Course Business

- Midterm assignment due on CourseWeb on October 24th at 1:30 PM– 2 weeks from today
  - Unsure if an article is suitable? Can run it by me

- Add-on packages to install for today:
  - emmeans (may have gotten this last week)
  - afex

- New dataset on CourseWeb

- Next 3 weeks:
  - This week: Finish categorical predictors
  - Next week: Categorical outcomes
  - 2 weeks: Discuss midterm projects
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Your research team is modeling the effect of out-of-class study on college achievement. You recruit a sample of 300 students. Each term over their college career, the students report the number of hours they spent studying for their final exam week that term as well as their GPA for that term. Your first model is:

```
model1 <- lmer(GPA ~ 1 + HoursOfStudy + (1|Subject), data=x)
```

But, your team thinks `HoursOfStudy` may show a stronger effect on GPA for some students than others (i.e., some people make better use of study time). How can your new model reflect this?

```
model2 <- lmer(GPA ~ 1 + HoursOfStudy + ???, data=x)
```

- Albert says, “We should use `(1|Subject) + (1|HoursOfStudy)` because we’re adding `HoursOfStudy` as another random effect.”
- Betsy says, “We can use `(1+HoursOfStudy|Subject)` to make both the intercept and slope different for each subject.”
- Carlos says, “We want to capture both subject differences and `HoursOfStudy` differences, so it’s `(1|Subject+HoursOfStudy)`”
- Dipika says, “`HoursOfStudy` is a between-subjects variable, so this question makes no sense.”
Distributed Practice!

- Your research team is modeling the effect of out-of-class study on college achievement. You recruit a sample of 300 students. Each term over their college career, the students report the number of hours they spent studying for their final exam week that term as well as their GPA for that term. Your first model is:
  \[
  \text{model1} \leftarrow \text{lmer}(\text{GPA} \sim 1 + \text{HoursOfStudy} + (1|\text{Subject}), \text{data}=\text{x})
  \]

But, your team thinks \text{HoursOfStudy} may show a stronger effect on \text{GPA} for some students than others (i.e., some people make better use of study time). How can your new model reflect this?
  \[
  \text{model2} \leftarrow \text{lmer}(\text{GPA} \sim 1 + \text{HoursOfStudy} + \text{???}, \text{data}=\text{x})
  \]

- Albert says, “We should use \text{(1|Subject)} + \text{(1|HoursOfStudy)} because we’re adding \text{HoursOfStudy} as another random effect.”
- Betsy says, “We can use \text{(1+HoursOfStudy|Subject)} to make both the intercept and slope different for each subject.”
- Carlos says, “We want to capture both subject differences and \text{HoursOfStudy} differences, so it’s \text{(1|Subject+HoursOfStudy)}”
- Dipika says, “\text{HoursOfStudy} is a between-subjects variable, so this question makes no sense.”
Distributed Practice!

- Alyssa is a chemistry professor experimenting with online quizzes. Half of her students take a quiz on the Web, and half take it on paper. In Alyssa’s R dataframe (called quizzes), that variable looks like this:

![Quiz Type](image)

- Alyssa is interested in:
  a. The *overall average* quiz score, and
  b. The effect of *Web* quizzes relative to paper quizzes

- Given the eventual model:
  - `model2<-lmer(Score ~ 1+QuizType + (1|Year), data=quizzes)`

- What R code will create contrasts for QuizType that will tell her both (a) and (b) in one model?
Distributed Practice!

- Alyssa is a chemistry professor experimenting with online quizzes. Half of her students take a quiz on the Web, and half take it on paper. In Alyssa’s R dataframe (called `quizzes`), that variable looks like this:

```
QuizType
Paper:50
Web   :50
```

Alyssa is interested in:

a. The overall average quiz score, and
b. The effect of Web quizzes relative to paper quizzes

- Given the eventual model:
  - `model2<-lmer(Score ~ 1+QuizType + (1|Year), data=quizzes)`

- What R code will create contrasts for QuizType that will tell her both (a) and (b) in one model?
  - `contrasts(quizzes$QuizType) <-`
Distributed Practice!

- Alyssa is a chemistry professor experimenting with online quizzes. Half of her students take a quiz on the Web, and half take it on paper. In Alyssa’s R dataframe (called quizzes), that variable looks like this:

- Alyssa is interested in:
  a. The **overall average** quiz score, and
  b. The effect of **Web** quizzes relative to paper quizzes

- Given the eventual model:
  - `model2<-lmer(Score ~ 1+QuizType + (1|Year), data=quizzes)`

- What R code will create contrasts for QuizType that will tell her both (a) and (b) in one model?
  - `contrasts(quizzes$QuizType) <- c(???, ???)`
Distributed Practice!

- Alyssa is a chemistry professor experimenting with online quizzes. Half of her students take a quiz on the Web, and half take it on paper. In Alyssa’s R dataframe (called quizzes), that variable looks like this:

<table>
<thead>
<tr>
<th>QuizType</th>
<th>Paper:50</th>
<th>Web :50</th>
</tr>
</thead>
</table>

Alyssa is interested in:
- a. The *overall average* quiz score, and
- b. The effect of *Web* quizzes relative to paper quizzes

- Given the eventual model:
  ```r
  model2<-lmer(Score ~ 1+QuizType + (1|Year), data=quizzes)
  ```
- What R code will create contrasts for QuizType that will tell her both (a) and (b) in one model?
  ```r
  contrasts(quizzes$QuizType) <- c(-0.5, 0.5)
  ```
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Alice in Um-derland (Fraundorf & Watson, 2011)

- disfluency.csv on CourseWeb

- How do disfluencies in speech (e.g., “uh”, “um”) change listener comprehension?

