Chapter 5

Lin 205C
Santiago Barreda

Chapter Precap

- Compare two groups and introduce the concept of between and within-subjects factors.
- Factor coding is presented, and treatment and sum coding are compared.
- Discuss the decomposition of variation in the dependent variable into effects associated with independent variables
- Discuss retrieving and manipulating the model posterior samples.
- The hypothesis function is introduced, and the retrieval and combination of 'random effect' parameters are explained.
- Finally, there is a discussion of outliers and robustness, and the t-distribution is presented.

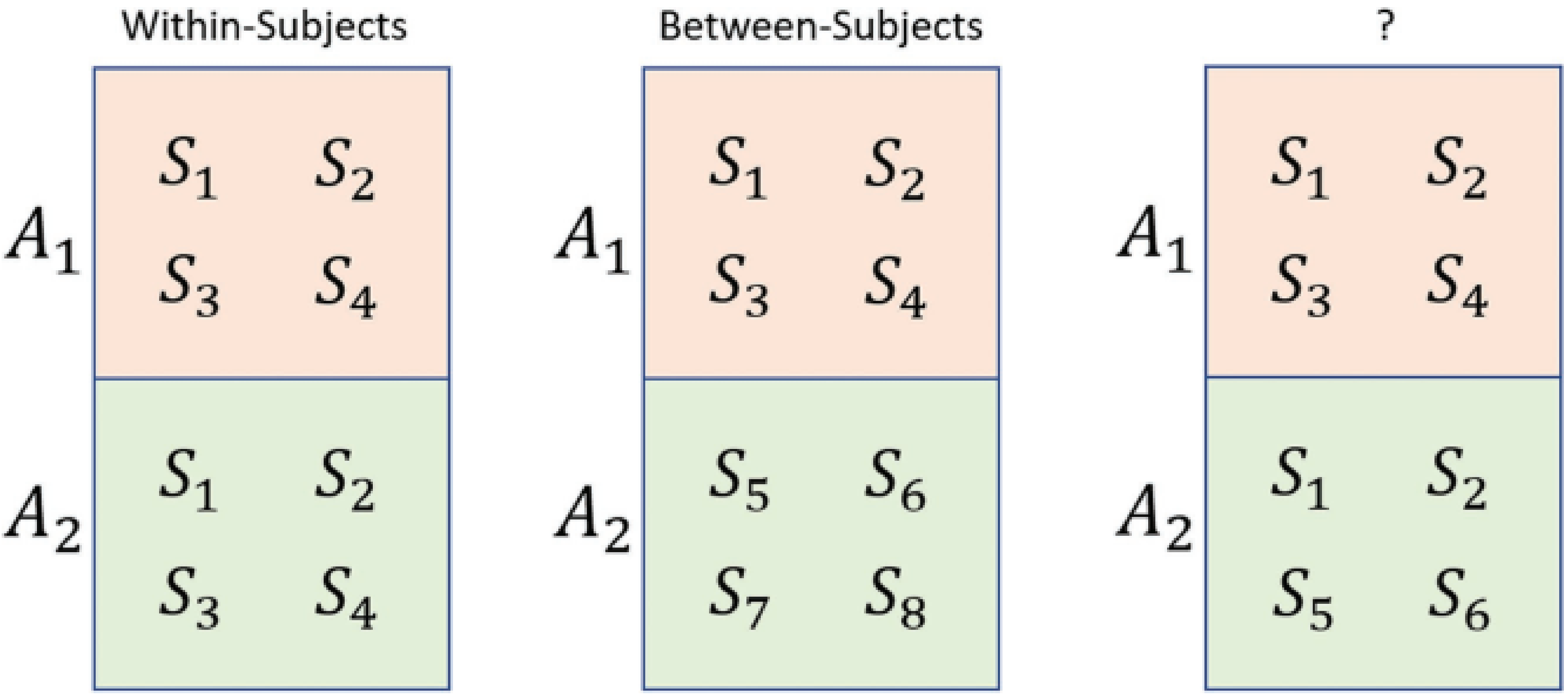
Comparing Two Groups

- We can observe under two different conditions.

- We can ask: Is the distribution of observations under one condition the same as the distribution under the other condition?

- Usually, people focus on: Is the mean the same for both conditions, i.e. does $\mu_{A_1} = \mu_{A_2}$ for conditions A_1 and A_2?

Between and Within Subjects



Data

```
# load packages and data
library (bmmb)
library (brms)
data (exp_data)
# exclude actual men and apparent men
notmen = exp_data[exp_data$C_v!='m' & exp_data$C!='m',]
```

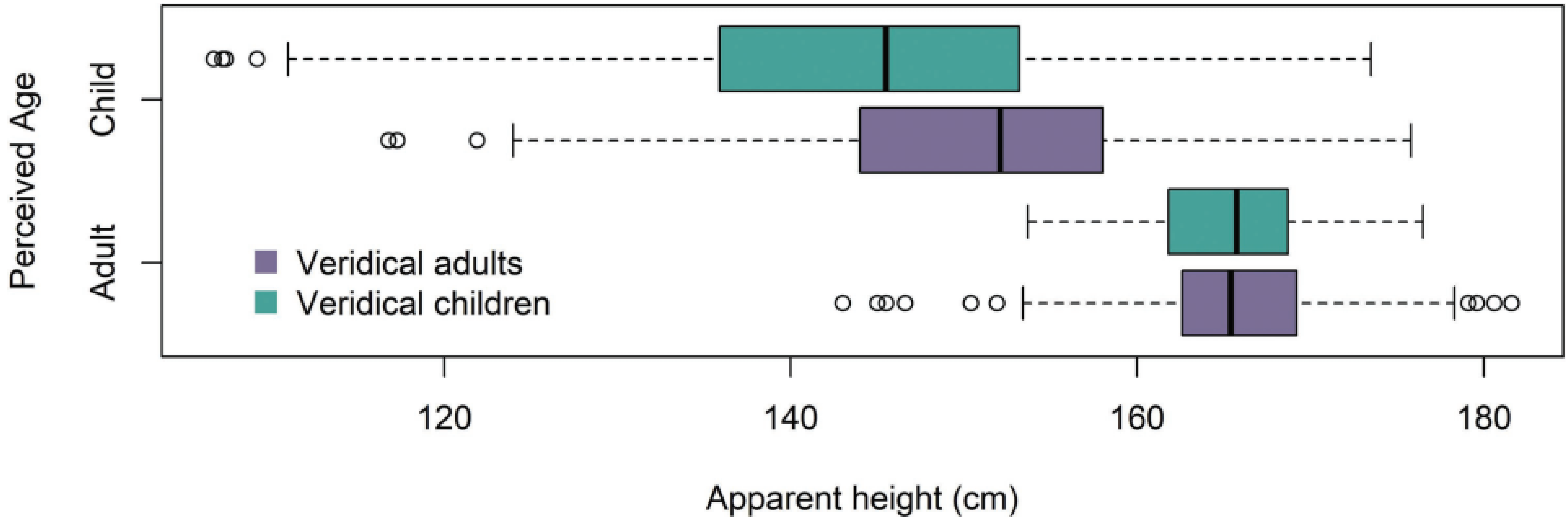
- `L`: An integer from 1 to 15 indicating which *listener* responded to the trial.
- `height`: A floating-point number representing the *height* (in centimeters) reported for the speaker on each trial.
- `S`: An integer from 1 to 139 indicating which *speaker* produced the trial stimulus.
- `A`: The *apparent age* of the speaker indicated by the listener, `a` (adult) or `c` (child).

