Linear Mixed-Effect Modeling in Kinesiology Research: A Tutorial in R

Jacob A Mota[1]*, Mehdi Rajeb[2], Kaiwen Man[2] and Kealey J Wohlgemuth[1]

[1]Department of Kinesiology and Sport Management, Texas Tech University, Lubbock, TX, USA

[2]Department of Educational Studies, University of Alabama, Tuscaloosa, AL, USA

*Corresponding author: Jacob A Mota, Assistant Professor of Kinesiology Director, Neuromuscular and Occupational Performance Laboratory, Department of Kinesiology and Sport Management, Texas Tech University, Box 43011, Lubbock, TX 79409, USA

ARTICLE INFO

ABSTRACT

The amount of kinesiology research produced has steadily risen over the previous decades. While much of the existing literature has utilized traditional hypothesis testing, the use of advanced statistical procedures is growing in popularity. Linear mixed-effect modeling is an extension of the traditional linear model and presents many advantages when used with study designs common in kinesiology or other intervention-based research with multiple time points. Like a linear model, mixed-effect modeling allows for the inclusion of fixed effects, and simultaneously incorporates random variabilities across subjects, termed random effects. Therefore, linear mixed-effect modeling has a broader range of applications compared to repeated measures analysis of variance. Unfortunately, in kinesiology, many research groups who are unfamiliar with this technique, or the software needed to run these models appropriately, may be slow to adapt. Therefore, the purpose of this report is to present the reader with a brief background on linear mixed effect modeling and walk them through an example analysis. Example R syntax is provided in addition to a Kinesiology specific dataset. This tutorial is not meant to serve as an in-depth discussion on the theory of linear mixed-effect modeling; it is geared towards being a primer on how to implement these procedures.

Keywords: Statistics; Exercise Science; Programming; Missing Data; Data Science

Introduction

The volume of kinesiology and strength and conditioning research has dramatically risen over the previous two decades. A concordant increase in repeated measure-based intervention trials has also been noted. Traditional null hypothesis testing has been the cornerstone statistical tool throughout this time. These common techniques (e.g., factorial analysis of variance [ANOVA], linear regression modeling), while popular and well implemented, are not without flaw. Specifically, traditional ANOVA's may be significantly impacted by missing data, the independence assumption, and unable to account for individual variability across participants. In accordance with the increased volume of research produced, there has been a concordant increased prevalence of advanced statistical such as linear mixed-effect modeling. Unlike traditional linear models (e.g., ANOVA), linear mixed-effect models investigate the "condition of interest", while simultaneously incorporating random variabilities across subjects that may be present within the data [1,2]. Linear mixed-models have been suggested as being superior to repeated measures ANOVA [3] for their ability to address missing data, clustering effects (which may bring bias in repeated measure ANOVA), or when the experimental design is unbalanced [4-6]. Unfortunately, this approach has not been fully adapted in the field of kinesiology, potentially due to the complexity of the modeling methodology. However, it has been suggested that guidance on proper implementations of novel statistical procedures may be useful [7,8]. Thus, the purpose of this manuscript is to serve as a tutorial on model genesis and interpretation. The authors intend to present the reader with the tools required to make their own decisions on how to build models which may fit their work best. We will carry out this task using the included kinesiology-specific example data.

Linear Mixed-Effect Modeling Theory

Traditional linear models (i.e., ANOVA) performs hypothesis testing by assessing the presence or absence of fixed main effects (e.g., "group" effect; "time" effect). This is often expanded to assess the presence of an interaction effect of two (or more) fixed main effects (e.g., group X time effect). Linear mixed-effect modeling expands upon traditional linear models to include both fixed and random effects [9,10]. In brief, the inclusion of random effects, instead of fixed effects alone, within a model may reduce:

A.   Type 1 and 2 error rates [5],

B.   Allow for missing data within a model [11], and

C.   May increase power in small sample research by better allocating degrees of freedom. For further reading on the advantages of including random effects within statistical modeling, the reader is led to other works [1,5,12,13].

Statistical Software

There are many powerful statistical software options available. Some employ "point-and-click" procedures (i.e., SPSS), while others use syntax-based approaches (i.e., SAS, R) to carry out instructed calculations. R is a programming language for statistical computing and graphics production. Unlike other statistical programs (e.g., SPSS, SAS), it is free and open source, which has a plethora of support available through the Comprehensive R Archive Network (CRAN). Due to the almost unlimited supporting packages available, R is endlessly expandable and capable of a wide variety of statistical and graphical tasks. R is also available on three popular computing operating systems (i.e., Windows, MacOS, UNIX). These features make it flexible across various operating systems and scientific disciplines, which may further the transparency and openness. However, some researchers may find "programming" in R intimidating.

Therefore, this tutorial provides investigators with not only the code, but hopefully an understanding of how to code linear mixed-effect models in R, how to appropriately prepare and visualize the data, and how to interpret the outcomes. The authors will attempt to annotate their code throughout, and the reader is encouraged to use the code we provide as a template for their future analyses. Please note, experienced R users may use a different syntax structure and/or style, and this is acceptable, as the end-product will remain unaffected. Other linear mixed-effect modeling packages may require different instructions; therefore the reader is directed to Bates, et al. [14] and Pinheiro, et al. [15] for additional details as necessary.

