STAT 631 Final Project - Bayesian Hierarchical Models

Daniel Yang (10163206)

Introduction

Insofar as the role of a statistician (and their dismal counterparts, the data scientist) goes, it comes with no controversy to claim a primary objective is to analyze data. Data analysis comes in many forms, but a common tool is to use regression. The simplest forms of regression are adequate for most purposes. However, we can improve model inference by considering hierarchical models, when appropriate. Hierarchical models are multilevel models, where the data of interest exhibit a kind of structured dependence with respect to a group. Examples of this nesting include people who are grouped in cities, students within schools, or in longitudinal studies where a person is repeatedly sampled over time. These kinds of dependence form a natural kind of hierarchy, hence the name. These types of models have grown ubiquitous in use given their ability to model more realistically and are computationally feasible by powerful MCMC algorithms. The value of hierarchical models will be shown in the upcoming examples.

Background

There are several reasons why one should consider a hierarchical model for datasets that exhibit a hierarchical structure. Namely, using a single-level model (and therefore treating all covariates) as indepedent tends to under-estimate standard errors and overstate statistical significance[1]. Additionally, the group effects themselves may be of interest for the modeler, for future research. Notationally in a hierarchical model, variates, covariates, and errors all

[1]See University of Bristol, "What are multilevel models and why should I use them?" website in references

have an additional index j specifying which group they belong to. Groups can be made to be specified having their own intercepts or slopes, and each having a mean and variance.

Bayesian hierarchical models do not differ that much in concept to traditional hierarchical models. In Bayesian hierarchical models, the prior distribution parameters themselves are dependent on a parameter(s) (known as hyperparameters) that follow an underlying distribution (known as hyperpriors). For example, if our prior distribution is a Bernoulli distribution with parameter p, and we chose to model p with a Beta distribution with parameters α, β, then our hyperprior is the beta distribution and our hyperparameters are α and β. In essence, the hyperparameters are parameters imposed by the group of which the individual belongs to. Then for a parameter θ and hyperparameter ϕ our (joint) posterior distribution can be rewritten as

$$\begin{aligned} p(\theta, \phi|y) &\propto p(\phi, \theta)p(y|\phi, \theta) \\ &= p(\phi, \theta)p(y|\theta) \end{aligned}$$

Since ϕ only affect y through θ.[2] This can be extended via hyperparameters having their own hyperparameters (known as hyperhyperparameters), and so on.

A Note on R packages Used

The estimates in a Bayesian hierarchical model is obtained implicitly by a MCMC. In particular, the R package 'brms' uses Hamiltonian MC (a variant of the Metropolis algorithm) to obtain estimates in models. From RStan's reference manual, this Monte Carlo "approximates a Hamiltonian dynamics simulation based on numerical integration which is then corrected by performing a Metropolis Acceptance step."[3]

[2] See BDA, page, 107.

[3] RStan's reference manual, 15.1. Hamiltonian Monte Carlo. See references for the manual

Example 1. Eight Schools

Our first Bayesian Hierarchical model is intended to be simple and based off of the relatively famous 'eight schools' paper from Rubin (1981), popularized by Gelman et al. in their textbook, Bayesian Data Analysis. This dataset contains information about a study performed on the effects of similar coaching programs in SAT scores. 8 different schools elected to employ coaching programs, which then reported their estimated average coaching effects and standard errors.