Confusable Voices

```
xtabs ( ~ bmmb::exp_data$C_v + bmmb::exp_data$C)
##                bmmb::exp_data$C
## bmmb::exp_data$C_v   b   g   m   w
##                  b 234 133   6  32
##                  g  79 184   0  22
##                  m  31   0 626  18
##                  w  97 109   3 511
```

```
xtabs (~ notmen$A_v + notmen$C)
##           notmen$C
## notmen$A_v   b   g   w
##          a  97 109 511
##          c 313 317  54
```

Apparent Age and Height



Research Questions

(Q1) How tall do speakers perceived as adult females sound?

(Q2) How tall do speakers perceived as children sound?

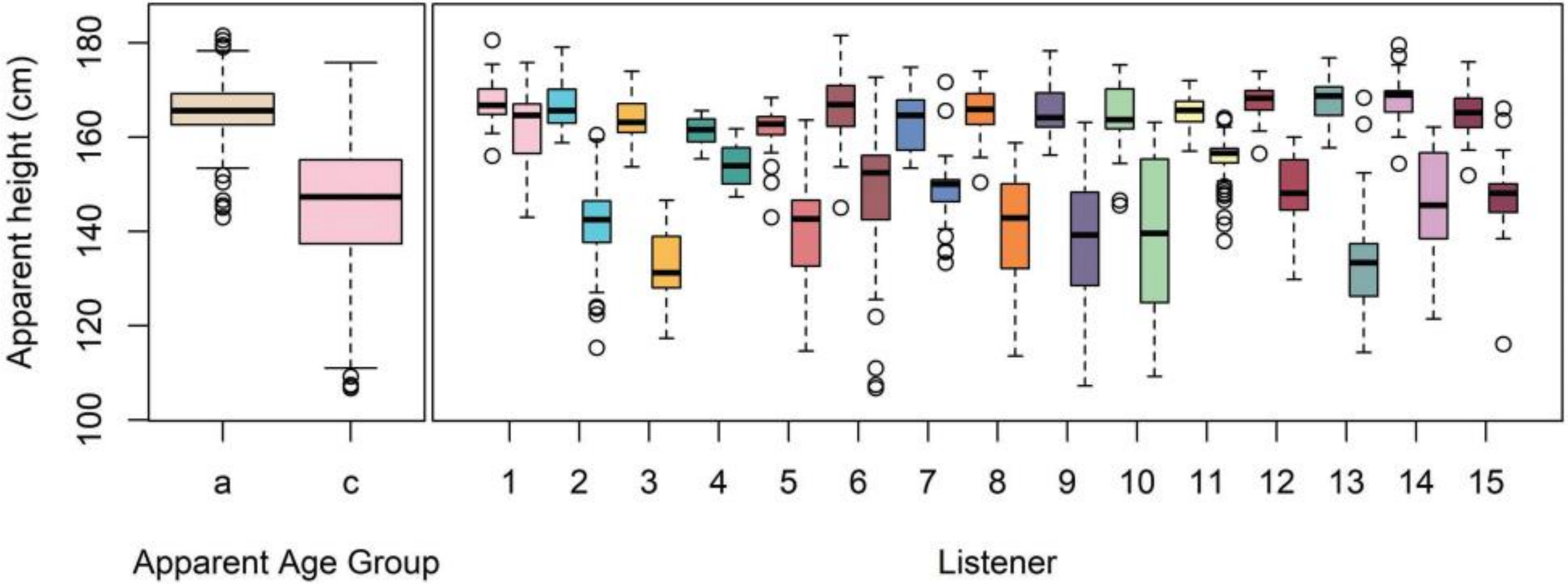
(Q3) What is the difference in apparent height associated with the perception of adultness?

Between and Within

```
xtabs (~ notmen$A + notmen$L)
##            notmen$L
## notmen$A  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
##        a 42 35 42 20 34 37 30 48 42 46 42 44 31 36 36
##        c 52 59 50 74 58 55 64 46 49 48 52 50 63 58 58
```

```
xtabs (~ notmen$A + notmen$S)[,40:50]
##            notmen$S
## notmen$A 40 41 42 43 44 45 46 92 93 94 95
##        a 11  0  0  0  0  0  0 11  7 15 14
##        c  4 15 15 15 15 15 15  4  8  0  1
```

Differences Between Two Groups



The Model Formula

- Our last model contained only an intercept ‘fixed’ effect.

```
height ~ 1 + (1|L) + (1|S)
```

- To include apparent age (A), we just add it outside the parentheses:

```
height ~ 1 + A + (1|L) + (1|S)
```

- We can omit the ‘1’ when another predictor is present:

```
height ~ A + (1|L) + (1|S)
```

Fitting the Model

- `Intercept`: this is a unique class, only for intercepts.
- `b`: This class includes all fixed-effect predictors *apart* from the intercept.
- `sd`: This is for standard deviation parameters related to 'batches' of parameters, e.g. `sd(Intercept)` for `L` (σ_L).
- `sigma`: the error term.

```
# Fit the model yourself
model = brms::brm (
  height ~ A + (1|L) + (1|S), data = notmen, chains = 4, cores = 4,
  warmup = 1000, iter = 3500, thin = 2,
  prior = c(brms::set_prior("normal(156, 12)", class = "Intercept"),
          brms::set_prior("normal(0, 12)", class = "b"),
          brms::set_prior("normal(0, 12)", class = "sd"),
          brms::set_prior("normal(0, 12)", class = "sigma")))
```

Interpreting the Model

```
# inspect model
bmmb::short_summary (model)
## Formula:  height ~ A + (1 | L) + (1 | S)
##
## Group-Level Effects:
## ~L (Number of levels: 15)
##               Estimate Est.Error l-95% CI u-95% CI
## sd(Intercept)     5.24      1.11     3.54     7.85
##
## ~S (Number of levels: 94)
##               Estimate Est.Error l-95% CI u-95% CI
## sd(Intercept)     3.59      0.43      2.8     4.51
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI
## Intercept   163.92      1.46   160.97   166.72
## Ac          -17.37      0.74   -18.81   -15.89
##
```

?

```
# calculate mean apparent height based on apparent adultness
tapply (notmen$height, notmen$A, mean)
##     a     c
## 165.5 145.4
```

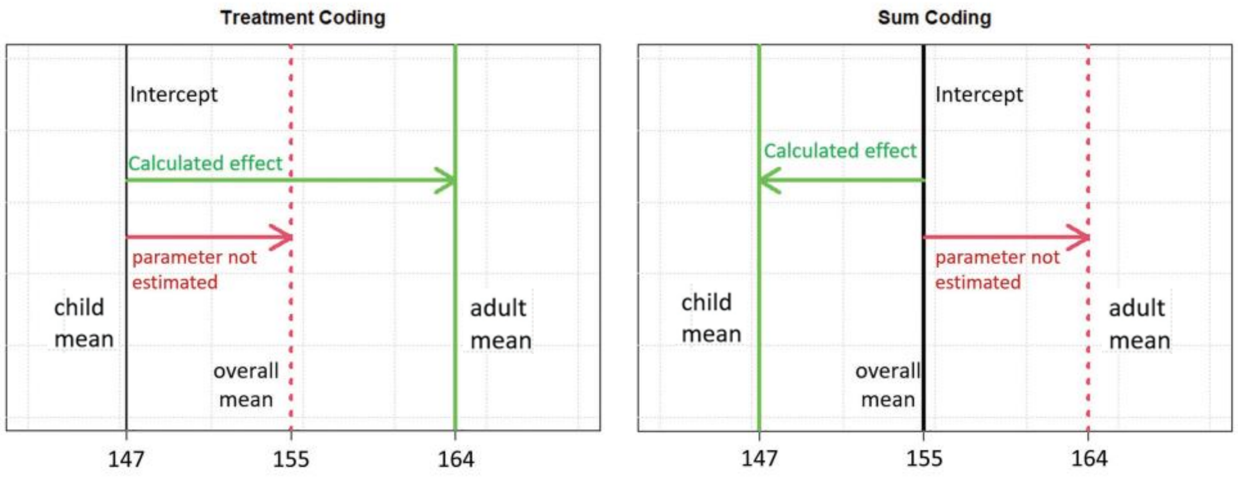
Contrasts

- Your model needs to pick a ‘reference’.
- All group parameters encode variation relative to this reference.
- The mathematical implementation of this is called ‘contrast coding’.

Treatment Coding

To interpret treatment-coded coefficients in a regression model:

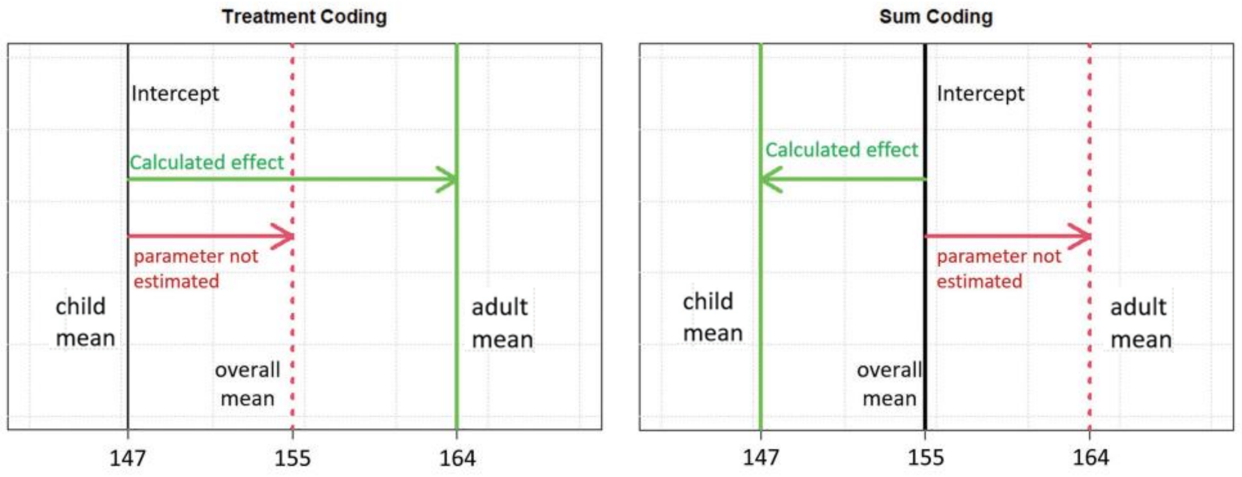
- The reference category mean is the ‘Intercept’ in the model.
- The value of the coefficients of any non-intercept group is equal to `group mean - Intercept (reference group mean)`.
- To recover the mean estimate for any non-intercept group, we add `group effect + Intercept (reference group mean)`.



Sum Coding

To interpret sum-coded coefficients in regression models:

- The mean of all your group means (the grand mean) is the 'Intercept' in the model.
- The value of the coefficients of any other group mean will be equal to `group mean - Intercept (grand mean)`.
- To recover the mean estimate for any other group, we add `group effect + Intercept (grand mean)`.



Description of Our Model

Notice the coding is not represented here!

$$\text{height}_{[i]} \sim \mathrm{N}\left(\mu_{[i]}, \sigma\right)$$

$$\mu_{[i]} = \text{Intercept} + A + L_{\left[\mathsf{L}_{[i]}\right]} + S_{\left[\mathsf{S}_{[i]}\right]}$$

Apparent speaker height is expected to vary according to a normal distribution with some unknown mean (μ) and standard deviation (σ). Means are expected to vary based on whether the listener identified the speaker as an adult or a child (A), and listener and speaker-dependent deviations from the mean (L,S). The listener and speaker effects were modeled as coming from a normal distribution with means of 0 and unknown standard deviations (σ_L, σ_S). The intercept was given a normal prior with a mean of 156 and a standard deviation of 12, and the remaining parameters were given normal priors centered at 0 with standard deviations of 12.

Priors :

$$L_{[\bullet]} \sim \mathrm{N}(0, \sigma_L)$$

$$S_{[\bullet]} \sim \mathrm{N}(0, \sigma_S)$$

$$\text{Intercept} \sim \mathrm{N}(156, 12)$$

$$A \sim \mathrm{N}(0, 12)$$

$$\sigma \sim \mathrm{N}(0, 12)$$

$$\sigma_L \sim \mathrm{N}(0, 12)$$

$$\sigma_S \sim \mathrm{N}(0, 12)$$

Sum Coding and Decomposing Variation

$$\text{height}_{[i]} \sim \mathrm{N}\left(\mu_{[i]}, \sigma\right)$$

$$\mu_{[i]} = \text{Intercept} + A + L_{\left[\mathsf{L}_{[i]}\right]} + S_{\left[\mathsf{S}_{[i]}\right]}$$

- Represent group means as deviations from the intercept.
- Represent the listener and speaker-specific deviations from the intercept $(L_{[\cdot]}, S_{[\cdot]})$ as being centered at 0, with standard deviations of σ_L and σ_S.
- Represent the random error (ε) as having a mean of 0 and a standard deviation of σ.

- When a group coefficient is 0 the group lies exactly at the intercept. In sum coding this is the grand mean (the mean of the means) indicating that the group is basically average.
- When a listener or speaker effect is 0 this listener/speaker is exactly average with respect to their group. This means there is nothing about this speaker's average that is unpredictable given knowledge of their group.
- When an error is 0 this production is exactly as expected for a given listener/speaker in a given group. This means that an observation contains no error since it was *exactly* predictable.

Priors :

$$L_{[\cdot]} \sim \mathrm{N}(0, \sigma_L)$$

$$S_{[\cdot]} \sim \mathrm{N}(0, \sigma_S)$$

$$\text{Intercept} \sim \mathrm{N}(156, 12)$$

$$A \sim \mathrm{N}(0, 12)$$

$$\sigma \sim \mathrm{N}(0, 12)$$

$$\sigma_L \sim \mathrm{N}(0, 12)$$

$$\sigma_S \sim \mathrm{N}(0, 12)$$

Sum Coding and Decomposing Variation

If we think of our predictors as representing deviations from some reference value, we can ‘break up’ any observed value into its component parts. For example, suppose that:

- The overall mean is 157 cm.
- The adult female mean is 165 cm (+8 over the intercept).
- A particular speaker has a mean apparent height of 170 cm (+5 over the adult female mean).

If we observe an apparent height judgment of 173 cm for this speaker, that suggests the following decomposition:

173 = 157 (Intercept) + 8 (adult female effect) + 5 (speaker effect) + 3 (error)

This reflects the following considerations:

- The average apparent height across the groups is 157 cm.
- The average for adult females is 8 cm above the overall mean (157 + 8 = 165).
- This speaker’s average apparent height is 5 cm above the average for adult females (157 + 8 + 5 = 170).
- This particular production is 3 cm higher than expected for this particular speaker (157 + 8 + 5 + 3 = 173).

Fitting the Model

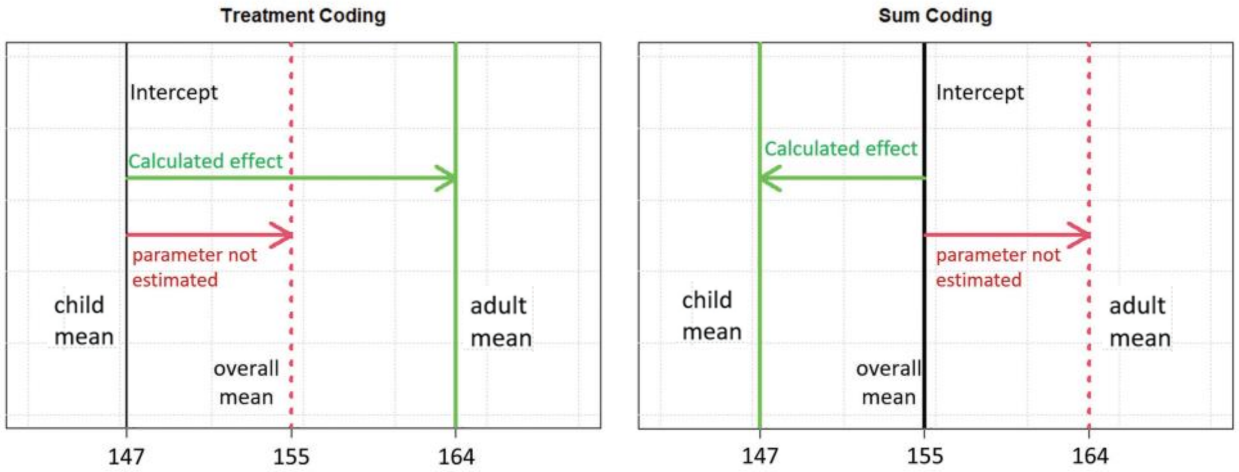
```
# to change to sum coding
options (contrasts = c('contr.sum','contr.sum'))
```

```
# Fit the model yourself
model_sum_coding =  brms::brm (
  height ~ A + (1|L) + (1|S), data = notmen, chains = 4, cores = 4,
  warmup = 1000, iter = 3500, thin = 2,
  prior = c(brms::set_prior("normal(156, 12)", class = "Intercept"),
            brms::set_prior("normal(0, 12)", class = "b"),
            brms::set_prior("normal(0, 12)", class = "sd"),
            brms::set_prior("normal(0, 12)", class = "sigma")))

# Or download it from the GitHub page:
model_sum_coding = bmmb::get_model ('5_model_sum_coding.RDS')
```

Comparison of Sum and Treatment Coding

```
# sum coding
brms::fixef (model_sum_coding)
##             Estimate Est.Error    Q2.5   Q97.5
## Intercept    155.218    1.4050 152.434 157.965
## A1             8.711    0.3602   8.003   9.416
```

```
# treatment coding
brms::fixef (model)
##             Estimate Est.Error   Q2.5  Q97.5
## Intercept     163.92    1.4580 160.97 166.72
## Ac            -17.37    0.7391 -18.81 -15.89
```



Variance of the Sum of Variables

$$\sigma^2_{x+y} = \sigma^2_x + \sigma^2_y + 2\rho\sigma_x\sigma_y$$

$$\sigma^2_{x-y} = \sigma^2_x - \sigma^2_y - 2\rho\sigma_x\sigma_y$$

You need to consider the correlation.

```
# 'marginal' variance of x
var (x)
## [1] 9.167

# 'marginal' variance of y
var (y)
## [1] 8.773

# variance of sum of x and y
var (x+y)
## [1] 0.6093

# variance of difference of x and y
var (x-y)
## [1] 35.27
```

Combine Before Summarizing

- Samples must be combined/transformed before summarizing them.
- Summarizing and then combining may give you a totally wrong credible interval (variance) estimate.
- Combining individual posterior samples preserves this information.

```
samples = brms::fixef (model_sum_coding, summary = FALSE)
head (samples)
##      variable
## draw Intercept    A1
##    1     153.2 9.040
##    2     154.7 8.365
##    3     152.2 8.835
##    4     156.0 9.003
##    5     154.7 8.196
##    6     155.9 8.305
```

Good

`mean(Intercept + A1)`

Bad

`mean(Intercept) + mean(A1)`

The Hypothesis Function

- Hypotheses are combinations of individual parameter samples.

- You can use bmmb::get_samples to get the individual samples of your hypotheses.

Model

Hypothesis

```
bmmb::short_hypothesis(model_sum_coding, "Intercept + A1 = 0")
##     Estimate Est.Error  Q2.5 Q97.5           hypothesis
## H1     163.9     1.465 161.1 166.8 (Intercept+A1) = 0
```

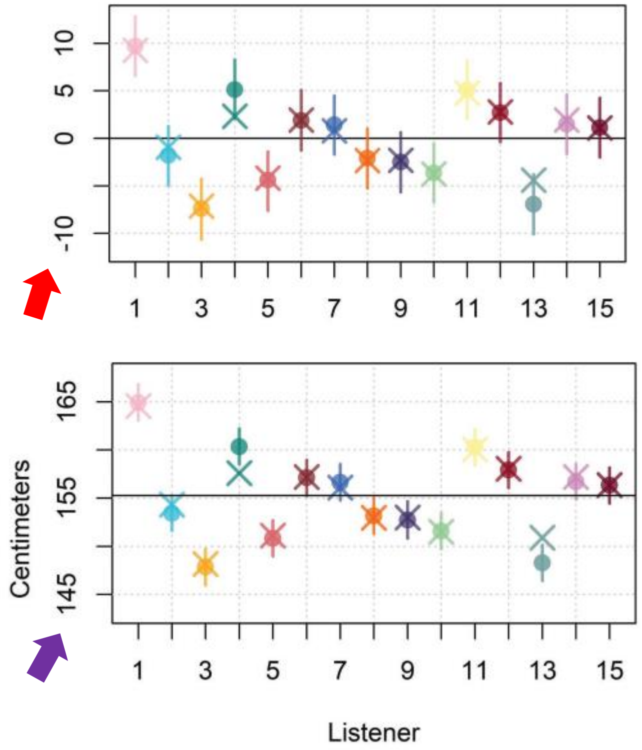
```
short_hypothesis(model_sum_coding,
                 c("Intercept = 0",           # overall mean
                   "Intercept + A1 = 0",      # adult mean
                   "Intercept - A1 = 0"))     # child mean
##     Estimate Est.Error  Q2.5 Q97.5           hypothesis
## H1     155.2     1.405 152.4 158.0      (Intercept) = 0
## H2     163.9     1.465 161.1 166.8 (Intercept+A1) = 0
## H3     146.5     1.435 143.7 149.3 (Intercept-A1) = 0
```

Using Hypothesis for Random Effects

```
short_hypothesis(model_sum_coding, "Intercept = 0")
##     Estimate Est.Error  Q2.5 Q97.5       hypothesis
## H1     155.2     1.405 152.4   158 (Intercept) = 0
```

```
short_hypothesis(model_sum_coding, "Intercept = 0",
                        scope = "ranef",group="L")[1:5,]
##     Estimate Est.Error    Q2.5  Q97.5      hypothesis group
## H1     9.690     1.585   6.603 12.832 (Intercept) = 0     1
## H2    -1.792     1.597  -4.941  1.297 (Intercept) = 0     2
## H3    -7.376     1.602 -10.636 -4.255 (Intercept) = 0     3
## H4     5.121     1.566   2.104  8.283 (Intercept) = 0     4
## H5    -4.393     1.570  -7.598 -1.383 (Intercept) = 0     5
```

```
short_hypothesis(model_sum_coding, "Intercept = 0",
                        scope = "coef",group="L")[1:5,]
##     Estimate Est.Error  Q2.5 Q97.5      hypothesis group
## H1     164.9    0.9516 163.1 166.8 (Intercept) = 0     1
## H2     153.4    0.9438 151.6 155.3 (Intercept) = 0     2
## H3     147.8    0.9585 146.0 149.8 (Intercept) = 0     3
## H4     160.3    0.9556 158.5 162.2 (Intercept) = 0     4
## H5     150.8    0.9475 149.0 152.7 (Intercept) = 0     5
```



Robustness

- Ferrari vs. Honda, which is ‘better’?

- A Honda is more ‘robust’: reliable and useful in a wider range of situations.

- Distributional robustness: Assume less, assume more ‘realistic’ things.

The (non-standardized) t-distribution

- Three parameters: Mean, scale, and nu (ν, degrees of freedom) parameter.

$$\mathrm{t}\left(\nu, \mu_{[i]}, \sigma\right)$$

- The variance of the t distribution is determined by ν.

$$\sigma = s \cdot \sqrt{\nu / (\nu - 2)}$$

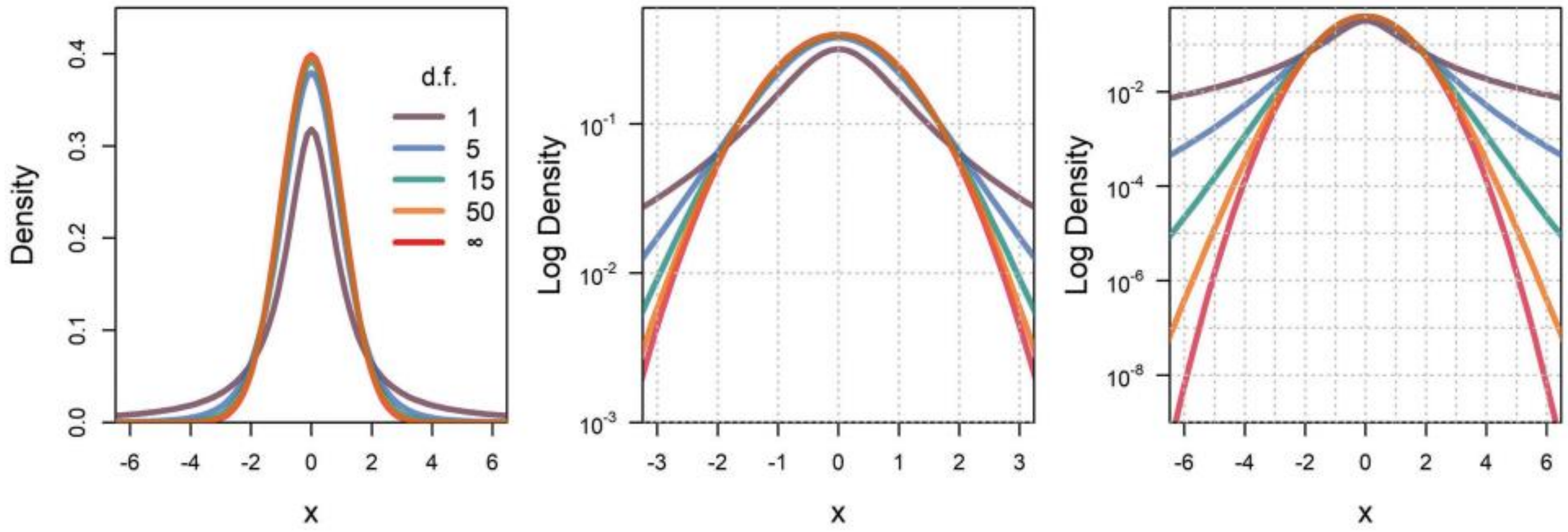
$$\sigma^2 = s^2 \cdot \nu / (\nu - 2)$$

- The 'scale' parameter scales up the variance (which is entirely determined by nu).