- Disfluencies more common with more difficult material, so might lead listeners to pay more attention

- But: Any benefit might be confounded with just having more time to process
  - Control: Speaker coughing, matched in duration
*Alice in Um-derland* (Fraundorf & Watson, 2011)

- `disfluency.csv` on CourseWeb

- Each participant hears stories based on *Alice in Wonderland*
  - Later, test recall of each chapter – scored from 0 to 10

- Conditions:
  - Some chapters told fluently (control)
  - Some chapters contain speech fillers
  - Some have coughs matched in duration to the fillers
  - Each subject hears some chapters in all 3 conditions
  - Each chapter heard in all 3 conditions across subjects
Alice in Um-derland (Fraundorf & Watson, 2011)

- Average memory score in each condition:
  - `tapply(disfluency$MemoryScore, disfluency$InterruptionType, mean)`
  - “Take MemoryScore, separate it out by InterruptionType, and give me the mean”

<table>
<thead>
<tr>
<th>Condition</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>6.05625</td>
</tr>
<tr>
<td>Cough</td>
<td>6.28750</td>
</tr>
<tr>
<td>Filler</td>
<td>7.66250</td>
</tr>
</tbody>
</table>
Factors with More Than 2 Levels

• How can we code a variable with three categories?
  • Control = 0, Cough = 1, Filler = 2?
  • Let’s imagine the equations:

<table>
<thead>
<tr>
<th>Category</th>
<th>Score equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>$\text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType}$</td>
</tr>
<tr>
<td>Cough</td>
<td>$\text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType}$</td>
</tr>
<tr>
<td>Filler</td>
<td>$\text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType}$</td>
</tr>
</tbody>
</table>
**Factors with More Than 2 Levels**

- How can we code a variable with three categories?
  - Control = 0, Cough = 1, Filler = 2?
- Let’s imagine the equations:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Score Equation</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>( \gamma_{000} + \gamma_{100} \times 0 )</td>
<td>0</td>
</tr>
<tr>
<td>Cough</td>
<td>( \gamma_{000} + \gamma_{100} \times 1 )</td>
<td>1</td>
</tr>
<tr>
<td>Filler</td>
<td>( \gamma_{000} + \gamma_{100} \times 2 )</td>
<td>2</td>
</tr>
</tbody>
</table>
Factors with More Than 2 Levels

• How can we code a variable with three categories?
  • Control = 0, Cough = 1, Filler = 2?
  • Let’s imagine the equations:

\[
\text{Score} = \gamma_{000} + \gamma_{100} \]

Control \hspace{1cm} \text{Score} = \gamma_{000} + \gamma_{100} \hspace{1cm} 0

\text{Cough} \hspace{1cm} \text{Score} = \gamma_{000} + \gamma_{100} \hspace{1cm} 1

\text{Filler} \hspace{1cm} \text{Score} = \gamma_{000} + \gamma_{100} \hspace{1cm} 2

• This coding scheme assumes Control & Cough differ by the same amount as Cough & Filler
  • Probably not true. Not a safe assumption
Factors with More Than 2 Levels

- To actually represent three levels, we need two sets of codes.
  - “InterruptionType1” and “InterruptionType2”
- If a factor has 3 levels, R automatically creates multiple sets of codes.
  - `contrasts(disfluency$InterruptionType)`

One set of codes (“InterruptionType1”). 1 for Cough, 0 for everything else.

Another, different set of codes (“InterruptionType2”). 1 for Filler, 0 for everything else.
Factors with More Than 2 Levels

• Annoying R “feature”: If you take a subset that includes only some levels…
  • disfluency.NoCoughs <- subset(disfluency, InterruptionType != 'Cough')
  • ...R still remembers all of the possible levels...

• Solution: Re-make into a factor with factor():
  • disfluency.NoCoughs$InterruptionType <- factor(disfluency.NoCoughs$InterruptionType)
Week 7: Coding Predictors II

- Factors with More than 2 Levels
  - Treatment Coding
  - Problem of Multiple Comparisons
  - Orthogonal Contrasts
    - Example
    - Implementation
    - Definition
    - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
**Treatment Coding With >2 Levels**

- The two sets of codes are 2 separate variables in the underlying regression equation:

  \[
  \text{Control: Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2}
  \]

  \[
  \text{Cough: Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2}
  \]

  \[
  \text{Filler: Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2}
  \]
Treatment Coding With >2 Levels

- The two sets of codes are 2 separate variables in the underlying regression equation:

\[
\text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2}
\]

<table>
<thead>
<tr>
<th>Control</th>
<th>[ \text{Score} = \gamma_{000} + \gamma_{100} \times 0 + \gamma_{200} \times 0 ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cough</td>
<td>[ \text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2} ]</td>
</tr>
<tr>
<td>Filler</td>
<td>[ \text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2} ]</td>
</tr>
</tbody>
</table>
Treatment Coding With >2 Levels

• The two sets of codes are 2 separate variables in the underlying regression equation:

Control

\[ \text{Score} = \gamma_{000} \]

Once again, the intercept is just performance in the baseline level (the one coded with all 0s)

Cough

\[ \text{Score} = \gamma_{000} + \gamma_{100} \times 1 + \gamma_{200} \times 0 \]

Filler

\[ \text{Score} = \gamma_{000} + \gamma_{100} \times \text{InterruptionType1} + \gamma_{200} \times \text{InterruptionType2} \]
Treatment Coding With >2 Levels

- The two sets of codes are 2 separate variables in the underlying regression equation:

\[
\text{Control:} \quad \text{Score} = \gamma_{000}
\]

- Once again, the intercept is just performance in the baseline level (the one coded with all 0s)

\[
\text{Cough:} \quad \text{Score} = \gamma_{000} + \gamma_{100}
\]

- InterruptionType1 = Difference between fluent story & coughs

\[
\text{Filler:} \quad \text{Score} = \gamma_{000} + \gamma_{100} \cdot 0 + \gamma_{200} \cdot 1
\]
Treatment Coding With >2 Levels

- The two sets of codes are 2 separate variables in the underlying regression equation:

Control: \[ \text{Score} = \gamma_{000} \]  
Once again, the intercept is just performance in the baseline level (the one coded with all 0s)

Cough: \[ \text{Score} = \gamma_{000} + \gamma_{100} \]  
InterruptionType1 = Difference between fluent story & coughs

Filler: \[ \text{Score} = \gamma_{000} + \gamma_{200} \]  
InterruptionType2 = Difference between fluent story & fillers
Treatment Coding: Model

- We want to analyze how well each chapter was recalled based on the InterruptionType condition
  - Each subject hears some chapters in each of the 3 conditions
  - Each chapter heard in all 3 conditions across subjects
- Finish the maximal random effects model:
  - `dummycode.Maximal <- lmer(MemoryScore ~ 1 + InterruptionType + (1 + InterruptionType|Subject) + (1 + InterruptionType|Chapter), data=disfluency)`
Treatment Coding: Model

- We want to analyze how well each chapter was recalled based on the InterruptionType condition.
  - Each subject hears some chapters in each of the 3 conditions.
  - Each chapter heard in all 3 conditions across subjects.
- Finish the maximal random effects model:
  - `dummycode.Maximal <- lmer(MemoryScore ~ 1 + InterruptionType + (1 + InterruptionType|Subject) + (1 + InterruptionType|Chapter), data=disfluency)`
Treatment Coding: Model

• We want to analyze how well each chapter was recalled based on the InterruptionType condition
  • Each subject hears some chapters in each of the 3 conditions
  • Each chapter heard in all 3 conditions across subjects
• Finish the maximal random effects model:
  • `dummycode.Maximal <- lmer(MemoryScore ~ 1 + InterruptionType +
                              (1 + InterruptionType|Subject) +
                              (1 + InterruptionType|Chapter),
                              data=disfluency)`
### Treatment Coding: Results

#### Intercept: Baseline score in the control condition

- Scaled residuals:
  - Min: -3.7128
  - 1Q: -0.6202
  - Median: 0.0797
  - Mean: 0.6132
  - Max: 2.5460

#### Cough effect: Numerically greater recall with coughs than control fluent condition ... but only marginally significant

#### Filler effect: Greater recall with speech fillers than control fluent condition
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
The Problem of Multiple Comparisons

• It would be nice to have a *direct* comparison of fillers vs. coughs

• Actually, there are a *lot* of other comparisons we could consider…
  • Fillers vs. coughs
  • Fluent story vs. any kind of interruption
  • Fillers vs. mean performance in this task
  • Cough vs. mean performance in this task

• But there is a problem running *too many* comparisons
Recap of Hypothesis Tests

• Under null hypothesis of no effect, extreme $z$- or $t$-values are improbable
Recap of Hypothesis Tests

- Under null hypothesis of no effect, extreme z- or t-values are *improbable*
- A t-value with < 5% probability: Significant evidence against null hypothesis
- But, *possible* (just unlikely) even if no real effect

![Normal distribution with z-scores and significance levels]
Recap of Hypothesis Tests

• Under null hypothesis of no effect, extreme \( z \)- or \( t \)-values are *improbable*
• A \( t \)-value with < 5% probability: Significant evidence against null hypothesis
• But, *possible* (just unlikely) even if no real effect
  • Could just result from chance! (sampling error)
  • We’d conclude that there is an effect, but it doesn’t really exist
  • *False positive* or *Type I error*
• 5% probability of this happening
• \( \alpha = .05 \)
Problem of Multiple Comparisons

- With $\alpha = .05$ (Type I error rate of 5%), we’d expect 1 in 20 comparisons to be significant just by chance
- Problem if we want to run lots of comparisons!
- Even if aren’t running 20 comparisons, Type I error will be inflated with >1 comparison
- Probability of fully avoiding Type I error:

\[
1 - \alpha \approx 90\%
\]
Problem of Multiple Comparisons

- With $\alpha = .05$ (Type I error rate of 5%), we’d expect 1 in 20 comparisons to be significant just by chance
- Problem if we want to run lots of comparisons!
- Even if aren’t running 20 comparisons, Type I error will be inflated with >1 comparison
- Probability of fully avoiding Type I error:

$$1 - \alpha \quad \text{(95%)} \quad \times \quad 1 - \alpha \quad \text{(95%)} \quad \times \quad \ldots \quad = (1-\alpha)^c$$

- **Familywise** error rate: $\alpha_{FW} = 1 - (1-\alpha)^c$
- Probability of making a Type I error somewhere
WE FOUND NO LINK BETWEEN PURPLE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BROWN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN PINK JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BLUE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TEAL JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN SALMON JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN RED JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TURQUOISE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN MAGENTA JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN YELLOW JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN GREY JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TAN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND A LINK BETWEEN CYAN JELLY BEANS AND ACNE (P < 0.05).
WE FOUND NO LINK BETWEEN GREEN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN MAUVE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BLACK JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN ORANGE JELLY BEANS AND ACNE (P > 0.05).

