Statistics Lab #4

PSYCH 710

4 Initialize R

Before starting this lab, please initialize R by entering the following commands at the prompt:

```r
options(contrasts = c("contr.sum", "contr.poly"))
options(digits = 6)
options(width = 70)
source(url("http://psycserv.mcmaster.ca/bennett/psy710/Rscripts/linear_contrast_v2.R"))
```

```r
## [1] "loading function linear.comparison"
```

### Table 1: Results of one-way ANOVA described in Section 4.1.

<table>
<thead>
<tr>
<th>Source</th>
<th>Df</th>
<th>SS</th>
<th>MS</th>
<th>F</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>group</td>
<td>2</td>
<td>187</td>
<td>93.3</td>
<td>3.73</td>
<td>0.037</td>
</tr>
<tr>
<td>Residuals</td>
<td>27</td>
<td>675</td>
<td>25.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.1 Example #1

This example is question #3 in the Exercises for Chapter 4 (and appears in my notes for that chapter). A psychologist collected data from three groups and conducted a one-way ANOVA to evaluate differences among the groups. The results from the ANOVA are shown in Table 1. The sample means are: \( \bar{Y}_1 = 12, \bar{Y}_2 = 10, \bar{Y}_3 = 6 \). MS\(_W\) = 25, and there are 10 subjects per group. We want to compare the average of the means of Groups 1 and 2 to the mean of Group 3. We will use contrast coefficients \( c=(1, 1, -2) \) to evaluate this comparison. The null and alternative hypotheses are

\[
\begin{align*}
H_0 & : (1)\mu_1 + (1)\mu_2 - (2)\mu_3 = 0 \\
H_1 & : (1)\mu_1 + (1)\mu_2 - (2)\mu_3 \neq 0
\end{align*}
\]

Because we only have the group means, and not the raw scores, we need to use the following equation to calculate \( \Psi \):

\[
\hat{\Psi} = \sum_{j=1}^{a} c_j \bar{Y}_j = (1)12 + (1)10 - (2)6 = 12 + 10 - 12 = 10
\]

Next, we use MS\(_W\) = 25 to calculate \( F \):

\[
F = \frac{(\Psi^2) / \sum_{j=1}^{a} (c_j^2 / n_j)}{MS_W} = \frac{10^2 / (1/10 + 1/10 + 4/10)}{25} = \frac{100/(6/10)}{25} = \frac{166.67}{25} = 6.67
\]

with numerator and denominator degrees of freedom of 1 and 30 – 3 = 27, respectively. In R, we can do these calculations with the following commands:
the.means <- c(12, 10, 6); # group means
the.weights <- c(1, 1, -2); # contrast weights
n <- 10; # number of subjects per group
sum(the.weights) # must sum to zero!

## [1] 0

MS.w <- 25; # Mean Square Within-cell
# in the next line, "sum(the.weights*the.means)" is the same as "(1*12) + (1*10) + (-2*6)"
(psi <- sum(the.weights*the.means) ) # psi is the value of the comparison

## [1] 10

the.numerator <- (psi^2) / (sum(the.weights^2) / n)
(F <- the.numerator / MS.w) # F statistic

## [1] 6.66667

If the null hypothesis is true, the probability of obtaining this F or one more extreme is

pf(6.67, 1, 27, lower.tail=FALSE)

## [1] 0.0155463

1 - pf(6.67, 1, 27)

## [1] 0.0155463

which is less than $\alpha = 0.05$, so we reject the null hypothesis in favor of the alternative. Note, by the way, that the two forms of the pf command give the same result.