We fit a Bayesian hierarchical model to this dataset. We treat each estimate belonging to their own group, hence we've included random intercepts. This hierarchical model has form

$$y_i \sim N(\theta_i, \sigma_i^2)$$

where the i denotes the group. Furthermore,

$$\theta_i \sim N(\mu, \tau)$$

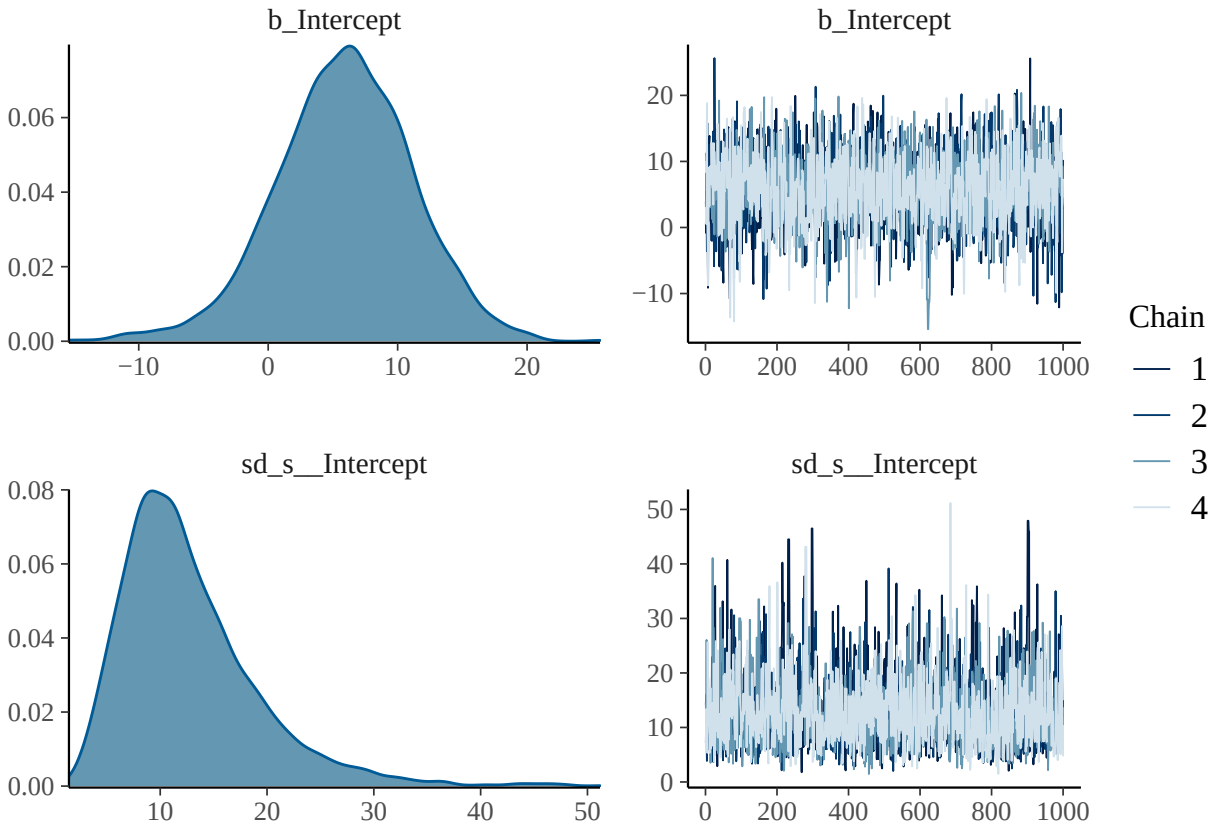
and μ, τ are the hyperparameters of this model. We assign μ a $N(0, 10)$ a normal prior and $\ln \tau \sim N(5, 1)$ a lognormal prior[4]. Below are the results, R code and outputs.