Getting Started in R

R may be used in conjunction with integrated development environments (IDE). An IDE is a software designed to facilitate program design to programmers (i.e., the reader). A popular and free IDE for R is RStudio, which has many additional tools that may help the reader overcome "coding" issues commonly feared. However, an IDE is not a replacement for R, nor does it make R a "point-and-click" software. Please note, an IDE is not required to use the code presented here, but the authors may highlight select areas that the reader may find helpful within RStudio. Included as an appendix file is the syntax used in this manuscript which can be opened in either R or RStudio.

Required Packages

The primary reason why R is so powerful is because the availability of a variety of built-in packages. A package is a bundle of pre-written code and functions which allow users to easily implement certain tools, tests, or graphics. For the purposes of our tutorial, our code requires the following packages: tidyverse, summarytools, plotrix, nlme, effects, & emmeans. These packages can be installed using the install.packages() function. In the below example, the authors install multiple packages at once with the use of the concatenate function, c(), within the install.packages() function where each package is within quotations (" ") and separated by a comma (,). Please note spelling and case is critical in R:

```
# text with the pound sign preceding it will not be read by R
# this feature is also great for notes at any time
## two pound signs usually precede R outputs
install.packages(c(«tidyverse»,»summarytools»,»plotrix»,
 «nlme», «effects», «emmeans»),
 dependencies = TRUE,
 repos = «http://cran.us.r-project.org»)
```

The reader will only need to install packages the first time they use R, or when they wish to use a new package. The authors will refrain from going into depth regarding all possible tools of each package. Rather, we will point the reader to what tools are from each package. The reader is encouraged to explore all aspects of each package on their own using "?packagename" function (i.e., "help" feature). Depending on which analyses the reader elects to run in the future, the reader may not require all these packages and/or may require others. After the required packages have been installed, they can be loaded by using library() for each package. The reader will need to do this task prior to using the functions within each package, but they will only require loading each package once within a given R session.

```
library(tidyverse) #multitude of tools for data wrangling
library(summarytools) #at a glance summary statistic
library(plotrix) #provides functions for custom figures
library(nlme) #package for linear mixed modeling
library(effects) #package to estimate effects of models
library(emmeans) #packages to estimate marginal means
```

Example Data Set

An example data set is included as an appendix file. The provided data is meant to serve as a useful example based on a common research design so that the reader can better utilize this tutorial. The example research design is a repeated measures study looking at the

effects of a resistance training intervention on muscle volume. The dataset (n = 40) includes a group (2 groups; intervention [n = 20], control [n = 20]) and time effects (5 time points; pre-, 3 mid-, and a post-testing). Time points are labeled as m0 (pre-testing), m1 (mid-test 1), m2 (mid-test 2), m3 (mid-test 3) and m4 (post-testing). The dependent variable is Muscle Volume and labeled as MuscleVol.

Loading in the Data

Loading data into R can be done in multiple ways. Our recommended way uses the read_excel() function from the readxl package to read in .csv files. Other file types (e.g., .txt, SAS, SPSS, Stata) are also compatible, but may require a different function. In RStudio, the reader is encouraged to use the "Import Dataset" tool found in the environment pane to help automate the process if desired. In the present example, we have loaded our example data into a data frame called "example.data", which is where R will call data from. Please note, we have named our example data as "example.data" for clarity, but the reader may rename this as they please. Also, the file path is specific to the lead author and will differ from that of the reader. Please carefully change this file path and use the specific file path for the desired file.

> *example.data <- readxl::read_excel(«C:/Users/jamota/2023 Mixed Effect Modeling Project/LME Example Data.xls")*

This syntax creates a data frame (example.data) from the "LME Example Data" file, found within the given path on the C: drive (the file location for the reader may vary). The data frame (example.data) will be used moving forward. If desired, the reader may wish to "see" the data. This can be done using view() to see all of the data. When you only wish to see some of the data, other functions may be more convenient; head() provides you with the first ten rows of data and tail() provides you with the final ten rows of data, colnames() gives you an output of all column names, which correspond to the available variables.

> *head(example.data)*
> *## # A tibble: 6 x 7*
> *## id group m0 m1 m2 m3 m4*
> *## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>*
> *## 1 1 1 190. 200. 206. 260. 236.*
> *## 2 2 1 195. 271. 225. 271. 272.*
> *## 3 6 1 280. 338. 371. 449. 486.*

> *## 4 8 1 244. 299. 279. 287. 350.*
> *## 5 10 1 251. 262. 257. 255. 264.*
> *## 6 11 1 143. 154. 170. 164. 209.*

At present the dataframe, example.data, is in wide format (i.e., each row of data contains all data for a given subject). However, many linear mixed-effects modeling approaches require data to be in long format (i.e., each row of the dataframe contains individual instances of data). R makes this process simple by using the reshape function:

> *example.data.long = reshape(data = as.data.frame(example.data),*
> *idvar = «id»,*
> *timevar = «measure»,*
> *v.names = «MuscleVol»,*
> *varying = list(c(«m0», «m1» ,»m2» ,»m3», «m4»)),*
> *times = c(0, 1, 2, 3, 4),*
> *direction = «long»)*

This step created a new data frame, example.data.long, which has data in long format. This was done using the reshape() function, apart of the tidyverse package. This formatting process also creates two new variables named "measure" and "MuscleVol" within the data frame example.data.long. The measure variable represents the categorical "time" variable corresponding to the aforementioned repeated measures assessments (i.e., pre-, 3 mid-, and a post-testing time-points). The MuscleVOL represents the muscle volume measure of interest. These two "new" variables can be named at the discretion of the reader. The data in both data frames are the same, just in different formats (i.e., wide vs. long). This can be confirmed again using the head() function:

> *head(example.data.long[,c(1,2,3,4)], n=10)*
> *## id group measure MuscleVol*
> *## 1 1 1 0 190.2513*
> *## 2 1 1 1 199.7280*
> *## 3 1 1 2 206.2445*
> *## 4 1 1 3 259.5733*
> *## 5 1 1 4 235.5904*
> *## 6 2 1 0 195.3600*
> *## 7 2 1 1 271.2829*
> *## 8 2 1 2 225.2292*
> *## 9 2 1 3 270.7539*
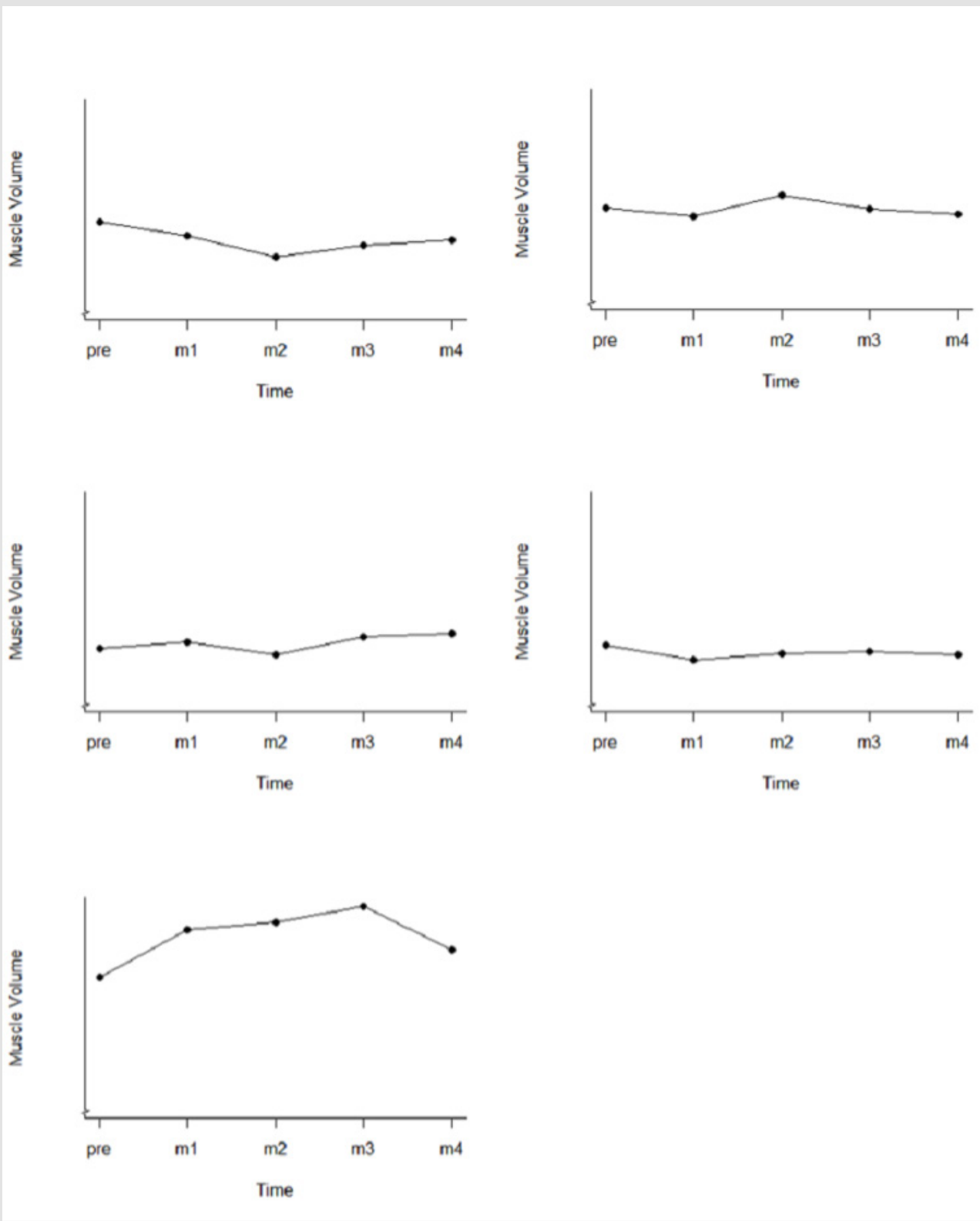> *## 10 2 1 4 272.2077*

Figure 1: Individual plots of 5 random samples across time in the example dataset.

