

Within-cases analysis of Normal Data

The grapefruit data

Grapefruit sales were measured for eight grocery stores at three price levels. The multivariate data format is more natural, but `lmer` wants each value of the response variable to be on a separate line.

Multivariate format

Store	sales1	sales2	sales3
1	62.1	61.3	60.8
2	58.2	57.9	55.1
3	51.6	49.2	46.2
4	53.7	51.5	48.3
5	61.4	58.7	56.6
6	58.5	57.2	54.3
7	46.8	43.2	41.5
8	51.2	49.8	47.9

Univariate format

Store	Price	Sales
1	1	62.1
1	2	61.3
1	3	60.8
2	1	58.2
2	2	57.9
2	3	55.1
3	1	51.6
3	2	49.2
3	3	46.2
4	1	53.7
4	2	51.5
4	3	48.3
5	1	61.4
5	2	58.7
5	3	56.6
6	1	58.5
6	2	57.2
6	3	54.3
7	1	46.8
7	2	43.2
7	3	41.5
8	1	51.2
8	2	49.8
8	3	47.9

These data are from a homework problem in *Applied linear statistical models* (2005, 5th edition) by Kutner et al.

```

> ##### Grapefruit #####
> # http://www.utstat.utoronto.ca/~brunner/data/illegal/GrapefruitData.html
>
> 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")
> # 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
> # Avoid the lmerTest package. It seems to have actual errors.
>
> fruit = read.table("grapefruit.data.txt",header=T) # Local copy
> head(fruit); attach(fruit)
  Store Price Sales
1     1     1  62.1
2     1     2  61.3
3     1     3  60.8
4     2     1  58.2
5     2     2  57.9
6     2     3  55.1
> table(Price,Store)
      Store
Price 1 2 3 4 5 6 7 8
     1 1 1 1 1 1 1 1
     2 1 1 1 1 1 1 1
     3 1 1 1 1 1 1 1
> Price = factor(Price) # Otherwise it's numeric
> aggregate(fruit,by=list(Price),FUN=mean)
  Group.1 Store Price  Sales
1         1     4.5    1 55.4375
2         2     4.5    2 53.6000
3         3     4.5    3 51.3375
> gfmixed = lmer(Sales ~ Price + (1 | Store) )
> summary(gfmixed)
Linear mixed model fit by REML ['lmerMod']
Formula: Sales ~ Price + (1 | Store)

REML criterion at convergence: 93.2

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.48527 -0.41034 -0.07546  0.53703  1.90090

Random effects:
 Groups   Name      Variance Std.Dev.
 Store   (Intercept) 35.2571  5.9378
 Residual              0.6837  0.8269
Number of obs: 24, groups: Store, 8

Fixed effects:
              Estimate Std. Error t value
(Intercept)  55.4375    2.1196  26.155
Price2       -1.8375    0.4134  -4.444
Price3       -4.1000    0.4134  -9.917

Correlation of Fixed Effects:
      (Intr) Price2
Price2 -0.098
Price3 -0.098  0.500

```

```

> anova(gfmixed)
Analysis of Variance Table
      Df Sum Sq Mean Sq F value
Price  2  67.481   33.74  49.346
> # No p-values
> Anova(gfmixed, test="F") # F test (from car package)
Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)

Response: Sales
      F Df Df.res      Pr(>F)
Price 49.346  2      14 0.0000004567 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
> # Compare naive fixed effects model
> anova(lm(Sales ~ Price))
Analysis of Variance Table

Response: Sales
      Df Sum Sq Mean Sq F value Pr(>F)
Price   2  67.48  33.740  0.9388 0.4069
Residuals 21 754.76  35.941