News
GREEN JELLY BEANS LINKED TO ACNE!
95% CONFIDENCE
ONLY 5% CHANCE OF COINCIDENCE!

Scientists...
The Problem of Multiple Comparisons

- Situation gets even more complicated if we use some of the same data in >1 comparison.

Maybe we underestimated performance in the fluent control condition.

$\alpha = .05$
The Problem of Multiple Comparisons

- Situation gets even more complicated if we use some of the same data in >1 comparison
- If wrong, other similar comparisons have a higher probability of being wrong
- They’re not independent

Maybe we underestimated performance in the fluent control condition

Would affect both the fluent vs coughs and fluent vs fillers comparisons
The Problem of Multiple Comparisons

- How many independent comparisons?
- Our model already showed us that:
  - Recall score is 0.22 points greater for coughs than fluent
  - Recall score is 1.61 points greater for fillers than fluent
  - Difference between coughs and fillers is already known: \(1.61 - 0.22 = 1.39\)
The Problem of Multiple Comparisons

• How many independent comparisons?
• Our model already showed us that:
  • Recall score is 0.22 points greater for coughs than fluent
  • Recall score is 1.61 points greater for fillers than fluent
  • Difference between coughs and fillers is already known: 1.61 – 0.22 = 1.39
• In general: With g levels, g-1 comparisons fully describe data
  • Could position all g conditions on a number line just based on g-1 comparisons
Here Comes Trouble!

- These reasons are why R doesn’t perform all possible comparisons between all levels
  - Not new, independent comparisons … inflated Type I error rate

- In fact, even our current comparisons aren’t totally independent
  - Filler vs control
  - Coughs vs control
  - If we have underestimated (or overestimated) performance in Control condition, both comparisons will be affected
Here Comes Trouble!

- These reasons are why R doesn’t perform all possible comparisons between all levels
  - Not new, independent comparisons … inflated Type I error rate

- So, here’s what let’s do:
  - First, let’s look at how to run comparisons that are *truly* independent
  - And, we can discuss how to control Type I error rate if we need comparisons that aren’t independent
Factors with More than 2 Levels
Treatment Coding
Problem of Multiple Comparisons
Orthogonal Contrasts
  • Example
  • Implementation
  • Definition
  • Practice
Overview of Coding Systems
Additional Tests
  • Testing an Overall Factor
  • Random Slopes
  • Post-Hoc Comparisons
Orthogonal Contrasts

- Another set of comparisons…

**CONTRAST 1**
- FLUENT
- CONTROL
- COUGHS
- FILLERS

Do **coughs** and **fillers** differ?

**CONTRAST 2**

Do interruptions (in general) differ from fluent speech?
Orthogonal Contrasts

- These comparisons are independent ("orthogonal")
- Knowing that interruptions differ from fluent speech doesn’t tell us anything about which type (if any) is better

**CONTRAST 1**
- FLUENT CONTROL
- COUGHS
- FILLERS

Do coughs and fillers differ?

**CONTRAST 2**
- FLUENT CONTROL
- Interruptions (in general)
- Fluent speech

Do interruptions (in general) differ from fluent speech?
Orthogonal Contrasts

• In each contrast, compares the positive-coded level(s) to the negative-coded level(s)
• Ignore the level(s) coded as zero

Contrast 1

Fluent Control
Coughs
Fillers

Contrast 2

Do coughs and fillers differ?

Do interruptions (in general) differ from fluent speech?
Orthogonal Contrasts

• In each contrast, compares the positive-coded level(s) to the negative-coded level(s)
• Ignore the level(s) coded as zero

**CONTRAST 1**

- **FLUENT CONTROL**
  - 0
- **COUGHS**
  - -1/2
- **FILLERS**
  - 1/2

Do coughs and fillers differ?

Centered around mean of 0!

**CONTRAST 2**

Do interruptions (in general) differ from fluent speech?

Ignore the level(s) coded as zero.
Orthogonal Contrasts

- In each contrast, compares the positive-coded level(s) to the negative-coded level(s).
- Ignore the level(s) coded as zero.

**CONTRAST 1**
- FLUENT CONTROL: 0
- COUGHS: -1/2
- FILLERS: 1/2

Do coughs and fillers differ?
Centered around mean of 0!

