Multilevel analysis of directors data using JAGS

Jasper Ginn

April 8, 2019

The **directors** dataset that is included with the `blm` library has a natural hierarchical structure with individuals nested in companies nested in industries nested in sectors. In this document, we will construct a multilevel model in JAGS using companies and individuals. Given that the data do not conform to the assumptions we need for a linear mixed-effect model, we will not run a full multilevel model (i.e. containing cross-level)

```
rm(list=ls())
# Get the directors data
library(blm)
data("directors")

# Preprocess directors data
library(dplyr)
directors <- directors %>%
  # Log compensation
  mutate(Compensation = log(Compensation)) %>%
  # Subtract mean from Age variable
  mutate(Age = Age - mean(Age),
         Gend = as.numeric(Male)-1,
         Male = as.numeric(Male) - mean(as.numeric(Male))) %>%
  # Sort data (makes it easier later)
  arrange(Sector)

# Jags data
dir_jags <- with(directors, list(## Outcome for individuals (level 1)

                                 compensation = Compensation,
```

```
                                # Age of individuals
                                age=Age,
                                # Gender of individuals
                                gender=Male,

                                ## Company-level variables (level 2)

                                # Company indicator
                                company=as.numeric(as.factor(Company)),
                                # Average age of directors for each company
                                avgAge=unname(tapply(Age, Company, mean)),
                                # Proportion of males for each company
                                avgMale=unname(tapply(Gend, Company, function(x) sum(x)/length(x))),

                                ## Group totals

                                # Number of individuals
                                n=nrow(directors),
                                # Number of companies
                                k=length(unique(Company)),

                                ## Number of predictors

                                # Number of individual-level predictors
                                p1=2,
                                # Number of company-level predictors
                                p2=2))
```

```
library(ggplot2)
library(forcats)
f1a <- ggplot(directors, aes(x=fct_reorder(Company, as.numeric(Sector)), y=Compensation, color=Sector))
  geom_point() +
  geom_boxplot() +
```

```
  geom_hline(yintercept = mean(directors$Compensation)) +
  theme_blm() +
  scale_x_discrete(name = "Companies") +
  scale_y_continuous(name = "Compensation (logged)") +
  theme(legend.position="none",
        axis.text.x = element_blank())

f1b <- ggplot(directors, aes(x=Age, y=Compensation, color=Company)) +
  geom_point() +
  theme_blm() +
  theme(legend.position="none") +
  geom_smooth(method="lm", se=FALSE)

library(gridExtra)
grid.arrange(f1a, f1b, ncol=2)
```