Data Visualization

Prior to model genesis, proper visualization of the dataset is encouraged. We will use the base R plotting function plot() and the function ggplot() from the ggplot2 package. Both are valuable tools for exploring and creating publication-ready graphics with endless customization options. Sometimes it may be difficult to see individuals' growth trajectories clearly when there are many observations in the data. In these scenarios it is wise to take a few random samples of a smaller size. Several individual's data connected by lines across time is called a «spaghetti plot» because it often resembles that famed pasta. To get the program to create this type of graph correctly, it is critical that all NAs be removed from the dataset before graphing.

```
set.seed(32039) #used to standardize random variables
#The following chunk of code creates a "for" loop to pull 5 random samples
for(g in 1:5){
par(ask = TRUE)
childid = sample(n.dat$id, 1, replace = F)
lab.example.data = c(«pre»,»m1»,»m2», «m3»,»m4»)
### make a spaghetti plot using the base plot() function
plot(n.dat$measure, n.dat$MuscleVol, type = «n»,
xlab = «Time», ylab = «Muscle Volume»,
bty = «l», cex.lab = 1, pch = 21, xaxt = «n», yaxt = «n»)
for (I in unique(childid)){
points(n.dat$measure[n.dat$id==i], n.dat$MuscleVol[n.dat$id==i],
pch = 19)
lines(n.dat$measure[n.dat$id==i],    n.dat$MuscleVol[n.dat$id==i],
type = «l», lty = 1, lwd = 1.5)
 }
axis(1, at = c(0, 1, 2, 3, 4), labels = lab.example.data, tick = TRUE)
axis(2, at = seq(0, 40, 10), labels = TRUE, tick = TRUE)
axis.break(axis = 2, breakpos = 115, bgcol = «white», breakcol =
"black»,
style = «zigzag», brw = 0.02)
}
```

The above script provides the reader with 5 random samples responses across time. This screening procedure may be helpful for some investigators wishing to capture random "snapshots" of their data (Figure 1). Though other investigators may be more interested in grouped mean responses below.

```
### make a spaghetti plot for group1
set.seed(32039) #used to standardize random variables
childid = sample(n.dat$id, 30, replace = F)
```

```
lab.example.data = c(«pre»,»m1»,»m2», «m3»,»m4»)
plot(n.dat$measure, n.dat$MuscleVol, type = «n»,
xlab = «Time Period», ylab = «Muscle Volume»,
bty = «l», cex.lab = 1, pch = 21, xaxt = «n», yaxt = «n»)
for (j in unique(childid)){
points(n.dat$measure[n.dat$id==j &n.dat$group==1], #note two =
signs
n.dat$MuscleVOL[n.dat$id==j &n.dat$group==1], pch = 19)
lines(n.dat$measure[n.dat$id==j &n.dat$group==1],
n.dat$MuscleVOL[n.dat$id==j &n.dat$group==1], type = «l», lty =
1,lwd = 1.5)
 }
axis(1, at = c(0,1,2,3,4), labels = lab.example.data, tick = TRUE)
axis(2, at = seq(0,40,10), labels = TRUE, tick = TRUE)
axis.break(axis = 2, breakpos = 115, bgcol = «white», breakcol =
«black»,
style = «zigzag», brw = 0.02)
### make a spaghetti plot for group2
set.seed(32039) #used to standardize random variables
childid=sample(n.dat$id, 30, replace = F)
lab.example.data = c(«pre»,»m1»,»m2», «m3»,»m4»)
plot(n.dat$measure, n.dat$MuscleVol, type = «n», xlab = «Time
Period»,
ylab = «Muscle Volume», bty = «l», cex.lab = 1, pch = 21,
xaxt = «n», yaxt = «n»)
for (h in unique(childid)){
points(n.dat$measure[n.dat$id==h &n.dat$group==2], #note two
= signs
n.dat$MuscleVOL[n.dat$id==h &n.dat$group==2 ], pch = 2)
lines(n.dat$measure[n.dat$id==h &n.dat$group==2],
n.dat$MuscleVOL[n.dat$id==h &n.dat$group==2],
type = «l», lty = 3, lwd = 1.5)
 }
axis(1, at = c(0,1,2,3,4), labels = lab.example.data, tick = TRUE)
axis(2, at = seq(0,40,10), labels = TRUE, tick = TRUE)
axis.break(axis = 2, breakpos = 115, bgcol = «white», breakcol =
«black»,
style = «zigzag», brw = 0.02)
```