Table 1: Posterior means and percentile intervals for informative priors on μ and τ

Parameter	Posterior Mean (SE)	95% Percentile Interval
μ	5.71 (5.45)	(-5.68, 16.10)
τ	13.10 (6.76)	(4.03, 30.36)
θ_1	14.47 (10.66)	(-3.90, 37.80)
θ_2	7.04 (7.75)	(-8.19, 22.20)
θ_3	2.67 (10.46)	(-19.64, 21.93)
θ_4	6.58 (8.48)	(-10.31, 22.84)
θ_5	1.87 (7.50)	(-13.03, 16.44)
θ_6	3.30 (8.35)	(-14.00, 19.46)
θ_7	12.68 (8.15)	(-2.49, 29.48)
θ_8	7.67 (10.72)	(-13.77, 29.37)

[4]these priors are chosen rather arbitrarily, but are based from an analysis done in a TensorFlow notebook, a Python ML library, to check if estimates obtained were accurate. See references for the Jupyter notebook.

```r
set.seed(1)
schooldata = tibble(y_treat = c(28,8,-3, 7, -1, 1, 18, 12), y_sd = c(15,10,16,11,9,11,10
options(mc.cores = parallel::detectCores())
schoolFit2 = brm(data = schooldata, family = gaussian(), formula = y_treat|se(y_sd) ~ 1
#summary(schoolFit2) #mu, tau
#coef(schoolFit2)$s #theta
plot(schoolFit2)
```



The trace plots for all chains look stationary and resemble the 'fuzzy caterpillar' look. Additionally, the marginal plots of μ and τ are shown. The overall answer to whether or not coaching programs improved SAT scores is statistically significant is suspected to be no, as the credibility intervals for θ_i covers 0 in the model.

Example 2. Epilepsy

While the last example implemented Bayesian hierarchical modeling to fit the data, it does not show explicitly when one should prefer to use it as a framework over single level model. Here, we explicate the advantage and consider another relatively famous dataset from Thall and Vail (1990), the epilepsy dataset. 59 epileptics were randomly assigned to either a treatment or a control group and had their number of seizure counts recorded every 2 weeks, for a total of 8 weeks. This dataset can be found in the R package MASS. Chains, iterations, and burn-in are the same as Example 1. We fit a Zero Inflated Poisson model given the rather large amount of 0 counts. Variables were included based on their significance (i.e., if the credibility interval of those estimates capture 0 or not). The regression model is

$$\Pr\left(y_{ij}=k\right)=\begin{cases}\pi_{ij}+\left(1-\pi_{ij}\right)\exp\left(-\lambda_{ij}\right) & \text{if } k=0\\ \left(1-\pi_{ij}\right)\frac{\lambda_{ij}^{y_{ij}}\exp(-\lambda_{ij})}{y_{ij}!} & \text{if } k>0\end{cases}$$