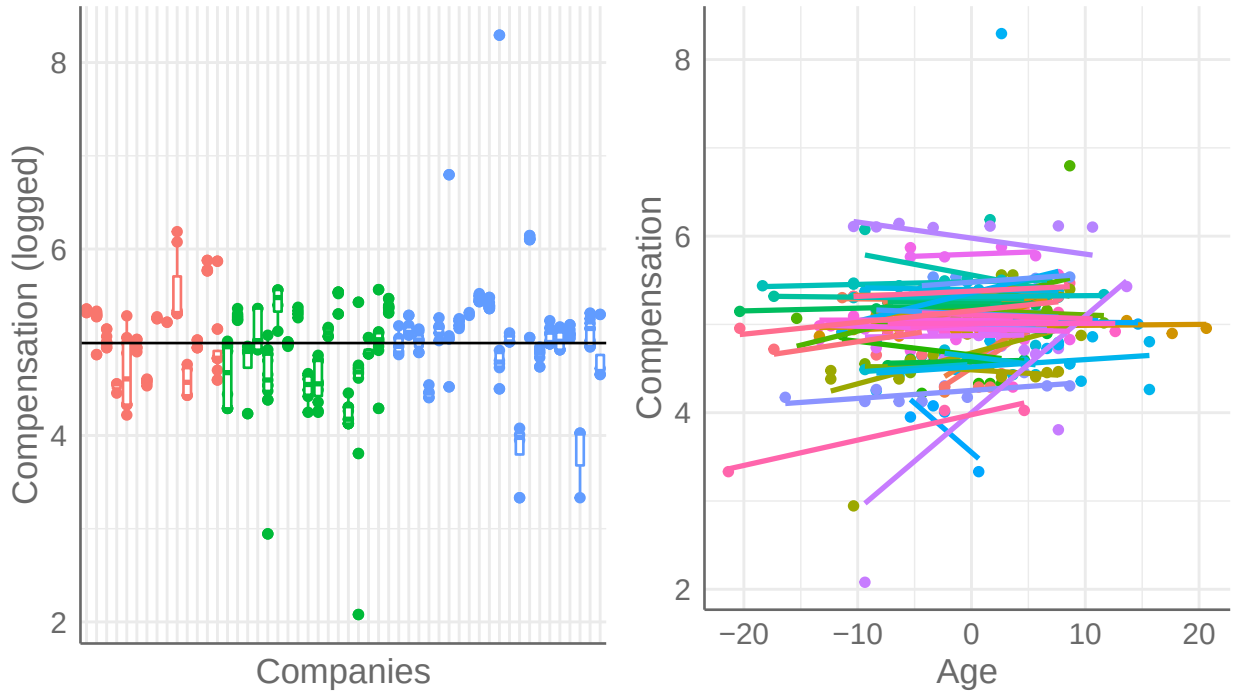

Figure 1: Within and between variance of director compensation across companies

From figure 1(a) and 1(b), it should be clear that the compensation variable across companies is not exactly

normally distributed, but we will ignore that for now. Tere appears to be variance in the intercepts for each company and some evidence for variance in the slopes.

The intercept-only model

The intercept-only model for the directors data is given by the following mixed-model equation:

$$\text{compensation}_{ijk} = \gamma_{00} + u_{0j} + e_{ij}$$

In words, this means we expect that a director's compensation is dependent on a grand mean over all companies (γ_{00}) with some error term that captures the variation across companies (u_{0j}) and an error term that captures variation across directors (e_{ij}).

Alternatively, we can specify the model as follows:

$$\begin{aligned}
\text{compensation}_{ijk} &\sim N(\beta_{0j}, \sigma_e^2) \\
\beta_{0j} &\sim N(\gamma_{00}, \sigma_{u_0}^2) \\
\gamma_{00} &\sim N(m_{00}, s_{00}^2) \\
\sigma_{u_0}^2 &\sim IG(\alpha_{2_0}, \beta_{2_0}) \\
\sigma_e^2 &\sim IG(\alpha_{1_0}, \beta_{1_0})
\end{aligned}$$

To run the model in JAGS, we construct the following code

```
library(rjags)
library(tidyr)


if(!"model1.rds" %in% list.files()) {


  # Initial values
  dir_inits <- list(
    init1 <- list(tau=runif(1), tau_u0=runif(1)),
    init2 <- list(tau=runif(1), tau_u0=runif(1))
  )
```

```
  # Specify model in JAGS
  mod_io <- jags.model("1_intercept_only.txt",
                       data = dir_jags,
                       inits = dir_inits,
                       n.chains=2)

  # Burn
  update(mod_io, n.iter=60000)

  # Draw samples
  params <- c("sigma_e", "sigma_u0", "gamma_00")
  # Run the chain
  resm1 <- coda.samples(mod_io, variable.names = params, n.iter=500000, thin = 10)

  # Save model
  saveRDS(resm1, "model1.rds")

  # DIC
  DICm1 <- dic.samples(mod_io, thin=5, n.iter=20000, type="pD")

  # Save
  saveRDS(DICm1, "model1_DIC.rds")

} else {

  resm1 <- readRDS(file="model1.rds")
  DICm1 <- readRDS(file="model1_DIC.rds")

}

# MAP values
MAPm1 <- apply(do.call(rbind.data.frame, resm1), 2, mean)
```

```r
# Variance explained
vl2 <- MAPm1[3]^2 / sum(MAPm1[-1]^2)
vl1 <- 1-vl2  # 42.6%
```

The values returned by JAGS allows us to calculate the *intraclass correlation coefficient*, defined as:

$$\rho = \frac{\sigma^2_{u_0}}{\sigma^2_e + \sigma^2_{u_0}}$$

We interpret it as follows:

1. Roughly 57.3% of the variance can be found at the level 2 (company) variable.
2. The expected correlation of two randomly picked directors in a given company is $\hat{r} = .573$.

In the case of the directors data, the majority of the variance comes from variation between companies. Consequently, 42.6% of the variance is within-group variance and a result of variance among individuals.

