A visualization of random effects model and fixed effects model

Miao Cai

2019-07-26

1 Random intercept model

1.1 Statistical model

In this section, we will set up the statistical model for a random-intercept model:

$$\begin{aligned} & y_{ij} = \beta_{0j} + \beta_1 * x_{1i} + \beta_2 * x_{2i} + \beta_3 * x_{3i} + \epsilon_i \\ & \beta_{0j} \sim N(\mu_0, \sigma_0^2) \end{aligned}$$

Where $i = 1, 2, \cdots, N$ is the index for individual observations, so the total sample size is N in this case. $j = 1, 2, \cdots, J$ is the index for groups, so there are J groups/clusters in this case. $\beta_{01}, \beta_{02}, \cdots, \beta_{0J}$ are the random intercepts in this model. $\beta_1, \beta_2, \cdots, \beta_K$ are the fixed-effects parameters, and in this case, $K = 3$. The number of predictor variables can vary, and this data generating process can be generalized using a more succinct matrix form style as:

$$\begin{aligned} & y_{ij} = \beta_{0j} + \mathbf{X}\beta + \epsilon_i \\ & \beta_{0j} \sim N(\mu_0, \sigma_0^2) \end{aligned}$$

where $\mathbf{X}$ is a $N \times K$ data matrix, and β is a $K \times 1$ column vector.

We assume that the random intercepts β_{0j} has a normal distribution with two hyperparameters: mean μ_0 and standard deviation σ_0.

1.2 Simulation parameters

1.2.1 Observations and groups

- The number of groups/clusters is $J = 10$,
- The number of observations per group is generated from a homogeneous Poisson process with the mean parameter of $\lambda = 40$.

1.2.2 Model parameters

- Hyperparameters: $\mu = 1, \sigma = 1$,
- Number of fixed parameters: $K = 3$,
- Fixed-effects parameter vector: $\beta = [1, -1, 0.5]'$
- Error term: $\epsilon_i \sim \text{Normal}(0, 5)$
- Data matrix:
  - $x_1 \sim \text{Normal}(2, 3)$,
  - $x_2 \sim \text{Poisson}(3)$,
  - $x_3 \sim \text{Gamma}(5, 5)$

1.3 Data simulation

```
pacman::p_load(lme4, broom, dplyr)
set.seed(666)

# Obs and groups
J = 10 # of the groups
lambda = 40 # mean number of obs per group
N_j = rpois(J, lambda) # number of obs per group
N = sum(N_j) # total number of obs

# Model parameters
mu0 = 1; sigma0 = 1 # Hyperparameters
b0_j = rnorm(J, mu0, sigma0) # Random intercepts for j-th group
b0_i = rep(b0_j, N_j) # Random intercepts for i-th obs

K = 3
b_vector = c(1, -1, 0.5)
epsilon = rnorm(N, 0, 5)

X = as.matrix(data.frame(
  x1 = rnorm(N, 2, 3),
  x2 = rpois(N, 3),
  x3 = rgamma(N, 5, 5)
))
id = rep(1:J, N_j)

# y = b0 + b1 + e
y = b0_i + X%*%b_vector + epsilon
dt = data.frame(id = id, y = y, X) # simulated data

rfit = function(fit){
 r = fit %>%
  broom::tidy() %>%
  mutate_if(is.numeric, round, 3)
 return(r)
}
```

2 Parameter estimation

2.1 Linear model

```
dt %>%
  lm(y ~ x1 + x2 + x3, data = .) %>%
  rfit()
```

```
## # A tibble: 4 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)     1.57     0.733      2.14   0.033
## 2 x1              1.03     0.078     13.2    0
## 3 x2             -1.11     0.136     -8.15    0
## 4 x3              0.75     0.531      1.41    0.159
```

2.2 Linear model with group-level fixed effects

```
dt %>%
  lm(y ~ factor(id) + x1 + x2 + x3, data = .) %>%
  rfit()
```

```
## # A tibble: 13 x 5
##    term         estimate std.error statistic p.value
##    <chr>           <dbl>     <dbl>     <dbl>   <dbl>
##  1 (Intercept)     1.33      0.971     1.37    0.17
##  2 factor(id)2    -1.61      0.967    -1.66    0.097
##  3 factor(id)3     2.28      1.05      2.17    0.03
##  4 factor(id)4     1.79      1.04      1.72    0.085
##  5 factor(id)5     0.213     1.07      0.2     0.842
##  6 factor(id)6    -0.542     1.10     -0.491   0.624
##  7 factor(id)7     0.692     0.999     0.693   0.489
##  8 factor(id)8     0.856     0.99      0.864   0.388
##  9 factor(id)9     1.38      1.04      1.33    0.183
## 10 factor(id)10   -0.309     1.03     -0.299   0.765
## 11 x1              0.991     0.079    12.6     0
## 12 x2             -1.13      0.135    -8.41    0
## 13 x3              0.709     0.523     1.36    0.176
```