### 4.2 Example #2

We will now compute contrasts on the set of (fictitious) blood pressure data listed in Table 4.1 in Maxwell and Delaney. The data, shown in Table 2, are from four groups of subjects who received four different treatments for hypertension: drug therapy (aDrug), biofeedback (bFeedback), dietary modification (cDiet), and a treatment that combines all aspects of the other treatments (dCombo). The dependent variable is blood pressure after treatment and is stored in bp$blood.

bp <- read.csv(url("http://psycserv.mcmaster.ca/bennett/psy710/datasets/maxwell_tab41.csv"))
names(bp)

## [1] "blood" "group"

levels(bp$group)

## [1] "aDrug" "bFeedback" "cDiet" "dCombo"

We are going to do three contrasts. First, we want to compare dCombo to the average of the other groups, so we will use a contrast c=(1/3, 1/3, 1/3, -1). Next, we will compare cDiet to the average of aDrug and bFeedback with a contrast c=(1/2, 1/2, -1, 0). Finally, we will compare aDrug to bFeedback with the contrast c=(1, -1, 0, 0). (Note that we must set the contrast for all groups, even ones that are not included in the contrast.) We first construct a list that has all three contrasts.
Table 2: Blood Pressure Data (Table 4.1)

<table>
<thead>
<tr>
<th>blood group</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 84.00 aDrug</td>
</tr>
<tr>
<td>2 95.00 aDrug</td>
</tr>
<tr>
<td>3 93.00 aDrug</td>
</tr>
<tr>
<td>4 104.00 aDrug</td>
</tr>
<tr>
<td>5 81.00 bFeedback</td>
</tr>
<tr>
<td>6 84.00 bFeedback</td>
</tr>
<tr>
<td>7 92.00 bFeedback</td>
</tr>
<tr>
<td>8 101.00 bFeedback</td>
</tr>
<tr>
<td>9 80.00 bFeedback</td>
</tr>
<tr>
<td>10 108.00 bFeedback</td>
</tr>
<tr>
<td>11 98.00 cDiet</td>
</tr>
<tr>
<td>12 95.00 cDiet</td>
</tr>
<tr>
<td>13 86.00 cDiet</td>
</tr>
<tr>
<td>14 87.00 cDiet</td>
</tr>
<tr>
<td>15 94.00 cDiet</td>
</tr>
<tr>
<td>16 91.00 dCombo</td>
</tr>
<tr>
<td>17 78.00 dCombo</td>
</tr>
<tr>
<td>18 85.00 dCombo</td>
</tr>
<tr>
<td>19 80.00 dCombo</td>
</tr>
<tr>
<td>20 81.00 dCombo</td>
</tr>
</tbody>
</table>

blood.contrasts <- list(c(1/3, 1/3, 1/3, -1), c(1/2, 1/2, -1, 0), c(1, -1, 0, 0))

And then we pass the `bp$blood`, `bp$group`, and `blood.contrasts` to the function `linear.comparison`:

```r
bp.results <- linear.comparison(bp$blood, bp$group, blood.contrasts, var.equal = TRUE)

# [1] "computing linear comparisons assuming equal variances among groups"
# [1] "C 1: F=4.815, t=2.194, p=0.043, psi=9.333, CI=(2.554,16.112), adj.CI= (-2.036,20.703)"
# [1] "C 2: F=0.012, t=0.110, p=0.913, psi=0.500, CI=(-7.767,8.767), adj.CI= (-11.601,12.601)"
# [1] "C 3: F=0.321, t=0.566, p=0.579, psi=3.000, CI=(-10.210,16.210), adj.CI= (-11.163,17.163)"

bp.results.unequal <- linear.comparison(bp$blood, bp$group, blood.contrasts, var.equal = FALSE)

# [1] "computing linear comparisons assuming unequal variances among groups"
# [1] "C 1: F=8.519, t=2.919, p=0.014, psi=9.333, CI=(2.298,16.369), adj.CI= (0.320,18.346)"
# [1] "C 2: F=0.016, t=0.128, p=0.900, psi=0.500, CI=(-8.009,9.009), adj.CI= (-10.361,11.361)"
# [1] "C 3: F=0.232, t=0.481, p=0.643, psi=3.000, CI=(-11.407,17.407), adj.CI= (-15.862,21.862)"
```