Finally, we note that 0 is not included in the credible interval of $\sigma^2_{u_0}$, which leads us to the conclusion that a multilevel model is appropriate. (95% CCI = $[.321, .503]$).

Level 1 variables

The second-level model is specified using the following mixed-model equation:

$$\text{compensation}_{ij} = \gamma_{00} + u_{0j} + \gamma_{10} \cdot \text{Age}_{ij} + \gamma_{20} \cdot \text{Male}_{ij} + e_{ij}$$

or, equivalently:

$$\text{compensation}_{ijk} \sim N(\beta_{0j} + \gamma_{10} \cdot \text{Age}_{ij} + \gamma_{20} \cdot \text{Gender}_{ij}, \sigma_e^2)$$

$$\beta_{0j} \sim N(\gamma_{00}, \sigma_{u_0}^2)$$

$$\gamma_{00} \sim N(m_{00}, s_{00}^2)$$

$$\gamma_{10} \sim N(m_{10}, s_{10}^2)$$

$$\gamma_{20} \sim N(m_{20}, s_{20}^2)$$

$$\sigma_{u_0}^2 \sim IG(\alpha_{2_0}, \beta_{2_0})$$

$$\sigma_e^2 \sim IG(\alpha_{1_0}, \beta_{1_0})$$

This model is implemented as follows (gender is removed here because its 95% CCI contains 0).

```
# Initial values
dir_inits <- list(
  init1 <- list(tau=runif(1), tau_u0=runif(1)),
  init2 <- list(tau=runif(1), tau_u0=runif(1))
)

if(!"model2a.rds" %in% list.files()) {

  # Specify model in JAGS
  mod_io <- jags.model("2a_level_one_predictors.txt",
                       data = dir_jags,
                       inits = dir_inits,
                       n.chains=2)

  # Burn
  update(mod_io, n.iter=60000)

  # Draw samples
  params <- c("sigma_e", "sigma_u0", "gamma_00", "gamma_10")
  # Run the chain
  resm2 <- coda.samples(mod_io, variable.names = params, n.iter=500000, thin = 10)
```

```
  # Save model
  saveRDS(resm2, "model2a.rds")

  # DIC
  DICm2 <- dic.samples(mod_io, thin=5, n.iter=20000, type="pD")

  # Save
  saveRDS(DICm2, "model2a_DIC.rds")

} else {

  resm2 <- readRDS(file="model2a.rds")
  DICm2 <- readRDS(file="model2a_DIC.rds")

}
# MAP values
MAPm2 <- apply(do.call(rbind.data.frame, resm2), 2, mean)

# Variance explained
ve_l1 <- ((MAPm1[2] - MAPm2[3]) / MAPm1[2]) * 100
ve_l2 <- ((MAPm1[3] - MAPm2[4]) / MAPm1[3]) * 100
```

Explained variance at each level:

1. Level 1

$$R^2_{L1} = \frac{\sigma^2_{e|\text{baseline}} - \sigma^2_{e|\text{model 1}}}{\sigma^2_{e|\text{baseline}}} = 1.34\%$$

2. Level 2

$$R^2_{L2} = \frac{\sigma^2_{u_0|\text{baseline}} - \sigma^2_{u_0|\text{model 1}}}{\sigma^2_{u_0|\text{baseline}}} = 0.76\%$$

Adding level-2 predictors

$$\text{compensation}_{ij} = \gamma_{00} + u_{0j} + \gamma_{10} \cdot \text{Age}_{ij} + \gamma_{01} \cdot \text{avgAge}_j + \gamma_{02} \cdot \text{avgMale}_j + e_{ij}$$

or, equivalently:

$$\begin{aligned}
\text{compensation}_{ij} &\sim N(\beta_{0j} + \gamma_{10} \cdot \text{Age}_{ij} + \gamma_{01} \cdot \text{avgAge}_j + \gamma_{02} \cdot \text{avgMale}_j, \sigma_e^2) \\
\beta_{0j} &\sim N(\gamma_{00}, \sigma_{u_0}^2) \\
\gamma_{00} &\sim N(m_{00}, s_{00}^2) \\
\gamma_{10} &\sim N(m_{10}, s_{10}^2) \\
\gamma_{01} &\sim N(m_{01}, s_{01}^2) \\
\gamma_{02} &\sim N(m_{02}, s_{02}^2) \\
\sigma_{u_0}^2 &\sim IG(\alpha_{2_0}, \beta_{2_0}) \\
\sigma_e^2 &\sim IG(\alpha_{1_0}, \beta_{1_0})
\end{aligned}$$

