Repeated measures analysis with R

Summary for experienced R users

The `lmer` function from the `lme4` package has a syntax like `lm`.

Add something like `+ (1|subject)` to the model for the random subject effect.

To get p-values, use the `car` package.

Avoid the `lmerTest` package.

For balanced designs, `Anova(dichotic, test="F")`

For unbalanced designs,

Set contrasts on the factors like this: `contrasts(Time) = contr.sum(5)`
And `BlueModel = lmer(Beck ~ Time*Drug*Treatment + (1|Subject))`
And `Anova(BlueModel, test="F", type="III")`

Or

```
BlueModel = lmer(Beck ~ Time*Drug*Treatment + (1|Subject),
   contrasts = list(Time="contr.sum", Drug="contr.sum",
                    Treatment="contr.sum"))
```

And `Anova(BlueModel, test="F", type="III")`

```
# Install packages if necessary. Only need to do this once.
# install.packages("lme4")
# install.packages("car")
# Load packages -- do this every time
library(lme4) # For the lmer function
library(car)  # For F and Wald tests with p-values
```

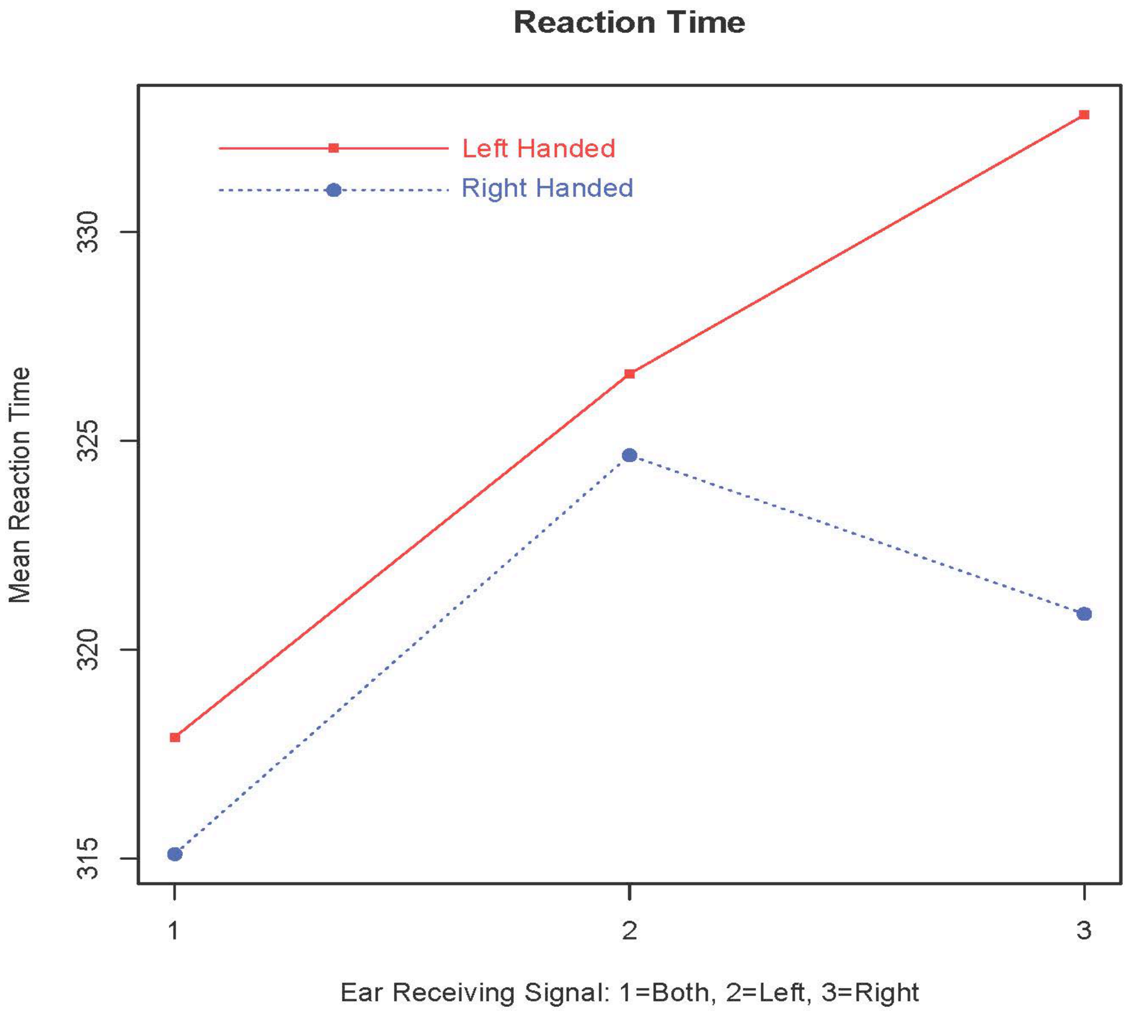
The `glmer` function from the `lme4` package has a syntax like `glm`.

```
hoops =  glmer(Hit ~ Spot*Hand + (1 | Subject), family=binomial)
Anova(hoops, type="III")
```

```
> # Hand-ear dichotic listening study
> # Install packages if necessary. Only need to do this once.
> # install.packages("lme4")
> # install.packages("car")
> # Load packages -- do this every time
> library(lme4) # For lmer function
Loading required package: Matrix
> library(car)  # For F-tests, likelihood ratio and Wald chi-squared tests
> # Read data into a data frame
> dichotic = 
read.table("http://www.utstat.toronto.edu/~brunner/data/legal/HandEar.data.txt")
> head(dichotic) # Look at the first few lines.
  subject handed   ear rtime
1       1   Left  Left   330
2       1   Left Right   327
3       1   Left  Both   303
4       2   Left  Left   294
5       2   Left Right   339
6       2   Left  Both   315
> attach(dichotic) # Make variable names available
> 
> # Sample sizes
> table(handed,ear)
       ear
handed  Both Left Right
  Left     20   20    20
  Right    20   20    20
> 
> # Treatment means
> aggregate(rtime,by=list(ear,handed),FUN=mean) # First changes fastest
  Group.1 Group.2      x
1    Both    Left 317.90
2    Left    Left 326.60
3   Right    Left 332.80
4    Both   Right 315.10
5    Left   Right 324.65
6   Right   Right 320.85
> 
> # Two-way table of means
> LeftHanded = meantable[1:3,3]; RightHanded =  meantable[4:6,3]
> TwoWay = rbind(LeftHanded,RightHanded)
> colnames(TwoWay) = c("BothEars","LeftEar","RightEar")
> TwoWay
> 

            BothEars LeftEar RightEar
LeftHanded     317.9  326.60   332.80
RightHanded    315.1  324.65   320.85
> 
> # Marginal means
>  aggregate(rtime,by=list(handed),FUN=mean)