In the above code chunks, the inclusion of the "n.dat$group==1" or "n.dat$group==2" arguments instruct the program to select groups as desired (Figure 2). The line type is adjusted with the "lty = " argument and the points can be modified using the "pch= " argument. This can be a quick resource for investigators wishing to examine, at a glance, the grouped mean responses across time. Many more options exist for plotting options but are outside the scope of this tutorial.
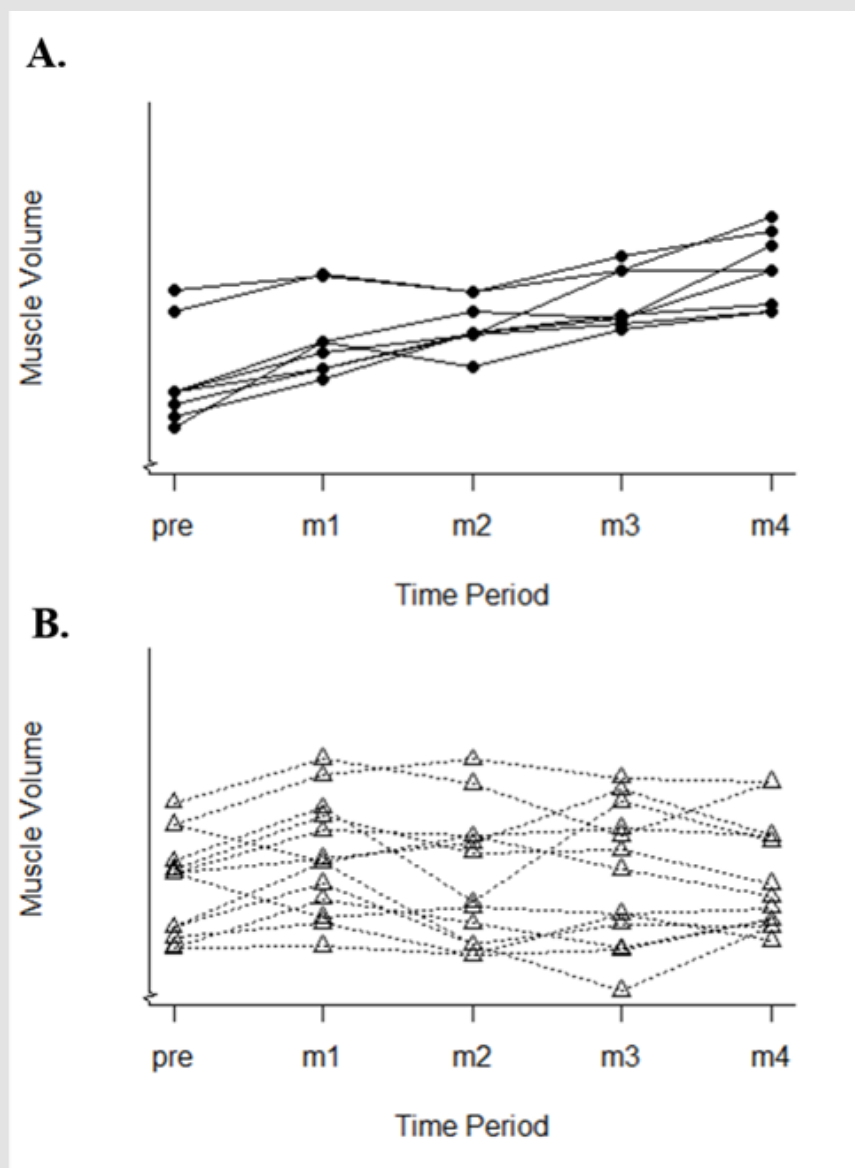
Figure 2: Spaghetti plot across time for those in group 1 (A.) and group 2 (B.).

Setting up the Linear Mixed-Effect Model

To set up the linear mixed-effect model, we suggest using the lme() function from the nlme package. Similar operations can also be done using other packages such as lme4, but please note the syntax will vary. To begin, we will instruct R to create a linear mixed-effect model into an object named "vol.mod.1". The choice of name was intentionally designed to be informative of the variable (vol = volume), procedure (i.e., mod = model), and iteration of syntax (1 = first iteration) though this may be named at the behest of the reader.

*vol.mod.1 <- lme(fixed = MuscleVol ~ measure + group + measure*group,*

data = example.data.long,
na.action = na.omit,
method = «REML»,
random = ~ 1 + measure|id)

The reader will note the inclusion of fixed effects (i.e., fixed = ...), specification of which data frame to use (i.e., data = ...), variance estimation (i.e., method = "REML"), followed by the random effect (i.e., random = ...). The random effect notation is separated by a pipe (|). The value on the left of the pipe is the effect to vary (i.e., intercept, slope), and the right side of the bar names the variable for which the random factor will vary (e.g., subject). The listed term (mesure|id) allows the intercepts (i.e., 1) and slopes to vary by subject (i.e., id)

across each measurement (i.e., measure). This model is stored in the object "vol.mod.1".

Linear Mixed-Effect Model Output

In the present example, the results from the linear mixed-effect model are stored in the object "vol.mod.1". The authors recommend using the summary() function to view the results. The summary() output provides the reader with indications of model fit, random effects, fixed effects, correlation structure, within-group residuals, and the number of observations and groups. We can examine some aspects of the summary(vol.mod.1) output further.