This model is implemented as follows

```
# Initial values
dir_inits <- list(
  init1 <- list(tau=runif(1), tau_u0=runif(1)),
  init2 <- list(tau=runif(1), tau_u0=runif(1))
)

if(!"model3.rds" %in% list.files()) {

  # Specify model in JAGS
  mod_io <- jags.model("3_level_two_predictors.txt",
                       data = dir_jags,
                       inits = dir_inits,
                       n.chains=2)

  # Burn
  update(mod_io, n.iter=60000)
```

```
  # Draw samples
  params <- c("sigma_e", "sigma_u0", "gamma_00", "gamma_10", "gamma_02")
  # Run the chain
  resm3 <- coda.samples(mod_io, variable.names = params, n.iter=500000, thin = 5)

  # DIC
  DICm3 <- dic.samples(mod_io, thin=5, n.iter=20000)

  # Save model
  saveRDS(resm3, "model3.rds")

  # Save
  saveRDS(DICm3, "model3_DIC.rds")



} else {

  # Load
  resm3 <- readRDS("model3.rds")
  DICm3 <- readRDS("model3_DIC.rds")

}

# MAP values
MAPm3 <- apply(do.call(rbind.data.frame, resm3), 2, mean)

# Variance explained at level 2
ve2 <- ((MAPm1[3] - MAPm3[5]) / MAPm1[3]) * 100
```

The variable 'AvgAge' is removed from the model since its 95% CCI contains 0. By adding the level 2 variables, the reduction in level-2 variance becomes:

$$R^2_{L2} = \frac{\sigma^2_{u_0|\text{baseline}} - \sigma^2_{u_0|\text{model 2}}}{\sigma^2_{u_0|\text{baseline}}} = 3.71\%$$

Random coefficient model

$$\text{compensation}_{ij} = \gamma_{00} + u_{0j} + \gamma_{10} \cdot \text{Age}_{ij} + \gamma_{20} \cdot \text{Male}_{ij} + u_{1j} \cdot \text{Age}_{ij} + u_{2j} \cdot \text{Male}_{ij} + \gamma_{02} \cdot \text{avgMale}_j + e_{ij}$$

or, equivalently:

$$\begin{aligned}
\text{compensation}_{ij} &\sim N(\beta_{0j} + \beta_{1j} \cdot \text{Age}_{ij} + \beta_{2j} \cdot \text{gender}_{ij} + \gamma_{02} \cdot \text{avgMale}_j, \sigma^2_e) \\
\beta_{0j} &\sim N(\gamma_{00}, \sigma^2_{u_0}) \\
\beta_{1j} &\sim N(\gamma_{10}, \sigma^2_{u_1}) \\
\beta_{2j} &\sim N(\gamma_{20}, \sigma^2_{u_2}) \\
\gamma_{00} &\sim N(m_{00}, s^2_{00}) \\
\gamma_{10} &\sim N(m_{00}, s^2_{00}) \\
\gamma_{20} &\sim N(m_{01}, s^2_{01}) \\
\gamma_{02} &\sim N(m_{02}, s^2_{02}) \\
\sigma^2_{u_0} &\sim IG(\alpha_{2_0}, \beta_{2_0}) \\
\sigma^2_{u_1} &\sim IG(\alpha_{2_1}, \beta_{2_1}) \\
\sigma^2_{u_2} &\sim IG(\alpha_{2_2}, \beta_{2_2}) \\
\sigma^2_e &\sim IG(\alpha_{1_0}, \beta_{1_0})
\end{aligned}$$

