CGS698C, Lectures 19-20: Bayesian hierarchical models

Himanshu Yadav

2024-07-07

Contents

See chapters 5 of the book "An Introduction to Bayesian Data Analysis for Cognitive Science (`https://vasishth.github.io/bayescogsci/book/`)" for reference.

1 Do observations from human experiements always follow the i.i.d assumption?

Human experiments are typically based on a sample of individuals from the population. **From each individual in the sample, repeated measurements are collected**. For example, in the word recognition experiments, we collect recognition times for multiple words from each participant, i.e., we collect repeated measures from each participant. Do such data follow the i.i.d./ assumption?

This question is important because in real-life datasets, we often see *clusters*. If ten players participate in a disc throwing game, their throw-distance scores would contain clusters. Some players might on average throw at longer distances than others; some players are more consistent across their attempts than others.

Similarly, in the word recognition times, the average recognition time could differ across individuals due to their varying exposure to language, reading habbits, etc. Some individuals may cluster as fast word readers and others as slow word readers.

Do observations with such clusters follow the i.i.d. assumption?

Not necessarily! These observations may not absolutely follow the i.i.d. assumption. However, they would follow a conditional i.i.d. assumption based on the underlying distribution of data.

For example, we cannot say that each word recognition time is independent and comes from a distribution p. But we can say that word recognition times for the participant j come from a distribution p_j such that each p_j depends on a population-level distribution q.

Thus, data with such clusters can be viewed as a sequence of i.i.d. random variables where the order of these random variables does not matter. More formally, such observations follow the **exchangeability assumption**.

2 Exchangeability

A sequence of random variables X_1, X_2, X_3, X_4, ... is said to be *exchangeable* if the joint probability distribution does not change when the order of the sequence $X_1, X_2, X_3, ...$ is altered.

Formally, an exchangeable sequence of random variables is a finite or infinite sequence $X_1, X_2, X_3, ...$ of random variables such that for any finite permutation of the indices $1, 2, 3, ...$, the joint probability distribution of the permuted sequence remains the same.

A sequence of random variables that are i.i.d, conditional on some underlying distributional form, is exchangeable.

How to model the exchangeable random variables?

3 Bayesian Hierarchical models

3.1 Varying intercepts model

Suppose, in an experiment, you are asked to identify a triangle among many other shapes shown on a screen. Your response time is being recorded.

The experimenter hypothesizes that the background color of the screen ("black" or "blue") affects your response time.

We do not have any further information about how exactly the background color affects the response time.

We can assume a linear relationship between the background color and the response time. Such that, **the mean response time changes as a linear function of the background color**.

However, some participants might be in general faster on this shape recognition task than others due to several reasons like their faster motor movements, faster visual processing, etc.

Suppose rt_i is the response time in the i^{th} observation. We can write:

$$rt_i \sim Normal(\mu_i, \sigma) \tag{1}$$

μ_i is the underlying average response time in the i^{th} observation; μ_i can be given by

$$\mu_i = \alpha_{subj[i]} + \beta X_i \tag{2}$$

where X_i is the background color in the i^{th} observation and can take values 0 (for "black") or 1 (for "blue"); $alpha_{subj[i]}$ is the intercept of the straight line for the subject who produced the i^{th}, and β is the slope of the straight line.

Now, the intercept parameter is not a single parameter anymore. We have a separate *alpha* for each subject. Hence, there are as many α as many subjects in data. But all these subject-level intercepts follow a certain distribution. For example, we can say

$$\alpha_{subj[i]} \sim Normal(\alpha, \tau) \tag{3}$$

where α is the population-level intercept and τ is the standard deviation of subject-level intercepts; τ depicts the extent of individual differences in the population.

You can set priors on α, β, σ, and τ.

$$\alpha \sim Normal(300, 50)$$

$$\beta \sim Normal(0, 20)$$

$$\sigma \sim Normal_+(0, 10)$$

$$\tau \sim Normal_+(0, 10)$$

You can estimate the parameters α, β, σ, and τ using **brms**; we are primarily interested in the estimates of β because we want to test the experimenter's hypothesis that said $\beta \neq 0$.

