Bayesian Thinking: Fundamentals, Regression and Multilevel Modeling

Jim Albert and Monika Hu

1/9/2023

Webinar 1-3: Regression Models for Continuous Data

1. Introduction: adding a continuous predictor variable
2. A simple linear regression for the CE sample
3. A multiple linear regression for the CE sample
4. Wrap-up and additional material

Section 1

Introduction: adding a continuous predictor variable

Review: the normal model

- When you have continuous outcomes, you can use a normal model:

$$Y_i \mid \mu, \sigma \overset{i.i.d.}{\sim} \textrm{Normal}(\mu, \sigma), \;\; i = 1, \cdots, n. \tag{1}$$

- Suppose now you have another continuous variable available, x_i. And you want to use the information in x_i to learn about Y_i.
  1. Y_i is the log of expenditure of CU's
  2. x_i is the log of total income of CU's
- Is the model in Equation (1) flexible to include x_i?

An observation specific mean

- We can adjust the model in Equation (1) to Equation (2), where the common mean μ is replaced by an observation specific mean μ_i:

$$Y_i \mid \mu_i, \sigma \overset{ind}{\sim} \text{Normal}(\mu_i, \sigma), \;\; i = 1, \cdots, n. \tag{2}$$

- How to link μ_i and x_i?

Linear relationship between the mean and the predictor

- One basic approach: use a linear relationship:

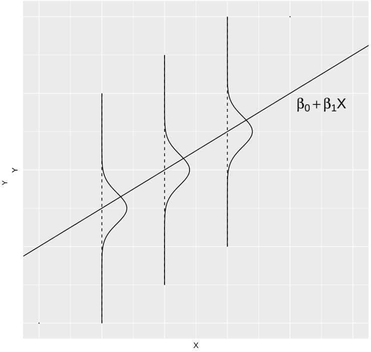
$$\mu_i = \beta_0 + \beta_1 x_i, \;\; i = 1, \cdots, n. \tag{3}$$

- x_i's are known constants.
- β_0 (intercept) and β_1 (slope) are unknown parameters.
- Bayesian approach:
  1. assign a prior distribution to $(\beta_0, \beta_1, \sigma)$
  2. perform inference
  3. summarize posterior distribution of these parameters

The simple linear regression model

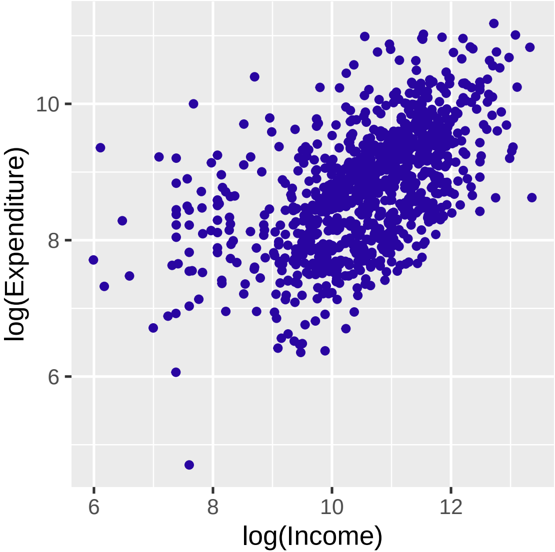
- To put everything together, a linear regression model:

$$Y_i \mid x_i, \beta_0, \beta_1, \sigma \overset{ind}{\sim} \text{Normal}(\beta_0 + \beta_1 x_i, \sigma), \;\; i = 1, \cdots, n. \tag{4}$$



#

The simple linear regression model cont'd



Section 2

A simple linear regression for the CE sample

The CE sample

The CE sample comes from the 2017 Q1 CE PUMD: 4 variables, 994 observations.

