610 R9 -- Two-way Repeated-measures Anova **Prof Colleen F. Moore**
Balanced Designs (complete data) only
For Psychology 610, University of Wisconsin--Madison

This uses the data of class HO#24, Prof Lopes's poker example. It is available on the course website as HO#24data.xls.

Contents of this tutorial:

I. Data setup and basic anova for 2-way repeated measures, using the 'car' package and 'Anova' (capital A). Yields sphericity tests and adjusted p-values.
II. Contrasts with partitioned error
III. Method using 'aov'. Requires either data manipulation in R or a spreadsheet so that each observation on a separate line. Allows use of 'model.tables' to obtain estimated effects, means, and standard errors.

'Quick Look' Summary of R Code:
Using 'Anova' in 'car' package:
```
> multmodel=lm(cbind(dv1,dv2,dv3)~1)
> model1=Anova(multmodel,idata=your.factors,idesign=A*B,type="III") # make a matrix that lays out the order of the factors and use that as 'idata'
> summary (model1, multivariate=F)
```

Using 'aov':
-- data is entered with one observation per line, plus conditions and observation number (subjects) for that data point
-- conditions are represented as 'factors' in R
```
> model=aov(dv ~ A*B + Error(person / (A*B)))  # specify error term  carefully!!
```

I. Data setup and anova using 'Anova' in 'car' package.

Here I am using the 2nd data example that begins on p. 4 of H.O. # 24.

A. Bring data in and do the analysis in two steps.
```
> library(car)  # activate the 'car' package in the environment
> your.data=read.table(pipe("pbpaste"),header=T)
> your.data
  partic A1B1 A2B1 A3B1 A4B1 A1B2 A2B2 A3B2 A4B2
1     p1    1    3    7   10    2    7    8    3
2     p2    2    4    6    9    4    4    7    4
3     p3    1    4    7    9    4    4    5    5
4     p4    1    3    8   10    3    6    4    4
5     p5    2    3    7    9    2    3    6    6
> attach(your.data)
```

Step 1: Calculate a multivariate model over all repeated measures.

> multmodel=lm(cbind(A1B1 ,A2B1, A3B1, A4B1 ,A1B2, A2B2, A3B2 ,A4B2)~1) # use 'lm' to calculate a multivariate model that uses only the intercept as predictor. 'cbind' makes a vector of all the data columns of the repeated measures variables.

Step 2. Set up a matrix of factor codes for the repeated measures variable to use inside 'Anova' (capital A). I laid these out in the excel spread sheet, and then pasted them into R from the clipboard

> ex2.idata=read.table(pipe("pbpaste"),header=T)

> ex2.idata

```
   A  B
1 A1 B1
2 A2 B1
3 A3 B1
4 A4 B1
5 A1 B2
6 A2 B2
7 A3 B2
8 A4 B2
```

> attach(ex2.idata)

Step 3. Carry out the analysis of variance using 'Anova' (capital A).

> ex2Anova=Anova(multmodel,idata=ex2.idata,idesign=~ A*B, type="III") # R will regard A and B as factors because they are labeled with letters in their names. Notice that the matrix 'ex2.idata' is laid out so it corresponds to the data format – as you go down the rows of ex2.idata, you go across the columns of the data in 'your.data'.

> summary(ex2Anova,multivariate=F) # check the dfs in the results to make sure the design is set up properly

```
Univariate Type III Repeated-Measures ANOVA Assuming Sphericity

                SS num Df Error SS den Df        F    Pr(>F)    
(Intercept) 970.22      1     0.65      4 5970.615 1.681e-07 ***
A           144.87      3    10.75     12   53.907 3.090e-07 ***
B             5.62      1     1.25      4   18.000 0.0132356 *  
A:B          70.67      3    22.95     12   12.318 0.0005653 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Mauchly Tests for Sphericity

    Test statistic p-value
A          0.60094 0.93084
A:B        0.57528 0.91838


Greenhouse-Geisser and Huynh-Feldt Corrections
 for Departure from Sphericity

     GG eps Pr(>F[GG])    
A   0.75291  7.441e-06 ***
A:B 0.79694   0.001739 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

    HF eps Pr(>F[HF])
```

```
A   1.7791   3.09e-07 ***
A:B 2.0619  0.0005653 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Warning message:
In summary.Anova.mlm(ex2Anova, multivariate = F) : HF eps > 1 treated as 1
```

II. Contrasts with partitioned error for two-way within

A. Interaction contrasts

This section ***uses the first data*** set on HO#24. I apologize for switching data sets back and forth, but I had an issue that I hope to solve later.

