

Applied Bayesian modeling - HW3 (except age at married exercises)

Score: The maximum number of points in this HW is 20 points, with 3 points extra credit. This HW counts for 1.5 HWs, so points are rescaled such that a “full pass” score is 150%. Specifically, for calculating your final HW score, the points will be rescaled to a maximum score of $(20+3)/20 \cdot 150\% = 172.5\%$.

Exercise 1: Fit a Bayesian model using brm and check and interpret the output (4 points)

We continue with IQ data, as introduced in HW2. For this HW, data set `iq_scores.csv` contains 10 iq-scores, sampled from a town in let's say, your favorite country.

Use brm to fit the following model to the IQ data:

$$\begin{aligned} y_i | \theta_i, \sigma^2 &\sim N(\theta_i, \sigma^2) (\text{independent}), \text{ for } i = 1, 2, \dots, n; \\ \mu &\sim N(100, 15^2); \\ \sigma &\sim \text{use the brm-default.} \end{aligned}$$

Then answer the following questions: (i) Plot a histogram of the posterior samples of μ and report a posterior point estimate and 80% CI for μ .

(ii) Plot a histogram of the posterior samples of σ and report a posterior point estimate and 80% CI for σ .

(iii) Can you report a posterior point estimate and 80% CI for μ/σ ? If yes, do so. If not, why not?

Note that the prior for μ can be specified with an additional argument in brm (as illustrated in optional material in module 5), as follows:

```
#mu_prior <- set_prior("normal(100,15)", class = "Intercept")
#fit <- brm(y ~ 1, prior = c(mu_prior),
#           your other usual arguments)
```

Solution

Note that I generated the data as follows:

```
#set.seed(1234567)
#y <- rnorm(10, 112, 13)
#write_csv(tibble(iq_score = y), "../data/iq_scores.csv")
```

Read in data

```
dat <-
  read_csv("../data/iq_scores.csv") %>%
  rename(y = iq_score)
mean(dat$y)
```

```
## [1] 111.0602
```

```
summary(dat$y)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    88.88  102.52  112.96   111.06  120.17   129.86
```

Model fitting

```
mu_prior <- set_prior("normal(100,15)", class = "Intercept")
fit <- brm(y ~ 1, prior = c(mu_prior),
          data = dat,
          seed = 12345, # set seed to make sampling reproducible
          chains = 4, iter = 1000, warmup = 500, cores = getOption("mc.cores", 4))
```

```
summary(fit)
```