The (non-standardized) t-distribution

- Nu determines the ‘normalness’ of the distribution.

- A smaller nu results in a pointier distribution with ‘fat tails’, i.e. more outliers.



Outliers

```
resids = residuals (model_sum_coding)[,1]
```

```
range (scale(resids))
## [1] -4.144  3.471
```

```
head (sort(scale(resids)))
## [1] -4.144 -3.846 -3.785 -3.679 -3.668 -3.613
```

```
mu = mean(resids)
sigma = sd(resids)

# probability of value smaller than smallest outlier
pnorm (min (resids),mu,sigma)
## [1] 1.704e-05
```

```
# sample size before outlier this big expected
1/pnorm (min (resids),mu,sigma)
## [1] 58680
```

Outliers

```r
# get maximum likelihood estimates of t parameters
# the 'lower' bounds are for the sd and df respectively
tparams = MASS::fitdistr (resids, 't', lower = c(0,1))
# check out mean, scale and nu. bottom row is standard errors
tparams
##       m         s         df
##    0.2902    7.1651    7.5753
```

```r
m = tparams[[1]][1]
s = tparams[[1]][2]
df = tparams[[1]][3]

# probability of value smaller than smallest outlier
bmmb::ptns (min (resids),m, s, df)
##         m
## 0.0007384

# sample size before outlier thie big expected
1/bmmb::ptns (min (resids),m, s, df)
##    m
## 1354
```

Model with t-Distributed Errors

- Can be basically the same as a model with normally-distributed errors.
- The nu (ν) parameter can be estimated from the data.

$$\text{height}_{[i]} \sim \text{t}\left(\nu, \mu_{[i]}, \sigma\right)$$
$$\mu_{[i]} = \text{Intercept} + A + L_{\left[\text{L}[i]\right]} + S_{\left[\text{S}[i]\right]}$$

Priors:

$$L_{[\bullet]} \sim \text{N}(0, \sigma_L)$$
$$S_{[\bullet]} \sim \text{N}(0, \sigma_S)$$

$$\text{Intercept} \sim \text{t}(3, 156, 12)$$
$$A \sim \text{t}(3, 0, 12)$$
$$\sigma \sim \text{t}(3, 0, 12)$$
$$\sigma_L \sim \text{t}(3, 0, 12)$$
$$\sigma_S \sim \text{t}(3, 0, 12)$$
$$\nu \sim \text{gamma}(2, 0.1)$$

$$\text{height}_{[i]} \sim \text{N}\left(\mu_{[i]}, \sigma\right)$$
$$\mu_{[i]} = \text{Intercept} + A + L_{\left[\text{L}[i]\right]} + S_{\left[\text{S}[i]\right]}$$

Priors:

$$L_{[\bullet]} \sim \text{N}(0, \sigma_L)$$
$$S_{[\bullet]} \sim \text{N}(0, \sigma_S)$$

$$\text{Intercept} \sim \text{N}(156, 12)$$
$$A \sim \text{N}(0, 12)$$
$$\sigma \sim \text{N}(0, 12)$$
$$\sigma_L \sim \text{N}(0, 12)$$
$$\sigma_S \sim \text{N}(0, 12)$$

Model with t-Distributed Errors

$$\text{height}_{[i]} \sim \text{t}\left(\nu, \mu_{[i]}, \sigma\right)$$
$$\mu_{[i]} = \text{Intercept} + A + L_{\left[\mathsf{L}_{[i]}\right]} + S_{\left[\mathsf{S}_{[i]}\right]}$$

Priors:

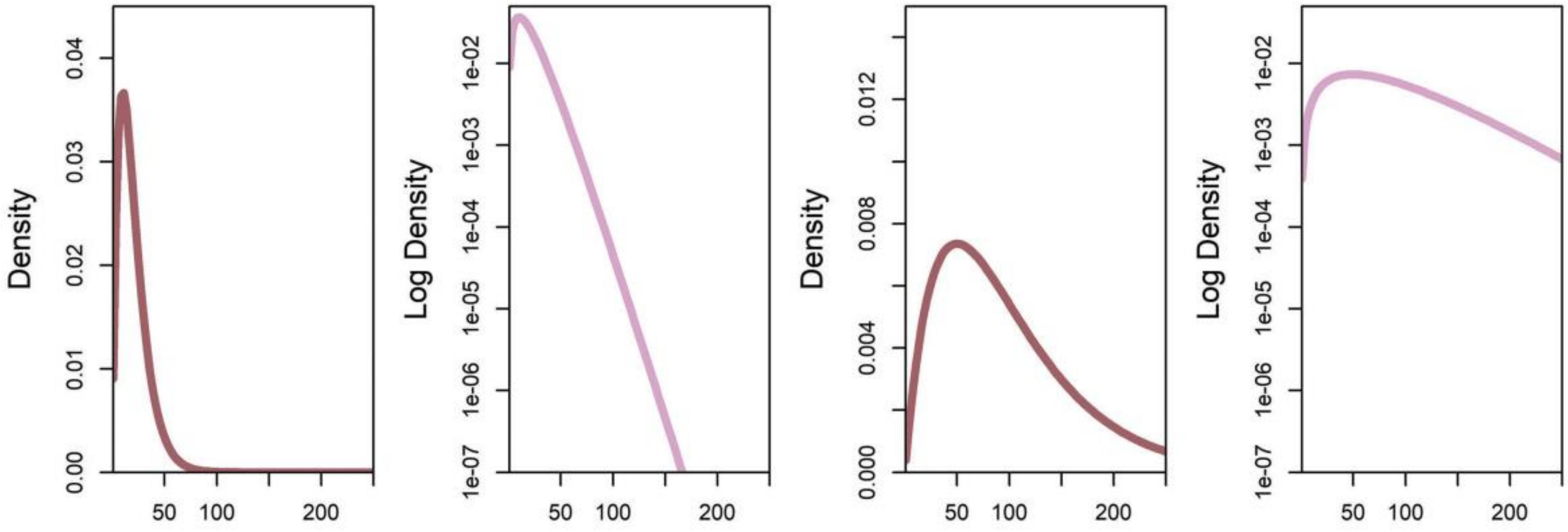
$$L_{[\bullet]} \sim \text{N}\left(0, \sigma_L\right)$$
$$S_{[\bullet]} \sim \text{N}\left(0, \sigma_S\right)$$

