Colleen Moore & Mike Amato
University of Wisconsin - Madison

1. Enter the Data

For this exercise we'll continue using the data from class handout #3, in which we are trying testing for variance in "score" due to "group". See handout R1 on entering data.

```
> data1 = read.table(file.choose(), header=T)
> attach(data1)
> group = as.factor(group)
> data1
  group score
1     1     8
2     1     6
3     1     7
...
```

Factor A				
level 1	level 2	level 3	level 4	level5
8	4	5	3	6
6	5	3	4	7
7	5	3	2	5
7	6	6	2	4
6	3	2	3	6

2. Do a specific pairwise test of two of the means

Is group1 significantly different from group2? First make a separate variable for the set of scores in each group. The following command makes a new vector, *group1*, which contains only the scores from participants in group1, and then does the same for *group2*.

```
> group1 = score[group=="1"]     #single equal sign means "set equal to"
> group2 = score[group=="2"]     #double means "evaluates as equal to"
> group1
[1] 8 6 7 7 6
> group2
[1] 4 5 5 6 3
> t.test(group1,group2, var.equal=T)

        Two Sample t-test

data:  group1 and group2
t = 3.4785, df = 8, p-value = 0.008338
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.741555 3.658445
sample estimates:
mean of x mean of y
      6.8       4.6
```

By default R assumes that variance is NOT equal, and so unless you say otherwise it will use the Welch method to adjust your degrees of freedom and your p-value. By setting *var.equal=T*, we are able to take advantage of our friend the pooled variance.

The above t-test compares the scores from group1with the scores from group2. The means of the two groups are significantly different, $t(8)= 3.4785$, $p<.01$.

If appropriate given our design, we could instead perform a paired t-test. Note that here we don't bother creating new variables like we did above. We could have, but this is quicker.

```
> t.test(score[group=="1"], score[group=="2"], paired=T)

        Paired t-test

data:  score[group == 1] and score[group == 2]
t = 3.773, df = 4, p-value = 0.01955
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.5810682 3.8189318
sample estimates:
mean of the differences
                    2.2
```

3. Create and test a set of Helmert contrasts

R can automatically create and then test a set of Helmert contrasts.
First we create the set of contrasts:

```
> coefs = contr.helmert(5)          #"5" because we have 5 groups
> coefs                             #displays the contrasts
  [,1] [,2] [,3] [,4]
1   -1   -1   -1   -1
2    1   -1   -1   -1
3    0    2   -1   -1
4    0    0    3   -1
5    0    0    0    4
```

Each column is a contrast.
Each row is a group.

R allows you to set the "contrasts" attribute of a factor. This attribute will just hang out in the background for now, waiting to spring into action when we perform an ANOVA.

```
> contrasts(group) = coefs          # sets the contrasts attribute of factor group
```

Run the anova...

```
> anova1 = aov(score~group)
```

This parameter says to show us each of the 4 contrasts of *group*.

...then ask for the summary, including the four contrasts.

```
> summary(anova1, split=list(group=c(1,2,3,4)) )
             Df Sum Sq Mean Sq F value    Pr(>F)
group         4 48.240  12.060  9.0000 0.0002508 ***
  group: C1   1 12.100  12.100  9.0299 0.0069971 **
  group: C2   1 12.033  12.033  8.9801 0.0071289 **
  group: C3   1 19.267  19.267 14.3781 0.0011441 **
  group: C4   1  4.840   4.840  3.6119 0.0718780 .
Residuals    20 26.800   1.340
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4. Test all polynomial trends

Similar to the Helmert contrasts, you can tell R to create contrasts to test for polynomial trends in your groups.

```
> coefs2 = contr.poly(5)            #"5" because we have 5 groups
> contrasts(group) = coefs2         #set the contrasts attribute of group
> anova2 = aov(score~group)         #do the anova
```

Let's label our contrasts this time. First we'll create a list of four labels for our four contrasts, then we'll use that list with the split parameter when we ask for the summary.

```
> polylabels = list("linear"=1, "quadratic"=2, "cubic"=3, "quartic"=4)

> summary(anova2, split=list(group=polylabels))
                    Df Sum Sq Mean Sq F value    Pr(>F)
group                4  48.24   12.06  9.0000 0.0002508 ***
  group: linear      1   8.82    8.82  6.5821 0.0184486 *
  group: quadratic   1  34.30   34.30 25.5970 5.997e-05 ***
  group: cubic       1   2.88    2.88  2.1493 0.1581860
  group: quartic     1   2.24    2.24  1.6716 0.2107755
Residuals           20  26.80    1.34
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5. Test a specific one-way contrast

Let's say that we're interested in testing the following contrasts:

group1 and group3 versus group2 and group5 (1,-1,1,0,-1)
group4 versus all others (-1,-1,-1,4,-1)

First make a matrix with the two sets of contrast coefficients.

```
> coefs= cbind( c(1,-1,1,0,-1), c(-1,-1,-1,4,-1) )
```

Set the "contrasts" attribute of factor *group*.

```
> contrasts(group) = coefs
```

Next create a list of names for the contrasts, to help you interpret the output. We are testing two contrasts, so we need to provide a label for contrast #1 and another label for contrast #2.

```
> contrastlabels = list("1and3_vs_4and5"=1, "4_vs_others"=2)
```

So go ahead and do that ANOVA now. Include the *split* parameter so R shows you the contrasts, using the contrast labels you gave it. Just for kicks, ask for the intercept too.

```
> summary(anova1, intercept=T, split=list(group=contrastlabels) )
```

```
                        Df Sum Sq Mean Sq  F value    Pr(>F)    
(Intercept)              1 556.96  556.96 415.6418 7.486e-15 ***
group                    4  48.24   12.06   9.0000 0.0002508 ***
  group: 1and3_vs_4and5  1   0.20    0.20   0.1493 0.7033289    
  group: 4_vs_others     1  23.04   23.04  17.1940 0.0004994 ***
Residuals               20  26.80    1.34                       
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So we find that there is not a significant difference in the first contrast F(1,20)= 0.15, p>.05, but there is in the second, F(1,20)=17.19, p<.001.