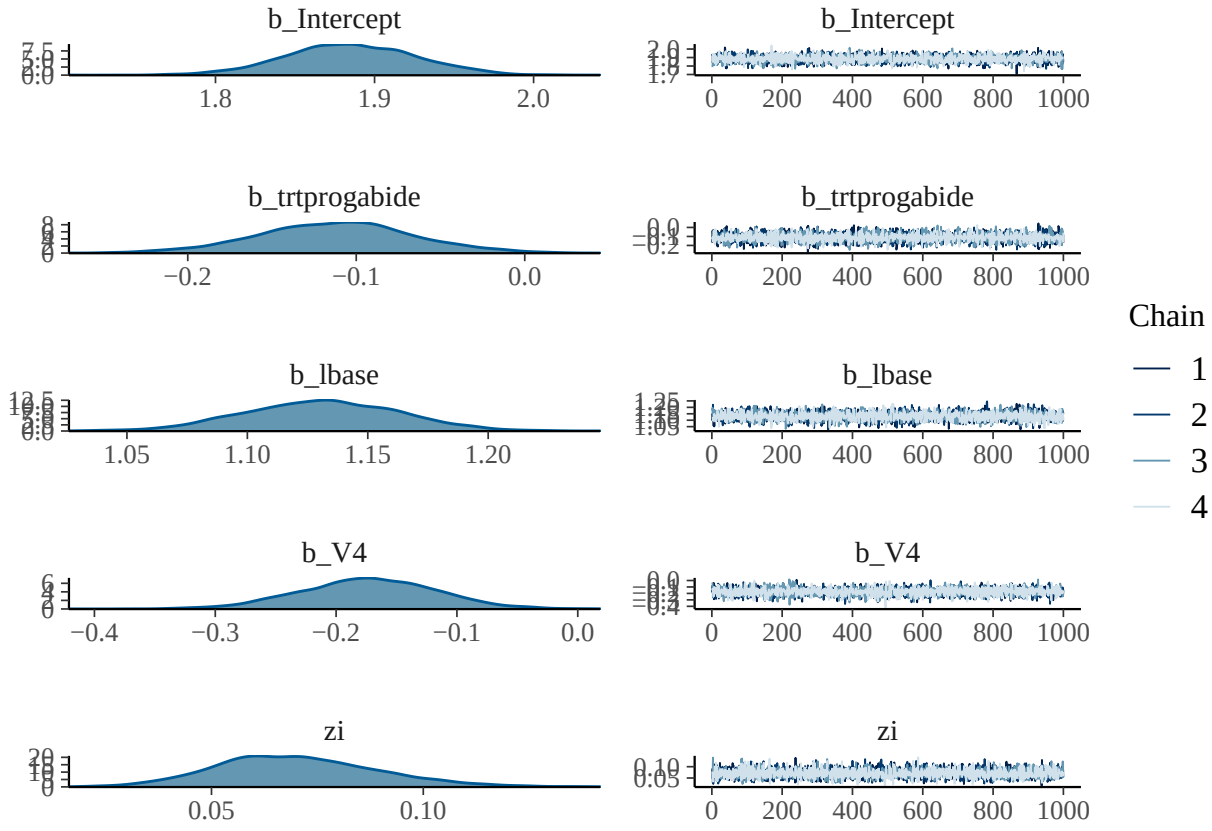
Where $\lambda_{ij}=\exp(\beta_{0j}+\beta_{1j}x_{1ij}\ldots+\beta_{p}x_{pij})$. Below is a non-hierarchical Bayesian model with a Normal(0,5) prior:

Table 2: Posterior means and percentile intervals for single-level model

Parameter	Posterior Mean (SE)	95% Percentile Interval
Intercept	1.88 (0.04)	(1.81, 1.96)
trt	-0.11 (0.05)	(-0.20, -0.02)
lbase	1.13 (0.03)	(1.07, 1.19)
v4	-0.17 (0.05)	(-0.28, -0.07)
π	0.07 (0.02)	(0.04, 0.11)

