.

Stan specifically uses a form of Markov chain Monte Carlo simulation called No-U-Turn Sampling. First, a quick review of ancient Markov chain Monte Carlo samplers. Metropolis-Hasitings only requires user to specify the numerator of Bayes Rule. However, only 22 percent of proposals ideally get accepted to get relatively big jumps in the sampling process. The effective sample size, as a result, can be essentially zero. Gibbs sampling, in general, forces a user to work out all full-conditional distributions. Jumps are always accepted in the sampling process, but they might not be very big. Effective sample size is low if the parameters are highly correlated. Stan, like Metropolis-Hastings, only requires the user to specify the numerator of Bayes Rule. Like Metrolopis-Hastings, but unlike Gibs sampling, proposals are joint. Unlike Metropolis-Hastings, but like Gibbs sampling, proposals are always accepted and tuning of proposals is (often) not required. Unlike both Metropolis-Hastings and Gibbs sampling, the effective sample size is typically 25 percent to 125% of the nominal number of draws from the posterior distribution. As mention, Stan utilizes No U-Turn Sampling (NUTS) as its sampler for the Markov Chain Monte Carlo Simulation. The location of Θ moving according to Hamiltonian physics at any instant would be a valid draw from the posterior distribution. The challenge became then determing when to stop as Θ moves indefinitely (in the absence of friction). Hoffman and Gelman (2014) proposed stopping when there is a U-Turn. In the sense that the footprints found in the monte carlo simulation turn around and start to head in the direction they just came from. After the U-Turn, one footprint is selected with probability proportional to the posterior kernel to be the realization of Θ on interation *s* and the process repeats itself S times. NUTS discretizes a continuous-time Hamiltonian process in order to solve a system of Ordinary Differential Equations (ODEs). These ODEs require a stepsize that is also tuned during the warmup phase of the monte carlo simulation.

Registering priors:

Model Methodology: