International Journal of TESOL Studies (2026)
Vol. 8 (3) 5-28 https://doi.org/10.58304/ijts.250620

Trends in Research Methods

A Method Tutorial in Using Linear Mixed-Effects Modeling in R to Understand Incidental Vocabulary Learning from Captioned Viewing

Mark Feng Teng
Macao Polytechnic University, Macau SAR China

Received: 15 March 2025 / Received in revised form: June 1 2025 / Accepted: 4 June 2025 / Available online: 20 June 2025

Abstract
This research method tutorial article provides a step-by-step guidance on how to carry out linear mixed-effects modeling using R. The guidance was through an example of exploring incidental vocabulary learning under captioned viewing. The data structure is carefully examined to determine an appropriate modeling approach. Both a full model and a simplified model are compared to identify any significant differences in their performance. Subsequently, the optimal model is selected based on the evaluation results. The chosen model is then thoroughly explained and interpreted to shed light on incidental vocabulary learning under captioned viewing. The study encompasses model evaluation and visualizations of the prediction results, providing a comprehensive assessment of the model's reliability and effectiveness. The aim is to demonstrate the practical application of mixed effects models, showcasing their value in real-world research scenarios for researchers in applied linguistics.

Keywords
Linear Mixed-Effects Modeling, R, incidental vocabulary learning, captioned viewing

1 Incidental Vocabulary Learning from Captioned Viewing

Captioned viewing plays a crucial role in enhancing the incidental vocabulary learning of English as a Foreign Language (EFL) learners. This is attributed to the concurrent presentation of both visual and verbal elements, fostering effective information processing and recall, ultimately leading to potential incidental vocabulary acquisition (Teng, 2021). Numerous empirical studies underscore the positive impact of captions on vocabulary learning. For instance, Teng (2022), employing MANCOVA, corroborated the significance of captions in incidental vocabulary learning. Notably, learners exposed to captions outperformed their non-captioned counterparts in terms of word form and meaning recognition and recall. Factors intrinsic to the learners, such as proficiency level and aptitude, are also instrumental in influencing incidental vocabulary learning from captioned videos. Teng's subsequent study in 2023, utilizing Two Conditional Linear Mixed Models (CLMM) analyses, delved into captioned video genres

*Corresponding author. Email: markteng@mpu.edu.mo

and repeated viewing, while considering working memory and vocabulary knowledge. The outcomes endorsed the distinctive impact of various video genres on incidental vocabulary learning, with a notable preference for the comedy genre. Repetition, specifically repeated viewing, demonstrated significance solely in the immediate form recognition test. Moreover, complex working memory emerged as a significant factor in delayed meaning recognition and recall. The breadth of vocabulary knowledge and comprehension also emerged as influential factors affecting incidental vocabulary learning performance. Teng and Cui (2023, 2025), employing mixed effects modeling, compared the incidental learning of single words and collocations under different captioning conditions, taking into account vocabulary knowledge and working memory. The findings supported the efficacy of full captions in enhancing both single word and collocation learning. However, the study emphasized the importance of considering individual differences, including working memory and prior vocabulary knowledge.

While existing research provides substantial evidence supporting the role of captions in vocabulary learning, some results remain inconclusive. The present study seeks to provide further insights into the domain of incidental vocabulary learning from captioned viewing by introducing a tutorial on Linear Mixed Effects Modeling (LMEM) as an increasingly popular method for analyzing incidental vocabulary learning from captioned videos. By incorporating LMEM into data analysis, researchers can better account for complex dependencies within the data, avoid violating the independence assumption present in multiple regression, and consequently, enhance the reliability of their findings in assessing incidental vocabulary learning from captioned viewing. In this way, the present study contributes to advancing the methodological framework for studying vocabulary acquisition in EFL learners through captioned videos.

2 What is Linear Mixed Effects Modeling (LMEM) and Why Is It Important?

LMEM offers a solution to the limitations of ANOVAs and regression analysis. LMEM allows researchers to examine the condition of interest while simultaneously taking into account variability both within and across participants. This approach is more robust, as it handles missing data and unbalanced designs more efficiently. Consequently, the removal of a single observation has a much smaller effect on the analysis (Brown, 2021).

One major advantage of LMEM is the possibility to accommodate continuous predictors, eliminating the need for categorizing or binning continuous variables. This enables researchers to model nonlinear relationships between predictors and outcomes, thereby preserving statistical power (Snijders & Bosker, 2012). Moreover, the fitted model provides coefficient estimates that indicate the magnitude and direction of the effects of interest, essential to an understanding of different predictors (Baayen, 2010).

LMEM can also be extended to handle a variety of response variables, such as categorical outcomes, via generalized linear mixed-effects models. This versatility is particularly useful for researchers working with diverse data types. Operating within this framework eases the transition to Bayesian analysis, further expanding the methodological options available for researchers (Brown, 2021).

Unlike ANOVAs, which tend to focus on categorical "significant versus nonsignificant" results, LMEM encourages a more nuanced understanding of the data. LMEM is more appropriate for analyzing complex data sets of having different research groups and considering different individual differences or factors. Hence, LMEM is a versatile and powerful alternative to ANOVAs and multiple regression, particularly when analyzing data with multiple trials and nested observations. By accommodating continuous predictors, handling missing data, and providing coefficient estimates for effect magnitudes and directions, LMEM offers a more comprehensive analytical tool for researchers dealing with intricate data structures.

3 How to Carry Out LMEM?

Carrying out LMEM involves several basic steps.

Data Preparation: Organize your data in a suitable format, with one row per observation and appropriate columns for the dependent variable, independent variables (fixed effects), and grouping variables (random effects).

Choose a Software: LMEM is typically conducted using specialized statistical software such as R (with packages like lme4) or Python (with libraries like statsmodels or lmerTest). In this tutorial, I chose R software that supports LMEM and is familiar to researchers in applied linguistics.

Model Specification: Formulate the LMEM equation by defining the fixed effects (independent variables) and random effects (grouping variables).

Fit the LMEM: Use the selected R software to fit the LMEM to the data.

Model Assessment: Evaluate the model fit by examining diagnostic plots, residual plots, and goodness-of-fit statistics.

4 Introducing the Dataset

In the following, I will begin by presenting the data that has been utilized as an illustration for LMEM. The data and R script used to generate the models described in this article are available via OSF, at https://osf.io/9qwsy/

4.1 Participants

The participants were 49 first-year English major students at a university in China, ranging from 18.1 to 19.7 in terms of age. The participants were gathered and then randomly allocated to one of the two conditions, i.e., caption viewing groups (n=25), and a control group that only took the tests. Their first language was Mandarin Chinese, and they were learning English as a foreign language. The participants described themselves as intermediate learners, e.g., the B1-B2 level based on Common European Framework of Reference (CEFR).

4.2 Video Selection

The selected video was a documentary about *Ancient World*. The video was selected from YouTube (https://www.youtube.com/watch?v=Ml7lgPw-X3E). The captions were checked and added through Amara. This video was about top ten enigmas of the ancient world. The topic, which was about a world full of mystery, folklore and legend, was selected on a pilot of 10 English major students. The students watched the video and agreed that the topic was quite interesting and that the content was suitable for their L2 learning. The video is one hour, six minutes, and 27 seconds long. The spoken language was in line with the written language. VocabProfile (https://www.lextutor.ca/) was adopted to determine the lexical profile. The script contained 8,039 running words. The 1,000-, 2,000-, and 3,000-word families covered 73.67%, 81.20%, and 89.01% of all running words in the script, respectively.

4.3 Target Words

We selected a total of 46 words as test items. Based on VocabProfile (https://www.lextutor.ca/), approximately 56% of all target words were beyond the 3,000 word level. Those words might be unknown to the participants.

4.4 Vocabulary Test and Scoring

The assessment of incidental vocabulary learning was performed through a meaning recognition test. This test was administered two times: pretest, and immediate posttest.
This meaning recognition test included 46 items. Each item had five options, with a correct meaning, three distractors, as well as an "I don't know this word" option. The "I don't know" option was to reduce learners' wild guessing. The participants were required to choose one option after hearing the target word. Each correct answer was given 1 point, while each incorrect answer was given 0 points. The maximum score for each test part was 46 points.

4.5 Research Questions and Data Analysis

The research question concerns whether captioned viewing led to a greater increase in incidental vocabulary learning. LMEM using the lme4 package was performed (Bates & Maechler, 2010) in the R language and environment (R Core Team, 2020). Compared with ANOVA and ANCOVA, LMEM provides advantages in including groups (control vs. experimental groups) and time (pretest and immediate posttest) in a single model while considering any potential variance due to individual differences through random effects. The fixed effects were group, time, and the interaction between group and time. The random effects were participants. Group and time were categorical variables.

4.6 Details and Procedures for Data Analysis

In the following, LMEM was used to evaluate the effects of conditions (Control and Caption viewing) on the performance of meaning recognition, while modeling the variation within participants (Pid) and items (item).