```

Dichotic listening

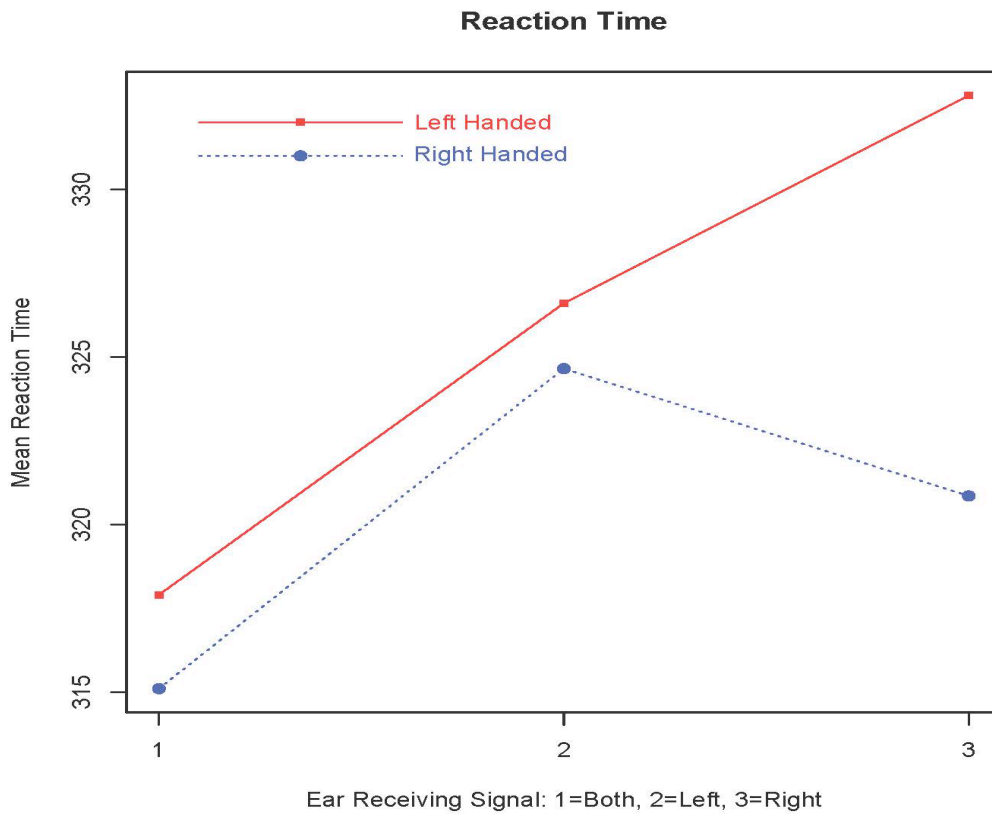
Left-handed and right-handed subjects push a key when they hear their names over background noise. They are wearing stereo headphones. The signal comes in the left ear, the right ear, or both. There are 50 trials in each condition, presented in a different random order for each subject. The response variable is median reaction time in milliseconds. Each subject contributes 3 medians.

```
> 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")
> # 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
> # Avoid the lmerTest package. It seems to have actual errors.
>
> # Read data into a data frame
> dichotic =
read.table("http://www.utstat.toronto.edu/~brunner/data/legal/HandEar.data.txt")
> head(dichotic); attach(dichotic)
  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
>
> # Sample sizes
> table(handed,ear)
      ear
handed Both Left Right
Left    20   20   20
Right   20   20   20
>
> meantable = aggregate(rtime,by=list(ear,handed),FUN=mean)
> colnames(meantable) = c("Ear","Handed","Mean RT")
> meantable
  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
>
> # 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
```

```

>
> addmargins(TwoWay,margin=c(1,2),FUN=mean) # Add marginal means to table
Margins computed over dimensions
in the following order:
1:
2:
      BothEars LeftEar RightEar    mean
LeftHanded   317.9 326.600  332.800 325.7667
RightHanded   315.1 324.650  320.850 320.2000
mean          316.5 325.625  326.825 322.9833
>
> # 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.
>
> # 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 Bonferroni corrected pairwise comparisons of marginal means for ear.
> # 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

The Dichotic Listening data are balanced, and tests from `lmer` (with the `car` package) are the classical exact Fs. These do not exist for unbalanced designs.

Unbalanced Data: Ana's Reaction Time Data

English and Spanish speaking children reacted to sentences of three types (Exclamation, Question, Statement) at three levels of context (Full, Low, None). The response variable is median reaction time in 50 trials. Each subject contributes 9 medians.

```
> 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")
> # 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
> # Avoid the lmerTest package. It seems to have actual errors.
>
> # Read data into a data frame
> rt = read.table("anaRT2.data.txt") # Edited to be more unbalanced than it really
is.
> head(rt,12) # First 12 lines
  PARTICIPANT Language sentencetype context  rtime
1          E01  English      Question   None 2955.0
2          E01  English Exclamation   None 2877.0
3          E01  English   Statement   None 2978.5
4          E01  English      Question   Low 3758.0
5          E01  English Exclamation   Low 5393.0
6          E01  English   Statement   Low 3380.5
7          E01  English      Question  Full 21058.5
8          E01  English Exclamation  Full 20875.0
9          E01  English   Statement  Full 17702.5
10         E02  English      Question   None 3271.0
11         E02  English Exclamation   None 3134.0
12         E02  English   Statement   None 3898.5
> # Notice Full context.
> summary(rt)
PARTICIPANT      Language      sentencetype context      rtime
E01      : 9  English:135  Exclamation:90  Full:90  Min.      : 2254
E02      : 9  Spanish:135  Question :90  Low :90  1st Qu.  : 3004
E03      : 9           Statement :90  None:90  Median   : 3679
E04      : 9
E05      : 9
E06      : 9
(Other):216
                        Mean      : 8783
                        3rd Qu.  :18221
                        Max.      :22734
                        NA's     :10
> rt = na.omit(rt) # Now it's unbalanced
> rt = subset(rt,context!='Full') # Full context gives MUCH slower reaction times
> attach(rt)
> table(sentencetype, context)
      context
sentencetype  Full Low None
Exclamation    0  24  30
Question       0  29  30
Statement      0  30  30
> context=factor(context)
> table(context, sentencetype)
      sentencetype
context Exclamation Question Statement
Low     24          29          30
None    30          30          30
```