Variable	Description
log(Expenditure)	Continuous; CU's total expenditures in last quarter (log)
log(Income)	Continuous; the amount of CU income before taxes in past 12 months (log)
Rural	Binary; the urban/rural status of CU: 0 = Urban, 1 = Rural
Race	Categorical; the race category of the reference person: 1 = White, 2 = Black, 3 = Native American, 4 = Asian, 5 = Pacific Islander, 6 = Multi-race

An SLR for the CE sample

- For now, we focus on a simple linear regression:

$$
\begin{aligned}
Y_i \mid \mu_i, \sigma &\overset{ind}{\sim} \text{Normal}(\mu_i, \sigma), \quad (5) \\
\mu_i &= \beta_0 + \beta_1 x_i. \quad (6)
\end{aligned}
$$

Variable	Description
log(Expenditure)	Continuous; CU's total expenditures in last quarter (log)
log(Income)	Continuous; the amount of CU income before taxes in past 12 months (log)

A weakly informative prior

- Assume know little about $(\beta_0, \beta_1, \sigma)$.
- Assuming independence: $g(\beta_0, \beta_1, \sigma) = g(\beta_0)g(\beta_1)g(\sigma)$.
- For example:

$$
\begin{aligned}
\beta_0 &\sim \text{Normal}(0, 10), \\
\beta_1 &\sim \text{Normal}(0, 10), \\
\sigma &\sim \text{Cauchy}(0, 1).
\end{aligned}
$$

Fitting the model

- Use the `brm()` function with `family = gaussian`.

```
library(brms)
SLR_fit <- brm(data = CEData, family = gaussian,
               log_TotalExp ~ 1 + log_TotalIncome,
               prior = c(prior(normal(0, 10), class = Intercept),
                         prior(normal(0, 10), class = b),
                         prior(cauchy(0, 1), class = sigma)),
               iter = 10000, warmup = 8000, chains = 2, seed = 123)
```

Saving posterior draws

- Save post as a matrix of simulated posterior draws

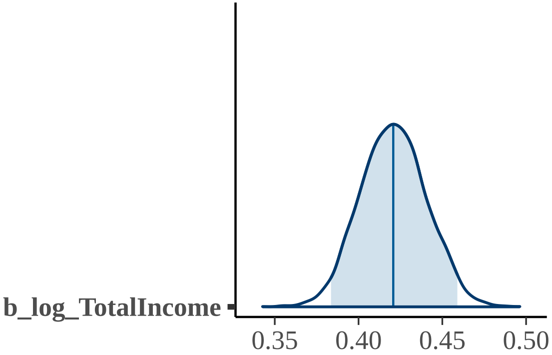
```
post <- as_draws_df(SLR_fit)
head(post)
```

```
# A draws_df: 6 iterations, 1 chains, and 5 variables
  b_Intercept b_log_TotalIncome sigma lprior  lp__
1         4.1              0.44  0.71   -7.7 -1097
2         4.0              0.45  0.70   -7.7 -1099
3         3.9              0.46  0.73   -7.7 -1099
4         4.0              0.45  0.72   -7.7 -1098
5         4.1              0.44  0.72   -7.7 -1097
6         4.3              0.43  0.72   -7.7 -1096
# ... hidden reserved variables {'.chain', '.iteration', '.draw'}
```

Posterior plots

- Function `mcmc_areas()` displays a density estimate of the simulated posterior draws with a specified credible interval.

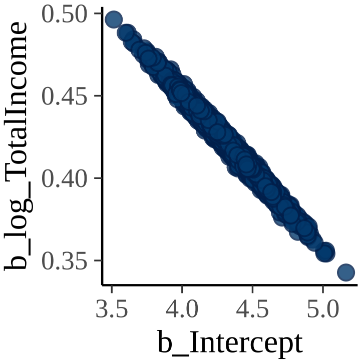
```
library(bayesplot)
mcmc_areas(post, pars = "b_log_TotalIncome", prob = 0.95)
```



Posterior plots cont'd

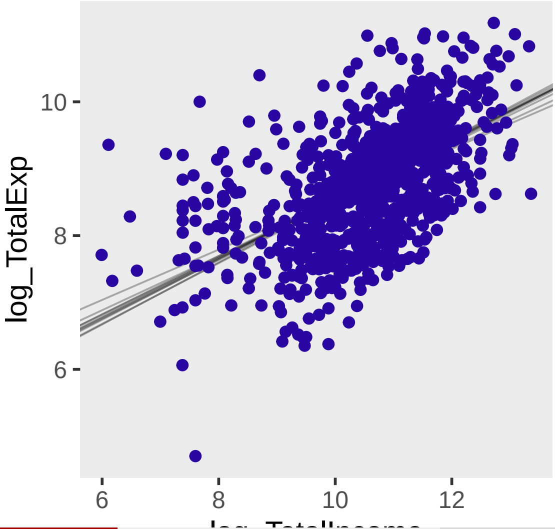
- Function `mcmc_scatter()` creates a simple scatterplot of two parameters.

```
mcmc_scatter(post, pars = c("b_Intercept", "b_log_TotalIncome"))
```



Plotting posterior inference against the data

- Plot the first 10 (β_0, β_1) fits to the data



Predictions

- Use the `predict()` function to make predictions of observed CUs.

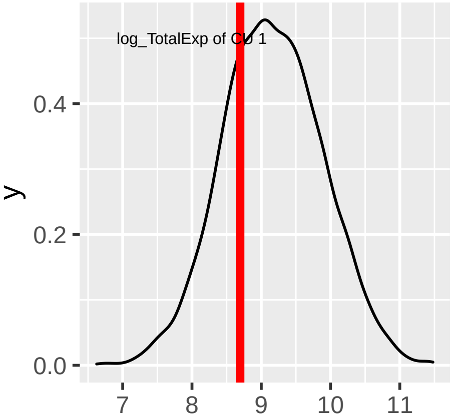
```
pred_logExp_obs <- predict(SLR_fit, newdata = CEData)
head(pred_logExp_obs)
```

```
     Estimate Est.Error     Q2.5     Q97.5
[1,] 9.157307 0.7293884 7.743278 10.542928
[2,] 8.571478 0.7240150 7.163445  9.953661
[3,] 9.090246 0.7239522 7.653539 10.470947
[4,] 9.343411 0.7261226 7.870521 10.762417
[5,] 9.283753 0.7234239 7.877661 10.703081
[6,] 8.706142 0.7286841 7.294871 10.147069
```

Predictions cont'd

- If we focus on one CU, i.e.g CU 1; set `summary = FALSE` to obtain predicted values.

```
pred_logExp_obs_1 <- predict(SLR_fit, newdata = CEData[1, ],
                             summary = FALSE)
```



Predictions cont'd

- Now suppose we get to know a new CU with log_TotalIncome = 10, and we want to predict its log_TotalExp

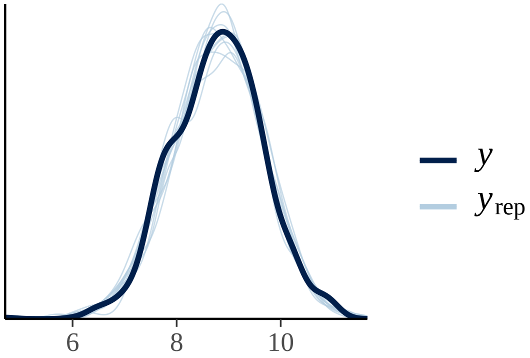
```
newdata <- data.frame(log_TotalIncome = c(10))
pred_logExp_new <- predict(SLR_fit, newdata = newdata)
pred_logExp_new
```

```
     Estimate Est.Error     Q2.5    Q97.5
[1,] 8.513021 0.7260283 7.109544 9.950892
```

Model checking

- Function pp_check() performs posterior predictive checks
  - plot density estimates for 10 replicated samples from the posterior predictive distribution and overlay the observed log income distribution

```
pp_check(SLR_fit)
```



Section 3

A multiple linear regression for the CE sample

Adding a binary predictor

Variable	Description
log(Expenditure)	Continuous; CU's total expenditures in last quarter (log)
Rural	Binary; the urban/rural status of CU: 0 = Urban, 1 = Rural

- Consider Rural as a binary categorical variable to classify two groups:
  - The urban group
  - The rural group
- Such classification puts an emphasis on the difference of the expected outcomes between the two groups.