In all the analyses described below, I will use a dummy coding (also known as treatment coding) scheme, where the control condition is set as the reference level and is therefore coded as 0, while the treatment condition is coded as 1. Thus, in the mixed-effects model, the regression coefficient associated with the intercept represents the estimated average score under the control condition (when the mode = 0), and the coefficients related to the mode effect represent how the average score changes under the treatment condition (when the mode = 1). We can also choose to use the Caption viewing condition as the reference level, in which case the intercept represents the estimated average meaning score under the Caption viewing condition (when the mode = 0), and the mode effect would indicate the change in this estimate under the control condition (when the mode = 1). Changing the coding scheme, whether by altering the reference level or completely switching to a different coding scheme (e.g., sum or deviation coding, which involves coding groups as -0.5 and 0.5 or -1 and 1, respectively, so that the intercept corresponds to the grand mean), will not change the model fit; it will only change the interpretation of the regression coefficients (but often lead to misconceptions about interactions).

The left side of Table 1 displays the first six rows of the required format for the data: ungrouped long format. For a useful tutorial on how to organize data in this format, I recommend the open textbook "R for Data Science" (Chapter 12.3.1) by Wickham and Grolemund (2017) and the tidyverse collection of R packages (Wickham et al., 2019). If you are tracking your own data, make sure it is in long format, where each row represents a separate observation (i.e., do not aggregate across participants or items). Note that in the first half of the table, each row among the first six corresponds to different items presented to the

same participant (Pid). In contrast, for score probability analysis, the data frame would include 2*46 rows per participant (number of time points * number of words), with only one mode per row, and the values in the column of the meaning recognition score reflect the average score for all words presented to that individual under the specified condition (right side of Table 1).

Table 1

The First Six Rows of the Required Format for the Data

```
> head(data2)
  Pid          item     trt time me
1 101 Accuse        Control  pre
2 101 Acoustic      Control  pre
3 101 Alignment     Control  pre
4 101 Besiege       Control  pre
5 101 Brutal        Control  pre
6 101 Coalition     Control  pre
```

```
> head(m_data)
# A tibble: 6 x 4
# Groups:   Pid, trt [3]
  Pid   trt     time  meaning_m
  <fct> <fct>   <fct>     <dbl>
1 101   Control pre      0.0435
2 101   Control post     0.0435
3 102   Control pre      0.109
4 102   Control post     0.109
5 103   Control pre      0
6 103   Control post     0
```

		Overall	Control (24)	Caption viewing (25)	*p*
	2254	948	177	771	
Pre	meaning	323 (14.3)	93 (8.4)	230 (20.0)	<0.001
Post	meaning	625 (27.7)	84 (7.6)	541 (47.0)	<0.001

The reason why mixed effects models are called “mixed” is because they simultaneously model both fixed and random effects. Fixed effects represent population-level (i.e., average) effects that are expected to persist across different experiments. Condition effects are typical fixed effects as they are expected to operate in a predictable manner across different samples of participants and items. In fact, in my example, the mode will be simulated as a fixed effect because I expect an average relationship between the mode and meaning scores that would reoccur if the same experiment was conducted with different samples of participants and items.

Fixed effects model the average trend, while random effects model the extent of variation in these trends at different levels of grouping factors, such as participants or items. Random effects consist of dependent data points within groups, where the observed values within a group come from the same higher-level population (such as individual participants or items) and are included in the mixed effects model to account for the fact that the behavior of specific participants or items may deviate from the average trend. Since random effects are discrete units sampled from some populations, they are inherently categorical. Therefore, if you want to determine whether an effect should be modeled as fixed or random, and it is essentially continuous, it should be noted that it cannot be modeled as a random effect and must be treated as a fixed effect. In our hypothetical experiment, participants and words are modeled as random effects because they are randomly sampled from their respective populations, and we want to account for the variability within these populations.

Including random effects for participants and items can address the issue of non-independence that often plagues multivariate regression, where some participants tend to score higher than others, and some items tend to receive higher scores than others. These random deviations from the average score are referred to as random intercepts. For example, the model may estimate an average score of 0.3 under certain conditions, but including random intercepts for each participant allows the model to estimate a fixed deviation for each participant from the average score. Therefore, if a participant tends to achieve particularly high scores, their individual intercept may be shifted upwards by 0.5 (i.e., the estimated

intercept would be 0.8). Similarly, including random intercepts for individual items allows the model to estimate the deviation of each item from the fixed intercept, reflecting that some words are generally easier to score than others. In contrast, in multivariate regression, the same regression line (including intercept and slope) applies to all participants and items, resulting in less accurate predictions compared to mixed effects regression, and often larger residual errors. Therefore, in mixed models, the fixed intercept estimates represent the average intercept, while the random intercepts allow for deviations from this average value for each participant and item.

Another source of variation that can be explained by mixed effects models is that variables simulated as fixed effects may actually have different effects on different participants (or items). In this example, some participants may show very small differences in scores between the control and treatment conditions, while others may show substantial differences. Similarly, some words may be more influenced by the pattern than others. To model this type of variation, I will include random slopes in the model specification. For example, in this tutorial, the model may estimate an effect of 0.321 for the Caption viewing mode on meaning, indicating that participants, on average, show a 0.25 increase in the word meaning recognition scores under the Caption viewing condition compared to the Control condition. However, one word may be strongly influenced by the mode (e.g., a score difference of 0.475 between Control and Caption viewing), while another word may be only weakly influenced by the mode (e.g., a score difference of 0.05 between Control and Caption viewing). These individual deviations from the average pattern effect are simulated through random slopes (note that simple mean differences described here are represented as slopes in the regression equation).

Patterns that arise within the context of fixed and random effects can be confusing, but it is important to remember that if an effect is assumed to persist across different participant samples, it is considered fixed. In this paper, the pattern is modeled as a fixed effect because we are modeling the joint influence of the pattern on the scores of participants and items. However, considering that participants are a random sample from the population of interest, the modal effects within participants represent a subset of possible interactions between the pattern and participants. In other words, the pattern itself is not a random effect, but its interaction with participants is random, and including random slopes for the pattern allows the model to estimate deviations of each participant from the overall (fixed) trend. (For a description of the distinction between fixed and random effects and when researchers may genuinely want to model participants as fixed effects, see Mirman, 2014).

One common question that arises is why seemingly systematic effects are referred to as random effects if the mixed effects model provides intercept and slope estimates for each participant. The answer lies in the fact that although an effect may be consistent within specific individuals (e.g., a participant may consistently respond with higher scores, showing a higher susceptibility to the pattern than the average level), the source of this variation is unknown and therefore considered random. If you find the use of the term "random" confusing, considering synonymous terms such as intercepts (or slopes) varying by participant (or item) may be helpful. However, given that these effects are most commonly referred to as random intercepts and slopes, I use that terminology here.

Before proceeding to the implementation in R, it is necessary to address another issue regarding the random effects structure in mixed effects modeling: determining which random slopes are reasonable in the research design. Considering the example again, I model the scores for words as a function of their difficulty. The difficulty of words is manipulated within participants. However, due to the inherent variability in the difficulty of words, it is a between-item variable. Since including item-specific random slopes describes the varying influence of related predictor variables across items, we cannot model the effect of word difficulty on specific items because each word has only one difficulty level; in other words, we cannot include item-specific random slopes. Conversely, if the predictor of interest is a within-item variable, I can include item-specific random slopes in the model, which would indicate that different words may be influenced differently by the predictor. In short, random slopes by participant and by item are applicable to within-subjects and within-items designs, respectively. Therefore, in the example of

word difficulty, the random effects structure may include random intercepts for participants and items, as well as random slopes by participant for word difficulty, but it cannot include random slopes by item for word difficulty.

Because including item-specific random slopes for word difficulty is unreasonable in this example, the random effects structure including random intercepts for participants and items, as well as random slopes by participant for word difficulty, represent the largest random effects structure justified by the design (see Barr et al., 2013; Matuschek et al., 2017). In cases where random slopes by participant and by item are both reasonable, the mixed effects model can include the simultaneous effects of random slopes by participant and by item (though it is important to note that the inclusion of random effects does not necessarily imply it is always the best approach. I will further discuss this issue in the sections on model building and convergence issues).

5 Examples and Implementation in R

First, the article utilizes R and R Studio, and the package used is lme4.

To begin, install the necessary packages.

```
if (!("lme4" %in% installed.packages())) install.packages("lme4")
if (!("tidyverse" %in% installed.packages())) install.packages("tidyverse")
if (!("afex" %in% installed.packages())) install.packages("afex")
```

Importing Packages

```
library(lme4)
library(tidyverse)
```

After downloading the data, import the data

```
data <- read.csv("data.csv")
```

This is the data transformed into a long format, and the table headers are as follows.

```
> head(data)
  X Pid          item     trt time meaning
1 1 101 Accuse        Control  pre       0
2 2 101 Acoustic      Control  pre       0
3 3 101 Alignment     Control  pre       0
4 4 101 Besiege       Control  pre       0
5 5 101 Brutal        Control  pre       0
6 6 101 Coalition     Control  pre       0
```

Looking at the table header, the Pid column represents the same participant for each row. Based on the participant (Pid) and item, the next step is to examine the scores of each participant at various testing phases under different conditions (Control, Caption viewing).

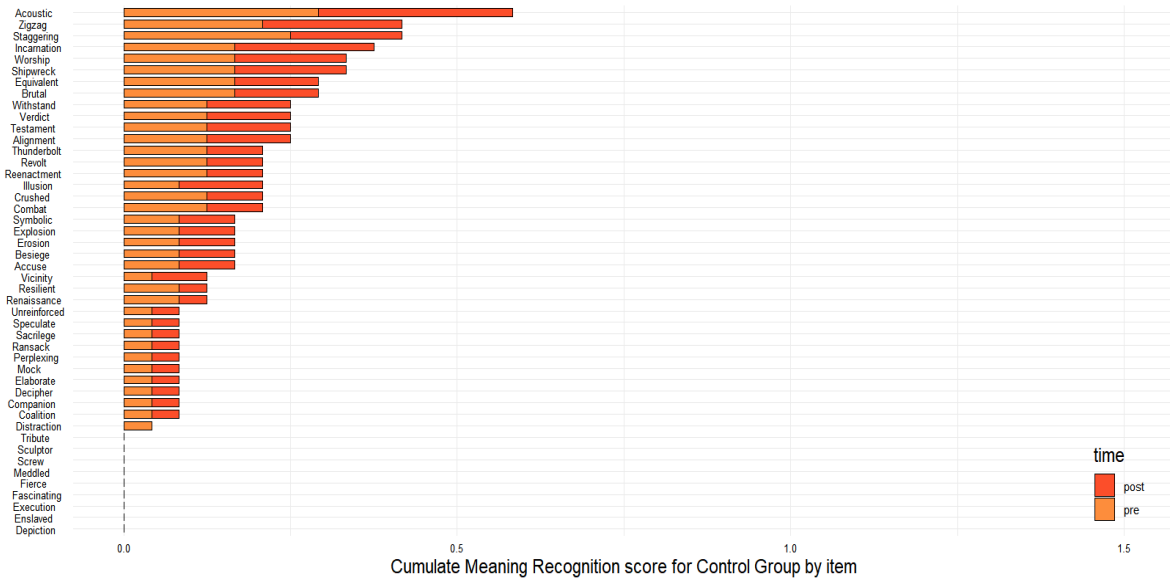
```
m_data=data %>%
 group_by(Pid,trt,time) %>%
```

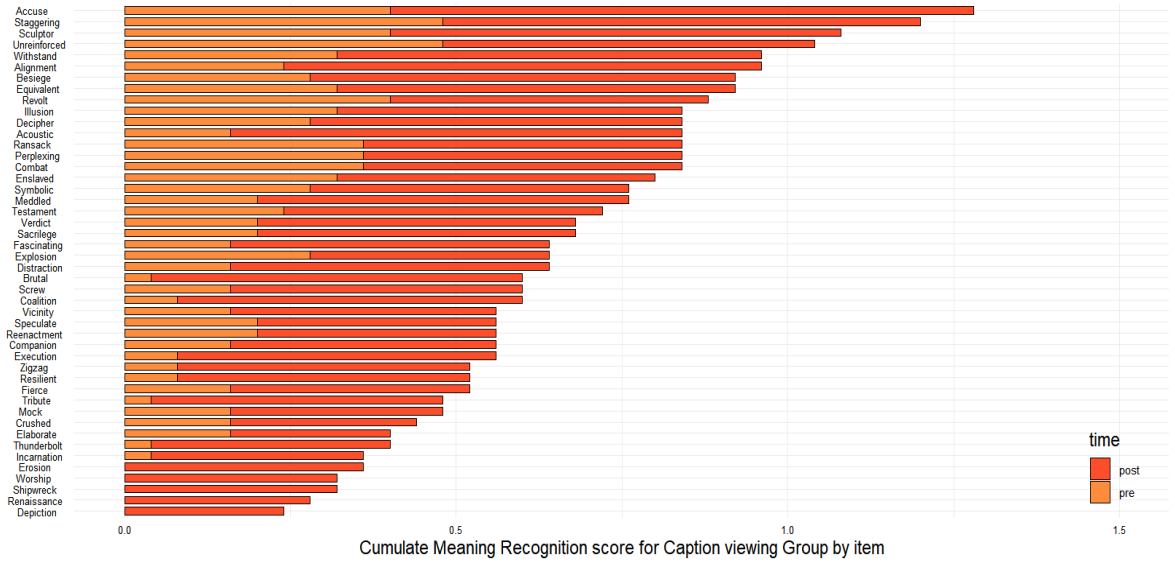
```
 summarise(meaning_m=mean(meaning))
head(m_data)
```

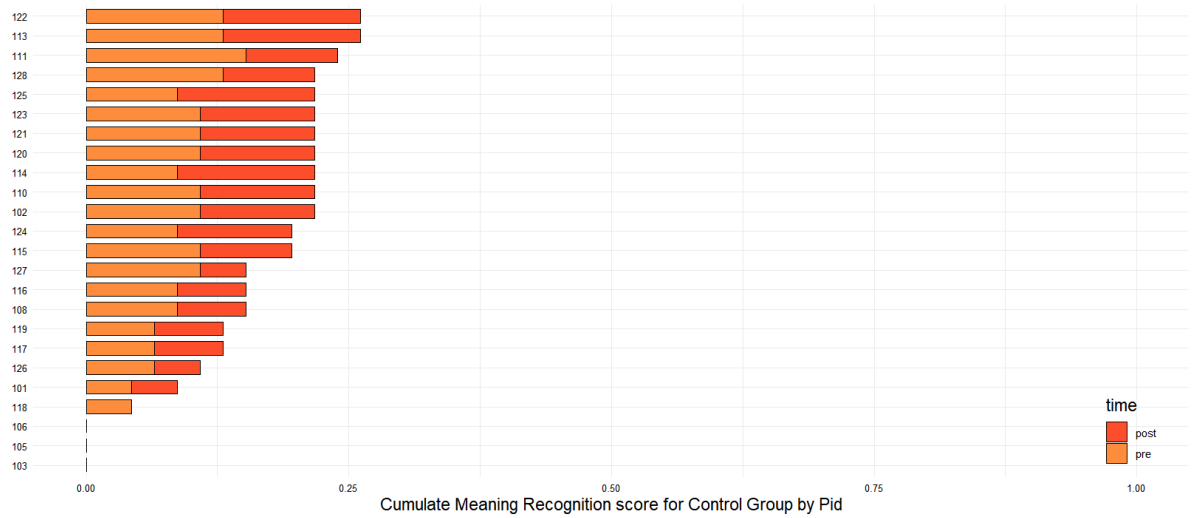
We can observe the average scores of each participant at different testing phases under each mode.

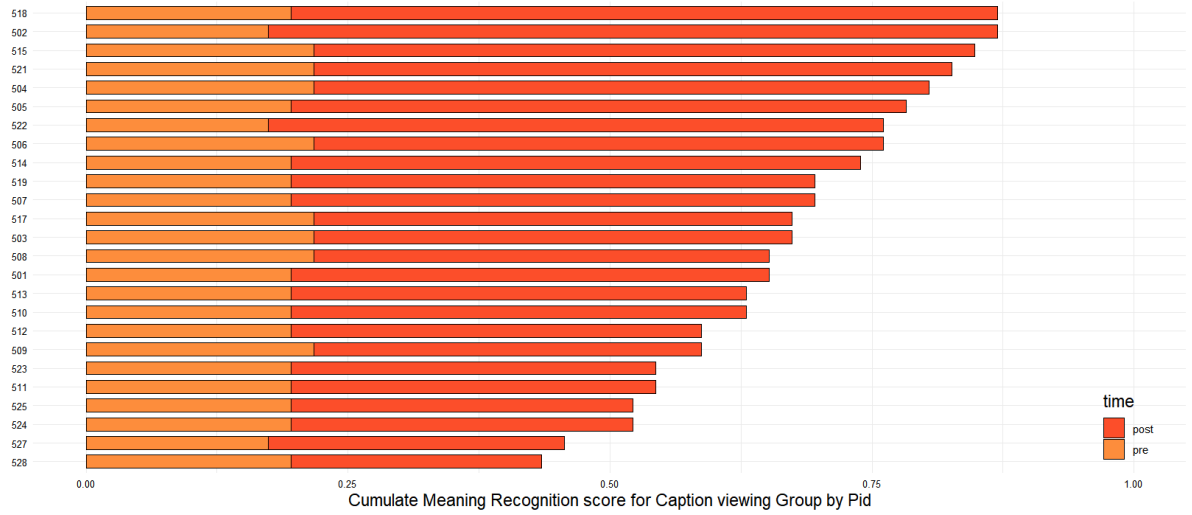
```
> head(m_data)
# A tibble: 6 x 4
# Groups:   Pid, trt [3]
  Pid   trt     time  meaning_m
  <fct> <fct>   <fct>     <dbl>
1 101   Control pre      0.0435
2 101   Control post     0.0435
3 102   Control pre      0.109
4 102   Control post     0.109
5 103   Control pre      0
6 103   Control post     0
```