We can list the results for comparison n using the syntax `bp.results[[n]]` (note the double brackets!). The results for each comparison are quite long, so I'll show only one complete record:

```r
bp.results[[1]]

# $contrast
# [1] 0.333333 0.333333 0.333333 -1.000000
#
# $F
# [1] 4.81505
#
# $t
```
The contrast coefficients are listed, as are the values of $F$, $t = \sqrt{F}$, $p$ and $\text{SS}_{\text{contrast}}$. Also listed are the value of $\hat{\Psi}$ and the 100$(1 - \alpha)$% confidence interval for $\Psi$. The last four values are measures of effect size and association strength, which we'll discuss later. The $F$ and $p$ values for all three contrasts are:

```r
for (kk in 1:3){
  print(sprintf("Contrast %2i, F = %.3f, p = %.3f", kk, bp.results[[kk]]$F, bp.results[[kk]]$p.2tailed))
}
```

```r
## [1] "Contrast 1, F = 4.815, p = 0.043"
## [1] "Contrast 2, F = 0.012, p = 0.913"
## [1] "Contrast 3, F = 0.321, p = 0.579"
```

The results from the contrasts assuming unequal group variances are:

```r
for (kk in 1:3){
  print(sprintf("Contrast %2i, F = %.3f, p = %.3f", kk, bp.results.unequal[[kk]]$F, bp.results.unequal[[kk]]$p.2tailed))
}
```

```r
## [1] "Contrast 1, F = 8.519, p = 0.014"
## [1] "Contrast 2, F = 0.016, p = 0.900"
## [1] "Contrast 3, F = 0.232, p = 0.643"
```
The contrast between dCombo and the other groups is significant, but the other contrasts are not. The value of the contrast is:

\[
\hat{\psi} = \frac{1}{3} (\mu_1 + \mu_2 + \mu_3) - \mu_4 = 9.333
\]

In other words, using the contrast that was calculated assuming equal group variances,

\[
\hat{\psi} = \frac{1}{3} (\mu_1 + \mu_2 + \mu_3) - \mu_4 = 9.333
\]

The confidence interval for the comparison is

Next, we will compute and evaluate the contrasts using the \texttt{lsmeans} and \texttt{contrast} commands in the \texttt{lsmeans} package:

Finally, we will evaluate the statistical significance of the contrasts using the \texttt{summary} commands on an \texttt{lm}:
Note that the results of the *t* tests listed in the regression summary table are the same as the ones produced by `lsmeans` and the equal-variance `linear.comparison` calculation presented above. In general, the values of the coefficients will not equal the values of psi. Finally, we will evaluate the linear contrasts using using the `split` and `aov` commands:

```r
c.mat <- cbind(c(1/3, 1/3, 1/3, -1), c(1/2, 1/2, -1, 0), c(1, -1, 0, 0)) # contrast matrix
contrasts(bp$group) <- c.mat # link contrasts with group factor
aov.01 <- aov(blood ~ group, data = bp) # aov, not lm
summary(aov.01, split = list(group = list(C1 = 1, C2 = 2, C3 = 3))) # summary(), not anova()
```

These results are similar but not identical to the results obtained previously. Why?

```r
with(bp, tapply(blood, group, length)) # list n in each group
```

### aDrug bFeedback cDiet dCombo
### 4 6 5 5

**Answer:** The `split` command is best used when we have equal *n* per group. In this case, we have unequal *n* and so the results are a bit off (though they aren’t too bad because we *almost* have equal *n*). If this is correct, then equating *n* across groups should yield the same results with the different commands...
aov.02 <- aov(blood~group,data=bp2)
ls.means.2 <- lsmeans(aov.02,"group")
contrast(ls.means.2, blood.contrasts, adjust="none")