$$\text{Intercept} \sim \text{t}(3, 156, 12)$$
$$A \sim \text{t}(3, 0, 12)$$
$$\sigma \sim \text{t}(3, 0, 12)$$
$$\sigma_L \sim \text{t}(3, 0, 12)$$
$$\sigma_S \sim \text{t}(3, 0, 12)$$
$$\nu \sim \text{gamma}(2, 0.1)$$

```r
# Fit the model yourself
options (contrasts = c('contr.sum','contr.sum'))
model_sum_coding_t =  brms::brm (
  height ~ A + (1|L) + (1|S), data = notmen, chains = 4,
  cores = 4, warmup = 1000, iter = 3500, thin = 2, family="student",
  prior = c(brms::set_prior("student_t(3, 156, 12)", class = "Intercept"),
    brms::set_prior("student_t(3, 0, 12)", class = "b"),
    brms::set_prior("student_t(3, 0, 12)", class = "sd"),
    brms::set_prior("gamma(2, 0.1)", class = "nu"),
    brms::set_prior("student_t(3, 0, 12)", class = "sigma")))

# Or download it from the GitHub page:
model_sum_coding_t = bmmb::get_model ('5_model_sum_coding_t.RDS')
```

Strange Priors

- Look at the shapes of the densities, see if these 'make sense'.
- Don't worry so much about the underlying math *per se*.



Inspecting the Model

```
# inspect model
bmmb::short_summary (model_sum_coding_t)
## Formula:  height ~ A + (1 | L) + (1 | S)
##
## Group-Level Effects:
## ~L (Number of levels: 15)
##                Estimate Est.Error l-95% CI u-95% CI
## sd(Intercept)      5.08      1.12     3.46     7.81
##
## ~S (Number of levels: 94)
##                Estimate Est.Error l-95% CI u-95% CI
## sd(Intercept)      3.36      0.42      2.6     4.21
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI
## Intercept   155.59      1.38   152.85   158.31
## A1            8.44      0.34     7.78     9.12
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI
## sigma     7.23      0.27     6.71     7.76
## nu        6.90      1.52     4.72    10.40
```

$$\text{height}_{[i]} \sim \mathrm{t}\left(\nu, \mu_{[i]}, \sigma\right)$$

$$\mu_{[i]} = \text{Intercept} + A + L_{\left[\mathsf{L}_{[i]}\right]} + S_{\left[\mathsf{S}_{[i]}\right]}$$

Priors:

$$L_{[\bullet]} \sim \mathrm{N}(0, \sigma_L)$$

$$S_{[\bullet]} \sim \mathrm{N}(0, \sigma_S)$$

$$\text{Intercept} \sim \mathrm{t}(3, 156, 12)$$

$$A \sim \mathrm{t}(3, 0, 12)$$

$$\sigma \sim \mathrm{t}(3, 0, 12)$$

$$\sigma_L \sim \mathrm{t}(3, 0, 12)$$

$$\sigma_S \sim \mathrm{t}(3, 0, 12)$$

$$\nu \sim \text{gamma}(2, 0.1)$$

Comparing Models

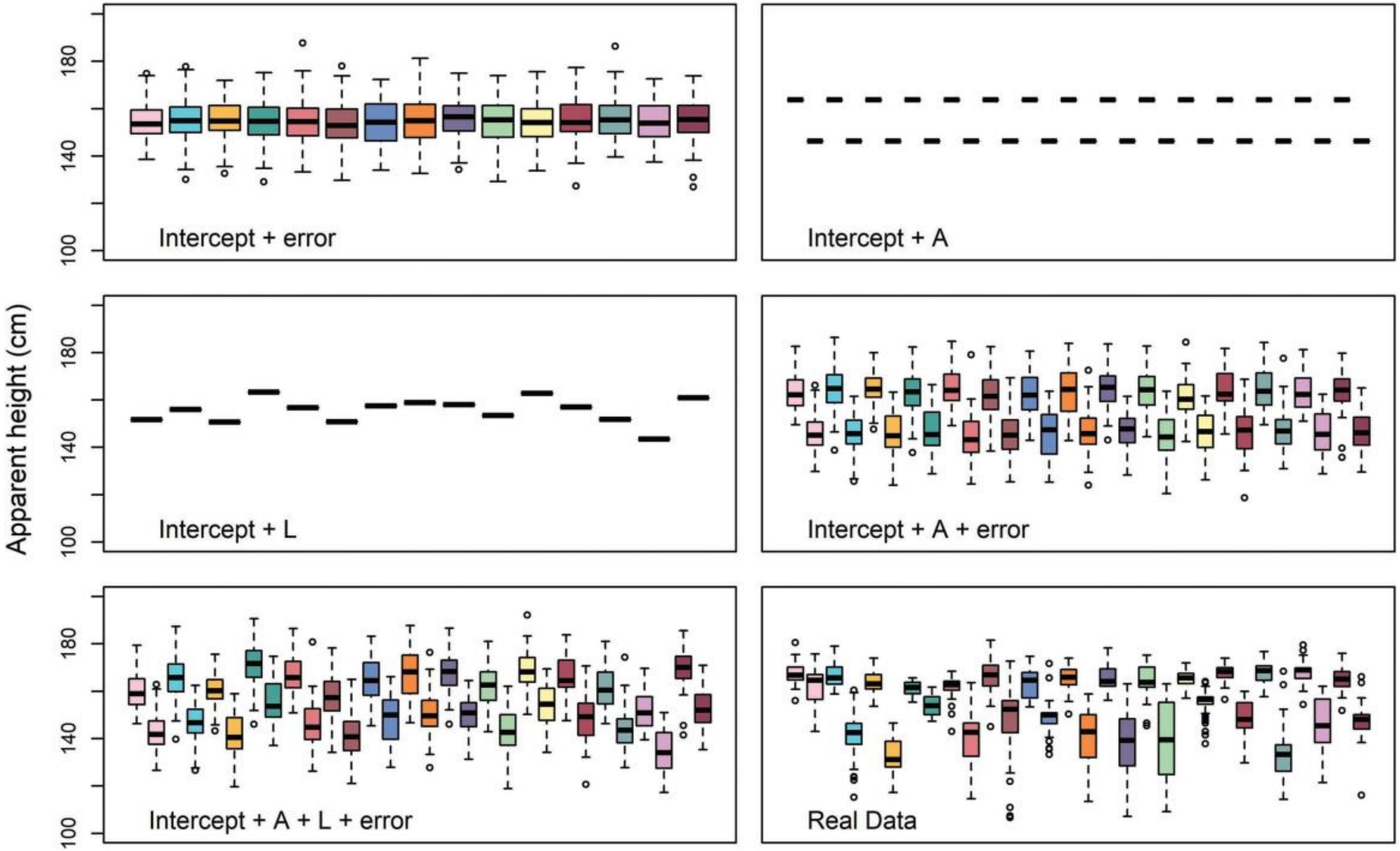
```
fixef (model_sum_coding)
##              Estimate Est.Error    Q2.5   Q97.5
## Intercept    155.218    1.4050 152.434 157.965
## A1             8.711    0.3602   8.003   9.416
fixef (model_sum_coding_t)
##              Estimate Est.Error   Q2.5   Q97.5
## Intercept    155.588    1.3803 152.85 158.311
## A1             8.443    0.3386   7.78   9.119
```