Visualizing the raw data









Encoding for the variable "trt" (mode): Control mode is encoded as 0, and Caption viewing mode is encoded as 1.

```
data$trt=factor (data$trt,level=c("Control","Caption viewing"))
```

Encoding for the variable test time ("time" (pre, post) within the same mode:
Pre-testing is encoded as 0, and post-testing is encoded as 1.

```
data$time=factor(data$time, levels=c("pre","post"))
```

For an experiment with a single independent variable and random intercepts but no random slopes for (crossed) participants (Pid) and items (item), the basic syntax for a mixed-effects model is as follows:

Linear mixed-effects model:

```
lmer(outcome ~ 1 + predictor + (1|Pid) + (1|iterm), data=data)
```

Generalized linear mixed-effects model:

```
glmer(outcome ~ 1 + predictor + (1|Pid) + (1|iterm),
   family=binomial(link='logit'),
   data=data)
```

The parts within parentheses are the random effects, while the parts outside the parentheses are the fixed effects. The vertical lines within the parentheses are called pipes, and they indicate that the effect on the left of the pipe varies by the grouping factor on the right of the pipe. When the outcome variable is a continuous variable following a normal distribution, a linear mixed-effects model is used. When the outcome variable is not a continuous variable or does not follow a normal distribution, a generalized linear mixed-effects model is used. In this case, the data follows a binomial distribution with values of 0 and 1 (Bernoulli distribution), so a generalized linear mixed-effects model with the family (gaussian) specifying logistic regression is used.

In this example, the intercept (represented by 1) varies by two grouping factors in the experiment: participants and items. Note that 1 is optional in the fixed effects part of the model specification because the fixed intercept is included by default, but it is not optional in the random effects part because there must be an indication of which effects are allowed to vary by each grouping factor (i.e., the region to the left of the pipe cannot be left empty). I recommend using 1 to indicate the intercept in both the fixed effects and random effects parts of the model specification to avoid confusion about when it needs to be included.

Finally, the "data" parameter represents the name of the R object containing the data, and the "glmer" part is the function used to fit the mixed-effects model (since you have installed the lme4 package, you have access to it).

So far, the model includes random intercepts but no random slopes. However, participants and items may differ in their response to the manipulated condition. Therefore, I will fit a model that includes random slopes for the mode by participants and by items. Not including random slopes would assume that all participants and items have the exact same response to the mode, which is an unreasonable assumption. While including both random intercepts and slopes by participants and items reflects the maximum random effects structure supported by the design, the decision to include random slopes by participants and items is also theoretically justified. Theoretical motivations should always be considered, as blindly maximizing may lead to model non-convergence and loss of statistical power (Matuschek et al., 2017). Note how the basic syntax for the model changes when we include slopes that vary by participants and items in the random-effects structure.

```
lmer(outcome ~ 1 + predictor + (1+predictor|Pid)+(1+predictor|item), data=data)
glmer(outcome ~ 1 + predictor + (1+predictor|Pid)+(1+predictor|item),
     family=binomial(link='logit'),
 data=data)
```

Here, the part within parentheses represents the intercept (represented by 1). In this case, the intercept is optional because its inclusion is implied by the presence of random slopes, but for clarity, it is included. The predictor is represented by "+ predictor". Both the intercept and predictor vary by participants (Pid) and items (item). In simple terms, this syntax means "use the data provided to predict the outcome based on the predictor and the random intercepts and slopes varying by participants and items."

The above model includes only one predictor, but if a model includes multiple predictors, researchers can decide which predictors can vary by participants or items. In other words, any fixed effect to the left of the pipe (within the inner parentheses) can be included as long as it is reasonable in the experimental design. For example, if we want to include a second predictor that varies by both participants and items

but there is no theoretical motivation to include a random slope for the second predictor, or if it is unreasonable to include a random slope for the second predictor given the experimental design (e.g., if the second predictor varies between items), the syntax would be as follows.

```
lmer(outcome ~ 1+predictor1 + predictor2 + (1+predictor1 + predictor2|Pid) + (1+ predictor1|item),
data=data)
```

```
glmer(outcome ~ 1+predictor1 + predictor2 + (1+predictor1 + predictor2|Pid) + (1+ predictor1|item),
      family=binomial(link='logit'),
 data=data)
```

In the example we are going to use, the complete model (i.e., the model that includes all the fixed effects of interest and all the theoretically motivated random effects) is specified as follows:

$$\log\left(\frac{P}{1-p}|\text{meaning} = 1\right)$$

$$= \beta_{00} * \text{trt}_{\text{Control}} * \text{time}_{\text{pre}} + \beta_{01} * \text{trt}_{\text{Control}} * \text{time}_{\text{post}} +$$

$$\beta_{10} * \text{trt}_{\text{Caption}} * \text{time}_{\text{pre}} + \beta_{11} * \text{trt}_{\text{Caption}} * \text{time}_{\text{post}} +$$

$$b_{1ijk} * \text{Pid}_{ik} + b_{2ijl} * \text{item}_{ijl} + \varepsilon_{ij}\epsilon$$

Here, i represents the level of trt, j represents the level of test time, β_ij is the estimated coefficient for the fixed effect, k is the participant identifier (Pid), l is the item identifier, b_1ijk and b_2ijl and are the random effects, and Pid_ik and item_ijl are the levels of the random effects.

```
meaning_full.mod <- glmer(meaning ~  trt:time +
            (1 + trt:time |Pid) +(1+trt:time |item),
            family=binomial(link='logit'),
            data = data)
```

Here, we are predicting the meaning recognition scores based on the fixed effects of intercept (1), trt (Control vs. treatment condition), and test time. I included random intercepts and slopes for participants (Pid = participant identifier) and items (item = stimulus). I used R to use the data frame named "data". This assigns a name to an object, which is a specific data structure stored in R, for future reference. Thus, with this line of code, I created a model and gave it an intuitive name so that we know what this object represents later on.

If you run this line of code in an R script, you may notice a warning message indicating that the model failed to converge. Generalized linear mixed-effects models can be computationally complex, especially when they have rich random effects structures, and convergence failure essentially means that a good fit to the data cannot be found within a reasonable number of iterations attempting to estimate model parameters. It is important not to report results from a non-converging model because the convergence warning indicates that the model has not obtained reliable estimates and therefore cannot be trusted.

When a model fails to converge, as a researcher, you have several options. One is to report degrees of freedom - the many seemingly harmless choices made during the research process that allow researchers to find statistically significant evidence consistent with any hypothesis (Simmons et al., 2011). Generally, you should leverage your specific domain knowledge and previous research to consider a priori which random effects are theoretically important to include in your model (e.g., ask yourself, "Does it make sense for the effect of the mode to vary by participant or item?") and only remove random effects as a last resort when all other attempts to resolve convergence issues have failed. If you must remove random effects, this decision should be documented and reported in your published manuscript and/or accompanying code.

The first step in addressing convergence issues is to consider your dataset and how your model relates to it, ensuring that your model has not been misspecified (e.g., including by-item varying slopes for a predictor that does not actually vary within items). It is also possible that convergence warnings arise from unbalanced data. If you have some participants or items with only a few observations, the model may struggle to estimate random slopes, and these participants or items may need to be removed to achieve convergence. While addressing convergence issues can feel tedious, remember that these warnings serve as friendly reminders to think deeply about your data and not model blindly. Assuming you have done these steps, the next step is to add control parameters to your model to patch up estimated nuts and bolts. There are many control parameters, and depending on the source of the convergence problem, some parameters may be more suitable or helpful than others. I suggest starting with adjusting the optimizer, the method by which the model finds the optimal solution. The following model specification is the same as above, except it includes a control parameter explicitly specifying the optimizer.

```
> meaning_full.mod1 <- glmer(meaning ~ Control_post +Caption_viewing_pre+Caption_viewing
_post+
+                              (1 + Control_post +Caption_viewing_pre+Caption_viewing_po
st |Pid) +
+                              (1+Control_post +Caption_viewing_pre+Caption_viewing_post
|item),
+                            family=binomial(link='logit'),
+                            control = glmerControl(optimizer = "bobyqa"),
+                            data = data2)
boundary (singular) fit: see help('isSingular')
> summary(meaning_full.mod1)
Generalized linear mixed model fit by maximum likelihood
  (Laplace Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula:
meaning ~ Control_post + Caption_viewing_pre + Caption_viewing_post +
    (1 + Control_post + Caption_viewing_pre + Caption_viewing_post |
        Pid) + (1 + Control_post + Caption_viewing_pre + Caption_viewing_post |
    item)
   Data: data2
Control: glmerControl(optimizer = "bobyqa")

     AIC      BIC   logLik deviance df.resid
  3878.7   4032.6  -1915.3   3830.7     4484

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.6788 -0.4555 -0.2739 -0.1503  6.0235
```

This model has converged, but how do I know which optimizer to choose? And what if the model does not converge successfully with that optimizer? When it comes to selecting an optimizer, I highly

recommend using the all_fit() function from the all_fit() package (Singmann & Kellen, 2019). This function takes a model as input, re-fits the model with various optimizers, and lets you know which optimizers produce warning messages. This package integrates well with lme4, so there is no need to change the syntax of your model before running this function. Below is the relevant code:

```
> allFit(meaning_reduced.mod4)
bobyqa : boundary (singular) fit: see help('isSingular')
[OK]
Nelder_Mead : [OK]
nlminbwrap : boundary (singular) fit: see help('isSingular')
[OK]
nloptwrap.NLOPT_LN_NELDERMEAD : boundary (singular) fit: see help('isSingular')
[OK]
nloptwrap.NLOPT_LN_BOBYQA : boundary (singular) fit: see help('isSingular')
[OK]
original model:
meaning ~ 1 + Caption_viewing_pre + Caption_viewing_post + (1 + Caption_viewi...
data:  data2
optimizers (5): bobyqa, Nelder_Mead, nlminbwrap, nloptwrap.NLOPT_LN_NELDERMEAD,nloptwra
p.NLO...
differences in negative log-likelihoods:
max= 0.000993 ; std dev= 0.00044
Warning message:
In checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv,  :
  Model failed to converge with max|grad| = 0.0626115 (tol = 0.002, component 1)
```

This output indicates that if there are convergence warnings or singular fits from the tested optimizers, it suggests that there are issues with the estimation. Therefore, none of these optimizers should produce reliable parameter estimates.