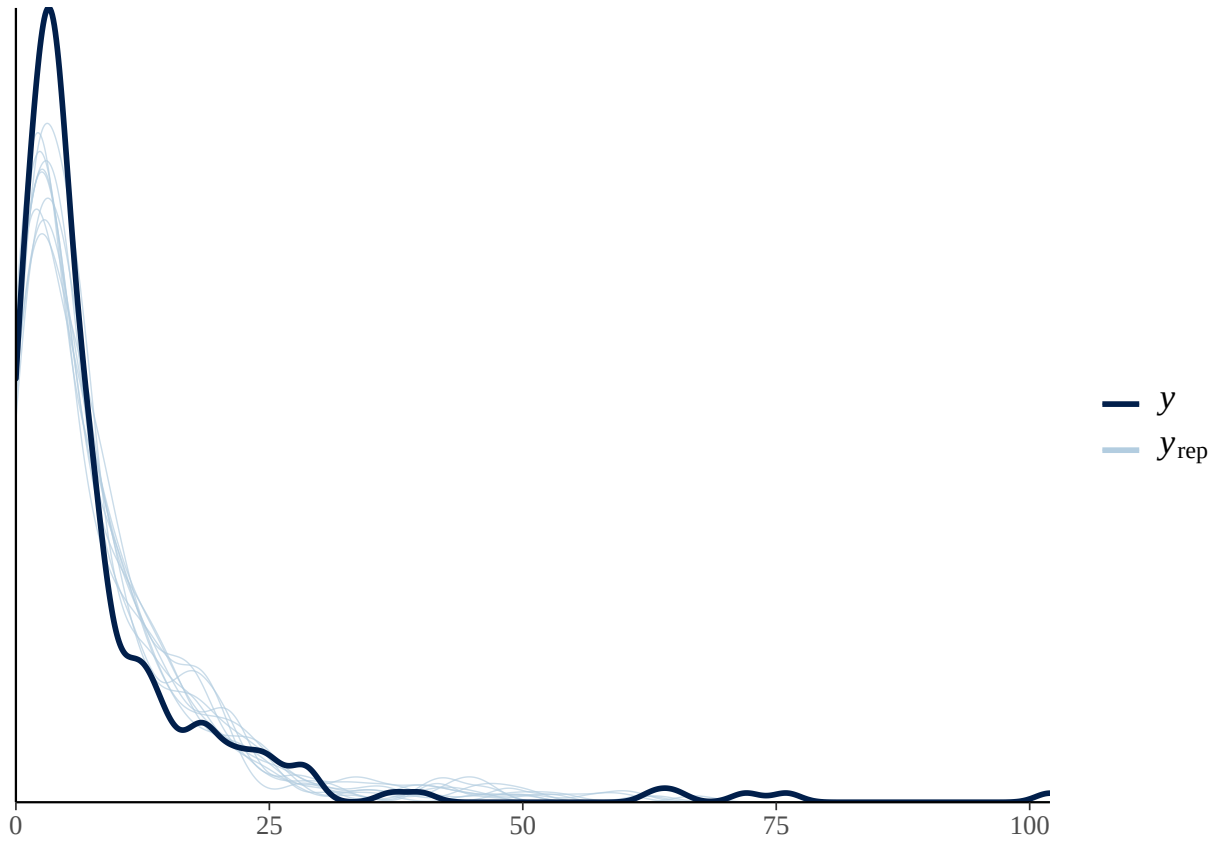
```
set.seed(-1)
data = epil
options(mc.cores = parallel::detectCores())


fitEpil1 = brm(y ~ 1 + trt + lbase + V4, data = data, family = zero_inflated_poisson(),

## Compiling the C++ model

## Start sampling
```

```
#summary(fitEpil1) #obtain estimates
plot(fitEpil1, ask = F)
```




```
pp_check(fitEpil1)
```

```
## Using 10 posterior samples for ppc type 'dens_overlay' by default.
```



Assessing the predictive fit, this model would be adequate for the dataset. However, if we considered the hierarchical model with the subjects being the group, we get a much better predictive fit:

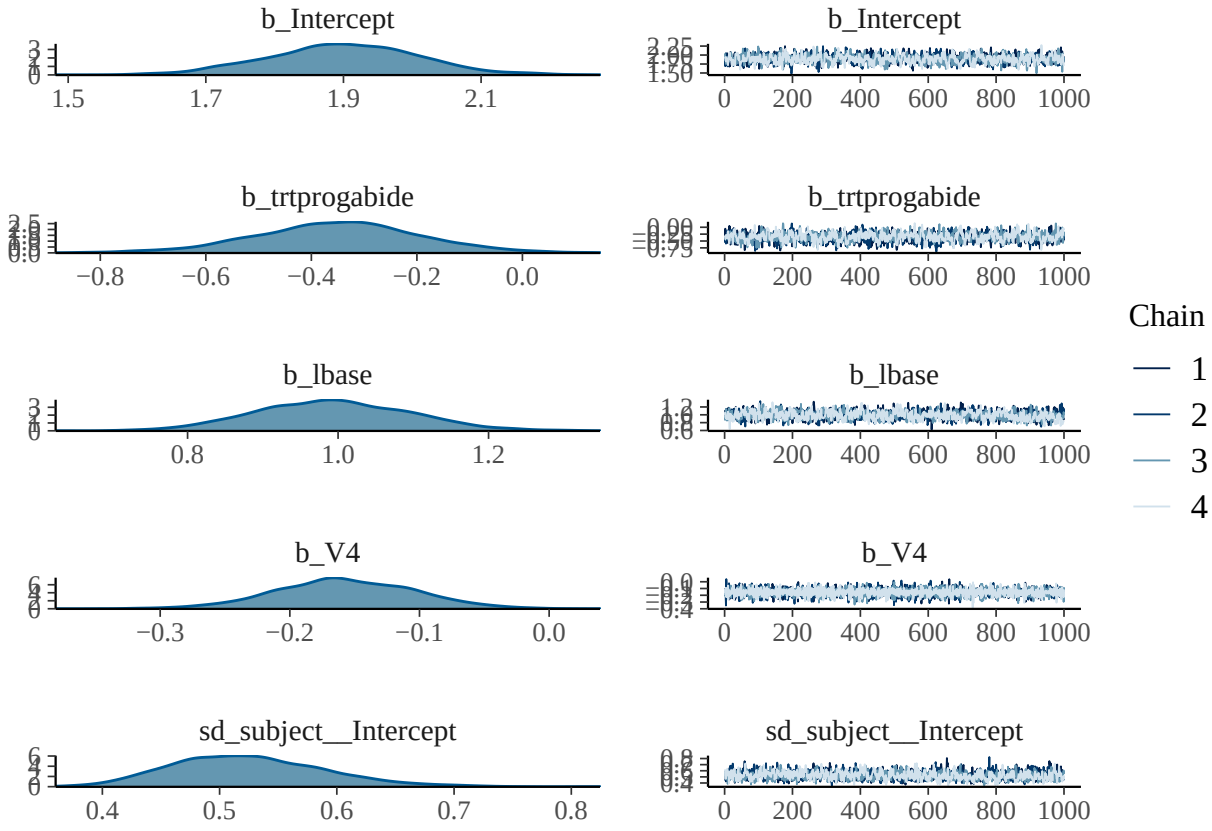
Table 3: Posterior means and percentile intervals for hierarchical model

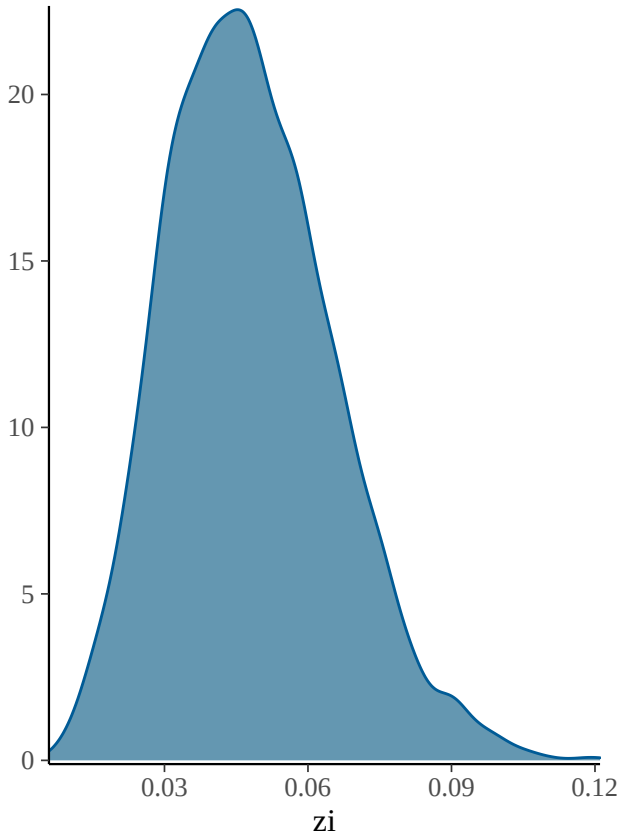
Parameter	Posterior Mean (SE)	95% Percentile Interval
τ	0.52 (0.07)	(0.41, 0.67)
Intercept	1.90 (0.11)	(1.68, 2.10)
trt	-0.36 (0.15)	(-0.66, -0.07)
lbase	1.00 (0.10)	(0.80, 1.19)
v4	-0.16 (0.06)	(-0.26, -0.05)
π	0.05 (0.02)	(0.02, 0.9)