**CONTRAST 2**
- Coefficients:
  - -2/3
  - 1/3

Do interruptions (in general) differ from fluent speech?
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Let’s Change the Contrasts

• As before, we use `<-` to change the contrasts

• Now, we’re trying to create a matrix of numbers

• Need to stick two columns together with `cbind`:
  
  ```r
  contrasts(disfluency$InterruptionType) <- cbind(c(0, -1/2, 1/2), c(-2/3, 1/3, 1/3))
  ```

  ![Contrast Matrix](image)
Naming the Contrasts

• Default contrast names are just “1” and “2”

\[
\begin{array}{ccc}
\text{[,1]} & \text{[,2]} \\
\text{Control} & 0.0 & -0.6666667 \\
\text{Cough} & -0.5 & 0.3333333 \\
\text{Filler} & 0.5 & 0.3333333 \\
\end{array}
\]

• We can change the names of these columns with `colnames()`
  - `colnames(contrasts(disfluency$InterruptionType)) <- c('FillerVsCough', 'InterruptionVsFluent')`

• Optional—it just makes the output easier to read
Orthogonal Contrasts: Results

- summary(orthogonal.Maximal)

### Fixed effects:

|                      | Estimate | Std. Error | df   | t value | Pr(>|t|) |
|----------------------|----------|------------|------|---------|----------|
| (Intercept)          | 6.66830  | 0.14395    | 31   | 46.374  | < 2e-16  |
| InterruptionTypeFillerVsCough | 1.37715  | 0.19927    | 15.76321 | 6.911  | 3.80e-06 *** |
| InterruptionType InterruptionVsFluent | 0.91769  | 0.09742    | 33.76044 | 9.420  | 5.64e-11 *** |

**Intercept:** Mean score across conditions (because these are centered)

**Contrast 1:** Fillers produce higher recall than coughs
→ *Not* just about the pause in speech

**Contrast 2:** Speech with pauses/interruptions better remembered than totally fluent speech
→ Effect of having more time
Which Model Fits Better?

- `anova(dummycode.Maximal, orthogonal.Maximal)`

- Overall model fit is *identical*!
  - Same total amount of variance explained
  - Changing coding schemes will *not* change the overall fit of the model
  - The same information is available to the model either way
  - We’re just dividing it up differently
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
What Makes Contrasts Orthogonal?
What Makes Contrasts Orthogonal?

- **Criterion 1:** Codes within contrast sum to 0

<table>
<thead>
<tr>
<th></th>
<th>CONTRAST 1</th>
<th>CONTRAST 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLUENT CONTROL</td>
<td>0</td>
<td>-0.67</td>
</tr>
<tr>
<td>COUGHS</td>
<td>-0.5</td>
<td>0.33</td>
</tr>
<tr>
<td>FILLERS</td>
<td>0.5</td>
<td>0.33</td>
</tr>
</tbody>
</table>

\[ \sum = 0 \]
What Makes Contrasts Orthogonal?

- **Criterion 1**: Codes within contrast sum to 0
- **and Criterion 2**:
  - Multiply codes for each level across contrasts
  - Then sum across the levels
  - Needs to sum to 0

Yes, orthogonal!
What Makes Contrasts Orthogonal?

- **Criterion 1:** Codes within contrast sum to 0
- **and** Criterion 2:
  - Multiply codes for each level across contrasts
  - Then sum across the levels
  - Needs to sum to 0

No, not orthogonal
What Makes Contrasts Orthogonal?

- **Criterion 1:** Codes within contrast sum to 0
- **and Criterion 2:**
  - Multiply codes for each level across contrasts
  - Then sum across the levels
  - Needs to sum to 0

Treatment codes are not orthogonal!
What Makes Contrasts Orthogonal?

- Multiply codes for each level across contrasts
- Then sum across the levels
- Needs to sum to 0
- Codes within a contrast must also sum to 0

- Interpretation given earlier…
  - Each contrast compares the + and – levels
  - And ignores the 0-coded levels
- …is valid only if each pair of contrasts is orthogonal
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Orthogonal Contrasts Practice

• Hitomi is a clinical psychologist investigated the effectiveness of talk therapy. She examines the severity of depressive symptoms in three groups: waitlisted controls, people receiving medication, and people receiving medication and talk therapy. In her data frame, depression, this TreatmentType variable looks like:

- Control: 100
- Medication: 100
- MedicationAndTalk: 100

• Hitomi wants to compare:
  • The two groups receiving any treatment vs. controls
  • Medication + talk therapy vs. medication only
• Create some R code to set these contrasts:
Hitomi is a clinical psychologist investigated the effectiveness of talk therapy. She examines the severity of depressive symptoms in three groups: waitlisted controls, people receiving medication, and people receiving medication \textit{and} talk therapy. In her dataframe, \textit{depression}, this \textit{TreatmentType} variable looks like:

<table>
<thead>
<tr>
<th>TreatmentType</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>100</td>
</tr>
<tr>
<td>Medication</td>
<td>100</td>
</tr>
<tr>
<td>MedicationAndTalk</td>
<td>100</td>
</tr>
</tbody>
</table>

Hitomi wants to compare:
- The two groups receiving any treatment vs. controls
- Medication + talk therapy vs. medication only

Create some R code to set these contrasts:
```r
c(contrasts(depression$TreatmentType) <-
cbind(c(-2/3, 1/3, 1/3), c(0, -0.5, 0.5)))
```
Orthogonal Contrasts Practice

- For each set of contrasts, decide whether it IS orthogonal or IS not orthogonal.