In this example, the model converged when I changed the optimizer, but that is not always the case. Sometimes, you may need to address convergence issues in a different way. One option is to force the correlation between random effects to be zero. Remember that in addition to estimating fixed and random effects, mixed-effects models also estimate the correlation between random effects. If you are willing to accept the possibility of zero correlation, this can reduce the computational complexity of the model and potentially improve convergence of parameter estimates. However, please note that it is best to perform likelihood ratio tests (described in detail in the next section) between different nested models with and without the correlation or examine confidence intervals around the correlation to determine if it is necessary to eliminate it.

To eliminate the correlation between two random effects in R, simply replace the '1' in the random effects specification with '0'. However, when you do this, the glmer() function no longer estimates the random intercept, so you need to make sure to include it back in the model specification. If you want to remove the correlation between the random intercept for participants and the random slope for conditions, the code would be as follows:

```
rt_full.mod <- lmer(meaning~ 1 + trt+
                      (0 + trt|Pid) + (1|Pid)+ (1 + trt|item),
                    data = data2)
```

Using (1|PID) represents the random intercept, while (0+trt|PID) represents the random slope. By separating them, we effectively remove the correlation between the two.

Other approaches to address convergence warnings include increasing the number of iterations before the model "gives up" searching for a solution (e.g., control = lmerControl(optCtr = list(maxfun = 1e9))), centering or scaling continuous predictor variables (or applying sum coding to categorical predictor variables), or using the control parameter control = lmerControl(calc.divs = FALSE) to remove some derivative calculations that occur after the solution is found. I also recommend entering convergence in the R console, which will open a help file providing additional suggestions for addressing convergence warnings.

Finally, it is possible that the model fails to converge simply because the random effects structure is too complex. In such cases, it is possible to selectively remove random effects based on model selection techniques (Matuschek et al., 2017). However, it should be emphasized that simplifying the random effects structure should only be considered as a last resort, and these decisions should be documented. The random effects structure should have theoretical justification, so it is generally preferable to attempt to maintain this structure unless all other attempts to address convergence issues have been unsuccessful.

Now that we have a model to work with, how do we determine if the pattern truly influences the scores? This is typically done by comparing the model that includes the effect of interest (such as the pattern) with a nested model that lacks that effect, using a likelihood-ratio test. If you obtain a small p-value from the likelihood-ratio test, it indicates that the full model provides a better fit to the data.

When conducting a likelihood-ratio test on our example, we are essentially asking whether a model that includes information about the presentation pattern is a better fit to the data compared to a model that lacks that information. Here is how you can do this in R: first, you establish the reduced model that lacks the fixed effect of the pattern but is otherwise identical to the full model (including any control parameters used). Then, you perform the test using the ANOVA( ) function (which actually performs a likelihood-ratio test rather than an analysis of variance):

```
> anova(meaning_full.mod1,meaning_reduced.mod)
Data: data2
Models:
meaning_reduced.mod: meaning ~ 1 + (1 + Control_post + Caption_viewing_pre + Caption_vie
wing_post | Pid) + (1 + Control_post + Caption_viewing_pre + Caption_viewing_post | ite
m)
meaning_full.mod1: meaning ~ Control_post + Caption_viewing_pre + Caption_viewing_post +
(1 + Control_post + Caption_viewing_pre + Caption_viewing_post | Pid) + (1 + Control_pos
t + Caption_viewing_pre + Caption_viewing_post | item)
                    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
meaning_reduced.mod   21 3954.9 4089.6 -1956.5   3912.9
meaning_full.mod1     24 3878.7 4032.6 -1915.3   3830.7 82.208  3  < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The output includes the likelihood-ratio test statistic and the corresponding p-value. If the p-value is below a chosen significance level (e.g., 0.05), it suggests that the inclusion of the pattern significantly improves the model fit.

The small p-value in the "Pr(>Chisq)" column indicates that the model including the pattern effect provides a better fit to the data compared to the model without the pattern effect. Therefore, the pattern effect is significant. I have added a red box around the p-value, the $\chi 2$ value (32.385) in the "Chisq" column, and the degrees of freedom for the test (1, found in the "Chisq Df" column) because these three values should be reported in your results section (I will come back to this point below).

Given that the full model includes three condition effects (patterns), testing can be relatively complex. However, performing likelihood-ratio tests becomes cumbersome. This is because these tests need to be conducted on nested models, so a reduced model lacking the effect of interest (sometimes referred to as

the null model) needs to be established for comparison. Another issue with this approach is that although the reduced models are only built for the purpose of comparison with the full model, it can be tempting to examine these intermediate models and consider them as plausible candidates for the "best model" (i.e., engaging in stepwise regression without being aware). Here, a full model with two fixed effects (pattern and test time) and a reduced model testing the significance of the pattern effect were established. In doing so, you may notice that the test time and pattern are both significant in both the reduced and full models.

```
> meaning_reduced.mod2 <- glmer(meaning ~ trt+
+                               (1 + trt |Pid) +
+                               (1+trt |item),
+                             family=binomial(link='logit'),
+                             control = glmerControl(optimizer = "bobyqa"),
+                             data = data2)
> anova(meaning_full.mod1,meaning_reduced.mod2)
Data: data2
Models:
meaning_reduced.mod2: meaning ~ trt + (1 + trt | Pid) + (1 + trt | item)
meaning_full.mod1: meaning ~ Control_post + Caption_viewing_pre + Caption_viewing_post +
(1 + Control_post + Caption_viewing_pre + Caption_viewing_post | Pid) + (1 + Control_pos
t + Caption_viewing_pre + Caption_viewing_post | item)
                      npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
meaning_reduced.mod2     8 4077.9 4129.2 -2031.0   4061.9
meaning_full.mod1       24 3878.7 4032.6 -1915.3   3830.7 231.22 16  < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> meaning_reduced.mod3 <- glmer(meaning ~ time+
+                                 (1 + time |Pid) +
+                                 (1+time |item),
+                               family=binomial(link='logit'),
+                               control = glmerControl(optimizer = "bobyqa"),
+                               data = data2)
boundary (singular) fit: see help('isSingular')
> anova(meaning_full.mod1,meaning_reduced.mod3)
Data: data2
Models:
meaning_reduced.mod3: meaning ~ time + (1 + time | Pid) + (1 + time | item)
meaning_full.mod1: meaning ~ Control_post + Caption_viewing_pre + Caption_viewing_post +
(1 + Control_post + Caption_viewing_pre + Caption_viewing_post | Pid) + (1 + Control_pos
t + Caption_viewing_pre + Caption_viewing_post | item)
                      npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
meaning_reduced.mod3     8 3982.4 4033.7 -1983.2   3966.4
meaning_full.mod1       24 3878.7 4032.6 -1915.3   3830.7 135.68 16  < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fortunately, the afex package has another convenient function that allows you to completely avoid this approach. The mixed ( ) function takes a model specification as input and when the parameter method = 'LRT' is included, it performs likelihood-ratio tests for all fixed (not random) effects in the model. Importantly, you will not see reduced models constructed solely for obtaining p-values, thus reducing the temptation for unintentional p-hacking. This function is particularly useful when your model has multiple fixed effects, but here's how you can implement this function in this example and what the output looks like (note that the $\chi 2$ value is different from what we obtained using the ANOVA( ) function because the two functions perform likelihood-ratio tests):

```
> mixed(meaning ~ 1+Control_post +Caption_viewing_pre+Caption_viewing_post+
+           (1 + Control_post +Caption_viewing_pre+Caption_viewing_post |Pid) +
+           (1+Control_post +Caption_viewing_pre+Caption_viewing_post |item),
+       family=binomial(link='logit'),
+       control = glmerControl(optimizer = "bobyqa"),
+       data = data2,
+       method="LRT")
Contrasts set to contr.sum for the following variables: Pid, item
Numerical variables NOT centered on 0: Control_post, Caption_viewing_pre, Caption_viewing_post
If in interactions, interpretation of lower order (e.g., main) effects difficult.
Fitting 4 (g)lmer() models:
[boundary (singular) fit: see help('isSingular')
.boundary (singular) fit: see help('isSingular')
.boundary (singular) fit: see help('isSingular')
.boundary (singular) fit: see help('isSingular')
.]
Mixed Model Anova Table (Type 3 tests, LRT-method)

Model: meaning ~ 1 + Control_post + Caption_viewing_pre + Caption_viewing_post +
Model:     (1 + Control_post + Caption_viewing_pre + Caption_viewing_post |
Model:         Pid) + (1 + Control_post + Caption_viewing_pre + Caption_viewing_post |
Model:     item)
Data: data2
Df full model: 24
                Effect df     Chisq p.value
1         Control_post  1      0.69    .407
2  Caption_viewing_pre  1 18.35 ***   <.001
3 Caption_viewing_post  1 75.53 ***   <.001
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘+’ 0.1 ‘ ’ 1
```