Step 1. Make contrast coefficients

Let's do the A-linear x B-linear interaction contrast that is on the class handout. Make a vector of the coefficients that is arranged the same way the data are. In R we have to think about which version of the data to use: long format or matrix version?? I think it will be easier to use the matrix version.

```
> LxLcoeff = c(-3,-1,1,3,0,0,0,0,3,1,-1,-3)  # make the contrast coeff's so they match up with the data matrix properly.
```

Step 2. Apply contrast coefficients to the individual scores

```
> LxLpsi=cbind(A1B1,A1B2,A1B3,A1B4,A2B1,A2B2,A2B3,A2B4,A3B1,A3B2,A3B3,A3B4) %*%LxLcoeff
        # use matrix multiplication to find the vector of psi-hats. Order matters in matrix multiplication !!
> LxLpsi   # here is the vector of psi-hats
     [,1]
[1,]  -24
[2,]  -19
[3,]  -21
[4,]  -25
[5,]  -21
```

Step 3. Test vector of psi-hats versus zero, by t-test or anova testing grand mean.

```
> t.test(LxLpsi)        # now do a t-test on it. Null H is that psi in pop =0.

        One Sample t-test

data:  LxLpsi
t = -20.0832, df = 4, p-value = 3.628e-05
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -25.04144 -18.95856
sample estimates:
mean of x
      -22
```

Or you can use 'aov' to test whether the grand mean is sig diff from zero. That's the test of the intercept in the table below.

```
> psi.aov = aov(LxLpsi~1) ; summary(psi.aov, intercept=T) # the '~1' tells R to test the intercept, and nothing else
```

```
            Df Sum Sq Mean Sq F value    Pr(>F)    
(Intercept)  1   2420    2420  403.33 3.628e-05 ***
Residuals    4     24       6                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

III. Alternative method using 'aov' by stacking data, etc.

This section also uses the first data set on H.O. #24. I rearranged the data in an excel spreadsheet so that each observation is on a separate line. With a small data set, this is easy to do. With a large data set it isn't easy. The disadvantage of using 'aov' for repeated-measures designs is that it doesn't give us the sphericity tests or the adjusted p-values.

A. Bring data into R.

```
> your.data=read.table(pipe("pbpaste"),header=T)
> your.data
   partic dv A B
1      p1  1 1 1
2      p2  2 1 1
3      p3  1 1 1
4      p4  1 1 1
5      p5  2 1 1
6      p1  1 1 2
7      p2  2 1 2
8      p3  2 1 2
9      p4  1 1 2
10     p5  2 1 2
11     p1  2 1 3
12     p2  3 1 3
13     p3  2 1 3
. . .
51     p1  7 3 3
52     p2  6 3 3
53     p3  7 3 3
54     p4  8 3 3
55     p5  7 3 3
56     p1 10 3 4
57     p2  9 3 4
58     p3  9 3 4
59     p4 10 3 4
60     p5  9 3 4
```

B. Create factors and carry out the anova

```
> attach(your.data)
> Afac=factor(A) # make a factor out of 'A' because I used numerical values
> Bfac=factor(B)
> within2way.aov1=aov(dv ~ Afac*Bfac + Error(partic / (Afac*Bfac)))  # specify the error term carefully
> summary(within2way.aov1)
```

```
Error: partic
          Df  Sum Sq Mean Sq F value Pr(>F)
Residuals  4 3.06667 0.76667         # This is the subjects effect

Error: partic:Afac
          Df  Sum Sq Mean Sq F value    Pr(>F)
Afac       2 109.200  54.600  94.273 2.745e-06 ***
Residuals  8   4.633   0.579           # this is A x subjects
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: partic:Bfac
          Df  Sum Sq Mean Sq F value    Pr(>F)
Bfac       3 194.183  64.728  135.48 1.611e-09 ***
Residuals 12   5.733   0.478      # this is B x subjects
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: partic:Afac:Bfac
          Df Sum Sq Mean Sq F value    Pr(>F)
Afac:Bfac  6 63.867  10.644   36.67 6.164e-11 ***
Residuals 24  6.967   0.290          # this is A x B x subjects
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Check that the error terms and df's are correct!!

C. Use 'model.tables' to obtain means, etc.

An advantage of using 'aov' is that then we can use 'model.tables' to find means and estimated effects.

```
> model.tables(within2way.aov1,"means", se=T)  # name the anova model just constructed above
Tables of means
Grand mean

3.65

 Afac
Afac
   1    2    3
2.05 3.55 5.35

 Bfac
Bfac
    1     2     3     4
1.533 2.467 4.400 6.200

 Afac:Bfac
    Bfac
Afac 1   2   3   4
   1 1.4 1.6 2.4 2.8
   2 1.6 2.4 3.8 6.4
   3 1.6 3.4 7.0 9.4
Warning message:
In model.tables.aovlist(within2way.aov1, "means", se = T) :
  SEs for type 'means' are not yet implemented
```

> model.tables(within2way.aov1, se=T) # the default is estimated effects, and with this we can also obtain estimated standard errors. These can the be used in a graph.

```
Tables of effects

 Afac
Afac
   1    2    3
-1.6 -0.1  1.7

 Bfac
Bfac
      1       2       3       4
-2.1167 -1.1833  0.7500  2.5500

 Afac:Bfac
    Bfac
Afac   1       2       3       4
   1  1.4667  0.7333 -0.4000 -1.8000
   2  0.1667  0.0333 -0.5000  0.3000
   3 -1.6333 -0.7667  0.9000  1.5000

Standard errors of effects
          Afac   Bfac Afac:Bfac
        0.1702 0.1785    0.2409
replic.     20     15         5
```

Note: a little hand calculation verifies that the estimated standard errors are the sqrt of (MSerror for the relevant source divided by the number of observations entering the mean for that source).