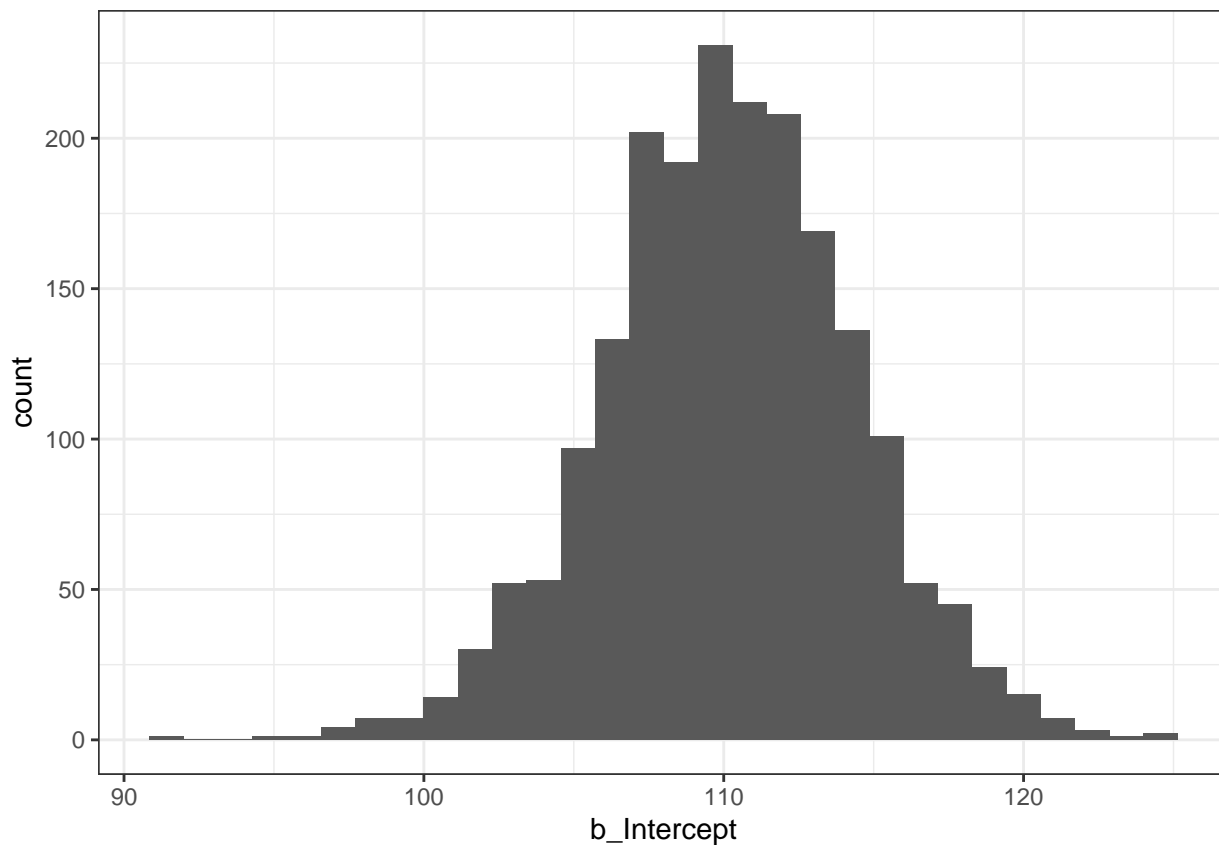
```
## Family: gaussian
## Links: mu = identity; sigma = identity
## Formula: y ~ 1
## Data: dat (Number of observations: 10)
## Draws: 4 chains, each with iter = 1000; warmup = 500; thin = 1;
##        total post-warmup draws = 2000
##
## Population-Level Effects:
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept    110.15      4.18   101.82   118.32 1.00     1052     1069
##
## Family Specific Parameters:
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      14.07      3.39    9.05    22.11 1.00     1171     1039
##
## 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).
```

Samples

```
samples <- as_draws_df(fit)
```

- (i) Plot a histogram of the posterior samples of μ and report a posterior point estimate and 80% CI for μ .

```
samples%>%
  ggplot(aes(x = b_Intercept)) +
  geom_histogram() +
  theme_bw()
```