Inferences based on posterior estimates

```
# Data
alpha <- 250
beta <- 20
sigma <- 10
tau <- 15

subj <- rep(1:10,each=10)
X <- rep(0:1,50)
rt <- rep(NA,100)
```

```r
dat <- data.frame(subj=subj,X=X,rt=rt)
alpha_j <- rnorm(10,alpha,tau)
dat$alpha_subj <- rep(alpha_j,each=10)
dat$subj <- factor(dat$subj)
for(i in 1:nrow(dat)){
  dat$rt[i] <- rnorm(1,dat$alpha_subj[i] + beta*X[i],sigma)
}
head(dat)
```

```
##   subj X       rt alpha_subj
## 1    1 0 255.7021   262.2446
## 2    1 1 288.1208   262.2446
## 3    1 0 262.6987   262.2446
## 4    1 1 270.9613   262.2446
## 5    1 0 266.9685   262.2446
## 6    1 1 279.7645   262.2446
```

```r
# Define priors
priors <- c(prior(normal(300, 50), class = Intercept),
            prior(normal(0, 20), class = b, coef=X),
            prior(normal(0, 10), class = sigma),
            prior(normal(0, 10), class = sd))

# Fit the model (estimate parameters)
mfit <-
  brm(formula = rt ~ 1+X + (1|subj),
      data=dat,
      family = gaussian(),
      prior = priors,
      chains = 4,cores = 4,
      iter = 2000,warmup = 1000)

save(mfit,file="FittedModels/Hierarchical-linear-regression.Rda")

summary(mfit)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: rt ~ 1 + X + (1 | subj)
##    Data: dat (Number of observations: 100)
##   Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup draws = 4000
##
## Group-Level Effects:
## ~subj (Number of levels: 10)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
```

```
## sd(Intercept)    13.69      3.06     8.91    20.67 1.01      931     1731
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept   254.21      4.52   245.29   263.45 1.01      727     1114
## X            18.26      1.90    14.46    21.87 1.00     3267     2539
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     9.53      0.72     8.26    11.05 1.00     2510     2555
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

Based on the above posterior estimates, you can say the following:

1. The data are consistent with the experimenter's hypothesis that background color affects the response times because the 95% credible interval for *beta* is completely in the positive direction and does not cross zero.
2. The data suggest that $\beta > 0$, i.e., the mean response time is higher when the background color is blue.

However, you cannot say that there is **evidence** for the experimenter's hypothesis. Because the evidence for any model assumption is always computed with respect to a baseline model assumption. No model is absolutely correct; a model can be relatively better than the other.

All you can say given the above results is that the data are consistent with what the experimenter predicted.

Individual differences in mean response times

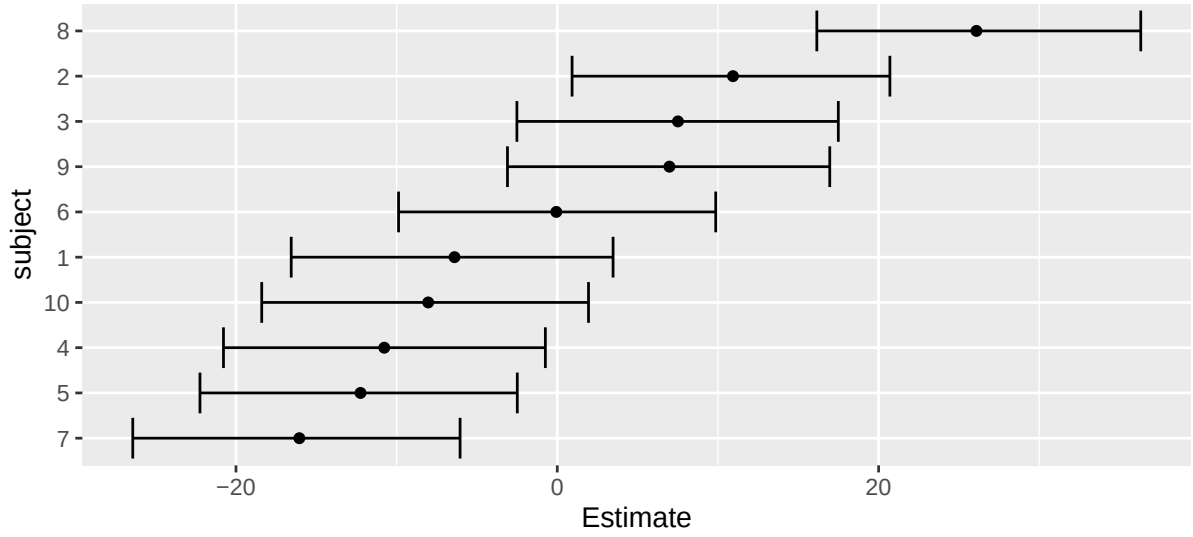
```
df.subj <- data.frame(matrix(nrow=10,ncol=4))
colnames(df.subj) <- c("subj","mean.alpha",
                       "lower.alpha","upper.alpha")
```