2.3 Random-effects model

```
dt %>%
  lmer(y ~ x1 + x2 + x3 + (1|id), data = .) %>%
  rfit()
```

```
## Warning in bind_rows_(x, .id): binding factor and character vector,
## coercing into character vector
```

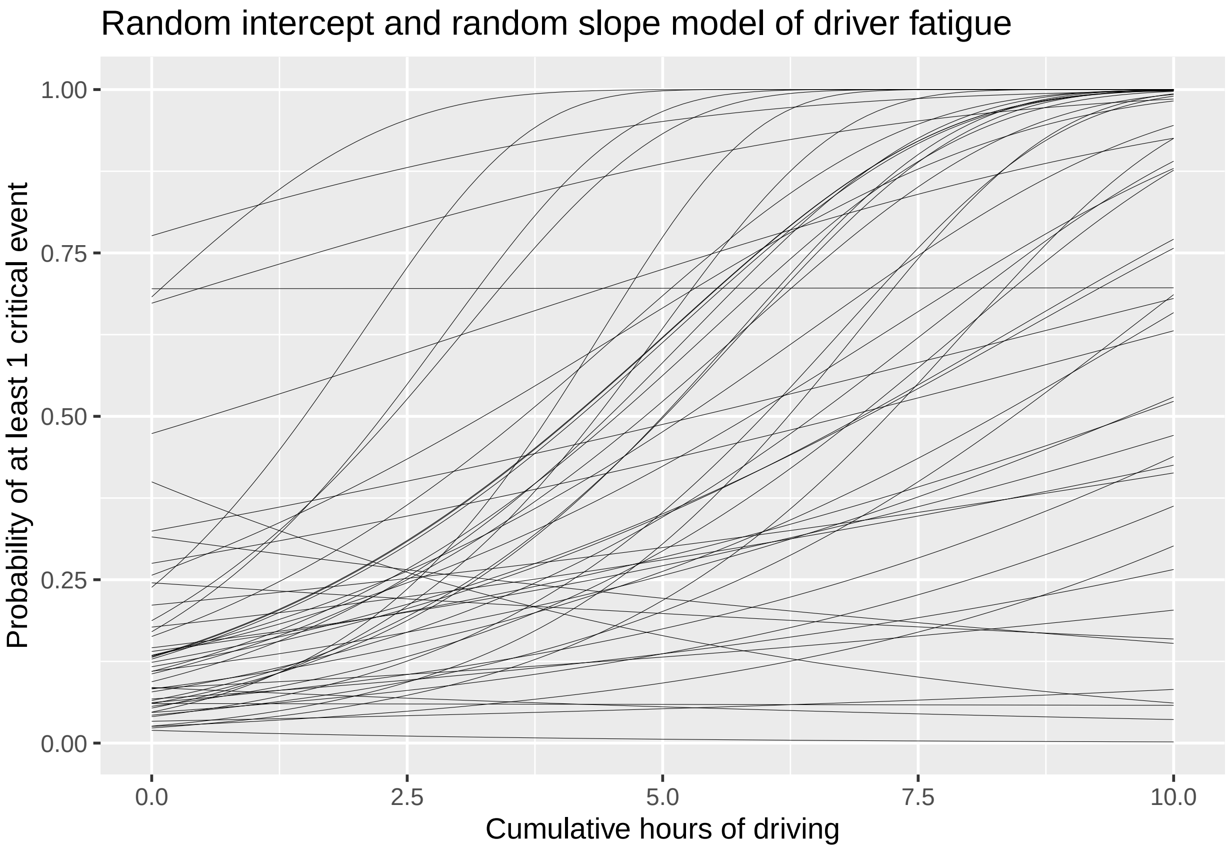
```
## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector
```

```
## # A tibble: 6 x 5
##   term                    estimate std.error statistic group
##   <chr>                      <dbl>     <dbl>     <dbl> <chr>
## 1 (Intercept)                1.73      0.78       2.21 fixed
## 2 x1                         1.00      0.078     12.9  fixed
## 3 x2                        -1.12      0.134     -8.37 fixed
## 4 x3                         0.723     0.523      1.38 fixed
## 5 sd_(Intercept).id          0.924    NA         NA    id
## 6 sd_Observation.Residual    4.70     NA         NA    Residual
```

2.4 Including Plots

You can also embed plots, for example:

```
## Loading required package: ggplot2
```


Random intercept and random slope model of driver fatigue
Probability of at least 1 critical event
1.00
0.75
0.50
0.25
0.00
0.0
2.5
5.0
7.5
10.0
Cumulative hours of driving