## contrast estimate  SE df t.ratio p.value
## C1  12.25  4.51 12  2.717  0.0187
## C2   4.12  4.78 12  0.863  0.4053
## C3  -1.25  5.52 12 -0.226  0.8247

# compare to linear.comparison:
tmp <- linear.comparison(bp2$blood, bp2$group, blood.contrasts, var.equal = TRUE)

## [1] "computing linear comparisons assuming equal variances among groups"
## [1] "C 1: F=7.383, t=2.717, p=0.019, psi=12.250, CI=(5.823,18.677), adj.CI= (-0.281,24.781)"
## [1] "C 2: F=0.744, t=0.863, p=0.405, psi=4.125, CI=(-5.313,13.563), adj.CI= (-9.166,17.416)"
## [1] "C 3: F=0.051, t=-0.226, p=0.825, psi=-1.250, CI=(-17.171,14.671), adj.CI= (-16.598,14.098)"

# use split:
summary(aov.02,split=list(group=list(C1=1,C2=2,C3=3) ) )

## Df Sum Sq Mean Sq  F value Pr(>F)
## group  3  499 166.33  2.73  0.091 .
## group: C1  1  450 450.00  7.38  0.019 *
## group: C2  1  45  45.00  0.74  0.405
## group: C3  1  3  3.00  0.06  0.825
## Residuals 12  732  61.00
## ---
## Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

# value of PSI = sqrt( SS.contrast * sum(weights^2) / n )
colSums(contrasts(bp2$group)^2)  # sum of squared contrast weights

## [1] 1.33333 1.50000 2.00000

n <- 4  # 4 per group
psil <- sqrt(450*1.333/n)
psil2 <- sqrt(45*1.5/n)
psil3 <- sqrt(3*2/n)

psi.values <- c(psil,psil2,psil3)  # same as contrast command in lsmeans

## [1] 12.24592  4.10792  1.22474

lm.02 <- lm(blood~group,data=bp2)
summary(lm.02)  # inspect the t and p values

##
## Call:
## lm(formula = blood ~ group, data = bp2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -15.25  -3.31  -0.50   4.12  12.75
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)  90.18811   1.95192  46.200   6.9e-15 ***
Answer: Note that the \( t \), \( F \), and \( p \) values for the three linear contrasts are consistent across methods. The important point is that unequal \( n \) affects the computations of linear contrasts by \texttt{split} and \texttt{aov}.

### 4.3 Cuckoo Data

The following text was taken from the Data and Story Library (http://lib.stat.cmu.edu/DASL):

A study by E.B. Chance in 1940 called The Truth About the Cuckoo demonstrated that cuckoos return year after year to the same territory and lay their eggs in the nests of a particular host species. Further, cuckoos appear to mate only within their territory. Therefore, geographical sub-species are developed, each with a dominant foster-parent species, and natural selection has ensured the survival of cuckoos most fitted to lay eggs that would be adopted by a particular foster-parent species.

Read the cuckoo data file into R. Note the following command should be typed on a single line:

```r
cuckoo <- read.csv("http://psycserv.mcmaster.ca/bennett/psy710/datasets/cuckoo2.csv")
```

The data frame contains the variable \texttt{egglength}, which is the length of a cuckoo’s egg (in mm) and the factor \texttt{hostbird}, which is the name of the host bird (i.e., the foster parent) of each egg. The levels of \texttt{hostbird} are indicated in Table 3.

<table>
<thead>
<tr>
<th>level</th>
<th>bird</th>
</tr>
</thead>
<tbody>
<tr>
<td>hdgesprw</td>
<td>Hedge Sparrow</td>
</tr>
<tr>
<td>mdwpipit</td>
<td>Meadow Pipit</td>
</tr>
<tr>
<td>piedwtail</td>
<td>Pied Wagtail</td>
</tr>
<tr>
<td>robin</td>
<td>Robin</td>
</tr>
<tr>
<td>treepipit</td>
<td>Tree Pipit</td>
</tr>
<tr>
<td>wren</td>
<td>Wren</td>
</tr>
</tbody>
</table>

Task: Cuckoos lay their eggs in the nests of other (host) birds. The eggs are then adopted and hatched by the host birds. If the hypotheses expressed in the quotation about egg laying behaviour in cuckoos are correct, then egg length ought to be associated with host bird. Your task is to evaluate that hypothesis with the data contained in the dataframe \texttt{cuckoo}.

1. Calculate the mean and standard deviation of \texttt{egglength} for each \texttt{hostbird}.

```r
( egg.m <- with(cuckoo,tapply(egglength,hostbird,mean)) )
```

<table>
<thead>
<tr>
<th>level</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>hdgesprw</td>
<td>23.1214</td>
</tr>
<tr>
<td>mdwpipit</td>
<td>21.4214</td>
</tr>
<tr>
<td>piedwtail</td>
<td>22.7643</td>
</tr>
<tr>
<td>robin</td>
<td>22.4357</td>
</tr>
<tr>
<td>treepipit</td>
<td>23.0214</td>
</tr>
<tr>
<td>wren</td>
<td>21.0500</td>
</tr>
</tbody>
</table>
2. Use the Analysis of Variance (ANOVA) to evaluate the hypothesis that egg length varies across host birds. Explain your results.

```r
koo.lm.01 <- lm(egglength~hostbird, data=cuckoo)
anova(koo.lm.01)

## Analysis of Variance Table
##  Df Sum Sq Mean Sq F value Pr(>F)
## hostbird 5 52.69 10.538 14.56 4.56e-10 ***
## Residuals 78 56.44 0.724
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Answer: The ANOVA tables produced by the `lm` and `aov` functions are the same. They both show the effect on goodness of fit – i.e., the sum of squared residuals – of adding the parameters associated `hostbird` ($\alpha_1...5$) to a model that includes only an intercept. Adding `hostbird` to the model reduces the sum-of-squared residuals by 52.69. Our $F$ test is significant ($F(5,78) = 14.56$, $p < .0001$), and therefore we reject the null hypothesis that average egg length does not vary among the host birds included in our analysis.

3. Equation 6 in the notes from Chapter 4 states that the $F$ statistic for a linear contrast is given by the equation

$$ F \leftarrow \frac{SS.\text{contrast}}{MS.w} $$

where

$$ SS.\text{contrast} \leftarrow \frac{(\psi^2)}{\sum\left(\frac{(c_j^2)}{n_j}\right)} $$

$\psi$ is the value of the linear comparison (i.e., a weighted combination of group means), $c_j$ represents the contrast weight for group $j$, $n_j$ is the number of observations in group $j$, and $MS.w$ is Mean Square
(within-cell), or Mean Square (residuals). Use these equations to conduct a linear contrast that evaluates the hypothesis that mean egg length in Wren hosts differs from the mean egg length in all other host birds.

**Answer:** The following answer will assume that the variance of the measurements is constant across groups, and therefore it is appropriate to use the equations listed in section 4.1 to perform the comparison. One complicating factor is that \( n \) varies across groups. Note below how \( n \) is represented by a vector of values rather than a single value.

\[
(\text{egg.n} <- \text{with(cuckoo,tapply(egglength,hostbird,length)))}) \quad \# n \text{ per group}
\]

\[
\begin{align*}
\text{##} & \quad \text{hdgesprw} \quad \text{mdwpipit} \quad \text{piedwtail} \quad \text{robin} \quad \text{treepipit} \quad \text{wren} \\
\text{##} & \quad 14 \quad 14 \quad 14 \quad 14 \quad 14 \quad 14
\end{align*}
\]

\[
(\text{egg.m} <- \text{with(cuckoo,tapply(egglength,hostbird,mean)))}) \quad \# \text{ group means}
\]

\[
\begin{align*}
\text{##} & \quad \text{hdgesprw} \quad \text{mdwpipit} \quad \text{piedwtail} \quad \text{robin} \quad \text{treepipit} \quad \text{wren} \\
\text{##} & \quad 23.1214 \quad 21.4214 \quad 22.7643 \quad 22.4357 \quad 23.0214 \quad 21.0500
\end{align*}
\]

\[
\text{levels(cuckoo$hostbird)}
\]

\[
\begin{align*}
\text{## [1]} & \quad \text{"hdgesprw"} \quad \text{"mdwpipit"} \quad \text{"piedwtail"} \quad \text{"robin"} \quad \text{"treepipit"} \\
\text{## [6]} & \quad \text{"wren"}
\end{align*}
\]

c1 <- c(-1,-1,-1,-1,-1,5) \quad \# \text{ weights}

MS.w <- 0.724 \quad \# \text{ from ANOVA table}

(\text{psi} <- \text{sum(c1 * egg.m)}) \quad \# \text{ value of PSI}

\[
\begin{align*}
\text{## [1]} & \quad -7.51429
\end{align*}
\]
Answer: Our comparison is significant ($F(1,78) = 36.39, p < 0.001$), and therefore we reject the null hypothesis in favor of the alternative, namely that mean egg length in the Wren group differs from the mean of the other groups.

Answer (continued): Now let’s compute the contrast using `lsmeans` and `linear.comparison`:

```r
# install.packages("lsmeans")
# library(lsmeans) # do this only once per session
# koo.aov.01 <- aov(egglength~hostbird,data=cuckoo) # did this earlier
ls.koo.01 <- lsmeans(koo.aov.01,"hostbird") # compute marginal means for hostbird groups
cweights <- list(wrenVSall = c(-1,-1,-1,-1,-1,5)) # store weights in list
(tmp.result <- contrast(ls.koo.01,cweights))
```

```r
## contrast estimate SE df t.ratio p.value
## wrenVSall -7.51 1.25 78 -6.035 <.0001
```

```r
tmp <- linear.comparison(cuckoo$egglength,cuckoo$hostbird,cweights, var.equal = TRUE)
```

```r
## [1] "computing linear comparisons assuming equal variances among groups"
## [1] "C 1: F=36.416, t=-6.035, p=0.000, psi=-7.514, CI=(-9.653,-5.376), adj.CI= (-9.993,-5.035)"
```

Answer (continued): Finally, let’s evaluate the contrast using `split` and `aov`:
Bennett, PJ

PSYCH 710

Lab #4

```r
c1 <- c(-1,-1,-1,-1,-1,5)
contrasts(cuckoo$hostbird) <- c1 # assign our contrast to hostbird
egg.aov <- aov(egglength~hostbird,data=cuckoo) # compute and store anova
# use split:
summary(egg.aov,split=list(hostbird=list(myContrast=1,others=2:5)) ) # list anova table

## Df Sum Sq Mean Sq F value Pr(>F)
## hostbird 5 52.7 10.54  14.6 4.6e-10 ***
## hostbird: myContrast 1 26.4 26.35  36.4 5.0e-08 ***
## hostbird: others 4 26.3  6.58  9.1 4.4e-06 ***
## Residuals 78 56.4  0.72
## ---
## Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

# value of PSI = sqrt( SS.contrast * sum( (weights^2) / egg.n ) )
( psi1 <- sqrt( 26.35 * (sum(c1^2/egg.n) ) ) ) # value of contrast