<table>
<thead>
<tr>
<th>[,1]</th>
<th>[,2]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Divorced</td>
<td>1</td>
</tr>
<tr>
<td>Married</td>
<td>0</td>
</tr>
<tr>
<td>Single</td>
<td>0</td>
</tr>
</tbody>
</table>

\textbf{(a)}

\begin{verbatim}
> contrasts(L2Vocab$RelationType)
     [,1]    [,2]
SimilarMeaning    0.3333333  0.5
SimilarSound      0.3333333 -0.5
Unrelated         -0.6666667  0.0
\end{verbatim}

\textbf{(b)}

<table>
<thead>
<tr>
<th>[,1]</th>
<th>[,2]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Young</td>
<td>-1</td>
</tr>
<tr>
<td>MiddleAge</td>
<td>1</td>
</tr>
<tr>
<td>Older</td>
<td>0</td>
</tr>
</tbody>
</table>

\textbf{(c)}

<table>
<thead>
<tr>
<th>Northeast</th>
<th>South</th>
<th>West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midwest</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Northeast</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>South</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>West</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\textbf{(d)}
Orthogonal Contrasts Practice

- For each set of contrasts, decide whether it IS orthogonal or IS not orthogonal.

(a) NOT orthogonal

(b) orthogonal

(c) orthogonal

(d) NOT orthogonal
Orthogonal Contrasts Practice

• Zebulon is a health psychologist examining the (potentially) protective effect of physical exercise on cognition in older adults. Measures of working memory are obtained from older adults from each of three groups: a control group that just stretches, a group that performs low-intensity exercises, and a group that performs moderate-intensity exercises.

• What is a reasonable set of orthogonal comparisons that Zebulon might make?
Orthogonal Contrasts Practice

• Zebulon is a health psychologist examining the (potentially) protective effect of physical exercise on cognition in older adults. Measures of working memory are obtained from older adults from each of three groups: a control group that just stretches, a group that performs low-intensity exercises, and a group that performs moderate-intensity exercises.

• What is a reasonable set of orthogonal comparisons that Zebulon might make?
  • One plausible set of comparisons is (a) control vs. any type of exercise, and (b) moderate- vs. low-intensity exercise. (Other answers may be possible.)
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Treatment / Dummy Coding

- Coding:
  - Baseline level always coded as 0
  - Each other level is coded as 1 in one of the $g-1$ contrasts

- Treatment coding is R’s default, but we might want to set this back (if we switched to something else but we want treatment coding back)

- Shortcut to this:
  - `contrasts(disfluency$InterruptionTime) <- contr.treatment(n=3)`
  - `n=3` because there are 3 groups
Treatment / Dummy Coding

• Coding:
  • Baseline level always coded as 0
  • Each other level is coded as 1 in one of the \( g - 1 \) contrasts
  • `contr.treatment()` in R

• Interpretation:
  • Each contrast compares one condition to the baseline

• Examples:
  • Compare each of 2 different interventions (talk therapy & medication) to control w/ no intervention
  • In reading time, compare each of helpful and unhelpful context to version with no context
Orthogonal Contrasts

• Coding:
  • Each contrast sums to 0
  • Product of weights across contrasts also sums to 0

• Interpretation:
  • Within each contrast, positively coded levels are compared to negative ones

• Examples:
  • Second language learning. Contrast 1 compares words related to 1st language with unrelated words. Contrast 2 compares two types of relations.
Helmert Contrasts

- **Coding:**
  - A subtype of orthogonal contrast
  - `contr.helmert()` in R

- **Interpretation:**
  - Each level is compared to the mean of all previous ones

- **Use when categories are ordered:**
  - Changes in time / across phases of an experiment
  - “Easy,” “medium,” or “hard” items
  - Control, mild anxiety, severe anxiety
Orthogonal Polynomials

• Coding:
  • A subtype of orthogonal contrast
  • `contr.poly()` in R

• Interpretation:
  • Is there a linear effect across levels?
  • Is there a quadratic effect across levels?
  • + cubic, quartic, etc…

• Use when categories are ordered and you’re interested in the form of the relation
  • Linear increase from low->medium->high arousal, or is medium arousal the best?
Sum Coding

• Coding:
  • Code one level as -0.5 (or as -1)
  • Each other level is coded as 0.5 (or 1) in one of the g-1 contrasts
  • \texttt{contr.sum()} in R

• Interpretation:
  • Each contrast compares one condition to the \textit{overall mean}

• Used when we don’t want to compare specific conditions & don’t have a clear baseline:
  • Compare students with various majors to the mean across majors
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice

- Overview of Coding Systems

- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Testing an Overall Factor

- So far, we have compared specific categories (levels) of a factor

- Sometimes, when we have >2 levels, we also just to want to ask if the factor matters at all
  - “Do interruptions affect speech comprehension?”
  - “Are there race/ethnicity differences in feelings of belonging in high school?”
  - “Do different persuasion techniques result in different consumer purchasing behavior?”
  - “Do people with different majors differ in the ease of transition to college?”