With only one binary predictor

- For simplicity, consider a simplified regression model with a single predictor: the binary indicator for rural area x_i.

$$\mu_i = \beta_0 + \beta_1 x_i = \begin{cases} \beta_0, & \text{the urban group;} \\ \beta_0 + \beta_1, & \text{the rural group.} \end{cases} \tag{7}$$

- The expected outcome μ_i for CUs in the urban group: β_0.
- The expected outcome μ_i for CUs in the rural group: $\beta_0 + \beta_1$.
- β_1 represents the change in the expected outcome μ_i from the urban group to the rural group.

The multiple linear regression model

$$
\begin{aligned}
Y_i \mid \mu_i, \sigma &\overset{ind}{\sim} \textrm{Normal}(\mu_i, \sigma), && (8)\\
\mu_i &= \beta_0 + \beta_1 x_{i,logIncome} + \beta_2 x_{i,Rural}. && (9)
\end{aligned}
$$

A weakly informative prior

- Assume know little about $(\beta_0, \beta_1, \beta_2, \sigma)$.

$$
\begin{aligned}
\beta_0 &\sim \text{Normal}(0, 10), \\
\beta_1 &\sim \text{Normal}(0, 10), \\
\beta_2 &\sim \text{Normal}(0, 10), \\
\sigma &\sim \text{Cauchy}(0, 1).
\end{aligned}
$$

Fitting the model

- Use the `brm()` function with `family = gaussian`.
- Use `as.factor()` for binary / categorical predictors.

```
MLR_fit <- brm(data = CEData, family = gaussian,
               log_TotalExp ~ 1 + log_TotalIncome + as.factor(Rural),
               prior = c(prior(normal(0, 10), class = Intercept),
                         prior(normal(0, 10), class = b),
                         prior(cauchy(0, 1), class = sigma)),
               iter = 10000, warmup = 8000, chains = 2, seed = 123)
```

Saving posterior draws

- Save post as a matrix of simulated posterior draws

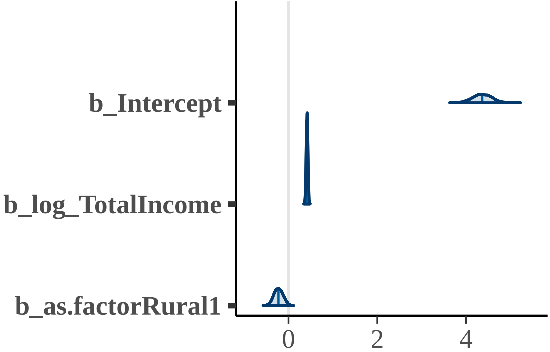
```
post_MLR <- as_draws_df(MLR_fit)
head(post_MLR)
```

```
# A draws_df: 6 iterations, 1 chains, and 6 variables
  b_Intercept b_log_TotalIncome b_as.factorRural1 sigma lprior  lp__
1         4.6              0.40             -0.25  0.73    -11 -1097
2         4.3              0.42             -0.27  0.71    -11 -1097
3         4.6              0.40             -0.22  0.72    -11 -1099
4         4.5              0.41             -0.22  0.74    -11 -1098
5         4.1              0.44             -0.19  0.73    -11 -1098
6         4.6              0.40             -0.28  0.72    -11 -1097
# ... hidden reserved variables {'.chain', '.iteration', '.draw'}
```

Posterior plots

- Function mcmc_areas() displays a density estimate of the simulated posterior draws with a specified credible interval.

```
mcmc_areas(post_MLR,
           pars = c("b_Intercept", "b_log_TotalIncome",
                    "b_as.factorRural1"),
           prob = 0.95)
```