Summary(vol.mod.1)

Linear mixed-effects model fit by REML
Data: example.data.long
AIC BIC logLik
2023.233 2049.458 -1003.617
##
Random effects:
Formula: ~1 + measure | id
Structure: General positive-definite, Log-Cholesky parametrization
StdDev Corr
(Intercept) 47.11404 (Intr)
measure 10.67873 0.142
Residual 26.38440
##
*## Fixed effects: MuscleVol ~ measure + group + measure * group*
Value Std.Error DF t-value p-value
(Intercept) 179.54271 25.677883 158 6.992115 0.0000
measure 48.54203 6.775856 158 7.163970 0.0000
group 29.74666 16.240119 38 1.831677 0.0748
measure:group -23.77184 4.285428 158 -5.547133 0.0000
Correlation:
(Intr) measure group
measure -0.097
group -0.949 0.092
measure:group 0.092 -0.949 -0.097
##
Standardized Within-Group Residuals:
Min Q1 Med Q3 Max
##-2.44098589 -0.48227564 -0.06876291 0.55239797 2.49035652
##
Number of Observations: 200
Number of Groups: 40

Assessing Model Fit

Linear mixed-effects model fit by REML
Data: example.data.long
AIC BIC logLik
2023.233 2049.458 -1003.617

In the analytical procedure of estimating a linear mixed-effect model, the first approach is to investigate one of the various model fit statistics (i.e., AIC, BIC, LogLik). In practice, Akaike's Information Criterion (AIC) is often used to understand the goodness of fit of a linear mixed-effect model [16]. However, when AIC is calculated from a single model it cannot be interpreted as a model fit measure. It is a relative fit measure, meaning multiple models could be compared based on multiple AICs, and the lowest AIC of any model indicates the best fit of that model. Said another way, the model with the least AIC value is usually considered as the optimized model. Hence, AIC is offering us a way to identify the optimized model if multiple predictor variables are available. To obtain better (lower) AIC, we should remember that parsimonious models are more advantageous (i.e., explaining outcomes with the least number of variables will provide us with lower AIC). For additional details on model fit, the reader is guided to Liu, et al. [17].

Random Effect Output and Interpretations

Random effects:
Formula: ~1 + measure | id
Structure: General positive-definite, Log-Cholesky parametrization
StdDev Corr
(Intercept) 47.11404 (Intr)
measure 10.67873 0.142
Residual 26.38440

Before examining the random effect, we remind the reader that the model is attempting to explain the variation of muscle volume (MuscleVol) by the intervention. MuscleVol was measured in five measurement periods, with individuals in both treatment and control groups. Despite the study design measuring intervention effects by grouping study participants into separate groups, it is valuable to understand the individual-level effects across all participants in the model, and to appreciate these individual effects are rather random. Hence, the random effects measures can provide insights about [16] the variance of intercepts across all individual subjects, and [4] the variability in the slope across all individual subjects [18].

In our example, we allowed individual intercepts and slopes to vary from individual to individual across each measurement period. The standard deviation associated with the intercept is 47.11, and it suggests that the variation in the individual (subject) intercepts is fitted with a standard deviation of 47.11 from the mean intercept 179.54 (the mean intercept is discussed in the fixed effect model section below). Secondly, we see there is another standard deviation listed just beside the "Measure" variable. The standard deviation of Measure provides us with the variation of the slope at individual levels across all measurement periods. In our example, the standard deviation associated with Measure is 10.68. This means variation in the slopes of all individuals within the dataset is estimated with

a standard deviation of 10.68 from the mean point 48.54 (the mean point is mentioned in the fixed effect model). For further reading, the reader is directed to the example provided by Brown [19].

Fixed Effect Output and Interpretations

*## Fixed effects: MuscleVol ~ measure + group + measure * group*
Value Std.Error DF t-value p-value
(Intercept) 179.54271 25.677883 158 6.992115 0.0000
measure 48.54203 6.775856 158 7.163970 0.0000
group 29.74666 16.240119 38 1.831677 0.0748
measure:group -23.77184 4.285428 158 -5.547133 0.0000

Fixed effect estimates obtained during the linear mixed-effect model estimation process are generally understood as we interpret traditional regression models, but the reader should be cautious when interpreting the average effects of each covariate in the presence of their interactions [19]. In the present example, MuscleVol was explained by measurement periods, groups, and measurement period by group interaction. Hence, the authors will attempt to interpret the effect of each factor and the significance of each explanatory variable considered in the model. The regression coefficients can be examined to understand the effects of each variable. The intercept of the model is often interpreted as the average level of the dependent variable when all other predictor variables (i.e., regression coefficients) assume value of 0s. The respective p-values of each regression coefficient are derived from a t-test and may be used to understand whether the factor variable has any significant effect on the dependent variables. The null hypothesis being tested here is that the regression coefficients (e.g., measure, group) = 0. Lower p-values (<0.05) indicate failure to accept the null hypothesis. In other words, when the null hypothesis is rejected there is not evidence to suggest the regression coefficient is meaningfully different than 0. For additional discussion on p-values. please see Thiese, et al. [20].

In our model, all factors (e.g., measure, group, and the interaction between measure and group) have a significant effect on the dependent variable, MuscleVol. The intercept value is 179.54, which indicates that on an average an individual in the study has MuscleVol of 179.54 cm^3, when there is no effect from measure, group, or their interaction. The coefficient value of the variable "group" is 29.75, which means individuals in group 1 had greater MuscleVol (29.75 cm^3) compared to individuals in group 2 when the measurement period was zero.

Interpretation of the interaction effects should be performed with caution, as it is heavily dependent on the coding structure [19]. For instance, the reader may be interested in the interaction term to examine the effect of one variable on the value of another variable. An estimated positive effect in interaction term could be interpreted as an increase in one variable will proportionally increase the other variable towards the outcome variable. In our example, we obtained

-23.77 as the coefficient of the interaction term. This indicates that if we increase one variable of the interaction by 1 unit, the effect of the other variable will decrease, which creates a joint negative marginal effect of 23.77 cm^3 to the muscle volume.