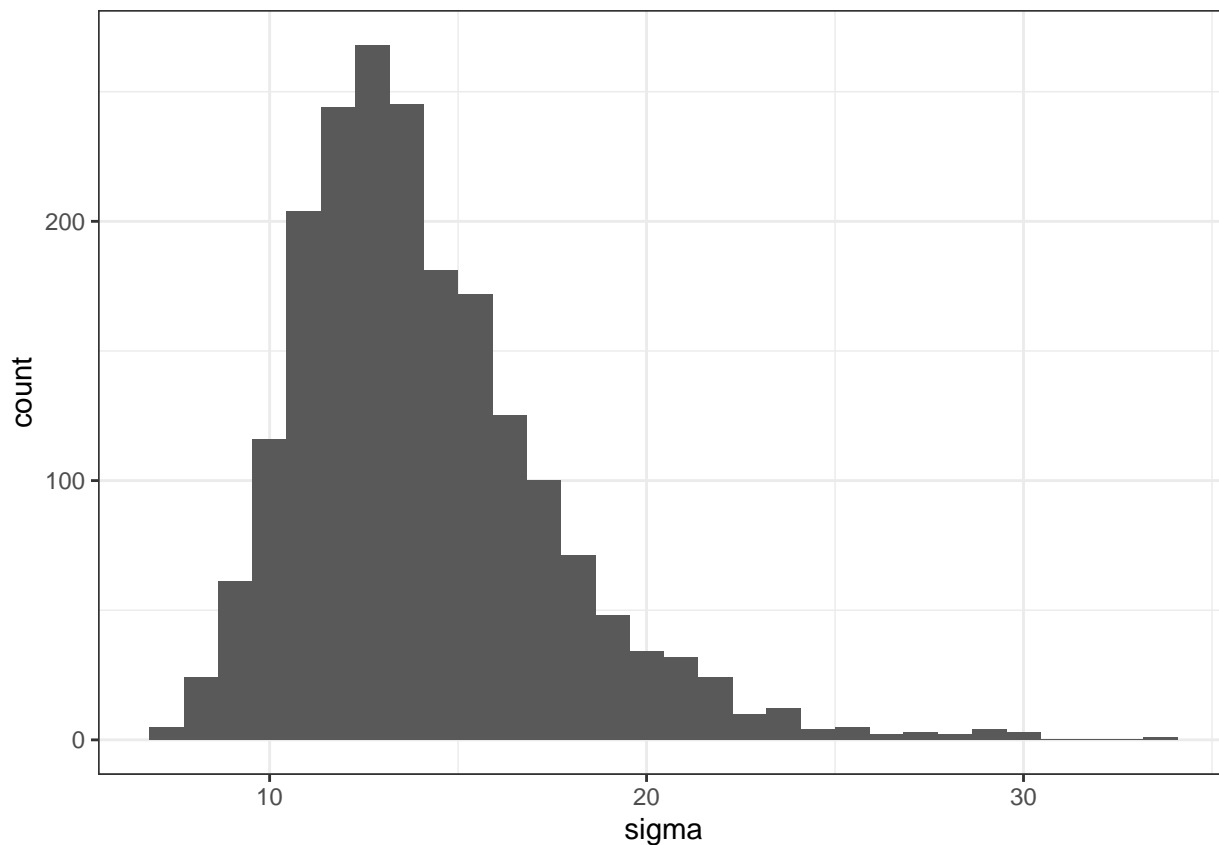
```
samples %>%
  select(b_Intercept) %>%
  point_interval( .point = mean, .width = 0.8)
```

```
## # A tibble: 1 x 6
##   b_Intercept .lower .upper .width .point .interval
##         <dbl> <dbl> <dbl> <dbl> <chr>  <chr>
## 1         110.   105.   115.   0.8 mean    qi
```

Point estimate is 110, with 80% CI (105, 115).

(ii) Plot a histogram of the posterior samples of σ and report a posterior point estimate and 80% CI for σ .

```
samples%>%
  ggplot(aes(x = sigma)) +
  geom_histogram() +
  theme_bw()
```



```
samples %>%
  select(sigma) %>%
  point_interval( .point = mean, .width = 0.8)
```

```
## # A tibble: 1 x 6
##   sigma .lower .upper .width .point .interval
##   <dbl> <dbl> <dbl> <dbl> <chr> <chr>
## 1  14.1  10.4  18.4   0.8 mean  qi
```

Point estimate is 14.1, 80% CI is (10.4, 18.4).

(iii) Can you report a posterior point estimate and 80% CI for μ/σ ? If yes, do so. If not, why not?

Yes we can, we can use the samples to create $\mu^{(s)}/\sigma^{(s)}$:

```
samples %>%
  mutate(muoversigma = b_Intercept/sigma) %>%
  select(muoversigma) %>%
  point_interval( .point = mean, .width = 0.8)
```

```
## # A tibble: 1 x 6
##   muoversigma .lower .upper .width .point .interval
##           <dbl> <dbl> <dbl> <dbl> <chr> <chr>
## 1         8.25  5.91  10.7   0.8 mean  qi
```

Point estimate given by 8.2, 80% CI (5.7, 10.7).