The likelihood-ratio test results for the full model indicate that Control_post is not significant. To compare the variance between the full model and the model without Control_post, we can remove Control_post from the full model and perform the variance comparison.

```
> anova(meaning_full.mod1,meaning_reduced.mod3)
Data: data2
Models:
meaning_reduced.mod3: meaning ~ 1 + Caption_viewing_pre + Caption_viewing_post + (1 + Caption_vie
wing_pre + Caption_viewing_post | Pid) + (1 + Caption_viewing_pre + Caption_viewing_post | item)
meaning_full.mod1: meaning ~ 1 + Control_post + Caption_viewing_pre + Caption_viewing_post + (1 +
Control_post + Caption_viewing_pre + Caption_viewing_post | Pid) + (1 + Control_post + Caption_vi
ewing_pre + Caption_viewing_post | item)
                    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
meaning_reduced.mod3   15 3861.5 3957.7 -1915.8   3831.5
meaning_full.mod1      24 3878.7 4032.6 -1915.3   3830.7 0.8254  9     0.9997
```

The reduced model does not show a significant difference compared to the full model, and it has a lower AIC. Therefore, the reduced model is selected as the optimal model. Once the optimal model is chosen, the fixed effects and random effects of the model can be interpreted.

6 Interpreting Fixed and Random Effects

Comparing the likelihood ratio tests between our full model and the reduced model indicates that the pattern effect and detection time are significant. However, it does not provide information about the direction or magnitude of these effects. So, how can we assess whether the caption viewing condition leads to higher or lower scores? And how can we gain further insights into the variability among different participants and items that our model estimates?

To answer these questions, we need to examine the output of the model using the summary() command. This output consists of two main sections. The upper section provides information about the random effects, while the lower section contains information about the fixed effects. The following code block implements the summary() command and displays a brief output related to interpreting the fixed effects:

```
> summary(meaning_reduced.mod3)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: meaning ~ 1 + Caption_viewing_pre + Caption_viewing_post + (1 +
    Caption_viewing_pre + Caption_viewing_post | Pid) + (1 +
    Caption_viewing_pre + Caption_viewing_post | item)
   Data: data2
Control: glmerControl(optimizer = "bobyqa")

     AIC      BIC   logLik deviance df.resid
  3861.5   3957.7  -1915.8   3831.5     4493

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.6780 -0.4550 -0.2741 -0.1536  6.3655


Fixed effects:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          -2.7920     0.1955 -14.283  < 2e-16 ***
Caption_viewing_pre   1.2322     0.2469   4.991 6.02e-07 ***
Caption_viewing_post  2.6632     0.2224  11.973  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


> b1
                    Estimate Std. Error z value Pr(>|z|)   p_m     L     U
(Intercept)           -2.792      0.195 -14.283        0 0.058 0.048 0.069
Caption_viewing_pre    1.232      0.247   4.991        0 0.774 0.728 0.814
Caption_viewing_post   2.663      0.222  11.973        0 0.935 0.920 0.947
```

To recap, I used a virtual coding scheme with a control condition and pretest as the reference level. Therefore, the intercept represents the estimated average score under the control condition. As there was no significant difference between the control pretest and posttest, the reduced variables were collapsed into the intercept term. The pattern effect 1 represents the adjustment of the intercept for the pretest under the Caption viewing condition, while the pattern effect 2 represents the adjustment of the intercept for the posttest under the Caption viewing condition. Therefore, the estimated average score under the control condition is 0.058, while the average score under the Caption viewing condition for the pretest is estimated to be 0.774, and for the posttest is estimated to be 0.935.

Now let us focus on the random effects section of the output.

```
Random effects:
 Groups Name                Variance Std.Dev. Corr
 Pid    (Intercept)         0.2922   0.5405
        Caption_viewing_pre 0.2820   0.5311   -1.00
        Caption_viewing_post 0.0589  0.2427   -0.64  0.65
 item   (Intercept)         0.6387   0.7992
        Caption_viewing_pre 1.3250   1.1511   -0.73
        Caption_viewing_post 0.6449  0.8031   -0.86  0.98
Number of obs: 4508, groups:  Pid, 49; item, 46
```

The "Groups" column lists the grouping factors (along with the residuals) that appear on the right side of the model specification. The "Name" column lists the effects grouped by each factor (i.e., the intercept and modal slopes that appear on the left side of the model specification). Each of these random intercepts and slopes has an associated variance (and standard deviation) estimate, which tells you the degree to which scores for specific stimuli and participants vary around the fixed intercept and slopes.

For example, the standard deviation of the random intercepts (highlighted in red in the above output) suggests that scores for specific items vary by approximately 0.5405 around the average intercept of -2.792. Similarly, the standard deviation of the random slopes for participants in the Caption viewing group (highlighted in red in the above output) indicates that the estimated slope for the pretest varies by approximately 0.531 around the average slope of 1.232. The estimated slope for the posttest among participants in the Caption viewing group varies by approximately 0.243 around the average slope of 2.663. Thus, the difference in slopes between pretest and posttest decreases, and the slope at posttest is more than twice as high as the slope at pretest, indicating that Caption viewing is highly beneficial for meaning recognition and improves convergence in participants' scores (slopes become steeper).

On the other hand, the slope at pretest is relatively flat and has a large variation. Individuals who are above 1 standard deviation from the mean have very steep slopes (indicating a difference of about 1.1 in the probability of obtaining the best and worst scores of 0.75). The coef() function in lme4 provides individual intercept and slope estimates for each participant and item, which not only helps to make the concept of random intercepts and slopes more concrete but also helps identify outliers. The code and brief output below show the estimates for the first four items and participants and the last four items and participants:

```
> b2=coef(meaning_reduced.mod3)
> b2
$Pid
    (Intercept) Caption_viewing_pre Caption_viewing_post
101      -3.083              1.5184                2.747
102      -2.450              0.8963                2.565
103      -3.649              2.0745                2.909
105      -3.649              2.0745                2.909
524      -3.267              1.6965                2.720
525      -3.267              1.6965                2.720
527      -3.413              1.8393                2.737
528      -3.559              1.9821                2.755

$item
            (Intercept) Caption_viewing_pre Caption_viewing_post
Accuse           -2.431            2.183681                3.054
Acoustic         -1.194           -0.155663                1.442
Alignment        -2.144            1.190405                2.443
Besiege          -2.540            1.609538                2.785
Vicinity         -2.850            1.114062                2.618
Withstand        -2.206            1.369605                2.556
Worship          -1.939           -0.828161                1.315
Zigzag           -1.672           -0.423094                1.448
```

This output indicates that the estimated intercept for the word "Acoustic" is -1.194 (probability of score 0.233), the estimated slope for Caption viewing at pretest is -0.156 (probability of score 0.461), and the estimated slope for Caption viewing at posttest is 1.442 (probability of score 0.809). These values are similar to the estimated fixed intercepts (-2.792, probability of score 0.058) and slopes (1.232, 2.663, probability of score 0.774 and 0.935). The participant section of the output shows that participants with ID numbers starting with 1 only appear in the control group, while those starting with 5 only appear in

the Caption viewing group. For example, participant 103 has an estimated intercept of -3.649 (probability of score 0.025) and estimated slopes of 2.0745 and 2.909 (probability of score 0.888 and 0.948) for pretest and posttest in the Caption viewing condition. This indicates that this individual's scores are lower than the average level in the control condition but higher than the average level in the Caption viewing condition, and the effect of the mode is smaller than the average level.

Please note that although we only looked at the estimated values for eight items and participants, it is evident that the variation in intercepts and slopes among different participants is smaller than the variation among different items. This observation is consistent with the standard deviations. Specifically, the standard deviations for random intercepts (0.541) and slopes (0.531, 0.243) across different participants are smaller than the standard deviations for random intercepts (0.799) and slopes (1.151, 0.803) across different items.

Although the assumed focus of the study is on fixed effects, the estimation of random effects itself is interesting and informative, and in some cases, it can provide insights into the key research question. Therefore, I argue that the core of the discussion lies in the random effects rather than the fixed effects.