Correlation Interpretations

Correlation:
(Intr) measure group
measure -0.097
group -0.949 0.092
measure:group 0.092 -0.949 -0.097##

The estimated correlation matrix from the fixed-effect model may not be interpreted as the correlation that is traditionally thought of, rather it provides approximate correlations between estimated regression coefficients. In this instance, a positive correlation of the effect of a variable with intercept ("Intr") indicates that a larger value of the intercept will influence a larger change in the slope of that variable, and a higher value correlation signifies a higher impact. A reverse interpretation could be done for negative values of these correlation coefficients. For example, the correlation between measure and intercept is -0.097, which indicates that with higher initial MuscleVol of an individual will lead to a slight decrease in the rate of gaining muscle volume. The other correlation coefficients could be interpreted in the similar manner, but the reader should be careful about interpreting these correlations of effects.

Post Hoc follow-Up

Given the interaction effect present, the reader may wish to explore this effect in more detail. A useful first step may be to closely examine the 95% confidence intervals (CI), which may be done using the intervals() function:

Approximate 95% confidence intervals
##
Fixed effects:
lower est. upper
(Intercept) 128.826525 179.54271 230.25889
measure 35.159091 48.54203 61.92497
group -3.129746 29.74666 62.62306
measure:group -32.235953 -23.77184 -15.30772
attr(,»label»)
[1] «Fixed effects:»

This function provides the reader with the estimated marginal effect (i.e., est.) and 95% CI (i.e., lower, upper) for each main effect (measure, group). The authors recommend the use of 95% CI which do not include 0 over relying on p-value thresholds, as this may make for easier interpretation of model parameters. In the above example, the interaction effect (measure:group) has an estimated marginal effect of -23.77 cm^3 and has a 95% CI of -32.24 – -15.31 cm^3. This

parameter suggests that there was a significant interaction effect present based on measurement number (i.e., testing day) and group inclusion (i.e., intervention vs. control). In order to get a clear idea of which direction these data may lead the reader, we encourage the use of the effect() function:

    effect(«measure*group», vol.mod.1)
    ##
    ## measure*group effect
    ## group
    ## measure 1   2
    ## 0 209.2894 239.0360
    ## 1 234.0596 240.0344
    ## 2 258.8297 241.0327
    ## 3 283.5999 242.0311
    ## 4 308.3701 243.0294

This function selects the interaction effect (measure*group) within our model of interest (vol.mod.1) and calculates the specific effects modeled. This output allows us to clearly see an effect for group 1 (i.e., intervention) measurements. To explain the interaction effect of measure across groups, we re-organized the marginalized effects obtained from the effect(measure* group) command (Table 1). In Table 1, all the marginal effects of different combinations of groups and measures are presented. In the present example, group 2 is observing a negative impact of time compared to group 1, and the rate of this effect is -23.77 cm^3 (i.e., interaction effect). The reader may also notice that at the initial level average muscle volume in the control group (239.04 cm^3) was higher than the intervention group (209.29 cm^3), but across time the muscle volume in those in the intervention group increased. The negative sign in the interaction effects arose due to coding only, which may encourage the reader to always display their data in tabular or graphical format. The authors believe this approach may increase clear outcomes associated with study future studies.

Table 1: Marginal effect of group across measurement time points.

Timepoint	Group		Difference in group wise muscle volume in different measurement periods	Rate of change in the difference (i.e., interaction effect)
	1	2		
0	209.29	239.04	-29.75	
1	234.06	240.03	-5.97	-23.77
2	258.83	241.03	17.8	-23.77
3	283.59	242.03	41.56	-23.77
4	308.37	243.03	65.34	-23.77

Note: Group 1 = intervention; Group 2 = control.

Common Pitfalls in Linear Mixed Modeling in R

There are certain aspects of linear mixed-effect modeling for longitudinal data that the reader may consider before modeling on their own. This tutorial is not probing into major theoretical modeling issues, rather attempting to provide a general guideline pointing out a few ideas.

A. Selecting variables for random effects requires a clear definition of the variable and its probable random effects. Due to mathematical limits in variability, it is sometimes disadvantageous to define a categorical variable with only two levels as the source of randomness (e.g., sex). Please see Brauer, et al. [21] for a detailed discussion.

B. The reader is encouraged to remain diligent when writing code. R is sensitive to case structure and placement of variables within functions. It is common for R users of all experiences to have errors in their code which may result in errors, especially when specifying differing factors or variables. The RStudio IDE may help some readers with keeping their code written without mistakes.

C. Some of the mixed-effect model estimates may not have the same meaning as traditional multiple regression models. The reader should express caution when interpreting model estimates, including interaction effect estimates, and remember these interpretations are made in the presence of random effects [19].

How to Report the Results from a Mixed-Effect Model? An Example

Arguably, the most critical portion of statistical modeling is the proper interpretation of the outcomes. Although the authors expanded upon each segment of the model above, we believe a sample report may assist help our readers in their future works. Therefore, when writing the results sections, we encourage the reader to remain parsimonious and specific.