```

>
> Ns = table(context, sentencetype, Language); Ns
, , Language = English

      sentencetype
context Exclamation Question Statement
Low      9          14          15
None     15         15          15

, , Language = Spanish

      sentencetype
context Exclamation Question Statement
Low      15         15          15
None     15         15          15

>
> meanz = aggregate(rtime, by=list(context,sentencetype,Language), FUN=mean)
> meanz
  Group.1      Group.2 Group.3      x
1   Low Exclamation English 3031.944
2   None Exclamation English 3013.833
3   Low  Question English 3073.964
4   None  Question English 3517.800
5   Low  Statement English 2840.833
6   None  Statement English 3632.833
7   Low Exclamation Spanish 3366.133
8   None Exclamation Spanish 3586.867
9   Low  Question Spanish 2955.700
10  None  Question Spanish 3745.933
11  Low  Statement Spanish 3264.233
12  None  Statement Spanish 3553.133
> meanz = meanz[,4]; dim(meanz) = c(2,3,2)
> meanz
, , 1

      [,1]      [,2]      [,3]
[1,] 3031.944 3073.964 2840.833
[2,] 3013.833 3517.800 3632.833

, , 2

      [,1]      [,2]      [,3]
[1,] 3366.133 2955.700 3264.233
[2,] 3586.867 3745.933 3553.133

>
> # Use nice labels from Ns
> meantable = Ns
> meantable[1,,1] = meanz[1,,1]
> meantable[2,,1] = meanz[2,,1]
> meantable[1,,2] = meanz[1,,2]
> meantable[2,,2] = meanz[2,,2]

```

```

> meantable
, , Language = English

      sentencetype
context Exclamation Question Statement
Low      3031.944 3073.964 2840.833
None     3013.833 3517.800 3632.833

, , Language = Spanish

      sentencetype
context Exclamation Question Statement
Low      3366.133 2955.700 3264.233
None     3586.867 3745.933 3553.133

>
> # For unbalanced designs, use effect coding (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.
>
> contrasts(Language) = contr.sum
> contrasts(context) = contr.sum
> contrasts(sentencetype) = contr.sum
> contrasts(sentencetype)
      [,1] [,2]
Exclamation    1    0
Question        0    1
Statement       -1   -1
>
> reaction1 = lmer(rtime ~ Language*sentencetype*context + (1 | PARTICIPANT) )
> Anova(reaction1, test="F", type="III")
Analysis of Deviance Table (Type III Wald F tests with Kenward-Roger df)

Response: rtime

```

	F	Df	Df.res	Pr(>F)	
(Intercept)	2682.3734	1	28.329	< 0.00000000000000022	***
Language	2.9058	1	28.329	0.099206	.
sentencetype	0.1935	2	134.152	0.824288	
context	29.6945	1	133.951	0.0000002349	***
Language:sentencetype	1.7813	2	134.152	0.172373	
Language:context	0.0937	1	133.951	0.760009	
sentencetype:context	5.1662	2	134.152	0.006896	**
Language:sentencetype:context	3.4458	2	134.152	0.034729	*

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

>
> # Marginal table of Sentence Type by Context
> scon = apply(meantable,c(1,2),mean); scon
      sentencetype
context Exclamation Question Statement
Low      3199.039 3014.832 3052.533
None     3300.350 3631.867 3592.983
> # Difference between differences
> scon[2,] - scon[1,] # None minus Low
Exclamation      Question      Statement
101.3111         617.0345         540.4500

>
> # What about the 3-way?
> meantable[2,,1] - meantable[1,,1] # None-Low for English
Exclamation      Question      Statement
-18.11111        443.83571        792.00000
> meantable[2,,2] - meantable[1,,2] # None-Low for Spanish
Exclamation      Question      Statement
220.7333         790.2333         288.9000
>

```