The last piece of information in the random effects output relates to the correlations between random effects. In the "Corr" column, it is shown that the correlation between the random intercepts of items and the random slopes of Caption viewing at pretest items is -0.73, while the correlation between the random intercepts of participants and the random slopes of Caption viewing at pretest for individual participants is -1. This means that items that are more difficult to score in the control condition tend to have steeper (more positive, more steep) slopes. The correlation between the random slopes of Caption viewing at pretest and posttest is 0.98 for items, and 0.65 for participants. This suggests that the convergence of vocabulary scores in the Caption viewing condition is higher than the convergence of scores among participants.

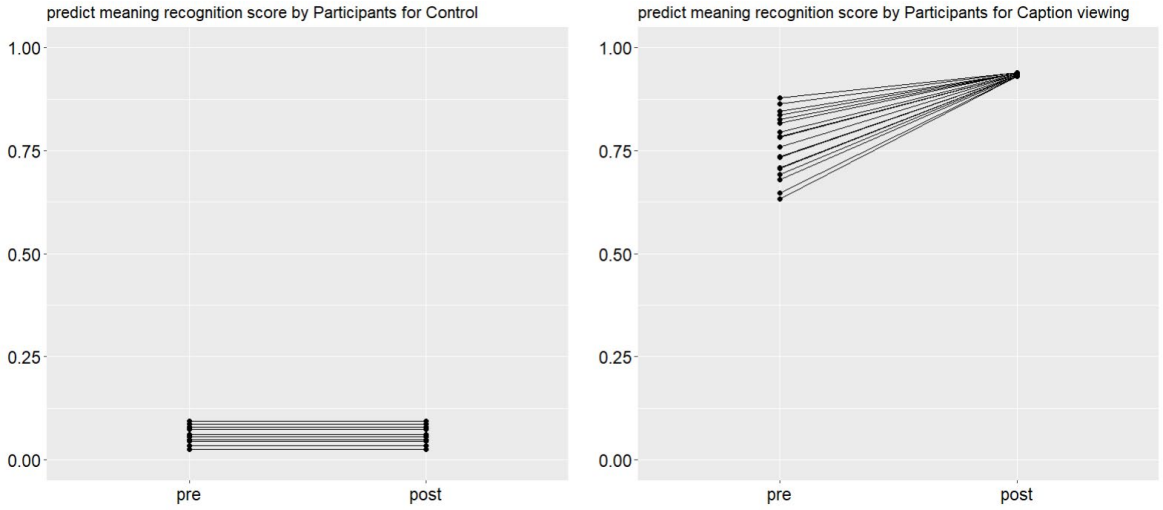
Finally, it should be noted that a model may encounter estimation issues (i.e., produce unreliable parameter estimates), and there may be no warning messages in the R console. The random effects section of the output contains some clues that can help you identify when this happens. One clue comes from the random effects correlations, which are set to -1.00 or 1.00 when they cannot be estimated. Another clue comes from the variance estimates, which are set to 0 when they cannot be estimated (i.e., when the variance and correlation parameters cannot be estimated, they are set to their boundary values; Bates et al., 2015). Although random effects correlations of -1.00 or 1.00 are usually accompanied by warnings of "singular fit," this is not always the case. Therefore, it is crucial to examine the random effects section of the model output to ensure smooth estimation.

Due to the different participants in different treatment groups, the issue of "singular fit" arises in the above analysis. To address this, a model that considers only random slopes for participants and ignores random intercepts is adopted. The reduced model is still preferred over the full model. The fixed effects part of the model is similar to the previous reduced model.

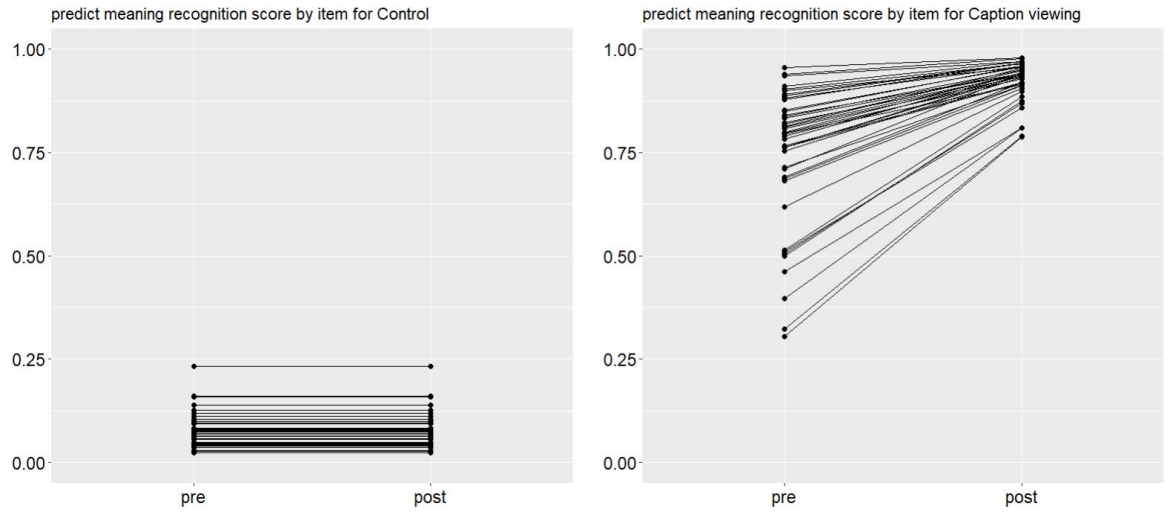
```
Fixed effects:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          -2.7920     0.1955 -14.283  < 2e-16 ***
Caption_viewing_pre   1.2322     0.2469   4.991 6.02e-07 ***
Caption_viewing_post  2.6632     0.2224  11.973  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) Cptn_vwng_pr
Cptn_vwng_pr -0.807
Cptn_vwng_ps -0.834  0.805
optimizer (bobyqa) convergence code: 0 (OK)
boundary (singular) fit: see help('isSingular')
```

In the random effects part of the model, it can be observed that individual differences have a relatively small impact on the scores, and the main source of score variation comes from the items, especially in the Caption viewing group for the pre-test and post-test. The standard deviation is greater than 1, indicating significant variability in scores.

```
Random effects:
 Groups Name                Variance Std.Dev. Corr
 Pid    (Intercept)         0.2922   0.5405
        Caption_viewing_pre  0.2820   0.5311   -1.00
        Caption_viewing_post 0.0589   0.2427   -0.64  0.65
 item   (Intercept)         0.6387   0.7992
        Caption_viewing_pre  1.3250   1.1511   -0.73
        Caption_viewing_post 0.6449   0.8031   -0.86  0.98
Number of obs: 4508, groups:  Pid, 49; item, 46
```



There is no significant difference between the pre-test and post-test scores for participants in the control group. In the Caption viewing group, there is a large individual difference in scores during the pre-test, but this difference becomes smaller during the post-test. Individuals who find the task easier have a flatter slope, while those who find it more challenging have a steeper slope. This suggests that captioned viewing is effective for all participants, with a stronger individual effect observed for those who struggle with the task.



In the control group, there is no significant difference between the pre-test and post-test items. The items that were easy to score in the pre-test remain easy in the post-test, while the items that were difficult to score in the pre-test remain difficult. The scores are concentrated, with only one item being easy to score with a probability of 0.25.

In the Caption viewing group, there is a larger gap between the easy and difficult items during the pre-test, but their probabilities are both above 0.25. During the post-test, the score differences become smaller and are concentrated above 0.75.

Overall, Caption viewing can reduce the score differences among individual participants as well as the score differences among items. It greatly improves the probability of scoring difficult items and benefits individuals who initially struggled with scoring. The impact of Caption viewing on item scores is much greater than its impact on individual scores.

```
> summary(meaning_reduced.mod4)Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [
glmerMod]
 Family: binomial  ( logit )
Formula: meaning ~ 1 + Caption_viewing_pre + Caption_viewing_post + (1 +
    Caption_viewing_pre + Caption_viewing_post | Pid) + (1 +
    Caption_viewing_pre + Caption_viewing_post | item)
   Data: data2
Control: glmerControl(optimizer = “bobyqa”)

     AIC      BIC   logLik deviance df.resid
  3861.5   3957.7  -1915.8   3831.5     4493

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.6780 -0.4550 -0.2741 -0.1536  6.3655

Random effects:
 Groups Name                Variance Std.Dev. Corr
 Pid      (Intercept)       0.2922   0.5405
          Caption_viewing_pre  0.2820   0.5311   -1.00
          Caption_viewing_post 0.0589   0.2427   -0.64  0.65
 item     (Intercept)       0.6387   0.7992
          Caption_viewing_pre  1.3250   1.1511   -0.73
          Caption_viewing_post 0.6449   0.8031   -0.86  0.98
Number of obs: 4508, groups:  Pid, 49; item, 46

Fixed effects:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          -2.7920     0.1955 -14.283  < 2e-16 ***
Caption_viewing_pre   1.2322     0.2469   4.991 6.02e-07 ***
Caption_viewing_post  2.6632     0.2224  11.973  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr) Cptn_vwng_pr
Cptn_vwng_pr -0.807
Cptn_vwng_ps -0.834  0.805
optimizer (bobyqa) convergence code: 0 (OK)
boundary (singular) fit: see help(‘isSingular’)
```

7 What Options Are There?

Collecting data in response to multiple trials is a common practice in experimental research. In such scenarios, it is important to choose the appropriate statistical analysis method. Repeated measures analyses of variance (ANOVAs) are often used in these cases. Standard ANOVAs and multiple regression violate a crucial assumption of many statistical tests: the independence assumption. This assumption stipulates that observations in a data set must be independent or should not be correlated. However, having multiple measurements taken on the same subject can result in the data not meeting this assumption. For example, in a reaction time study with multiple measurements, reaction times within a given participant and within an item will certainly be correlated. Thus, repeated measures ANOVAs are preferrable to standard ANOVAs and multiple regression in accounting for repeated measurements on the same subject.