Sample Results

The outcomes of the linear mixed-effect model suggest that there is a significant measure by group interaction effect present, with an estimated marginal effect of -23.77 cm^3 (95% CI = -32.24 – -15.31 cm^3). The model also revealed significant main effects for measurement period (estimated marginal mean = 48.54 cm^3; 95% CI = 35.16 – 61.92 cm^3), but not group (estimated marginal mean = 29.75 cm^3; 95% CI = -3.13 – 62.62 cm^3). When looking at the measure by group interaction effect more closely, the estimated marginal effects suggest an increase in muscle volume of 47.3% in the intervention group and a 1.67% change in muscle volume for those in the control group. This sample model report is based on only basic estimates that the reader may think of incorporating in his or her research. However, there are many other estimates mentioned in the model output which may also be reported based on the necessity and the pattern of the research questions.

Conclusion and Final Notes

Over the next two decades, linear mixed-effect models may gain momentum in various fields [22]. The modeling approach is capable of analyzing complex data structures incorporating random variance components [9]. Conveniently, many software are currently available to overcome the complexity of this modeling approach [23]. More importantly, this modeling approach is well equipped to manage longitudinal data common within kinesiology research and sport science settings, especially for data collected from experimental designs with random effects spanning over multiple time periods. Furthermore, mixed modeling approach may overcome the issues of missing data and unbalanced designs. Hence, this tutorial aims to aid our readers to design a longitudinal experiment and help them to appropriately estimate effects for the data. Based on the advantages discussed above, the authors urge research groups to consider this modeling approach in their future works.

Acknowledgment

Conflict of Interest

The authors declare no conflicts of interest.

References

1. Bates D, Machler M, Bolker B, Walker S (2014) Fitting linear mixed-effects models using lme4. arXiv preprint arXiv: 1406: 5823.

2. West BT, Welch KB, Galecki AT (2006) Linear mixed models: a practical guide using statistical software. Chapman and Hall/CRC.

3. McCulloch CE (2005) Repeated Measures Anova, RIP? Chance 18(3): 29-33.

4. Baayen R (2008) Analyzing Linguistic Data: A Practical Introduction to Statistics Using R. Cambridge University Press.

5. 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.

6. Pinheiro J, Bates D (2006) Mixed-effects models in S and S-PLUS. Springer science & business media.

7. Beck TW (2013) The importance of a priori sample size estimation in strength and conditioning research. J Strength Cond Res 27(8): 2323-2337.

8. Casals M, Girabent Farres M, Carrasco JL (2014) Methodological quality and reporting of generalized linear mixed models in clinical medicine (2000–2012): a systematic review. PloS one 9(11): e112653.

9. Nagle C (2018) An Introduction to Fitting and Evaluating Mixed-effects Models in R.

10. Singmann H, Kellen D (2019) An introduction to mixed models for experimental psychology, in: New methods in cognitive psychology. Routledge p. 4-31.

11. Ibrahim JG, Chu H, Chen MH (2012) Missing data in clinical studies: issues and methods. Journal of clinical oncology 30(26): 3297-3303.

12. Luke SG (2017) Evaluating significance in linear mixed-effects models in R. Behavior research methods 49(4): 1494-1502.

13. Winter B (2013) A very basic tutorial for performing linear mixed effects analyses. arXiv preprint arXiv:1308: 5499.

14. Bates D, Sarkar D, Bates MD, Matrix L (2007) The lme4 package. R package version 2: 74.

15. Pinheiro J, Bates D, DebRoy S, Sarkar D, Heisterkamp S, et al. (2017) Package 'nlme'. Linear and nonlinear mixed effects models, version 3.

16. Akaike H (1998) Information theory and an extension of the maximum likelihood principle. Selected papers of hirotugu akaike Springer pp. 199-213.

17. Liu S, Rovine MJ, Molenaar P (2012) Selecting a linear mixed model for longitudinal data: repeated measures analysis of variance, covariance pattern model, and growth curve approaches. Psychological methods 17(1): 15-30.

18. Borenstein M, Hedges LV, Higgins JP, Rothstein HR (2010) A basic introduction to fixed-effect and random-effects models for meta-analysis. Research synthesis methods 1(2): 97-111.

19. Brown VA (2021) An Introduction to Linear Mixed-Effects Modeling in R. Advances in Methods and Practices in Psychological Science 4(1): 2515245920960351.

20. Thiese MS, Ronna B, Ott U (2016) P value interpretations and considerations. Journal of thoracic disease 8(9): E928-E931.

21. Brauer M, Curtin JJ (2018) Linear mixed-effects models and the analysis of nonindependent data: A unified framework to analyze categorical and continuous independent variables that vary within-subjects and/or within-items. Psychological Methods 23(3): 389-411.

22. Kuznetsova A, Brockhoff PB, Christensen RH (2017) lmerTest package: tests in linear mixed effects models. Journal of statistical software 82(13): 1-26.

23. Tanaka E, Hui FK (2019) Symbolic Formulae for Linear Mixed Models. Presented at Research School on Statistics and Data Science p. 3-21.

Assets of Publishing with us

- Global archiving of articles
- *Imm*ediate, unrestricted online access
- Rigorous Peer Review Process
- Authors Retain Copyrights
- Unique DOI for all articles

https://biomedres.us/