We can run this model as follows

```
# Initial values
dir_inits <- list(
  init1 <- list(tau=runif(1), tau_u0=runif(1), tau_u1=runif(1), tau_u2=runif(1)),
  init2 <- list(tau=runif(1), tau_u0=runif(1), tau_u1=runif(1), tau_u2=runif(1))
)


if(!"model4.rds" %in% list.files()) {
```

```
  # Specify model in JAGS
  mod_io <- jags.model("4_random_coefficients.txt",
                       data = dir_jags,
                       inits = dir_inits,
                       n.chains=2)

  # Burn
  update(mod_io, n.iter=60000)

  # Draw samples
  params <- c("sigma_e", "sigma_u0", "sigma_u1", "sigma_u2",
              "gamma_00", "gamma_10", "gamma_20", "gamma_02")
  # Run the chain
  resm4 <- coda.samples(mod_io, variable.names = params, n.iter=500000, thin = 10)

  # DIC model 4
  DICm4 <- dic.samples(mod_io, thin=5, n.iter=20000)

  # Save model
  saveRDS(resm4, "model4.rds")

  # Save
  saveRDS(DICm4, "model4_DIC.rds")

} else {

  # Load
  resm4 <- readRDS("model4.rds")
  DICm4 <- readRDS("model4_DIC.rds")

}
```

```r
# MAP values
MAPm4 <- apply(do.call(rbind.data.frame, resm4), 2, mean)
```

The model fit is worsening but we do observe that the variance components do not contain 0 in their 95% CCI. This fact also means that we need to keep the main effects in the model.

The cross-level interaction

Finally, we run the full multilevel model containing a cross-level interaction:

$$\begin{aligned}\text{compensation}_{ij} =& \gamma_{00} + u_{0j} + \\ & \gamma_{10} \cdot \text{Age}_{ij} + \gamma_{11} \cdot \text{Age}_{ij} \cdot \text{avgAge}_j + \gamma_{12} \cdot \text{Age}_{ij} \cdot \text{avgMale}_j + \\ & \gamma_{20} \cdot \text{Male}_{ij} + \gamma_{21} \cdot \text{Male}_{ij} \cdot \text{avgAge}_j + \gamma_{22} \cdot \text{Male}_{ij} \cdot \text{avgMale}_j + \\ & u_{1j} \cdot \text{Age}_{ij} + u_{2j} \cdot \text{Male}_{ij} + \\ & \gamma_{01} \cdot \text{avgAge}_j + \gamma_{02} \cdot \text{avgMale}_j + \\ & e_{ij}\end{aligned}$$

or, equivalently:

$$\begin{aligned}
\text{compensation}_{ij} \sim N(&\beta_{0j} + \beta_{1j} \cdot \text{Age}_{ij} + \beta_{2j} \cdot \text{gender}_{ij} + \gamma_{02} \cdot \text{avgMale}_j + \gamma_{11} \cdot \text{Age}_{ij} \cdot \text{avgAge}_j + \\
&\gamma_{01} \cdot \text{avgAge}_j \gamma_{12} \cdot \text{Age}_{ij} \cdot \text{avgMale}_j + \gamma_{21} \cdot \text{Male}_{ij} \cdot \text{avgAge}_j + \\
&\gamma_{22} \cdot \text{Male}_{ij} \cdot \text{avgMale}_j, \sigma_e^2) \\
\beta_{0j} &\sim N(\gamma_{00}, \sigma_{u_0}^2) \\
\beta_{1j} &\sim N(\gamma_{10}, \sigma_{u_1}^2) \\
\beta_{2j} &\sim N(\gamma_{20}, \sigma_{u_2}^2) \\
\gamma_{00} &\sim N(m_{00}, s_{00}^2) \\
\gamma_{10} &\sim N(m_{00}, s_{00}^2) \\
\gamma_{20} &\sim N(m_{01}, s_{01}^2) \\
\gamma_{01} &\sim N(m_{01}, s_{01}^2) \\
\gamma_{02} &\sim N(m_{02}, s_{02}^2) \\
\gamma_{11} &\sim N(m_{11}, s_{11}^2) \\
\gamma_{12} &\sim N(m_{12}, s_{12}^2) \\
\gamma_{21} &\sim N(m_{21}, s_{21}^2) \\
\gamma_{22} &\sim N(m_{22}, s_{22}^2) \\
\sigma_{u_0}^2 &\sim IG(\alpha_{2_0}, \beta_{2_0}) \\
\sigma_{u_1}^2 &\sim IG(\alpha_{2_1}, \beta_{2_1}) \\
\sigma_{u_2}^2 &\sim IG(\alpha_{2_2}, \beta_{2_2}) \\
\sigma_e^2 &\sim IG(\alpha_{1_0}, \beta_{1_0})
\end{aligned}$$

```
# Initial values
dir_inits <- list(
  init1 <- list(tau=runif(1), tau_u0=runif(1), tau_u1=runif(1), tau_u2=runif(1)),
  init2 <- list(tau=runif(1), tau_u0=runif(1), tau_u1=runif(1), tau_u2=runif(1))
)


if(!"model5.rds" %in% list.files()) {
  # Specify model in JAGS
  mod_io <- jags.model("5_cross_level_interaction.txt",
                       data = dir_jags,
```

```
                        inits = dir_inits,
                        n.chains=2)

  # Burn
  update(mod_io, n.iter=60000)

  # Draw samples
  params <- c("sigma_e", "sigma_u0", "sigma_u1", "sigma_u2",
              "gamma_00", "gamma_10", "gamma_20", "gamma_01", "gamma_02",
              "gamma_21", "gamma_11", "gamma_22", "gamma_12")
  # Run the chain
  resm5 <- coda.samples(mod_io, variable.names = params, n.iter=500000, thin = 10)

  # All variance estimates > 0 so most likely significant
  DICm5 <- dic.samples(mod_io, thin=5, n.iter=20000)

  # Save model
  saveRDS(resm5, "model5.rds")

  # Save
  saveRDS(DICm5, "model5_DIC.rds")

} else {

  resm5 <- readRDS("model5.rds")
  DICm5 <- readRDS("model5_DIC.rds")

}

# MAP values
MAPm5 <- apply(do.call(rbind.data.frame, resm5), 2, mean)
```

```r
# SLope variance explained
s1 <- (MAPm4[7] - MAPm5[12]) / MAPm4[7]
s2 <- (MAPm4[8] - MAPm5[13]) / MAPm4[8]
```