  Group.1        x
1    Left 325.7667
2   Right 320.2000
>  aggregate(rtime,by=list(ear),FUN=mean)
  Group.1       x
1    Both 316.500
2    Left 325.625
3   Right 326.825
```

```
> # Plot the means
> Means = meantable[,3] # All the rows, column 3
> lhand = Means[1:3]; rhand = Means[4:6]
> Ear = c(1:3,1:3)
> # Invisible points at first, x axis points at 1,2,3; see help(plot)
> plot(Ear,Means,pch=" ",xaxp=c(1,3,2),
+     xlab="Ear Receiving Signal: 1=Both, 2=Left, 3=Right",
+     ylab="Mean Reaction Time")
> title("Reaction Time")
> points(1:3,lhand,col='red',pch=15) # Red squares
> points(1:3,rhand,col='blue',pch=19) # Blue circles
>  lines(1:3,lhand,lty=1,col='red'); lines(1:3,rhand,lty=3,col='blue')
> # Annotate the plot
> x1 = c(1.1,1.6); y1 = c(332,332); lines(x1,y1,lty=1,col='red')
>  points(1.35,332,col='red',pch=15)
> text(1.80,332,'Left Handed',col='red')
> x2 = c(1.1,1.6); y2 = c(331,331); lines(x2,y2,lty=3,col='blue')
>  points(1.35,331,col='blue',pch=19)
> text(1.82,331,'Right Handed',col='blue')
```



```
> # Naive fixed effects analysis
> anova(lm(rtime ~ handed*ear))

Analysis of Variance Table

Response: rtime
            Df Sum Sq Mean Sq F value  Pr(>F)
handed       1    930  929.63  1.8556 0.17582
ear          2   2551 1275.41  2.5458 0.08286 .
handed:ear   2    615  307.41  0.6136 0.54317
Residuals  114  57113  500.99
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Repeated measures with a mixed model
> dichotic = lmer(rtime ~ handed*ear + (1 | subject))
> Anova(dichotic, test="F") # F tests (from car package)
Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)

Response: rtime
                F Df Df.res  Pr(>F)
handed     0.9706  1     38 0.33075
ear        4.6787  2     76 0.01213 *
handed:ear 1.1277  2     76 0.32914
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Multiple comparisons: Which marginal means are different?
>
> # The combination variable HandEar will have 6 values
> n = length(rtime); n
[1] 120
> HandEar = character(n) # A character-valued variable of length n
> for(j in 1:n) HandEar[j] = paste(handed[j],ear[j],sep='')
> HandEar = factor(HandEar) # Maybe would be interpreted as a factor anyway
>  head(data.frame(handed,ear,HandEar))
  handed   ear   HandEar
1   Left  Left  LeftLeft
2   Left Right LeftRight
3   Left  Both  LeftBoth
4   Left  Left  LeftLeft
5   Left Right LeftRight
6   Left  Both  LeftBoth
> table(HandEar) # Sample sizes
HandEar
  LeftBoth   LeftLeft  LeftRight  RightBoth  RightLeft RightRight
        20         20         20         20         20         20
>
> # Want a table of means in a similar format.
> meantable # Again
    Ear Handed Mean RT
1  Both   Left  317.90
2  Left   Left  326.60
3 Right   Left  332.80
4  Both  Right  315.10
5  Left  Right  324.65
6 Right  Right  320.85
> ComboMeans = meantable[,3] # All the rows, 3d column
> names(ComboMeans) = sort(unique(HandEar)); ComboMeans
  LeftBoth   LeftLeft  LeftRight  RightBoth  RightLeft RightRight
    317.90     326.60     332.80     315.10     324.65     320.85
```

```
>
> # For a no-intercept model on a combination variable, the regression
> # coefficients are the treatment combination means. Details omitted.
>
> # Fit a no-intercept model
> ComboModel = lmer(rtime ~ 0 + HandEar + (1 | subject))
>
> # Contrast matrix for testing Ear, just as a check
> CM1 = rbind(c(1, -1,  0,  1, -1,  0), # Both - Left
+             c(0,  1, -1,  0,  1, -1)) # Left - Right
> colnames(CM1) = sort(unique(HandEar)) # Makes it easier to look at
> CM1
     LeftBoth LeftLeft LeftRight RightBoth RightLeft RightRight
[1,]        1       -1         0         1        -1          0
[2,]        0        1        -1         0         1         -1
> linearHypothesis(ComboModel,CM1,test="F") # Compare F = 4.6787
Linear hypothesis test

Hypothesis:
HandEarLeftBoth - HandEarLeftLeft  + HandEarRightBoth - HandEarRightLeft = 0
HandEarLeftLeft - HandEarLeftRight  + HandEarRightLeft - HandEarRightRight = 0

Model 1: restricted model
Model 2: rtime ~ 0 + HandEar + (1 | subject)

  Res.Df Df      F  Pr(>F)  
1     78                    
2     76  2 4.6787 0.01213 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
>
> # Now 3 pairwise comparisons of marginal means for ear.
> # Use a Bonferroni correction, meaning compare p-value to
> # 0.05 divided by the number of tests: 0.05/3 = 0.01666667
>
> bothVSleft = c(1, -1,  0,  1, -1,  0)
>  linearHypothesis(ComboModel,bothVSleft,test="F")
Linear hypothesis test

Hypothesis:
HandEarLeftBoth - HandEarLeftLeft  + HandEarRightBoth - HandEarRightLeft = 0

Model 1: restricted model
Model 2: rtime ~ 0 + HandEar + (1 | subject)

  Res.Df Df     F  Pr(>F)
1     77
2     76  1 6.109 0.01569 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> bothVSright = c(1,  0, -1,  1,  0, -1)
>  linearHypothesis(ComboModel,bothVSright,test="F")
Linear hypothesis test

Hypothesis:
HandEarLeftBoth - HandEarLeftRight  + HandEarRightBoth - HandEarRightRight = 0

Model 1: restricted model
Model 2: rtime ~ 0 + HandEar + (1 | subject)

  Res.Df Df      F   Pr(>F)
1     77
2     76  1 7.8214 0.006536 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> leftVSright = c(0,  1, -1,  0,  1, -1)
> linearHypothesis(ComboModel,leftVSright,test="F")
Linear hypothesis test

Hypothesis:
HandEarLeftLeft - HandEarLeftRight  + HandEarRightLeft - HandEarRightRight = 0

Model 1: restricted model
Model 2: rtime ~ 0 + HandEar + (1 | subject)

  Res.Df Df      F Pr(>F)
1     77
2     76  1 0.1056  0.746
>
>
```

