

Mixed ANOVA Design in R

Ax(Bxs)

Bruce Dudek

2023-04-17

Contents

1	Introduction	2
2	The R Environment	2
3	Load the Primary Data Set	2
4	Descriptive Statistics and EDA	3
5	Perform the base ANOVA for the two-factor Mixed design	8
5.1	Use the base system <code>aov</code> function	8
5.2	Use the EZ package for the mixed ANOVA	10
5.3	The rstatix package provides another approach	11
5.4	Use the afex package for the basic ANOVA	12
6	Assumptions	15
6.1	Box's M test for homogeneity of covariance matrices across groups	15
7	Additional Methods following up on Omnibus ANOVAs: Contrasts and Simple Effects.	15
7.1	Interaction Contrasts	15
7.2	Main Effect Contrasts	16
7.3	Follow up analyses using emmeans for Simple Main Effects	16
7.3.1	First extract the matrix of means for SME	16
7.3.2	Simple effects of Angle at levels of age.	17
7.3.3	Simple main effects of Age, the BG factor at levels of the repeated measure factor. . .	18
7.3.4	Contrasts on simple main effects	19
7.4	A manual approach to performing trend analysis on the Simple Main Effects	19
7.5	Comments on use of emmeans	20
7.6	Simple main effects using the rstatix omnibus model fit object	21
7.6.1	Contrasts in the rstatix models	22
7.7	Are there other approaches to followup analyses?	22
8	Linear Mixed Effects approaches.	22
8.1	Using lme	22
8.1.1	Finding the appropriate omnibus model	22
8.1.2	Simple main effects on the lme object	23
8.2	Linear Mixed Models with lmer and phia	23
9	Short summary and recommendations	24
10	Documentation for Reproducibility	24

1 Introduction

This tutorial document attempts to outline multiple methods for analysis of a “mixed” between-within subjects design. This document covers a simple design with one BG factor and one WG factor. It is intended for students in the APSY510/511 statistics sequence at the University at Albany, but can have general utility for researchers and students with some training in repeated measures analyses.

The general philosophy in approaching factorial designs is to follow omnibus analyses with examination of a mix of contrasts and simple effects. In this mixed design, this means that simple effects (and contrasts) can be either effects of the BG factor or of the WG (repeated) factor. Such contrasts can be performed on interactions, simple main effects, or main effects. These kinds of effects are best approached with a strategy that involves use error terms tailored to the specific effect - usually called “specific error terms”. These types of follow up analyses are not easily accomplished with the tools available in R. Some approaches outlined below have led to questionable error term choices. Others are appropriate. The best practices approach using specific error terms was strongly recommended by Boik (1981) and Rogan et al (Rogan, Keselman, & Mendoza, 1979) More recent writers/textbooks have urged this approach and it has been adopted in major commercial software. These are challenging in R and a primary reason why results from R will often not match those that users may be accustomed to from SPSS (MANOVA and UNIANOVA), and SAS. An analogous issue arises when attempting to test post hoc questions with pairwise comparison methods - error term choices are challenging in R. This document lays out strategies for several approaches - no single integrated/comprehensive set of functions provides results that make sense at the time of writing of this document.

This document uses the Maxwell/Delaney/Kelley (2017) textbook example (3rd ed, section beginning pg 688). We have already covered the approach to this design with SPSS MANOVA/UNIANOVA/Mixed.

2 The R Environment

Several packages are required for the work in this document. They are loaded here, but comments/reminders are placed in some code chunks so that it is clear which package some of the functions come from. Either functions are specified with the `pkgname::functionname` code style or the package is mentioned in accompanying text.

```
library(afex,quietly=TRUE, warn.conflicts=FALSE)
library(car,quietly=TRUE, warn.conflicts=FALSE)
library(rstatix,quietly=TRUE, warn.conflicts=FALSE)
library(Rmisc,quietly=TRUE, warn.conflicts=FALSE)
library(emmeans,quietly=TRUE, warn.conflicts=FALSE)
library(ggplot2,quietly=TRUE, warn.conflicts=FALSE)
library(gt,quietly=TRUE, warn.conflicts=FALSE)
library(knitr,quietly=TRUE, warn.conflicts=FALSE)
library(dplyr,quietly=TRUE, warn.conflicts=FALSE)
library(ez,quietly=TRUE, warn.conflicts=FALSE)
library(phia,quietly=TRUE, warn.conflicts=FALSE)
library(nlme,quietly=TRUE, warn.conflicts=FALSE)
library(lme4,quietly=TRUE, warn.conflicts=FALSE)
library(psych,quietly=TRUE, warn.conflicts=FALSE)
```

3 Load the Primary Data Set

The long-format data file is available in .csv format: “mixedmd_long.csv”

```

#mixed1.df <- read.csv(file.choose(),header=T, stringsAsFactors=TRUE)
mixed1.df <- read.csv("mixedmd_long.csv",header=T, stringsAsFactors=TRUE)
# change the snum variable to a factor variable (was numeric)
mixed1.df$snum <- as.factor(mixed1.df$snum)
mixed1.df$angle <- ordered(mixed1.df$angle, levels=c("zero", "four", "eight"))
# look at the structure
str(mixed1.df)

```

```

## 'data.frame': 57 obs. of 4 variables:
## $ snum : Factor w/ 19 levels "2","3","4","5",...: 1 1 1 2 2 2 3 3 3 4 ...
## $ age : Factor w/ 2 levels "older","young": 2 2 2 2 2 2 2 2 2 ...
## $ angle: Ord.factor w/ 3 levels "zero"<"four"<...: 1 2 3 1 2 3 1 2 3 1 ...
## $ rt : int 390 480 540 570 630 660 450 660 720 510 ...

```

Next, it is helpful to change the order of levels of the factors to match how we saw them ordered in our previous analyses - recalling that R orders them alphabetically by default. This will also help with graph labels.

```

mixed1.df$age <- ordered(mixed1.df$age, levels=c("young", "older"))
mixed1.df$angle <- ordered(mixed1.df$angle, levels=c("zero", "four", "eight"))
str(mixed1.df)

```

```

## 'data.frame': 57 obs. of 4 variables:
## $ snum : Factor w/ 19 levels "2","3","4","5",...: 1 1 1 2 2 2 3 3 3 4 ...
## $ age : Ord.factor w/ 2 levels "young"<"older": 1 1 1 1 1 1 1 1 1 ...
## $ angle: Ord.factor w/ 3 levels "zero"<"four"<...: 1 2 3 1 2 3 1 2 3 1 ...
## $ rt : int 390 480 540 570 630 660 450 660 720 510 ...

```

Examine the first and last few cases in the data frame.

```
psych::headtail(mixed1.df)
```

```
## Warning: headtail is deprecated. Please use the headTail function
```

```

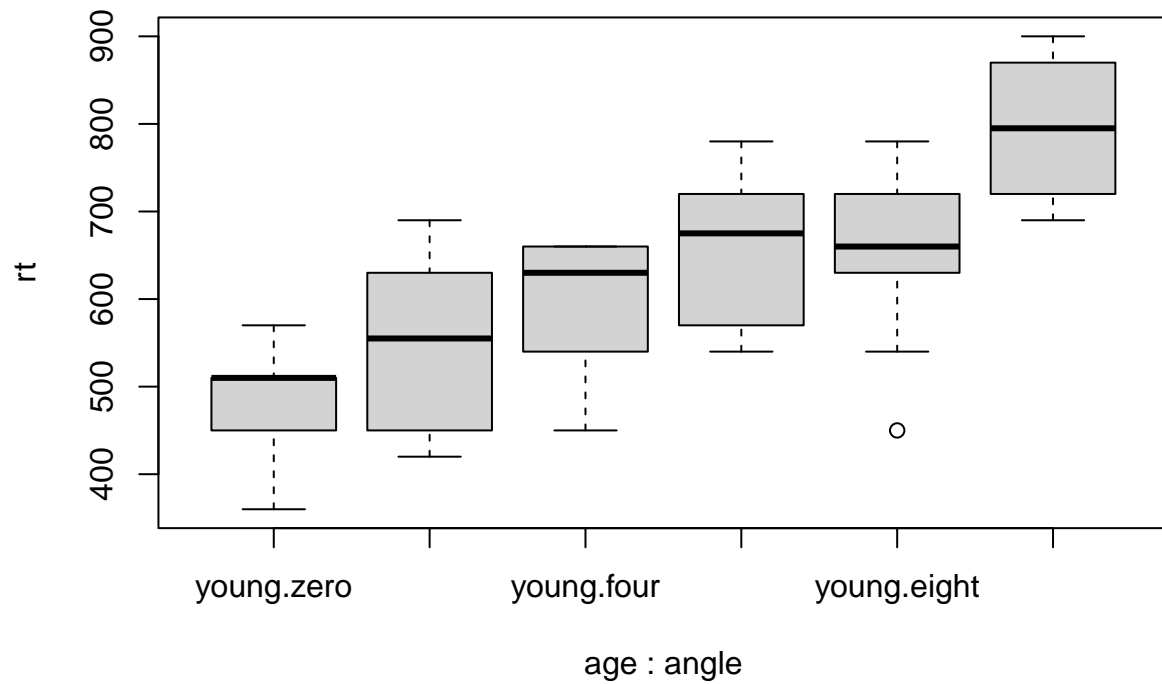
##      snum  age angle  rt
## 1      2 young  zero 390
## 2      2 young  four 480
## 3      2 young eight 540
## 4      3 young  zero 570
## ... <NA> <NA> <NA> ...
## 54     19 older eight 900
## 55     20 older  zero 510
## 56     20 older  four 690
## 57     20 older eight 810

```

4 Descriptive Statistics and EDA

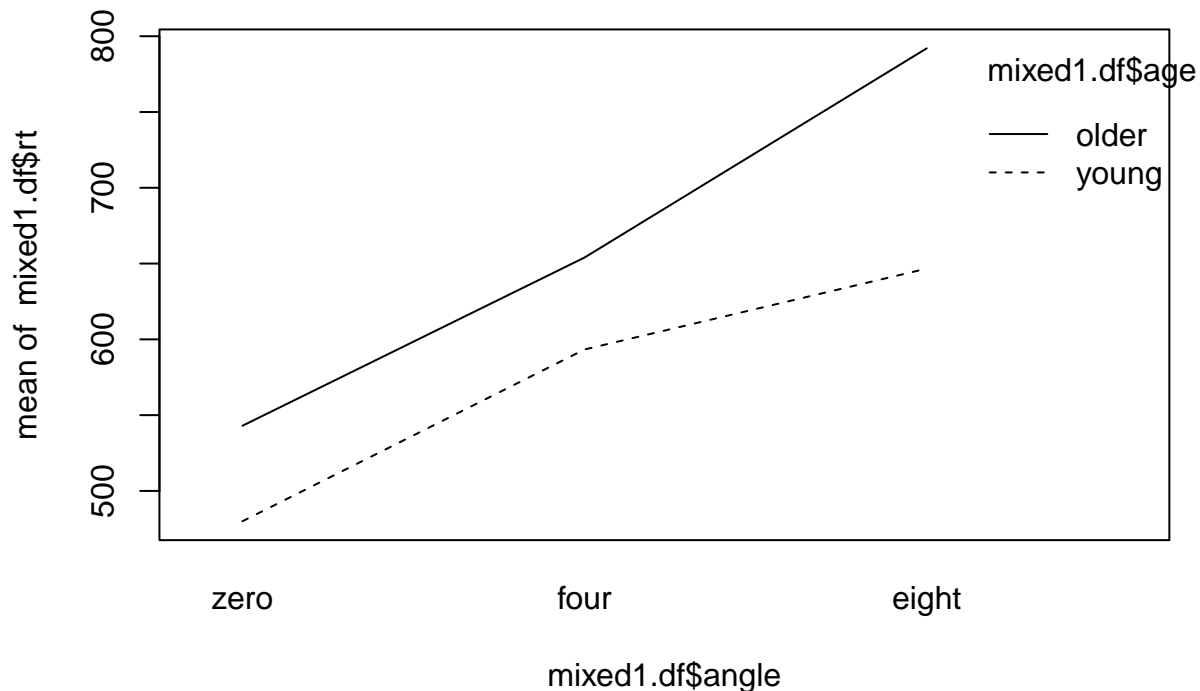
We can quickly obtain a box plot with base system graphics. It is not publication quality, but it is a quick way to examine the data set. Notice that the condition labels are not all visible. This is because they are too wide. A better approach would be use of ggplot to create a clustered boxplot graph - see below.

```
boxplot(rt~age*angle,data=mixed1.df)
```



A quick way to obtain a line graph is possible with the `interaction.plot` function. Again, not publication quality, but a good EDA tool. Note that a line graph is acceptable here since the X axis is a numeric scale.

```
interaction.plot(mixed1.df$angle, mixed1.df$age, mixed1.df$rt)
```



Numerical summaries can be obtained a variety of ways, and will be needed for **ggplot2** graphs. Initially, we will use the `summarySE` function from the **Rmisc** package. This function provides standard std errors and CI's that may not be appropriate for repeated measures designs. The means are in the column labeled as the dv name

```
summary1 <- summarySE(data=mixed1.df,
  measurevar="rt",
  groupvars=c("age", "angle"),
  conf.interval=.95)
summary1
```

```
##   age angle  N      rt      sd      se      ci
## 1 young  zero  9 480.0000 67.08204 22.36068 51.56382
## 2 young  four  9 593.3333 83.21658 27.73886 63.96593
## 3 young  eight  9 646.6667 99.62429 33.20810 76.57801
## 4 older  zero 10 543.0000 98.43780 31.12876 70.41816
## 5 older  four 10 654.0000 85.79044 27.12932 61.37079
## 6 older  eight 10 792.0000 75.09993 23.74868 53.72326
```

A second approach is available that implements the Morey 2008 correction of std errors and CIs for repeated measures designs. The `summarySEwithin` function from **Rmisc** is built for this purpose. However, it also produces means from “normed” data and these are not the means that we wish to plot. A better version of `summarySEwithin` is available:

[http://www.cookbook-r.com/Graphs/Plotting_means_and_error_bars_\(ggplot2\)/](http://www.cookbook-r.com/Graphs/Plotting_means_and_error_bars_(ggplot2)/)

Code for this function and the accompanying `normedDataWithin` helper function are sourced here from files shared with this implementation tutorial. Both raw and normed means are available in the object produced

by summarySEwithin.

```
source("normedDataWithin.R")
source("summarySEwithin.R")
summary2 <- summarySEwithin(data=mixed1.df,
                             measurevar="rt",
                             betweenvars="age",
                             withinvars="angle",
                             idvar="snum",
                             conf.interval=.95)
```

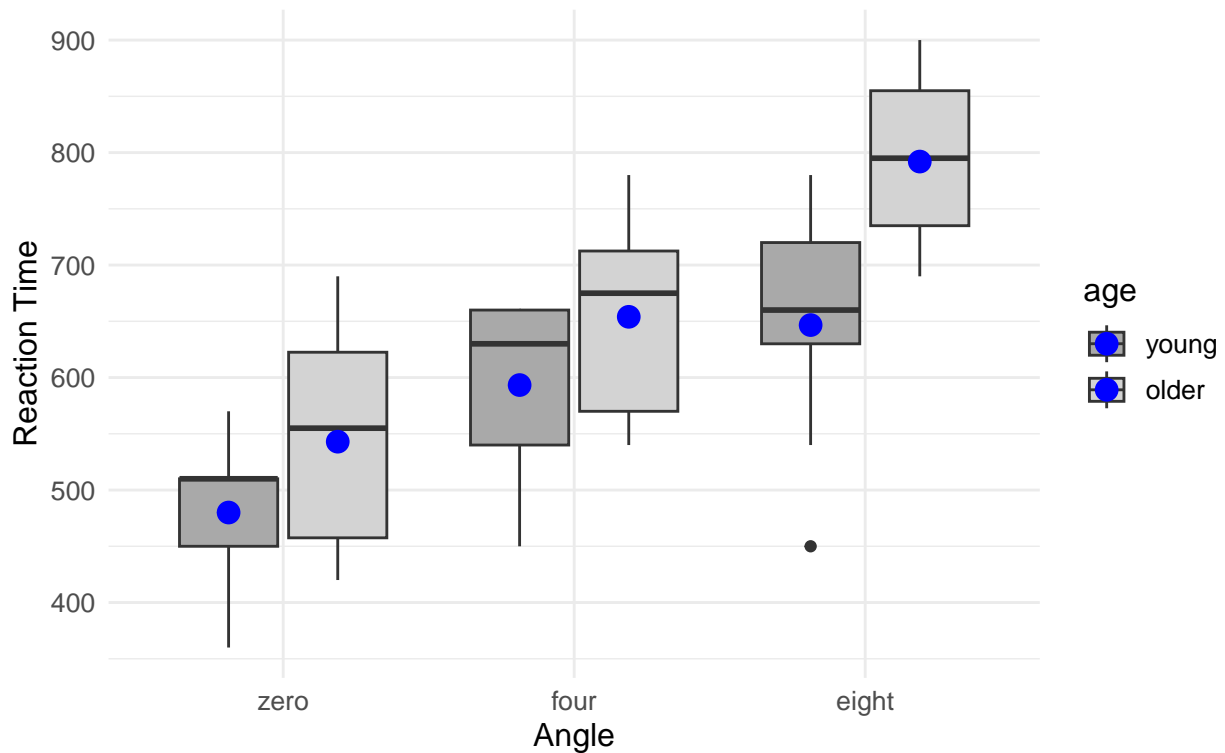
summary2

```
##   age angle  N      rt rt_norm      sd      se      ci
## 1 older eight 10 792.0000 749.5263 32.88870 10.400321 23.52716
## 2 older  four 10 654.0000 611.5263 29.69287  9.389711 21.24100
## 3 older  zero 10 543.0000 500.5263 42.03173 13.291601 30.06769
## 4 young eight  9 646.6667 693.8596 46.63690 15.545632 35.84829
## 5 young  four  9 593.3333 640.5263 36.74235 12.247449 28.24267
## 6 young  zero  9 480.0000 527.1930 45.41476 15.138252 34.90887
```

A boxplot drawn with **ggplot2** is customized to also display the mean of each condition.

```
ggplot(data = mixed1.df, aes(x=angle, y = rt, fill=age)) +
  geom_boxplot() +
  geom_point(
    data = summary1,
    aes(y = rt, x = angle),
    position = position_dodge(width = .75),
    color = "blue",
    size = 3.5) +
  scale_fill_manual(values = c("darkgray", "lightgray")) +
  xlab("Angle") +
  ylab("Reaction Time") +
  ggtitle(" Boxplot of Reaction Time\n Plus Means as Blue Points") +
  theme_minimal() +
  theme(text =element_text(size = 12))
```

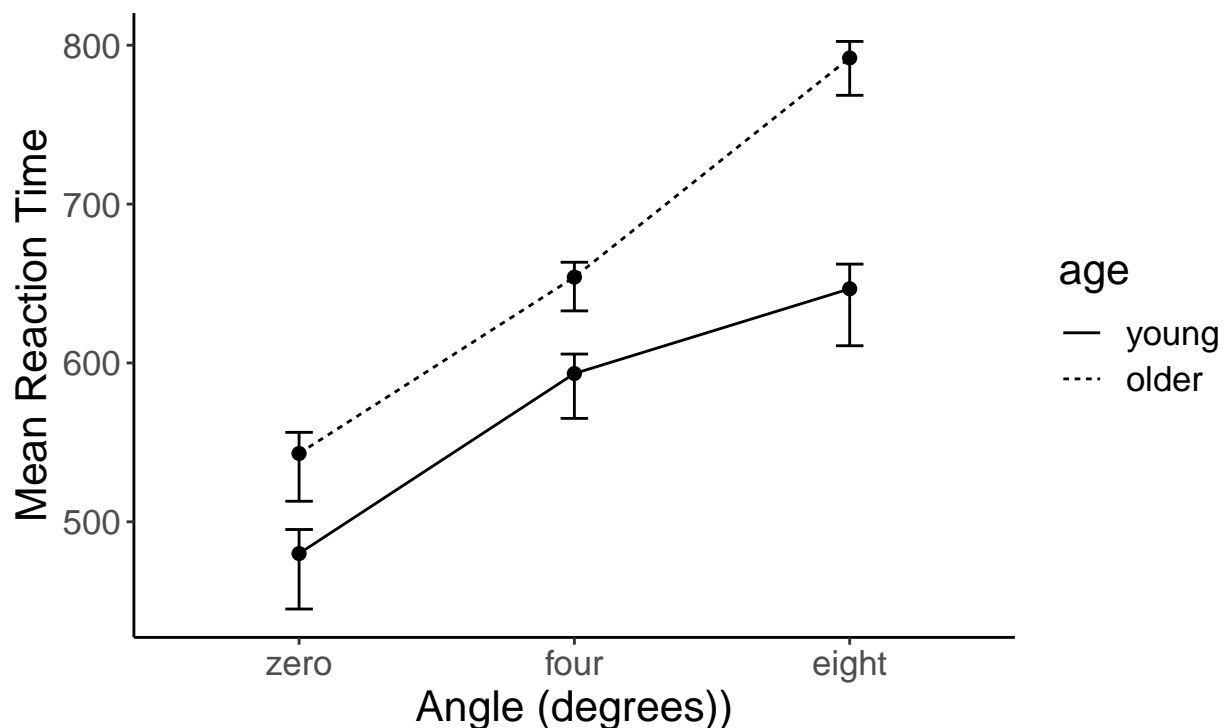
Boxplot of Reaction Time
Plus Means as Blue Points



Here is a ggplot version of a line graph with error bars that are the 95% CI's based on these adjusted within subject standard errors (Morey, 2008)

```
ggplot(summary2, aes(x=angle, y=rt, group=age)) +
# expand_limits(y=c(5,18)) +
# expand_limits(x=c(0,2.5)) +
# scale_y_continuous(breaks=0:4*4) +
# scale_x_continuous(breaks=0:5*.5) +
geom_line(aes(linetype=age)) + geom_point(size=2) +
geom_errorbar(aes(ymin=rt-ci, ymax=rt+se), colour="black", width=.1)+
xlab("Angle (degrees)")+
ylab("Mean Reaction Time")+
ggtitle("Age and Angle Effects\n on reaction time")+
theme_classic()+
theme(text = element_text(size=16))
```

Age and Angle Effects on reaction time



5 Perform the base ANOVA for the two-factor Mixed design

Several approaches are possible. I prefer the `afex` approach, although the `rstatix` approach provides some additional flexibility.

5.1 Use the base system `aov` function

The base system `aov` function can provide the omnibus F tests. However, the anova summary table provides Type I SS. If type III SS are desired (typical), it is unfortunate that the `Anova` function from the `car` package will not work on an `aov` object of this type. Other methods are employed below to obtain Type III SS decompositions.

```
fit1.aov <- aov(rt~age*angle + Error(snum/angle),mixed1.df)
summary(fit1.aov)

##
## Error: snum
##           Df Sum Sq Mean Sq F value Pr(>F)
## age         1 114254  114254    6.017 0.0253 *
## Residuals  17 322830   18990
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: snum:angle
##           Df Sum Sq Mean Sq F value Pr(>F)
## angle       2 419589  209795 136.700 < 2e-16 ***
```



```
## age:angle 2 22031 11015 7.177 0.00251 **
## Residuals 34 52180 1535
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can extract other useful information from the fit object

```
print(model.tables(fit1.aov,"means"),digits=3) #report the means of each level
```

```
## Tables of means
## Grand mean
##
## 620.5263
##
## age
##   young older
##   573 663
## rep 27 30
##
## angle
##   zero four eight
##   513 625 723
## rep 19 19 19
##
## age:angle
##   angle
## age   zero four eight
## young 480 593 647
## rep    9 9 9
## older 543 654 792
## rep   10 10 10
```

```
print(model.tables(fit1.aov,"effects", n=TRUE),digits=3) # report deviation effects for each level
```

```
## Tables of effects
##
## age
##   young older
##   -47.2 42.5
## rep 27.0 30.0
##
## angle
##   zero four eight
##   -107 4.74 103
## rep 19 19.00 19
##
## age:angle
##   angle
## age   zero four eight
## young 14.04 15.26 -29.30
## rep    9.00 9.00 9.00
## older -12.63 -13.74 26.37
## rep   10.00 10.00 10.00
```

5.2 Use the EZ package for the mixed ANOVA

The “type” argument in the `ezANOVA` function permits specification of SS decomposition type and we use type III here. Note that the SS and F test for the main effect of the repeated factor is different than in `fit1` above. One nice thing about the `ezANOVA` function is that it provides a test of the sphericity assumption and the generalized effect size statistic.

Note that there are two sphericity tests, one for the error term for the main effect of angle and the other for the interaction error term. The reader may notice that the Mauchly test statistic values are identical for these two tests. I think that this probably occurs because this is a fabricated textbook data set and the algorithm for generating the data led to both of those source interactions with “subject” following a pattern. The coincidental identical values would not be expected with real data sets.

```
fit2.ez <- ezANOVA(  
  data=mixed1.df,  
  dv=. (rt),  
  wid=. (snum),  
  within=. (angle),  
  between=. (age),  
  type=3,  
  detailed=TRUE,  
  return_aov=TRUE  
)
```

```
## Warning: Data is unbalanced (unequal N per group). Make sure you specified a  
## well-considered value for the type argument to ezANOVA().
```

```
print(fit2.ez)
```

```
## $ANOVA  
##      Effect DFn DFd      SSn      SSd      F      p p<.05  
## 1 (Intercept) 1 17 21721075.26 322830 1143.816496 4.914989e-17 *  
## 2      age 1 17 114254.21 322830 6.016546 2.526540e-02 *  
## 3      angle 2 34 410072.63 52180 133.599746 7.845482e-17 *  
## 4 age:angle 2 34 22030.53 52180 7.177442 2.509895e-03 *  
##      ges  
## 1 0.98302822  
## 2 0.23352252  
## 3 0.52233054  
## 4 0.05548685  
##  
## $`Mauchly's Test for Sphericity`  
##      Effect      W      p p<.05  
## 3      angle 0.9069528 0.4578019  
## 4 age:angle 0.9069528 0.4578019  
##  
## $`Sphericity Corrections`  
##      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF] p[HF]<.05  
## 3      angle 0.9148736 1.390544e-15 * 1.019609 7.845482e-17 *  
## 4 age:angle 0.9148736 3.424878e-03 * 1.019609 2.509895e-03 *  
##  
## $aov  
##  
## Call:  
## aov(formula = formula(aov_formula), data = data)  
##
```

```

## Grand Mean: 620.5263
##
## Stratum 1: snum
##
## Terms:
##              age Residuals
## Sum of Squares 114254.2 322830.0
## Deg. of Freedom    1      17
##
## Residual standard error: 137.8042
## Estimated effects are balanced
##
## Stratum 2: snum:angle
##
## Terms:
##              angle age:angle Residuals
## Sum of Squares 419589.5 22030.5 52180.0
## Deg. of Freedom    2      2      34
##
## Residual standard error: 39.17532
## Estimated effects may be unbalanced

```

5.3 The rstatix package provides another approach

A relatively new package, `rstatix` appears to offer a flexible approach. The `get_anova` function permits direct control of p value adjustment with GG, HF or none.

A tutorial is available:

<https://www.datanovia.com/en/lessons/repeated-measures-anova-in-r/#computation-1>

```

fit6.aovtest <- anova_test(data=mixed1.df,
                           dv=rt,
                           wid=snum,
                           within=angle,
                           between=age)
fit6.aovtest

```

```

## ANOVA Table (type III tests)
##
## $ANOVA
##      Effect DFn DFd      F      p p<.05 ges
## 1      age   1  17   6.017 2.50e-02 * 0.234
## 2     angle   2  34 133.600 7.85e-17 * 0.522
## 3 age:angle   2  34   7.177 3.00e-03 * 0.055
##
## $`Mauchly's Test for Sphericity`
##      Effect      W      p p<.05
## 1     angle 0.907 0.458
## 2 age:angle 0.907 0.458
##
## $`Sphericity Corrections`
##      Effect  GGe      DF[GG]      p[GG] p[GG]<.05  HFe      DF[HF]      p[HF]
## 1     angle 0.915 1.83, 31.11 1.39e-15 * 1.02 2.04, 34.67 7.85e-17
## 2 age:angle 0.915 1.83, 31.11 3.00e-03 * 1.02 2.04, 34.67 3.00e-03
##      p[HF]<.05

```

```
## 1      *
## 2      *
```

Or obtain briefer results summaries:

```
get_anova_table(fit6.aovtest, correction = "GG")
```

```
## ANOVA Table (type III tests)
##
##      Effect  DFn  DFd      F      p p<.05  ges
## 1      age  1.00 17.00   6.017 2.50e-02 * 0.234
## 2     angle  1.83 31.11 133.600 1.39e-15 * 0.522
## 3 age:angle  1.83 31.11   7.177 3.00e-03 * 0.055
```

5.4 Use the afex package for the basic ANOVA

I have come to prefer the **afex** package for its flexibility and ease of use with the **emmeans** package for followup analyses. Those follow up analyses turn out to be problematic as seen in a later section here.

We will use the “car” flavor of the **afex** approach with the `aov_car` function. Note that the default approach produces an ANOVA summary table that utilizes the Greenhouse-geisser correction method, so df are fractional. The generalized effect size statistic is provided.

```
# these functions need the BG factor set to zero sum contrasts
# we will use effect coding here, but with only two levels it is also contrast coding
contrasts(mixed1.df$age) <- contr.sum
contrasts(mixed1.df$age)
```

```
##      [,1]
## young   1
## older  -1
```

```
contrasts(mixed1.df$angle) <- contr.sum
contrasts(mixed1.df$angle)
```

```
##      [,1] [,2]
## zero    1    0
## four    0    1
## eight   -1   -1
```

```
fit3.afex <- aov_car(rt~age*angle + Error(snum/angle), data=mixed1.df)
```

```
## Contrasts set to contr.sum for the following variables: age
```

To obtain the basic ANOVA summary table plus the Mauchly sphericity test, plus information on the GG and HF epsilons we can pass the afex fit object to `summary`.

```
summary(fit3.afex)
```

```
## Warning in summary.Anova.mlm(object$Anova, multivariate = FALSE): HF eps > 1
## treated as 1
```

```
##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##      Sum Sq num Df Error SS den Df  F value  Pr(>F)
## (Intercept) 21721075      1  322830      17 1143.8165 < 2e-16 ***
## age          114254      1  322830      17   6.0165 0.02527 *
## angle        410073      2   52180      34 133.5997 < 2e-16 ***
## age:angle     22031      2   52180      34   7.1774 0.00251 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Mauchly Tests for Sphericity
##
##          Test statistic p-value
## angle          0.90695  0.4578
## age:angle      0.90695  0.4578
##
##
## Greenhouse-Geisser and Huynh-Feldt Corrections
## for Departure from Sphericity
##
##          GG eps Pr(>F[GG])
## angle      0.91487  1.391e-15 ***
## age:angle  0.91487   0.003425 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##          HF eps  Pr(>F[HF])
## angle      1.019609 7.845482e-17
## age:angle  1.019609 2.509895e-03
```

If we want the ANOVA summary table with GG corrections, then we can just ask for the fit object.

```
anova(fit3.afex, correction="GG")
```

```
## Anova Table (Type 3 tests)
##
## Response: rt
##          num Df den Df      MSE      F      ges      Pr(>F)
## age          1.0000 17.000 18990.0   6.0165 0.23352 0.025265 *
## angle        1.8297 31.106  1677.5 133.5997 0.52233 1.391e-15 ***
## age:angle    1.8297 31.106  1677.5   7.1774 0.05549 0.003425 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can provide a more nicely formatted table. First without the GG correction and then with it.

```
gt(nice(anova(fit3.afex, correction="none")))
```

Effect	df	MSE	F	ges	p.value
age	1, 17	18990.00	6.02 *	.234	.025
angle	2, 34	1534.71	133.60 ***	.522	<.001
age:angle	2, 34	1534.71	7.18 **	.055	.003

```
gt(nice(anova(fit3.afex, correction="GG")))
```

Effect	df	MSE	F	ges	p.value
age	1, 17	18990.00	6.02 *	.234	.025
angle	1.83, 31.11	1677.51	133.60 ***	.522	<.001
age:angle	1.83, 31.11	1677.51	7.18 **	.055	.003

It is possible to produce an unadjusted table of F tests where the GG correction is not employed in a different way. I also show here how to obtain partial eta squared instead of ges as the effect size statistic. In addition, the ANOVA summary table can be formatted in a more pleasing manner with `nice` from `afex` and `gt` from the `gt` package.

```
fit4.afex <- aov_car(rt~age*angle + Error(snum/angle), data=mixed1.df,
  anova_table = list(es = "pes",
    correction="none"))
```

```
## Contrasts set to contr.sum for the following variables: age
```

```
gt(nice(fit4.afex))
```

Effect	df	MSE	F	pes	p.value
age	1, 17	18990.00	6.02 *	.261	.025
angle	2, 34	1534.71	133.60 ***	.887	<.001
age:angle	2, 34	1534.71	7.18 **	.297	.003

A more traditional full ANOVA summary table can be obtained as well, although without effect size statistics. This analysis presents the unadjusted df and p values, but does show the sphericity test and epsilon values. The warning is issued to alert the analyst that the Huynh-Feldt epsilon correction factor exceeds the theoretical value of 1.0 in this data set (there is not much non-sphericity)

```
fit5.afex <- aov_car(rt~age*angle + Error(snum/angle), data=mixed1.df,
  return="univariate")
```

```
## Contrasts set to contr.sum for the following variables: age
```

```
## Warning in summary.Anova.mlm(Anova.out, multivariate = FALSE): HF eps > 1
```

```
## treated as 1
```

```
fit5.afex
```

```
##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##           Sum Sq num Df Error SS den Df   F value   Pr(>F)
## (Intercept) 21721075     1  322830    17 1143.8165 < 2e-16 ***
## age          114254     1  322830    17   6.0165 0.02527 *
## angle        410073     2   52180    34 133.5997 < 2e-16 ***
## age:angle     22031     2   52180    34   7.1774 0.00251 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Mauchly Tests for Sphericity
##
##           Test statistic p-value
## angle          0.90695  0.4578
## age:angle       0.90695  0.4578
##
##
## Greenhouse-Geisser and Huynh-Feldt Corrections
## for Departure from Sphericity
##
##           GG eps Pr(>F[GG])
```

```
## angle      0.91487 1.391e-15 ***
## age:angle 0.91487 0.003425 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##           HF eps  Pr(>F[HF])
## angle      1.019609 7.845482e-17
## age:angle 1.019609 2.509895e-03
```

6 Assumptions

The sphericity assumption is recognized to be a critical one in repeated measures analyses, and is provided with the EZ and AFEX methods above.

6.1 Box's M test for homogeneity of covariance matrices across groups

An additional assumption for mixed designs is known as homogeneity of covariance matrices across groups. A test of that null is provided by Box's M test. It requires a wide format version of the data frame which is available here.

The test statistic is a chi-square value.

There is some debate about how important the “significance” of this test is. As seen in this link, Tabachnick and Fidell argue that unless $p < .001$ and sample sizes are unequal, this assumption can be ignored.

https://en.wikiversity.org/wiki/Box%27s_M

```
mixedwide.df <- read.csv("mixedmd_wide.csv", stringsAsFactors=T)
head(mixedwide.df)
```

```
##      age snum zero four eight
## 1 young   2  390  480   540
## 2 young   3  570  630   660
## 3 young   4  450  660   720
## 4 young   5  510  660   630
## 5 young   6  360  450   450
## 6 young   7  510  600   720
```

```
box_m(mixedwide.df[, 3:5, drop = FALSE], mixedwide.df$age)
```

```
## # A tibble: 1 x 4
##   statistic p.value parameter method
##   <dbl>    <dbl>    <dbl> <chr>
## 1      4.51  0.609          6 Box's M-test for Homogeneity of Covariance Matric~
```

7 Additional Methods following up on Omnibus ANOVAs: Contrasts and Simple Effects.

Using the `afex_car` fit object, `emmeans` provides facilities for several followup analyses in this design. Methods for obtaining simple main effects and contrasts on each of the main effect, simple main effect and interaction terms are pursued here. **Warning: Simple main effect analyses seem flawed.**

7.1 Interaction Contrasts

This `emmeans` approach does yield appropriate tests of the interaction contrasts. We might call the two effects `Angle_linear` by Age and `Angle_quadratic` by Age. The squares of the t-values match the F values that we

generated with other methods (SPSS and manually), when specific error terms are used for each contrast.

```
emm1.int <- emmeans(fit3.afex, specs=c("angle","age"))
#emm1.int
contrast(emm1.int, interaction=c(angle="poly", age="trt.vs.ctrl"))

## angle_poly age_trt.vs.ctrl estimate SE df t.ratio p.value
## linear older - young 82.3 28.8 17 2.857 0.0109
## quadratic older - young 87.0 37.4 17 2.329 0.0325
```

7.2 Main Effect Contrasts

This approach yields appropriate tests of the main effect contrasts (on Angle). We might call the two effects Angle_linear and Angle_quadratic. The squares of the t-values match the F values that we generated with other methods (SPSS and manually), when specific error terms are used for each contrast.

```
emm1.angle <- emmeans(fit3.afex, specs="angle")
emm1.angle
```

```
## angle emmean SE df lower.CL upper.CL
## zero 512 19.6 17 470 553
## four 624 19.4 17 583 665
## eight 719 20.1 17 677 762
##
## Results are averaged over the levels of: age
## Confidence level used: 0.95
```

```
contrast(emm1.angle, "poly")
```

```
## contrast estimate SE df t.ratio p.value
## linear 207.8 14.4 17 14.422 <.0001
## quadratic -16.5 18.7 17 -0.883 0.3894
##
## Results are averaged over the levels of: age
```

7.3 Follow up analyses using emmeans for Simple Main Effects

This approach follows from `emmeans` documentation, but gives problematic results.

7.3.1 First extract the matrix of means for SME

The standard method of using `emmeans` requires extracting the grid of means, etc; the `test` and `contrast` functions can then work on this grid. In this first analysis, we obtain the means for the levels of angle separately in each of the two age groups - setting up analyses of simple main effects. Note that we could reverse the specification of the two IVs and obtain the alternate set of simple effects (and do so below).

However, this first table of descriptive statistics is problematic. In the young group, the df should be 8, and in the older group it should be 9 since sample sizes were 9 and 10, respectively. This issue is probably a core error that means the simple main effect tests are also flawed. In fact, it appears that the standard errors tabled here are also incorrect. E.g., the std error for the zero angle young condition should be the sd (67.802) divided by 3 (square root of sample size). This computation yields a value of 22.36068 as was also see from the application of `summareSE` to the data frame in an earlier section on graphing.

Unfortunately, any analyses based on this grid are likely to be flawed, but the approach is worked through in case corrections in the `emmeans` function emerge in later versions.

Perhaps there is some pooling happening that is not explained in the `afex` or `emmeans` documentation and vignettes.


```
emm1.sme <- emmeans(fit3.afex, "angle", by="age")
emm1.sme
```

```
## age = young:
## angle emmean SE df lower.CL upper.CL
## zero 480 28.4 17 420 540
## four 593 28.2 17 534 653
## eight 647 29.2 17 585 708
##
## age = older:
## angle emmean SE df lower.CL upper.CL
## zero 543 26.9 17 486 600
## four 654 26.7 17 598 710
## eight 792 27.7 17 734 850
##
## Confidence level used: 0.95
```

7.3.2 Simple effects of Angle at levels of age.

This set of simple main effects of the repeated factor at levels of age is obtained with the “joint” argument in the `test` function. However, the full SS/MS summary table is not provided and the effect and error df and F values are not as expected (an omnibus error term would have 34 df and a specific error term in the first level of age should have 16 df and 18 in the second). More important, the results suggest that each SME has 3 numerator df, but they should be 2 df each (three levels of angle). Since the full set of SS is not provided, it is not clear what the analysis is doing - **This cannot be a recommended approach for SME of the repeated factor at this version of emmeans.**

```
#test(contrast(emm1.sme), joint=TRUE, adjust="none")
test(emm1.sme, joint=TRUE, adjust="none")
```

```
## age df1 df2 F.ratio p.value
## young 3 17 170.844 <.0001
## older 3 17 274.324 <.0001
```

It is also possible to obtain t-tests for pairs of levels of the repeated measure variable at levels of age, as well as p-value adjustments in the bonferroni-sidak-holm-fdr family. However, these are not simply done as paired samples t-tests (which would provide use of specific error terms). Instead, the **potentially flawed error terms are used** - we can tell because of the 17 df that each t is reported with. In the “young” group there were 9 cases and in the “old” group sample size was 10, so it appears that the error was pooled from within each of the two groups giving 17 df. But documentation on this is lacking. See additional comments below.

```
pairs(emm1.sme, adjust="tukey")
```

```
## age = young:
## contrast estimate SE df t.ratio p.value
## zero - four -113.3 18.1 17 -6.264 <.0001
## zero - eight -166.7 20.9 17 -7.971 <.0001
## four - eight -53.3 16.1 17 -3.317 0.0108
##
## age = older:
## contrast estimate SE df t.ratio p.value
## zero - four -111.0 17.2 17 -6.467 <.0001
## zero - eight -249.0 19.8 17 -12.553 <.0001
## four - eight -138.0 15.3 17 -9.046 <.0001
##
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
#pairs(emm1.sme, adjust="sidak")
```

7.3.3 Simple main effects of Age, the BG factor at levels of the repeated measure factor.

For this design, with only two levels of the grouping factor (age), the F tests appear to be correct, using a specific error of s/Age separately at each level of Angle. This is treating the SME tests as essentially three separate independent samples t-tests done at the three levels of angle.

The approach is justifiable, but extension to larger designs should be carefully evaluated. Some apprehension exists here too since the same issue with std errors and df in the `emmeans` exists here as it did for the other grid separating by age.

```
emm2.sme <- emmeans(fit3.afex, "age", by="angle")
emm2.sme
```

```
## angle = zero:
## age  emmean  SE df lower.CL upper.CL
## young  480 28.4 17    420    540
## older  543 26.9 17    486    600
##
## angle = four:
## age  emmean  SE df lower.CL upper.CL
## young  593 28.2 17    534    653
## older  654 26.7 17    598    710
##
## angle = eight:
## age  emmean  SE df lower.CL upper.CL
## young  647 29.2 17    585    708
## older  792 27.7 17    734    850
##
## Confidence level used: 0.95
```

```
test(contrast(emm2.sme), joint=TRUE, adjust="none")
```

```
## angle df1 df2 F.ratio p.value note
## zero   1  17  2.594 0.1257 d
## four   1  17  2.436 0.1370 d
## eight  1  17 13.067 0.0021 d
##
## d: df1 reduced due to linear dependence
```

This characterization can be verified with the pairwise tests approach, where the t values are the square roots of the F's seen above.

```
pairs(emm2.sme)
```

```
## angle = zero:
## contrast      estimate    SE df t.ratio p.value
## young - older   -63.0 39.1 17  -1.611  0.1257
##
## angle = four:
## contrast      estimate    SE df t.ratio p.value
## young - older   -60.7 38.9 17  -1.561  0.1370
##
## angle = eight:
## contrast      estimate    SE df t.ratio p.value
## young - older  -145.3 40.2 17  -3.615  0.0021
```

7.3.4 Contrasts on simple main effects

Here, we attempt to examine trend contrasts for angle applied to each of the two simple main effects (of angle at levels of age). This would use the same `emmeans` grid of means used above. Although output is provided, the `df` for the error terms is puzzling. They are both 17, even though the sample sizes in each level of age are not the same. If the error terms were `lin` by `s/group 1` and `quad` by `s/group1` the `df` should have been 8 for each of the tests at `age=young`. For the contrasts in the older group they should have been 9, so it appears that some pooling has been done.

This approach cannot be recommended. It appears that `emmeans` has not accurately implemented F and t-tests that employ rational error terms.

```
test(contrast(emm1.sme, "poly"), adjust="none")
```

```
## age = young:
## contrast estimate SE df t.ratio p.value
## linear      167 20.9 17   7.971 <.0001
## quadratic   -60 27.1 17  -2.214 0.0408
##
## age = older:
## contrast estimate SE df t.ratio p.value
## linear      249 19.8 17  12.553 <.0001
## quadratic    27 25.7 17   1.050 0.3084
```

7.4 A manual approach to performing trend analysis on the Simple Main Effects

One possible approach to performing SME contrasts (trend on the angle variable separately in each age group) would be to perform a 1-way repeated measures ANOVA separately in each of the two age groups. This is a justifiable method since the error terms would be specific to each group and each trend component, and that is justifiable.

The manual approach is to employ subsetting the data frame by age group. The omnibus F test produced here is a test of the simple main effect of angle at the “young” level of age. It uses an error term specific to that group and the result can be defended as appropriate. A GG or HF correction can be applied if needed.

```
young <- subset(mixed1.df, age=="young")
fityoung.afex <- aov_car(rt~angle + Error(snum/angle), data=young,
                       anova_table = list(correction="none"))
anova(fityoung.afex, correction="none")
```

```
## Anova Table (Type 3 tests)
##
## Response: rt
##      num Df den Df    MSE      F    ges    Pr(>F)
## angle     2    16 1862.5 35.007 0.43293 1.434e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now, the `emmeans` grid is extracted for use by the `contrast` function.

```
young.emm <- emmeans(fityoung.afex, "angle")
young.emm
```

```
## angle emmean SE df lower.CL upper.CL
## zero     480 22.4 8     428     532
## four     593 27.7 8     529     657
## eight    647 33.2 8     570     723
##
```

```
## Confidence level used: 0.95
```

The table above is what the emmeans grid should have looked like above, where the flawed SE and df values emerged. Now we perform contrast tests on this “young” grid.

These tests of the two trend components use error terms specific to both the contrast and the “young” group - appropriately.

```
test(contrast(young.emm, "poly"), adjust="none")
```

```
## contrast estimate SE df t.ratio p.value
## linear          167 23  8   7.255 0.0001
## quadratic       -60 30  8  -2.000 0.0805
```

The process can be repeated for the “older” age group.

```
older <- subset(mixed1.df, age=="older")
fitolder.afex <- aov_car(rt~angle + Error(snum/angle), data=older,
                        anova_table = list(correction="none"))
anova(fitolder.afex, correction="none")
```

```
## Anova Table (Type 3 tests)
##
## Response: rt
##      num Df den Df    MSE      F    ges    Pr(>F)
## angle    2   18 1243.3 125.16 0.60381 2.752e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now, the emmeans grid is extracted for use by the contrast function.

```
older.emm <- emmeans(fitolder.afex, "angle")
older.emm
```

```
## angle emmean SE df lower.CL upper.CL
## zero    543 31.1  9    473    613
## four    654 27.1  9    593    715
## eight   792 23.7  9    738    846
##
## Confidence level used: 0.95
```

The table above is what the emmeans grid should have looked like above, where the flawed SE and df values emerged. Now we perform contrast tests on this “older” grid.

These tests of the two trend components use error terms specific to both the contrast and the “young” group - appropriately.

```
test(contrast(older.emm, "poly"), adjust="none")
```

```
## contrast estimate SE df t.ratio p.value
## linear          249 17.9  9  13.898 <.0001
## quadratic        27 23.0  9   1.174 0.2706
```

This manual approach works here and is approachable. However, extrapolation to larger designs (more levels or more factors) would not be efficient - too bad that `emmeans` is not yet ready for prime time with a more direct analysis of the full design fit object.

7.5 Comments on use of `emmeans`

We can be confident in the results of analyses that produce main effect contrasts and interaction contrasts. But the work to produce simple main effects and their contrasts revealed that the `emmeans` approach is not yet

fully developed - several effects produced uninterpretable results. Perhaps I am not yet fully conversant with the `emmeans` capabilities, but the documentation than I have found does not directly distinguish repeated measures approaches from the BG approaches where `emmeans` is fully functional.

7.6 Simple main effects using the `rstatix` omnibus model fit object

The approach to simple main effects outlined here is a parallel to the manual subsetting shown above where the data set was split and an analysis was performed at each level of the subset factor.

For the effect of age, in the first set of analyses, this is tantamount to using specific error terms `s/Age` at levels of angle and matches our SPSS MANOVA approach. In this section, the subsetting is done by using tidyverse methods ("`group_by`") and pipes

The `rstatix` functions work well in the tidyverse style of programming, using pipes. The user should read this code as

1. start with the `mixed1.df` data frame and assign the results of operations done on that data frame to a new object (a data frame - actually a "tibble") called `simple.age`.
2. subset the data by "angle", the repeated measures factor (thus into three subsets for the three levels of angle) - accomplished with the `group_by` function.
3. perform the anova on those subset data frames - running a simple BG anova on the age factor.
4. obtain the results table(s)
5. adjust the p values for having performed several tests.

This gives the three simple main effects of age at the three levels of angle.

```
simple.age <- mixed1.df %>%
  group_by(angle) %>%
  anova_test(dv=rt, wid=snum, between=age) %>%
  get_anova_table() %>%
  adjust_pvalue(method="holm")
simple.age

## # A tibble: 3 x 9
##   angle Effect  DFn  DFd    F    p `p<.05`  ges p.adj
##   <ord> <chr>  <dbl> <dbl> <dbl> <dbl> <chr>  <dbl> <dbl>
## 1 zero  age      1    17  2.59 0.126 ""      0.132 0.252
## 2 four  age      1    17  2.44 0.137 ""      0.125 0.252
## 3 eight age      1    17 13.1  0.002 "*"     0.435 0.006
```

Next, we can evaluate the simple effects of angle at levels of age. These use specific error terms, and can be adjusted for non-sphericity. These error terms are also specific: Angle by `s/Age1` and Angle by `s/Age2`, respectively. The approach is tantamount to doing two separate 1-factor repeated measures analyses in the young and older groups. This is probably a justifiable method for testing simple main effects in this kind of mixed design. Note that there is difficulty in obtaining correctly tested simple main effects in the `emmeans` section above, so this may be the only way to obtain reasonable tests of SME.

```
simple.angle <- mixed1.df %>%
  group_by(age) %>%
  anova_test(dv=rt, wid=snum, within=angle) %>%
  #get_anova_table(correction="GG") %>%
  get_anova_table(correction="none") %>%
  adjust_pvalue(method="holm")
simple.angle

## # A tibble: 2 x 9
```

```
##   age   Effect   DFn   DFd     F       p `p<.05`   ges   p.adj
##   <ord> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>
## 1 young angle     2    16  35.0 1.43e- 6 *    0.433 1.43e- 6
## 2 older angle     2    18 125.  2.75e-11 *    0.604 5.5 e-11
```

7.6.1 Contrasts in the rstatix models

I have not yet discovered a way to employ contrasts using the `rstatix` approach. Pairwise comparisons of marginals can be done in additive models. Refer to the tutorial link above.

7.7 Are there other approaches to followup analyses?

Modeling on some strategies found the 1-factor repeated measures tutorial, it is possible to use other approaches, especially with contrasts. Foremost among these is use of the `ghlt` function. Not yet included in this document.

8 Linear Mixed Effects approaches.

This section outlines some rudimentary mixed effects modeling strategies.

8.1 Using lme

A modeling strategy for the omnibus model is outlined here, along with possible follow up analyses.

8.1.1 Finding the appropriate omnibus model

Initially, this set of models permits evaluation of an intercept-only model, two single main effect models, an additive model and the full (inteaaction model). Comparison with the `anova` function reveals that the full model fits best since it has the lowest information criteria values and since the likelihood ratio test against the additive model is significant.

```
fit8.lme.intercept <- lme(rt ~1, random=~1 | snum/angle,
                        data=mixed1.df, method="REML")
fit8.lme.angle <- lme(rt ~ angle, random=~1 | snum/angle,
                     data=mixed1.df, method="REML")
fit8.lme.age <- lme(rt ~ age, random=~1 | snum/angle,
                   data=mixed1.df, method="REML")
fit8.lme.additive <- lme(rt ~ age + angle, random=~1 | snum/angle,
                        data=mixed1.df, method="REML")
fit8.lme.full <- lme(rt ~ age*angle, random=~1 | snum/angle,
                    data=mixed1.df, method="REML")
anova(fit8.lme.intercept, fit8.lme.angle, fit8.lme.age,
      fit8.lme.additive, fit8.lme.full)
```

```
## Warning in anova.lme(fit8.lme.intercept, fit8.lme.angle, fit8.lme.age,
## fit8.lme.additive, : fitted objects with different fixed effects. REML
## comparisons are not meaningful.

##           Model df      AIC      BIC   logLik   Test  L.Ratio p-value
## fit8.lme.intercept    1  4 712.6667 720.7681 -352.3334
## fit8.lme.angle        2  6 632.7522 644.6861 -310.3761 1 vs 2 83.91453 <.0001
## fit8.lme.age          3  5 701.5923 711.6290 -345.7962 2 vs 3 70.84011 <.0001
## fit8.lme.additive     4  7 621.6778 635.4698 -303.8389 3 vs 4 83.91453 <.0001
## fit8.lme.full         5  9 601.6900 619.0765 -291.8450 4 vs 5 23.98774 <.0001
```

Although this modeling sequence is appropriate, Note that with the default settings from above, the `lme` fit largely reproduces the GLM outcome seen with `aov`, `ez`, `afex`, and `rstatix`. What we have not done here is evaluate the covariance structure (default is compound symmetry) or try alternative structures - that extension is for advanced coursework.

```
anova(fit8.lme.full)

##           numDF denDF   F-value p-value
## (Intercept)     1    34 1155.7670 <.0001
## age             1    17   6.0165 0.0253
## angle           2    34 136.7003 <.0001
## age:angle       2    34   7.1774 0.0025
```

8.1.2 Simple main effects on the `lme` object

The `phia` package function `testInteractions` will work on an `lme` object, unlike with an `aov` object. here is the simple main effect of treatment group at levels of angle. This approach produces a Chi-square test statistics which is a different approach than the standard F test methodology and is beyond the scope of this document.

```
testInteractions(fit8.lme.full, fixed="angle", across="age", adjustment="none")

## Chisq Test:
## P-value adjustment method: none
##           Value Df   Chisq Pr(>Chisq)
## zero  -63.000  1  2.5568 0.1098204
## four  -60.667  1  2.3709 0.1236144
## eight -145.333  1 13.6065 0.0002254 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

8.2 Linear Mixed Models with `lmer` and `phia`

The `lmer` function from `lme4` provides another way of approaching repeated measures mixed models from the LME perspective. This initial/superficial application of `lmer` once again replicates the GLM outcome, but does not employ full LME modeling or alternative covariance structure exploration.

```
fit9.lmer <- lmer(rt ~ age*angle + (1|snm), data=mixed1.df)
anova(fit9.lmer)

## Type III Analysis of Variance Table with Satterthwaite's method
##           Sum Sq Mean Sq NumDF DenDF   F value   Pr(>F)
## age           9234     9234     1    17   6.0165 0.02527 *
## angle        410073  205036     2    34 133.5997 < 2e-16 ***
## age:angle    22031   11015     2    34   7.1774 0.00251 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The object produces will also work with the `testInteractions` object from the `phia` package and Simple main effects are shown here.

BE CAREFUL. These chi squared test statistics are tested with two-tailed p values and the reason is not clear - I am leary of this.

```
testInteractions(fit9.lmer, fixed="angle", across="age")

## Chisq Test:
## P-value adjustment method: holm
##           Value Df   Chisq Pr(>Chisq)
```

```
## zero -63.000 1 2.5568 0.2196409
## four -60.667 1 2.3709 0.2196409
## eight -145.333 1 13.6065 0.0006762 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

pchisq(2.5568, df=1, lower.tail=F)*2

## [1] 0.2196413
```

9 Short summary and recommendations

The inability to find simple ways of evaluating single df contrasts (orthogonal or not) and employ specific error terms is an ongoing disappointment that I have about using R for repeated measures models - either GLM or LME approaches. Those issues will only become more complex with larger designs having more levels of two factors or more factors.

The basics of the mixed models analysis are nicely provided by the **afex** and **rstatix** approaches. Simple effects are also available with **rstatix** as are some pairwise comparisons with **emmeans**. The SPSS methodology that we have covered is still more efficient and capable for contrasts and SME.

10 Documentation for Reproducibility

[Ver 1.0. April 17, 2023

R software products such as this markdown document should be simple to reproduce, if the code is available. But it is also important to document the exact versions of the R installation, the OS, and the R packages in place when the document is created.

```
sessionInfo()

## R version 4.2.2 (2022-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] psych_2.2.9      nlme_3.1-161    phia_0.2-1      ez_4.4-0
## [5] dplyr_1.0.10     knitr_1.41      gt_0.8.0        ggplot2_3.4.0
## [9] emmeans_1.8.3    Rmisc_1.5.1     plyr_1.8.8      lattice_0.20-45
## [13] rstatix_0.7.1    car_3.1-1       carData_3.0-5   afex_1.2-0
## [17] lme4_1.1-31      Matrix_1.5-3
##
## loaded via a namespace (and not attached):
```



```

## [1] tidyr_1.2.1          splines_4.2.2          assertthat_0.2.1
## [4] highr_0.10            yaml_2.3.6            numDeriv_2016.8-1.1
## [7] pillar_1.8.1          backports_1.4.1       glue_1.6.2
## [10] digest_0.6.31         minqa_1.2.5           colorspace_2.0-3
## [13] sandwich_3.0-2        htmltools_0.5.4       pkgconfig_2.0.3
## [16] broom_1.0.2           purrr_1.0.0           xtable_1.8-4
## [19] mvtnorm_1.1-3         scales_1.2.1          tibble_3.1.8
## [22] mgcv_1.8-41           farver_2.1.1          generics_0.1.3
## [25] ellipsis_0.3.2        TH.data_1.1-1         withr_2.5.0
## [28] cli_3.5.0             mnormt_2.1.1          survival_3.4-0
## [31] magrittr_2.0.3        estimability_1.4.1    evaluate_0.19
## [34] fansi_1.0.3           MASS_7.3-58.1         tools_4.2.2
## [37] lifecycle_1.0.3      multcomp_1.4-20       stringr_1.5.0
## [40] munsell_0.5.0         compiler_4.2.2        rlang_1.0.6
## [43] grid_4.2.2            nloptr_2.0.3          rstudioapi_0.14
## [46] labeling_0.4.2        rmarkdown_2.19        boot_1.3-28.1
## [49] gtable_0.3.1         codetools_0.2-18     lmerTest_3.1-3
## [52] abind_1.4-5           DBI_1.1.3             reshape2_1.4.4
## [55] R6_2.5.1             zoo_1.8-11           fastmap_1.1.0
## [58] utf8_1.2.2           stringi_1.7.8         parallel_4.2.2
## [61] Rcpp_1.0.9           vctrs_0.5.1          tidysselect_1.2.0
## [64] xfun_0.36            coda_0.19-4

```

References

- Boik, R. J. (1981). A priori tests in repeated measures designs: Effects of nonsphericity. *Psychometrika*, *46*, 241–255. Journal Article.
- Maxwell, S. E., Delaney, H. D., & Kelley, K. (2017). *Designing experiments and analyzing data : A model comparison perspective* (Third edition /, pp. pages cm). Book, New York, NY: Routledge.
- Rogan, J. C., Keselman, H. J., & Mendoza, J. L. (1979). Analysis of repeated measurements. *British Journal of Mathematical and Statistical Psychology*, *32*, 269–286. Journal Article.