However, repeated measures ANOVAs have some limitations. Although they can model either participant- or item-level variability, they cannot simultaneously take both sources of variability into account. This means that observations within a condition must be collapsed across either items or participants, causing a loss of important information about variability within participants or items, which in turn reduces statistical power (Barr, 2008). Repeated measures ANOVAs also deal with missing observations via listwise deletion, which can reduce sample size, leading to inflated standard error estimates and reduced statistical power (Enders, 2010). Furthermore, repeated ANOVAs assume that the dependent variable is continuous, and the independent variables are categorical. Experiments with categorical outcomes or continuous predictors require alternative techniques or adjustments, potentially reducing statistical power and making it difficult to model nonlinear relationships between predictors and outcomes (Royston et al., 2005). Finally, while repeated ANOVAs indicate whether an effect is significant, they do not provide information about the magnitude or direction of the effect, such as individual coefficient estimates for each predictor indicating growth or trajectory. Thus, when experimental studies involve subjects with a hierarchical structure, linear mixed-effects modeling (LMEM) is preferred over repeated measures ANOVAs.

8 What Issues Might There be and How Can We Resolve Them?

Model Advantages: LMEMs are more flexible, and they are more flexible compared to traditional linear models. The basic assumptions of the model are as follows: (1) The residuals can deviate from a normal distribution, (2) The variance can be homogeneous or heterogeneous, and (3) The samples are not independent (there are repeated measurements and internal correlations).

Disadvantages: The models can become overly complex, time-consuming, and may encounter convergence issues or warnings. It is difficult to perform model selection due to its complexity.

Overcoming Limitations: Efforts can be made to remove unrelated factors, eliminate non-significant variables, and simplify the model to achieve optimization. The model considers both the average effect of fixed factors on the dependent variable and the individual differences due to random factors. Thus, LMEM increases the difficulty of interpretation but makes the model more scientifically reasonable. It is challenging to explain the impact of fixed and random factors on the dependent variable. To address the limitations, researchers can adopt the following strategies: refining model specification, increasing the sample size, applying data transformation, identifying outliers, and exploring alternative modeling approaches, such as using generalized linear mixed-effects models (GLMM) for non-normally distributed data or hierarchical Bayesian models for complex data structures.

9 Concluding Remarks

In recent years, LMEM has gained significant popularity as a valuable tool for analyzing experimental data. These models offer distinct advantages due to the possibility to effectively capture and account for the variability that exists not only among participants but also among the items being studied. This comprehensive modeling approach sets them apart from other commonly employed statistical techniques, granting them greater flexibility and enhanced analytical power.

One key strength of LMEM is the capacity to handle missing observations in a robust manner. They possess the capability to handle data with missing values, allowing researchers to retain valuable information even in the presence of incomplete datasets. Additionally, these models seamlessly incorporate continuous predictors, enabling researchers to incorporate a wide range of relevant factors into their analyses.

A major benefit of LMEM is the ability to provide estimates of the average effects of predictors on outcomes, while also allowing for the examination of effects specific to individual participants and items. By accounting for both participant-specific and item-specific variations, researchers gain a more nuanced understanding of the underlying factors influencing the observed outcomes. Furthermore, LMEM can easily be extended to encompass categorical outcomes, providing researchers with a versatile framework for modeling various types of dependent variables commonly encountered in applied linguistics research.

The primary objective of this tutorial article is to offer an accessible and practical overview of LMEM to applied linguists who may be less familiar with this analytical approach. Through a detailed case study focusing on the potential of incidental vocabulary learning under captioned viewing, I aim to illustrate the practical application of mixed effects models in real-world research scenarios. The article emphasizes the fundamental concepts of mixed effects models, highlights their advantages over alternative analysis techniques, and provides step-by-step guidance on implementing these models using the statistical programming language R. It is my hope that this article will serve as a valuable resource, empowering researchers in applied linguists to leverage the full potential of mixed effects models in their research endeavors.

Availability of data and materials

The data and R script used to generate the models described in this article are available via OSF, at https://osf.io/9qwsy/

References

Baayen, R. H. (2010). A real experiment is a factorial experiment. *The Mental Lexicon, 5*(1), 149–157.

Barr, D. J. (2008). Analyzing "visual world" eyetracking data using multilevel logistic regression. *Journal of Memory and Language, 59*(4), 457–474.

Barr D. J., Levy R., Scheepers C., Tily H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language, 68*(3), 255–278.

Bates, D., & Maechler, M. (2010). Matrix: Sparse and dense matrix classes and methods. R package version 0.999375-43. http://cran. r-project. org/package= Matrix.

Bates D., Mächler M., Bolker B., Walker S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software, 67*(1), 1-48.

Brown, V. (2021). An Introduction to Linear Mixed-Effects Modeling in R. *Advances in Methods and Practices in Psychological Science, 4,* 1–19.

Enders, C. K. (2010). *Applied missing data analysis.* Guilford Press.

Matuschek H., Kliegl R., Vasishth S., Baayen H., Bates D. (2017). Balancing Type I error and power in linear mixed models. *Journal of Memory and Language, 94,* 305–315.

Mirman D. (2014). Growth curve analysis and visualization using R. Chapman and Hall.

R Core Team. (2020). R: A language and environment for statistical computing [Computer software]. R Foundation for Statistical Computing. http://www.R-project.org/

Royston, P., Altman, D. G., & Sauerbrei, W. (2005). Dichotomizing continuous predictors in multiple regression: A bad idea. *Statistics in Medicine, 25*(1), 127–141.

Simmons J. P., Nelson L. D., Simonsohn U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science, 22*(11), 1359–1366.

Singmann, H., & Kellen, D. (2019). An introduction to mixed models for experimental psychology. In D. Spieler & E. Schumacher (eds.), *New methods in cognitive psychology* (pp. 4-31). Routledge.

Snijders, T. A. B., & Bosker, R. J. (2012). *Multilevel analysis: An introduction to basic and advanced multilevel modeling* (2nd ed.). SAGE Publications.

Teng, M. F. (2021). *Language learning through captioned videos: Incidental EFL vocabulary acquisition.* Routledge.

Teng, M. F. (2022). Incidental L2 vocabulary learning from viewing captioned videos: Effects of learner-related factors. *System, 105,* 102736.

Teng, M. F. (2023). Incidental vocabulary learning from captioned video genres: vocabulary knowledge, comprehension, repetition, and working memory. *Computer Assisted Language Learning.* doi.org/10.1080/09588221.2023.2275158

Teng, M. F., & Cui, Y. (2023). Comparing incidental learning of single words and collocations from different captioning conditions: The role of vocabulary knowledge and working memory. *Journal of Computer Assisted Learning.* DOI: 10.1111/jcal.12910

Teng, M. F., & Cui, Y. (2025). Second language collocation learning through captioned videos: how do learners' vocabulary knowledge and working memory affect learning? *Computer Assisted Language Learning.* https://doi.org/10.1080/09588221.2025.2497495

Wickham H., Grolemund G. (2017). *R for data science: Import, tidy, transform, visualize, and model data* (1st ed.). O'Reilly Media.

Wickham H., Averick M., Bryan J., Chang W., McGowan L. D., François R., Grolemund G., Hayes A., Henry L., Hester J., Kuhn M., Pedersen T. L., Miller E., Bache S. M., Müller K., Ooms J., Robinson D., Seidel D. P., Spinu V., . . . Yutani H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686.

Mark Feng Teng is Professor in Applied Linguistics at Macau Polytechnic University. He was the recipient of the 2017 Best Paper Award from the Hong Kong Association for Applied Linguistics (HAAL). His research portfolio mainly focuses on L2 vocabulary acquisition, Reading and writing. His publications have appeared in international journals, including *Applied Linguistics, TESOL Quarterly, Language Teaching Research, System, Applied Linguistics Review, Computer Assisted Language Learning, Computers & Education, Foreign Language Annals,* and *IRAL*, among others. His recent monographs were published by Cambridge, Routledge, Springer, and Bloomsbury. He is currently editor-in-chief for International Journal of TESOL Studies (IJTS). ORCID: https://orcid.org/0000-0002-5134-8504