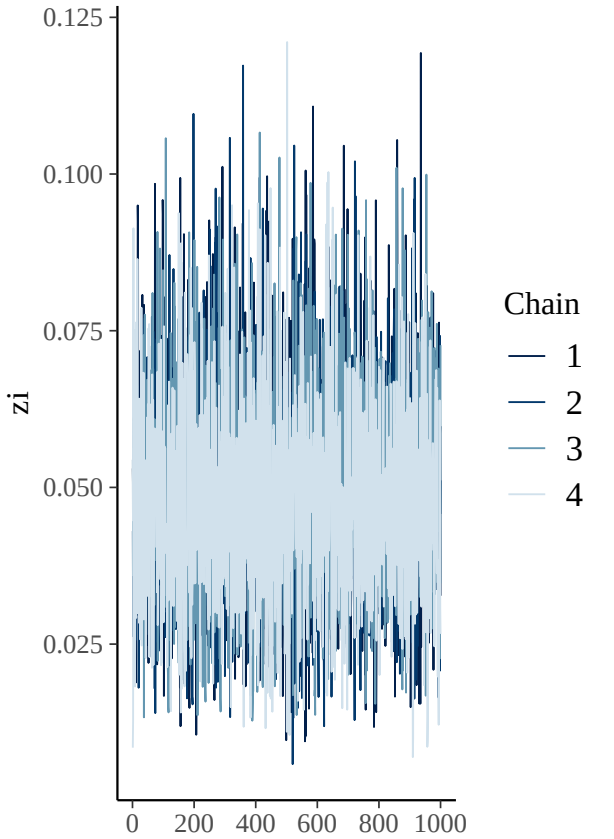
```
set.seed(1)
options(mc.cores = parallel::detectCores())
fitEpil2 = brm(y ~ 1 + trt + lbase + V4 + (1|subject), data = data, family = zero_inflat

## Compiling the C++ model
```

```
## Start sampling
```

```
plot(fitEpil2)
```