Exercise 2: Compare and contrast the MCMC diagnostics of two different model fits

Continue with the IQ data from Q1 (with y = IQ scores) to fit the model as specified below. Present and briefly summarize resulting MCMC diagnostics (traceplots, Rhat, effective sample sizes) and comment on whether this model fit can be used for summarizing information regarding μ . If not, why not?

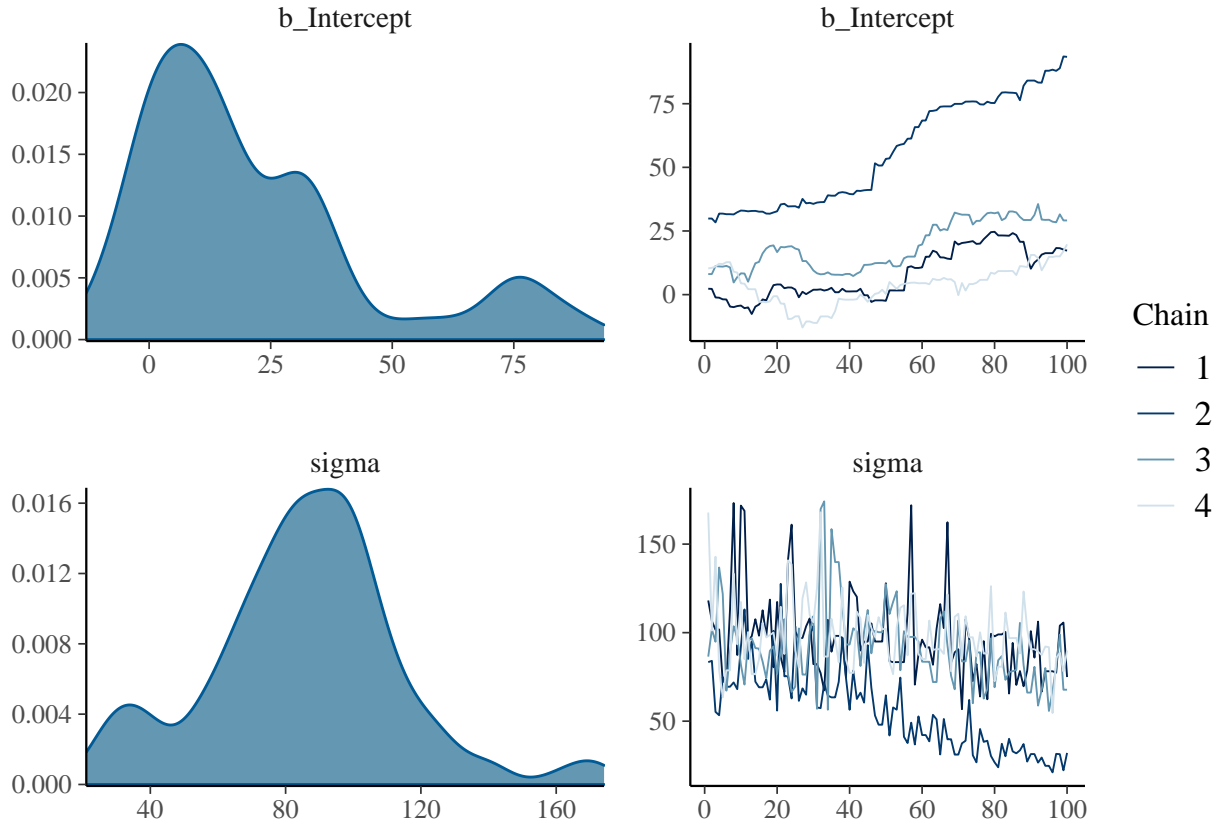
```
fit_bad <- brm(y ~ 1, data = dat,
  chains = 4, iter = 200, cores = getOption("mc.cores", 4),
  control = list(adapt_delta = 0.6, max_treedepth = 4),
  seed = 12,
  # these are NOT recommended options, trying to create problems here!
)
```

```
summary(fit_bad)
```

```
## Family: gaussian
## Links: mu = identity; sigma = identity
## Formula: y ~ 1
## Data: dat (Number of observations: 10)
## Draws: 4 chains, each with iter = 200; warmup = 100; thin = 1;
## total post-warmup draws = 400
##
## Population-Level Effects:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept    21.62     24.42   -8.56   83.35 2.72         5        12
##
## Family Specific Parameters:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma     85.73     28.45   30.16  158.55 1.45         8        12
##
## 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).
```