```
subj_intercept <-
    paste0("r_subj[",
           as.character(unique(dat$subj)),
           ",Intercept]")

alpha_by_subj <-
  posterior_summary(mfit,
                    variable = subj_intercept) %>%
  as.data.frame() %>%
```

```
  mutate(subject = 1:n()) %>%
  ## reorder plot by magnitude of mean:
  arrange(Estimate) %>%
  mutate(subject = factor(subject,
                          levels = subject))

head(alpha_by_subj)
```

```
##                          Estimate Est.Error       Q2.5     Q97.5 subject
## r_subj[7,Intercept]  -16.05416535  5.180237 -26.423356 -6.049878       7
## r_subj[5,Intercept]  -12.24801474  5.067560 -22.240933 -2.489652       5
## r_subj[4,Intercept]  -10.76883360  5.122033 -20.777500 -0.741585       4
## r_subj[10,Intercept]  -8.03450503  5.093207 -18.396426  1.944337      10
## r_subj[1,Intercept]   -6.39736523  5.110081 -16.561749  3.472756       1
## r_subj[6,Intercept]   -0.06172739  5.067714  -9.879491  9.861899       6
```

```
ggplot(alpha_by_subj,
       aes(x = Estimate,
           xmin = Q2.5, xmax = Q97.5,
           y = subject)) +
  geom_point() +
  geom_errorbarh()
```



3.2 *Correlated varying intercepts varying slopes model*

```
# Data
load("Data/df_pupil.rda")
head(df_pupil)
```

```
## # A tibble: 6 x 4
```

```
##    subj trial  load p_size
##   <int> <int> <int>  <dbl>
## 1   701     1     2  1021.
## 2   701     2     1   951.
## 3   701     3     5  1064.
## 4   701     4     4   913.
## 5   701     5     0   603.
## 6   701     6     3   826.
```

The model

$$p\_size_i \sim Normal(\mu_i, \sigma) \tag{4}$$

$$\mu_i = \alpha_{subj[i]} + \beta_{subj[i]} \cdot load_i \tag{5}$$

$$\begin{pmatrix} \alpha_{subj[i]} \\ \beta_{subj[i]} \end{pmatrix} \sim BivariateNormal\left(\begin{pmatrix} \alpha \\ \beta \end{pmatrix}, \begin{pmatrix} \tau_\alpha^2 & \rho\tau_\alpha\tau_\beta \\ \rho\tau_\alpha\tau_\beta & \tau_\beta^2 \end{pmatrix}\right) \tag{6}$$

$$\alpha \sim Normal(500, 100)$$

$$\beta \sim Normal(0, 25)$$

$$\sigma \sim Normal(0, 50)$$

$$\tau_\alpha \sim Normal(0, 50)$$

$$\tau_\beta \sim Normal(0, 50)$$

$$\rho \sim LKJ(2)$$

```
fit_pupil_full <-
  brm(p_size ~ 1 + load + (1 + load | subj),
    data=df_pupil,
    family=gaussian(),
    prior = c(prior(normal(500,100), class=Intercept),
              prior(normal(0,25),class=b),
              prior(normal(0,50),class=sd),
              prior(normal(0,50),class=sigma),
              prior(lkj(2),class=cor)),
    cores = 4)
save(fit_pupil_full,file="Correlated_intercept_slopes_model_pupil_size.rda")
```