Simulating Two-Group Data

```
n_listeners = 15
n_speakers = 94 # must be even!

# don't run this line if you want a new simulated dataset.
set.seed(1)

# this is the value of our intercept
Intercept = 155

# this is a vector of adultness fixed effects
A_ = c(8.7, -8.7)

# this is a vector indicating the adultness group
A = rep(1:2, (n_listeners*n_speakers/2))

# this is a vector of 15 listener effects
L_ = rnorm (n_listeners, 0, 5.2)
```

```
# this is a vector indicating the listener
L = rep (1:n_listeners, each = n_speakers)

# this is a vector of 94 speaker effects
S_ = rnorm (n_speakers, 0, 3.6)

# this is a vector indicating the speaker
S = rep (1:n_speakers, each = n_listeners)

# this vector contains the error
epsilon = rnorm (n_speakers*n_listeners, 0, 8.6)

# the sum of the above components equals our observations
height_rep = Intercept + A_[A] + L_[L] + S_[S] + epsilon
```

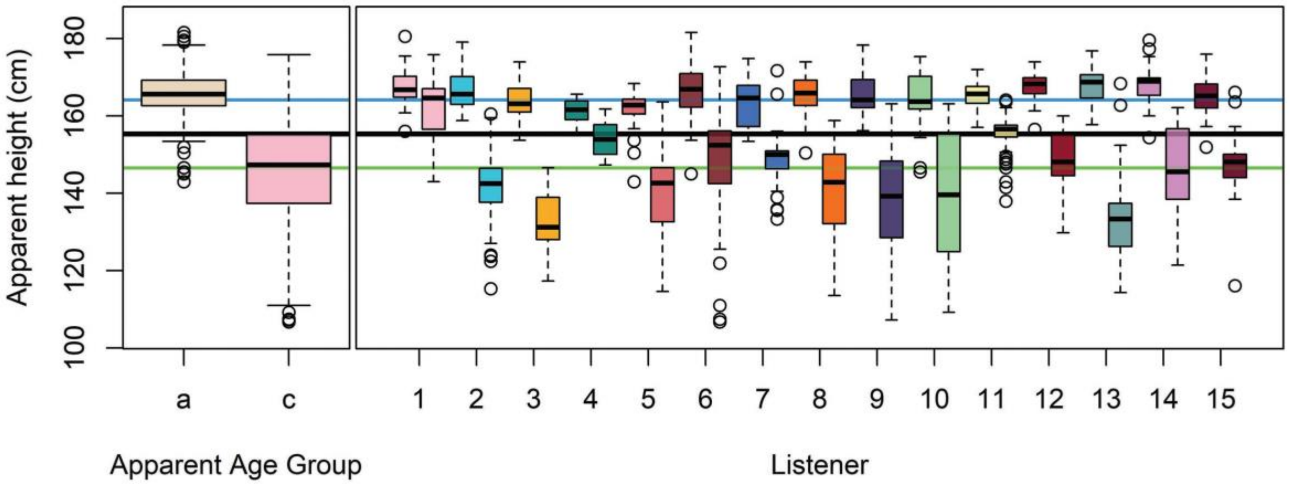
Simulating Two-Group Data



Answering Our Research Questions

(Q1) How tall do speakers perceived as adult females sound?
(Q2) How tall do speakers perceived as children sound?
(Q3) What is the difference in apparent height associated with the perception of adultness?

The overall mean apparent height across all speakers was 156 cm (s.d. = 1.38, 95% C.I = [152.85, 158.31]). We found a difference of 17 cm (s.d. = 0.68, 95% C.I = [15.56, 18.24]) in apparent height associated with the perception of adultness in speakers. The standard deviation of the listener and speaker effects were 5.1 cm (s.d = 1.1, 95% CI = [3.5, 7.8]) and 3.4 cm (s.d = 0.4, 95% CI = [2.6, 4.2]) respectively. Overall, results indicate a reliable difference in apparent speaker height due to apparent age, which is larger than the expected random variation in apparent height judgments due to variation between speakers and listeners.



Traditionalist’s Corner: lmer

```
library (lme4)
lmer_model = lmer (height ~ A + (1|L) + (1|S), data = notmen)

summary (lmer_model)
## Linear mixed model fit by REML ['lmerMod']
## Formula: height ~ A + (1 | L) + (1 | S)
##    Data: notmen
##
## REML criterion at convergence: 10161
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  S        (Intercept) 12.4     3.52
##  L        (Intercept) 22.9     4.79
##  Residual             73.5     8.57
## Number of obs: 1401, groups:  S, 94; L, 15
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  155.193      1.310   118.5
## A1             8.729      0.312    27.9
##
## Correlation of Fixed Effects:
##    (Intr)
## A1 0.046
```

```
fixef (model_sum_coding)
##           Estimate Est.Error    Q2.5   Q97.5
## Intercept  155.218    1.4050 152.434 157.965
## A1           8.711    0.3602   8.003   9.416
```

Exercises

Use the data in ‘exp_ex’ to do one of the following. You may also use your own data to answer a related question. In any case, describe the model, present and explain the results, and include at least two figures.

1. Easy: Analyze the (pre-ft) model that’s exactly like model_sum_coding_t, except using the data in exp_ex (bmmb::get_model("5_model_sum_coding_t_ex.RDS")).

2. Medium: Fit a model like model_sum_coding_t, but comparing any two groups across resonance levels.

3. Hard: Fit two models like model_sum_coding_t, but compare any two groups across resonance levels. Compare results across models to think about group differences.