- Often asked in ANOVA / experimental contexts
Testing an Overall Factor

- `anova(orthogonal.Maximal)`
  - Requires `lmerTest` to be loaded

Notes:
- Since this tests the overall contribution of the factor, not affected by how you code the individual levels
- For a 2-level factor, identical to the main-effect test you get with effects-coding
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
Random Slopes

• Last week, we saw you could simplify the random-effects structure by omitting random correlations with
  • e.g., correlation between random slope & random intercept
  • \((1 + \text{InterruptionType} || \text{Subject})\)
Random Slopes

• For a factor with >2 levels, correlations between the contrasts will still be included
  • `lmer(MemoryScore ~ 1 + InterruptionType + (1 + InterruptionType||Subject) + (1 + InterruptionType||Chapter)`

• To completely eliminate them, load package `afex` and use `lmer_alt()` instead of `lmer()`
  • `lmer_alt(MemoryScore ~ 1 + InterruptionType + (1 + InterruptionType||Subject) + (1 + InterruptionType||Chapter)`

<table>
<thead>
<tr>
<th>Subject</th>
<th>(Intercept)</th>
<th>Subject.1 re1.InterruptionTypeFillerVsCough</th>
<th>Subject.2 re1.InterruptionTypeInterruptionVsFluent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.40713</td>
<td>0.08204</td>
<td>0.00000</td>
</tr>
<tr>
<td></td>
<td>0.6381</td>
<td>0.2864</td>
<td>0.00000</td>
</tr>
</tbody>
</table>
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons
**Post-hoc Comparisons**

- Last week we looked at *aphasia.csv*:
  - Response times (RT) in a sentence verification task
  - Effect of `SubjectType` (aphasia vs. control)
  - Effect of `SentenceType` (active vs. passive)
  - And their interaction

- Maximal model was:
  ```r
  Model.Maximal <- lmer(RT ~ 1 + SentenceType * SubjectType + (1 + SentenceType|Subject) + (1 + SubjectType|Item),
                       data = aphasia)
  ```
Post-hoc Comparisons

- With treatment coding, we get estimates of simple-effects:

Random effects:
- Groups: Name, Variance, Std.Dev., Corr
- Item (Intercept): 47858.30, 218.765
- SubjectType1: 55.88, 7.475, -1.00
- Subject: (Intercept): 43287.45, 208.056
- SentenceType1: 1443.33, 37.991, 0.27
- Residual: 6850.06, 82.765
- Number of obs: 960, groups: Item, 32; Subject, 30

Fixed effects:
- Estimate, Std. Error, df, t value, Pr(>|t|)
  - (Intercept): 1716.01, 76.85, 57.43, 22.330, < 2e-16, ***
  - SentenceType1: 577.17, 78.33, 30.94, 7.368, 2.73e-08, ***
  - SubjectType1: 84.52, 76.37, 28.03, 1.107, 0.278
  - SentenceType1:SubjectType1: 188.75, 17.71, 28.82, 10.659, 1.64e-11, ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
- (Intr) SntnT1 SbjcT1
- SentencTyp1: -0.478
- SubjectTyp1: -0.514, 0.002
- SntncT1:ST1: -0.008, 0.258, 0.165

Intercept: RT for healthy controls, active voice sentences
Not a significant RT difference for aphasics (among active sentences)
Significant RT difference for passive sentences (among healthy controls)
Significant special effect of aphasia + passive sentence
Post-hoc Comparisons

- The estimates from a model are enough to fully describe differences among conditions
- With simple effects:

  - ACTIVE, CONTROL: RT ≈ 1716 ms
  - PASSIVE, CONTROL: RT ≈ 2293 ms
  - Passive simple effect +577 ms

Subject Type
- Aphasia
- Control

Sentence Type
- Active
- Passive
Post-hoc Comparisons

- The estimates from a model are enough to *fully describe* differences among conditions
- With simple effects:

  - **ACTIVE, APHASIA**
    - RT ≈ 1801 ms
  - **ACTIVE, CONTROL**
    - RT ≈ 1716 ms
  - **PASSIVE, CONTROL**
    - RT ≈ 2293 ms

  Aphasia simple effect +85 ms
  Passive simple effect +577 ms
Post-hoc Comparisons

- The estimates from a model are enough to fully describe differences among conditions
- With simple effects:

<table>
<thead>
<tr>
<th>Subject Type</th>
<th>Sentence Type</th>
<th>Active, Aphasia</th>
<th>RT ≈ 1801 ms</th>
<th>Passive, Aphasia</th>
<th>RT ≈ 2547 ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aphasia</td>
<td>Active</td>
<td>ACTIVE, Aphasia</td>
<td>+577 ms</td>
<td>PASSIVE, Aphasia</td>
<td>+85 ms</td>
</tr>
<tr>
<td>Control</td>
<td>Passive</td>
<td>ACTIVE, Control</td>
<td>+189 ms</td>
<td>PASSIVE, Control</td>
<td>+85 ms</td>
</tr>
</tbody>
</table>

Interaction effect: +189 ms
Post-hoc Comparisons

- But, sometimes we want to compare individual combinations (e.g., people w/ aphasia seeing active vs passive sentences)
  - i.e., individual cells

<table>
<thead>
<tr>
<th>SubjectType</th>
<th>SentenceType</th>
<th>Active</th>
<th>Passive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aphasia</td>
<td>Active</td>
<td>RT ≈ 1801 ms</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>Passive</td>
<td>RT ≈ 2547 ms</td>
<td></td>
</tr>
<tr>
<td>Aphasia</td>
<td>Active</td>
<td>RT ≈ 1716 ms</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>Passive</td>
<td>RT ≈ 2293 ms</td>
<td></td>
</tr>
</tbody>
</table>
Post-hoc Comparisons: Tukey Test