We can now estimate how much variance of the slopes we are explaining

1. Level 1

$$R^2_{\text{Slope},L1} = \frac{\sigma^2_{u_{1j}|\text{baseline}} - \sigma^2_{u_{1j}|\text{model 1}}}{\sigma^2_{u_{1j}|\text{baseline}}} = -1.8\%$$

2. Level 2

$$R^2_{\text{Slope},L1} = \frac{\sigma^2_{u_{2j}|\text{baseline}} - \sigma^2_{u_{2j}|\text{model 1}}}{\sigma^2_{u_{2j}|\text{baseline}}} = 4\%$$

We are explaining some of the variance in the gender slope but none of the variance in the age slope.

Model comparison

```r
# Table with DIC values
DICs <- data.frame(
  "DIC" = c(sum(DICm1$deviance + DICm1$penalty),
            sum(DICm2$deviance + DICm2$penalty),
            sum(DICm3$deviance + DICm3$penalty),
            sum(DICm4$deviance + DICm4$penalty),
            sum(DICm5$deviance + DICm5$penalty)),
  "pD" = c(sum(DICm1$penalty),
           sum(DICm2$penalty),
           sum(DICm3$penalty),
           sum(DICm4$penalty),
           sum(DICm5$penalty)),
  row.names = c("Intercept-only",
```

```
                "Predictors level 1",
                "Predictors levels 1 + 2",
                "Random coefficient",
                "Cross-level interaction")
)
# Cat
knitr::kable(DICs)
```

	DIC	pD
Intercept-only	288.6683	47.26153
Predictors level 1	280.2960	48.17404
Predictors levels 1 + 2	280.2528	48.14529
Random coefficient	287.8816	84.85971
Cross-level interaction	288.9189	86.21852

As we can see from the results, we are not explaining a lot of the variance in our data. Based on the ICC, it's definitely worth doing a multilevel analysis, but we clearly need better predictors at both levels.

With respect to the final model, we cannot really distinguish between models 2 (level 1 variables) and 3 (level 1 and 2 variables) when looking at model fit. However, we usually prefer more parsimonious models so our final model is model 2.

```
# Results from model 3
final_mod <- do.call(rbind.data.frame, resm2)
```

References

Lynch, S. M. (2007). Introduction to applied Bayesian statistics and estimation for social scientists. Springer Science & Business Media. Chapter 9.2

Aarts, E. (2019). Introduction to multilevel analysis and the basic two-level regression model, week 1 notes [powerpoint presentation]. *Introduction to Multilevel Analysis*, Utrecht University.