Rhat is 2.7 and 1.45, much larger than 1, suggesting that chains have not mixed well. Effective samples range from 5 to 12, much lower than the desired ~100 effective samples per chain, suggesting high autocorrelation within chains.

```
plot(fit_bad, variable = c("b_Intercept", "sigma"))
```



Traceplots confirm what we saw in the diagnostics: chains do not mix well and may not have converged away from the initial values to the target distribution.

The samples obtained from this model fit cannot be used for inference about μ because diagnostics show that there are issues with the sampling: the chains may not yet have converged away from the initial values to the target distribution and effective sample size is estimated to be low.

Exercise 5 (extra credit, 3 points)

Obtain the joint distribution of $(y_1, y_2) | \mu_\alpha, \sigma_\alpha^2, \sigma_y^2$ in the setting where $j[1] \neq j[2]$ and in the setting where $j[1] = j[2]$.

Hints:

- First consider the univariate distribution of $y_i | \mu_\alpha, \sigma_\alpha^2, \sigma_y^2$.
- You may find it helpful to write y_i as the sum of parameters (thus random variables) μ_α , a parameter to capture the variability across counties, and a parameter to capture variability across observations.

Solution

We can write $y_i = \mu_\alpha + \eta_{j[i]} + \varepsilon_i$, where $\varepsilon_i | \sigma_y^2 \sim N(0, \sigma_y^2)$ captures the variability across observations. For the univariate distribution, we get that $y_i | \mu_\alpha, \sigma_\alpha^2, \sigma_y^2$ is the sum of normal random variables, hence normally distributed. The conditional mean is given by

$$E(y_i | \mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = \mu_\alpha + 0 + 0 = \mu_\alpha.$$

The variance is given by (using independence between the 3 parts): $Var(y_i|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = Var(\mu_\alpha|\mu_\alpha) + Var(\eta_{j[i]}|\sigma_\alpha) + Var(\varepsilon_i|\sigma_y) = 0 + \sigma_\alpha^2 + \sigma_y^2$. We get

$$y_i|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2 \sim N(\mu_\alpha, \sigma_\alpha^2 + \sigma_y^2).$$

To get the joint density, we first note that this is a normal distribution as well, given the univariate distributions are normal. The covariance is obtained as follows: $Cov(y_1, y_2|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = Cov(\mu_\alpha + \eta_{j[1]} + \varepsilon_1, \mu_\alpha + \eta_{j[2]} + \varepsilon_2|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = Cov(\eta_{j[1]}, \eta_{j[2]}|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2)$. In the last step, we removed μ_α (because it's being conditioned on) and used that the η s are independent of the ε 's, and the ε s are independent of each other.

For $j[1] \neq j[2]$, $\eta_{j[1]}$ and $\eta_{j[2]}$ are independent, and we get that $Cov(\eta_{j[1]}, \eta_{j[2]}|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = 0$. For $j[1] = j[2]$, we get $Cov(\eta_{j[1]}, \eta_{j[2]}|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = Cov(\eta_{j[1]}, \eta_{j[1]}|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2) = \sigma_\alpha^2$.

Combining all of the above, we get

$$(y_1, y_2)|\mu_\alpha, \sigma_\alpha^2, \sigma_y^2 \sim N((\mu_\alpha, \mu_\alpha), \mathbf{\Sigma}),$$

where variance-covariance matrix $\mathbf{\Sigma}$ is given by $\Sigma_{11} = \Sigma_{22} = \sigma_\alpha^2 + \sigma_y^2$ and $\Sigma_{12} = \Sigma_{21} = \sigma_\alpha^2$ for $j[1] = j[2]$ and 0 otherwise.