- But, sometimes we want to compare individual combinations (e.g., people with aphasia seeing active vs passive sentences)
  - i.e., individual cells
- `emmeans(Model.Maximal, pairwise~SentenceType*SubjectType)`
  - Requires `emmeans` package to be loaded
    - `library(emmeans)`
  - Which two cells don’t significantly differ?
  - Uses Tukey test to correct for multiple comparisons so overall α still = .05

<table>
<thead>
<tr>
<th>contrasts</th>
<th>estimate</th>
<th>SE</th>
<th>df</th>
<th>t.ratio</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active, Aphasia - Passive, Aphasia</td>
<td>-765.91793</td>
<td>75.72157</td>
<td>31.01</td>
<td>-10.115</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Active, Aphasia - Active, Control</td>
<td>84.52172</td>
<td>76.36908</td>
<td>28.03</td>
<td>1.107</td>
<td>0.6884</td>
</tr>
<tr>
<td>Active, Aphasia - Passive, Control</td>
<td>-492.64500</td>
<td>109.50199</td>
<td>58.18</td>
<td>-4.499</td>
<td>0.0002</td>
</tr>
<tr>
<td>Passive, Aphasia - Active, Control</td>
<td>850.43965</td>
<td>109.50199</td>
<td>58.18</td>
<td>7.766</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Passive, Aphasia - Passive, Control</td>
<td>273.27293</td>
<td>81.19964</td>
<td>28.03</td>
<td>3.365</td>
<td>0.0113</td>
</tr>
<tr>
<td>Active, Control - Passive, Control</td>
<td>-577.16673</td>
<td>78.33003</td>
<td>30.94</td>
<td>-7.368</td>
<td>&lt;.0001</td>
</tr>
</tbody>
</table>

Comparisons of each pair of cells

Name of the model not the original dataframe

The independent variables (for now, all of them)
Post-hoc Comparisons: Cell Means

- `emmeans` also returns estimated means and std. errors for each cell of the design
  - Great for descriptives write-up
  - *Estimated* means controlling for random effects (esp. relevant when dealing with unbalanced data)

```
$emmeans
  SentenceType SubjectType  emmean    SE   df lower.CL upper.CL
Active     Aphasia   1800.528 75.52878  57.27 1649.300 1951.756
Passive   Aphasia   2566.446 78.00750  56.74 2410.224 2722.669
Active     Control   1716.007 76.84736  57.43 1562.147 1869.866
Passive   Control   2293.173 79.28486  57.13 2134.416 2451.931
```
Post-hoc Comparisons: Cell Means

• Also possible to test whether each of these estimated cell means significantly differs from 0
  • `ls_means(Model.Maximal)`
  • Silly in case of RTs, but could be relevant for some other DVs (e.g., preference)

| Least Squares Means table: | Estimate | Std. Error | df | t value | lower | upper | Pr(>|t|) |
|-----------------------------|----------|------------|----|---------|-------|-------|----------|
| SubjectTypeAphasia          | 2183.487 | 66.794     | 49.5| 32.690  | 2049.295 | 2317.680 | < 2.2e-16  |
| SubjectTypeControl          | 2004.590 | 67.542     | 50.5| 29.679  | 1868.964 | 2140.216 | < 2.2e-16  |
| SentenceTypeActive          | 1758.267 | 65.932     | 52.9| 26.668  | 1626.021 | 1890.513 | < 2.2e-16  |
| SentenceTypePassive         | 2429.810 | 67.359     | 54.5| 36.072  | 2294.788 | 2564.831 | < 2.2e-16  |
| SubjectTypeAphasia:SentenceTypeActive | 1800.528 | 75.529     | 57.3| 23.839  | 1649.300 | 1951.756 | < 2.2e-16  |
| SubjectTypeControl:SentenceTypeActive | 1716.007 | 76.847     | 57.4| 22.330  | 1562.147 | 1869.866 | < 2.2e-16  |
| SubjectTypeAphasia:SentenceTypePassive | 2566.446 | 78.007     | 56.7| 32.900  | 2410.224 | 2722.669 | < 2.2e-16  |
| SubjectTypeControl:SentenceTypePassive | 2293.173 | 79.285     | 57.1| 28.923  | 2134.416 | 2451.931 | < 2.2e-16  |
Post-hoc Comparisons: Marginal Means

- `emmeans` can also give us marginal means:
  - `emmeans(Model.Maximal, pairwise~SubjectType)`

Effect of one variable *averaging over* the other
- e.g., aphasic participants (averaging over all sentence types) vs. controls (averaging over all sentence types)
- These are what *main effects* are testing

```
emmeans
SentenceType  emmean    SE    df  lower.CL upper.CL
Active       1758.267  65.93176  52.94  1626.021  1890.513
Passive     2429.810  67.35936  54.45  2294.788  2564.831

Results are averaged over the levels of: SubjectType
Degrees-of-freedom method: kenward-roger
Confidence level used: 0.95
```
Week 7: Coding Predictors II

- Factors with More than 2 Levels
- Treatment Coding
- Problem of Multiple Comparisons
- Orthogonal Contrasts
  - Example
  - Implementation
  - Definition
  - Practice
- Overview of Coding Systems
- Additional Tests
  - Testing an Overall Factor
  - Random Slopes
  - Post-Hoc Comparisons