Predictions

- Use the `predict()` function to make predictions of observed CUs.

```
pred_logExp_obs <- predict(MLR_fit, newdata = CEData)
head(pred_logExp_obs)
```

```
     Estimate Est.Error     Q2.5    Q97.5
[1,] 9.163233 0.7177108 7.770345 10.55731
[2,] 8.587449 0.7184731 7.175813 10.02002
[3,] 9.080891 0.7308250 7.667061 10.49721
[4,] 9.344484 0.7316361 7.925087 10.77281
[5,] 9.279236 0.7121874 7.884744 10.70832
[6,] 8.707668 0.7280782 7.297315 10.13008
```

Predictions cont'd

- Now suppose we get to know two new CU with log_TotalExp = 10, one is rural and the other is urban, and we want to predict its log_TotalIncome.
- Can also use the `posterior_predict()` function.

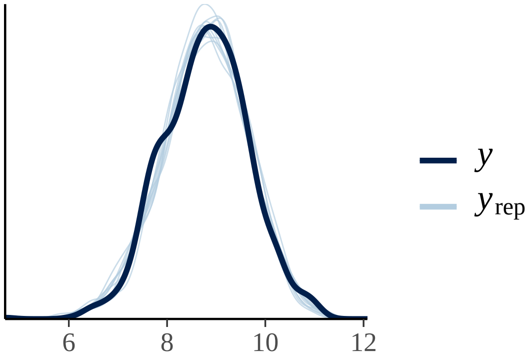
```
newdata <- data.frame(log_TotalIncome = c(10, 10), Rural = c(1, 0))
pred_logExp_new <- posterior_predict(MLR_fit, newdata = newdata)
apply(pred_logExp_new, 2, summary)
```

```
             [,1]      [,2]
Min.     5.259465  6.232784
1st Qu.  7.839439  8.078532
Median   8.349848  8.548410
Mean     8.332961  8.551773
3rd Qu.  8.836337  9.043449
Max.    10.804520 10.838913
```

Model checking

- Function `pp_check()` plots density estimates for 10 replicated samples from the posterior predictive distribution and overlay the observed log income distribution.

```
pp_check(MLR_fit)
```



Section 4

Wrap-up and additional material

Wrap-up

- Bayesian linear regression:
    - Linear relationship between the expected outcome and the predictor(s)
    - Continuous predictors, binary predictors
    - Using the `brms` package; prior choices
- Bayesian inferences
    - Bayesian hypothesis testing and credible interval
    - Bayesian prediction
    - Posterior predictive checks

Additional material: adding a categorical predictor

Variable	Description
log(Expenditure)	Continuous; CU's total expenditures in last quarter (log)
Race	Categorical; the race category of the reference person: 1 = White, 2 = Black, 3 = Native American, 4 = Asian, 5 = Pacific Islander, 6 = Multi-race

- It is common to consider it as a categorical variable to classify multiple groups:
  - How many groups? What are the groups?
- Such classification puts an emphasis on the difference of the expected outcomes between one group to the reference group.

With only one categorical predictor

- For simplicity, consider a simplified regression model with a single predictor: the race category of the reference person x_i.

$$\begin{aligned} \mu_i &= \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_4 x_{i,4} + \beta_5 x_{i,5} \\ &= \begin{cases} \beta_0, & \text{White;} \\ \beta_0 + \beta_1, & \text{Black;} \\ \beta_0 + \beta_2, & \text{Native American;} \\ \beta_0 + \beta_3, & \text{Asian;} \\ \beta_0 + \beta_4, & \text{Pacific Islander;} \\ \beta_0 + \beta_5, & \text{Multi-race.} \end{cases} \end{aligned} \tag{10}$$

- What is the expected outcome μ_i for CUs in the White group?
- What is the expected outcome μ_i for CUs in the Asian group?