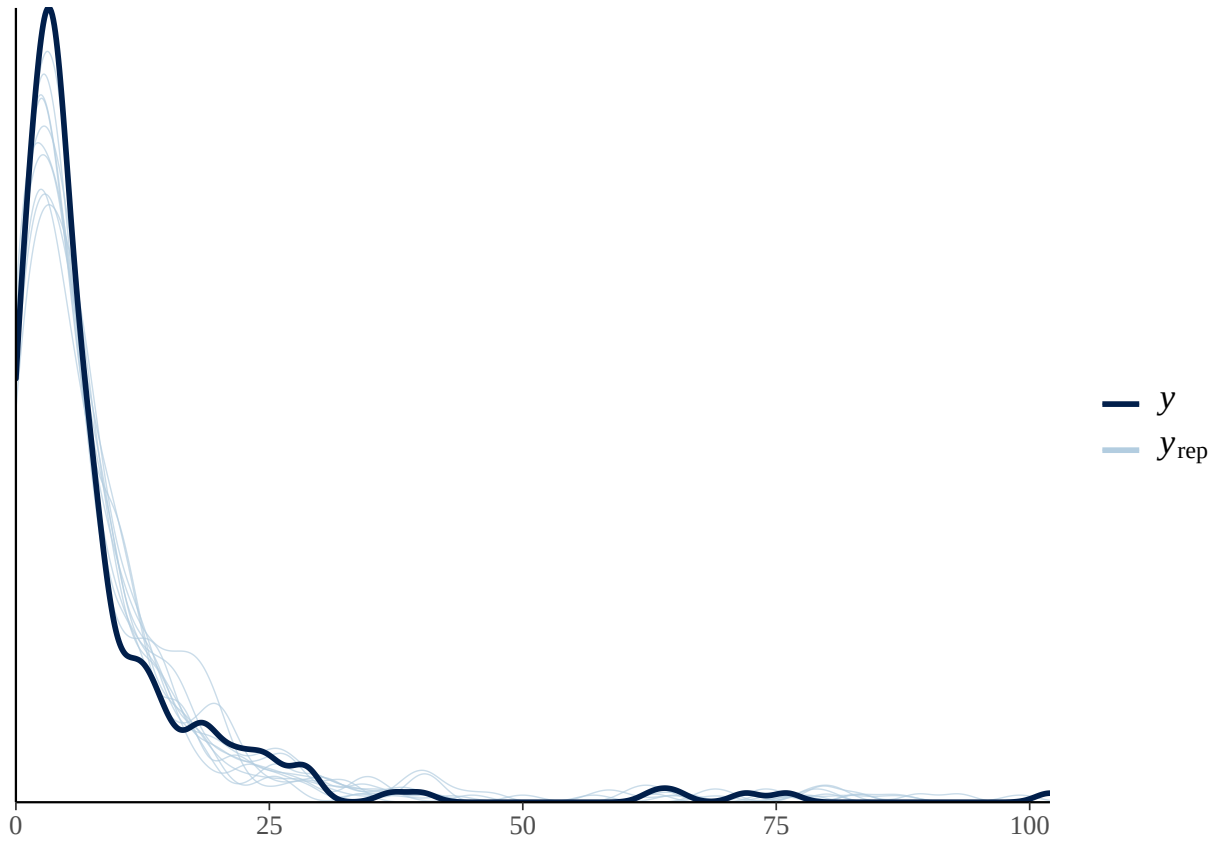







```
#summary(fitEpil2) #estimates
pp_check(fitEpil2)
```

```
## Using 10 posterior samples for ppc type 'dens_overlay' by default.
```



Conclusion

Depending on the nature of the dataset analyzed, it is easy to see how one could make more complicated hierarchical models such as incorporating different kinds of groups, levels beyond two, random slopes and intercepts. Further diagnostics can also be explored graphically and model comparisons can be made. However, even simple models in the present examples can lead to reasonable inference, and the idea illustrated is essentially the same: set up a model, add covariates and groupings, and specify priors. As a result, Bayesian hierarchical models should be considered, whenever hierarchies are involved: simple to begin, and flexible enough to model complex dependencies.

References

Bürkner P (2018). "Advanced Bayesian Multilevel Modeling with the R Package brms." The R Journal, 10(1), 395–411. doi: 10.32614/RJ-2018-017.

Stan Development Team. 2018. Stan Modeling Language Users Guide and Reference Manual, Version 2.18.0. http://mc-stan.org

Guo, J., et al. "rstan: R interface to stan [Computer software manual]." (2015).

Rubin, Donald B. "Estimation in parallel randomized experiments." Journal of Educational Statistics 6.4 (1981): 377-401.

Gelman, Andrew, et al. Bayesian data analysis. CRC press, 2013.

University of Bristol of Bristol. (n.d.). Centre for Multilevel Modelling. Retrieved April 20, 2020, from http://www.bristol.ac.uk/cmm/learning/multilevel-models/what-why.html

Tensorflow. "Tensorflow/Probability." GitHub, 20 Apr. 2020, github.com/tensorflow/probability/blob/master/tensorflow_probability/examples/jupyter_notebooks/Eight_Schools.ipynb.

Thall, Peter F., and Stephen C. Vail. "Some covariance models for longitudinal count data with overdispersion." Biometrics (1990): 657-671.