Baayen, Davidson and Bates (2008)

```
> # Baayen, Davidson and Bates (2008) Data (Item is a random effect)
>
> # Reproduce Baayen, Davidson and Bates numbers
> rm(list=ls()); options(scipen=999) # To avoid scientific notation
> # Install packages if necessary. Only need to do this once.
> # install.packages("lme4")
> # install.packages("car")
> # install.packages("tables")
> # Load packages -- do this every time
> library(lme4) # For lmer function
Loading required package: Matrix
> library(car)  # For F-tests, likelihood ratio and Wald chi-squared tests
> library(tables) # For nice-looking tables
Loading required package: Hmisc
Loading required package: lattice
Loading required package: survival
Loading required package: Formula
Loading required package: ggplot2

This data.table install has not detected OpenMP support. It will work but slower in
single threaded mode.


Attaching package: ‘Hmisc’

The following objects are masked from ‘package:base’:

    format.pval, round.POSIXt, trunc.POSIXt, units

> # Read data into a data frame
>
> rt = read.table("BaayenDavidsonBates.Data.txt",header=T)
> rt; attach(rt)
   Subject Item Treatment ReactionTime
1       s1   w1      Long          466
2       s1   w2      Long          520
3       s1   w3      Long          502
4       s1   w1     Short          475
5       s1   w2     Short          494
6       s1   w3     Short          490
7       s2   w1      Long          516
8       s2   w2      Long          566
9       s2   w3      Long          577
10      s2   w1     Short          491
11      s2   w2     Short          544
12      s2   w3     Short          526
13      s3   w1      Long          484
14      s3   w2      Long          529
15      s3   w3      Long          539
16      s3   w1     Short          470
17      s3   w2     Short          511
18      s3   w3     Short          528

> table(Treatment, Item) # Sample sizes
         Item
Treatment w1 w2 w3
    Long   3  3  3
    Short  3  3  3
```

```
>
> # In the tabular syntax, + means stick it together
> # Row descriptions are on the left of ~
> tabular(Treatment ~ ReactionTime*(mean+sd))

           ReactionTime
 Treatment mean         sd
 Long      522.1        35.91
 Short     503.2        25.41
> tabular(Item ~ ReactionTime*(mean+sd))

      ReactionTime
 Item mean         sd
 w1   483.7        18.29
 w2   527.3        25.33
 w3   527.0        30.46
>
> # Fit a mixed model: Treatment effect is fixed, Subject and Item are random
> # and independent.
> mixedmod = lmer(ReactionTime ~ Treatment + (1|Item) + (1|Subject) )
> summary(mixedmod)

Linear mixed model fit by REML ['lmerMod']
Formula: ReactionTime ~ Treatment + (1 | Item) + (1 | Subject)

REML criterion at convergence: 141.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.1932 -0.6004 -0.2425  0.4418  1.7307

Random effects:
 Groups   Name        Variance Std.Dev.
 Item     (Intercept) 607.9    24.66
 Subject  (Intercept) 499.4    22.35
 Residual             137.3    11.72
Number of obs: 18, groups:  Item, 3; Subject, 3

Fixed effects:
               Estimate Std. Error t value
(Intercept)     522.111     19.605  26.632
TreatmentShort  -18.889      5.525  -3.419

Correlation of Fixed Effects:
            (Intr)
TretmntShrt -0.141

> Anova(mixedmod, test="F")
Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)

Response: ReactionTime
              F Df Df.res   Pr(>F)
Treatment 11.69  1     12 0.005087 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
>
```

Unbalanced designs: Beat the Blues study

This R data set contains data from a longitudinal clinical trial of an interactive, multimedia program known as "Beat the Blues" designed to deliver cognitive behavioural therapy to depressed patients via a computer terminal. Patients with depression recruited in primary care were randomised to either the Beating the Blues program, or to "Treatment as Usual" (TAU). The variables are

- `id`: Patient identification code
- `drug`: Did the patient take anti-depressant drugs (No or Yes).
- `length`: The length of the current episode of depression, a factor with values <6m (less than six months) and >6m (more than six months).
- `treatment`: Treatment group, a factor with levels TAU (treatment as usual) and BtheB (Beat the Blues)
- `bdi_pre`: Beck Depression Inventory score before treatment.
- `bdi_2m`: Beck Depression Inventory score after two months
- `bdi_4m`: Beck Depression Inventory score after four months
- `bdi_6m`: Beck Depression Inventory score after six months
- `bdi_8m`: Beck Depression Inventory score after eight months

```
> # Install packages if necessary. Only need to do this once.
> # install.packages("lme4")
> # install.packages("car")
> # Load packages -- do this every time
> library(lme4) # For lmer function
Loading required package: Matrix
> library(car)  # For F-tests, likelihood ratio and Wald chi-squared tests
>
> # install.packages("HSAUR") # Handbook of Statistical Analyses Using R
> library(HSAUR)
Loading required package: tools
> attach(BtheB); head(BtheB) # Beat the Blues data
  drug length treatment bdi.pre bdi.2m bdi.4m bdi.6m bdi.8m
1   No    >6m       TAU      29      2      2     NA     NA
2  Yes    >6m     BtheB      32     16     24     17     20
3  Yes    <6m       TAU      25     20     NA     NA     NA
4   No    >6m     BtheB      21     17     16     10      9
5  Yes    >6m     BtheB      26     23     NA     NA     NA
6  Yes    <6m     BtheB       7      0      0      0      0
>
> # Format is wrong -- need one line per response for lmer
```

```
> # Format is wrong -- need one line per response for lmer
> Subject = Drug = Treatment = Beck = Time = NULL
> n = length(drug); n
[1] 100
> for(j in 1:n)
+     {
+     Time = c(Time, 0)
+         Beck = c(Beck,bdi.pre[j])
+         Subject = c(Subject, j)
+         Drug = c(Drug, as.character(drug[j]))
+         Treatment = c(Treatment, as.character(treatment[j]))
+     Time = c(Time, 2)
+         Beck = c(Beck,bdi.2m[j])
+         Subject = c(Subject, j)
+         Drug = c(Drug, as.character(drug[j]))
+         Treatment = c(Treatment, as.character(treatment[j]))
+     Time = c(Time, 4)
+         Beck = c(Beck,bdi.4m[j])
+         Subject = c(Subject, j)
+         Drug = c(Drug, as.character(drug[j]))
+         Treatment = c(Treatment, as.character(treatment[j]))
+     Time = c(Time, 6)
+         Beck = c(Beck,bdi.6m[j])
+         Subject = c(Subject, j)
+         Drug = c(Drug, as.character(drug[j]))
+         Treatment = c(Treatment, as.character(treatment[j]))
+     Time = c(Time, 8)
+         Beck = c(Beck,bdi.8m[j])
+         Subject = c(Subject, j)
+         Drug = c(Drug, as.character(drug[j]))
+         Treatment = c(Treatment, as.character(treatment[j]))
+     }
> Time = factor(Time) # Otherwise it's numeric
> blues = data.frame(Subject,Time,Drug,Treatment,Beck)
> dim(blues) # Number of rows and columns
[1] 500   5
> head(blues)
  Subject Time Drug Treatment Beck
1       1    0   No       TAU   29
2       1    2   No       TAU    2
3       1    4   No       TAU    2
4       1    6   No       TAU   NA
5       1    8   No       TAU   NA
6       2    0  Yes     BtheB   32
```

```
>
> # Sample sizes
> table(Treatment,Time,Drug)
, , Drug = No

         Time
Treatment  0  2  4  6  8
    BtheB 22 22 22 22 22
    TAU   34 34 34 34 34

, , Drug = Yes

         Time
Treatment  0  2  4  6  8
    BtheB 30 30 30 30 30
    TAU   14 14 14 14 14

> table(Treatment,Drug) # Unbalanced
         Drug
Treatment  No Yes
    BtheB 110 150
    TAU   170  70
>
> # For unbalanced designs, set contrasts to contr.sum on all factors and
> # use the type="III" (Type III sums of squares) option on the Anova function.
> # This way, tests of main effects and interactions will correspond to what you
> # would get by testing contrasts on a combination variable, which is usually
> # what you want. Details are definitely omitted.
>
> BlueModel = lmer(Beck ~ Time*Drug*Treatment + (1 | Subject),
+     contrasts = list(Time="contr.sum", Drug="contr.sum", Treatment="contr.sum"))
> Anova(BlueModel, test="F", type="III") # Satterthwaite adjustment to df
Analysis of Deviance Table (Type III Wald F tests with Kenward-Roger df)

Response: Beck
                          F Df  Df.res   Pr(>F)
(Intercept)        241.0743  1  99.215 < 2e-16 ***
Time                37.9453  4 269.910 < 2e-16 ***
Drug                 0.1434  1  99.215 0.70571
Treatment            3.3556  1  99.215 0.06998 .
Time:Drug            3.0517  4 269.910 0.01748 *
Time:Treatment       0.4104  4 269.910 0.80108
Drug:Treatment       1.1329  1  99.215 0.28974
Time:Drug:Treatment  1.8645  4 269.910 0.11693
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

>
> # This calls for a 2-way table of means
> # install.packages("apaTables")
> library(apaTables) # For 2-way tables of means and Sds

Warning message:
Installed Rcpp (0.12.8) different from Rcpp used to build dplyr (0.12.11).
Please reinstall dplyr to avoid random crashes or undefined behavior.
```

```
> apa.2way.table(Drug,Time,Beck,data=blues, show.marginal.means = T)


Means and standard deviations for Beck as a function of a 2(Drug) X 5(Time) design

           Time
              0           2           4           6           8       Marginal
     Drug     M    SD     M    SD     M    SD     M    SD     M    SD        M    SD
       No 21.55  8.97 17.69 10.98 15.63 10.78 13.88 11.96 12.00 10.19    16.99 10.91
      Yes 25.59 12.58 15.90 10.57 13.75 13.13 11.38 10.14 10.12  8.25    16.51 12.58
 Marginal 23.33 10.84 16.92 10.79 14.81 11.82 12.76 11.15 11.13  9.31

Note. M and SD represent mean and standard deviation, respectively.
Marginal indicates the means and standard deviations pertaining to main effects.

>
> # I actually edited the table by hand to get rid of the standard deviations
```

			Time			
Drug	**0**	**2**	**4**	**6**	**8**	**Marginal**
No	**21.55**	**17.69**	**15.63**	**13.88**	**12.00**	**16.99**
Yes	**25.59**	**15.90**	**13.75**	**11.38**	**10.12**	**16.51**
Marginal	**23.33**	**16.92**	**14.81**	**12.765**	**11.13**	