## [1] 7.51427
```

**Answer:** Notice that the $t$, $F$, and $p$ values for our contrast are consistent across methods because the design is balanced.

### 4.4 Sensorimotor learning task

An experiment was conducted to evaluate the effects of the duration of the inter-trial interval on learning/performance in a sensorimotor task. On each trial, a spot moved along a random, curvilinear path on a computer screen, and the subject’s task was to use a computer mouse to keep a cursor on the dot. To increase task difficulty, the relation between the mouse’s horizontal motion and the cursor’s horizontal motion was reversed: leftward mouse movement produced rightward cursor movement and vice versa. There were 10 trials and each trial lasted one minute. The dependent variable was the duration (in seconds) in which the cursor was located on the target spot. The independent variable was the duration of the inter-trial interval, which was 0, 20, 40, or 60 seconds. (N.B. In the 0 second condition, trial N+1 started immediately following trial N). The duration of the inter-trial duration was a between-subject variable: each subject was assigned randomly to one condition with the constraint that there were 5 subjects per group. The data are stored in the data frame `tracklearn`: the dependent variable is `y` and the independent variable is the factor `itd` (inter-trial duration). The group means are plotted in Figure 2.

Read the data file into R using the following command:

```
load(url("http://www.psychology.mcmaster.ca/bennett/psy710/datasets/tracklearn.Rdata"))
```

1. Calculate the mean and standard deviation of `y` for each `itd`.

```
( group.mean <- with(tracklearn,tapply(y,itd,mean)) )

## d0  d20  d40  d60
##  8.0 14.0 19.2 18.0

( group.sd <- with(tracklearn,tapply(y,itd,sd)) )

## d0  d20  d40  d60
## 2.91548 2.64675 3.42053 4.00000
```

2. The following commands confirm that `itd` is an *ordered* factor and list its levels and contrasts:

```r
class(tracklearn$itd)

## [1] "ordered" "factor"

levels(tracklearn$itd)

## [1] "d0"  "d20"  "d40"  "d60"

unique(tracklearn$itd)

## [1] d0  d20  d40  d60

# each column contains weights for one contrast
# .L == linear; .Q == quadratic; .C == cubic

contrasts(tracklearn$itd)

## .L .Q .C
## [1,] -0.670820 0.5 -0.223607
## [2,] -0.223607 -0.5 0.670820
## [3,] 0.223607 -0.5 -0.670820
## [4,] 0.670820 0.5 0.223607

(a) Verify that the contrasts for *itd* are orthogonal.

```r
# verify that the contrasts are orthogonal:
lin.weights <- contrasts(tracklearn$itd)[,1]
quad.weights <- contrasts(tracklearn$itd)[,2]
cubic.weights <- contrasts(tracklearn$itd)[,3]
# following assumes equal n per group:
sum(lin.weights*quad.weights) # should be zero
## [1] 0
sum(lin.weights*cubic.weights) # should be zero
## [1] 2.77556e-17
sum(quad.weights*cubic.weights) # should be zero
## [1] 9.71445e-17
```

(b) Imagine we added a fifth inter-trial interval of 100 seconds to our experiment. Use `contr.poly` to calculate the weights for the linear trend for our new experiment.

```r
contr.poly(n=5,scores=c(0,20,40,60,100) ) # note unequal spacing of our inter-trial intervals
```

```r
## .L .Q .C .^4
## [1,] -0.5718628 0.557647 -0.374753 0.1468446
## [2,] -0.3119251 -0.115991 0.621301 -0.5506673
## [3,] -0.0519875 -0.459501 0.216962 0.7342231
## [4,] 0.2079501 -0.472885 -0.631162 -0.3671115
## [5,] 0.7278253 0.490729 0.167653 0.0367112
```

```r
(lin.trend.weights <- contr.poly(n=5,scores=c(0,20,40,60,100) )[,1] ) # take values in 1st column
## [1] -0.5718628 -0.3119251 -0.0519875 0.2079501 0.7278253
```
3. Use `aov` to perform an ANOVA to evaluate the null hypothesis that performance is the same in all groups.

```r
# use aov command:
track.aov.01 <- aov(y~itd,data=tracklearn)
anova(track.aov.01)
```

---

4. Take the result of your `aov` command and then split the between-groups sum-of-squares to evaluate the linear, quadratic, and cubic trends. What null hypotheses are being evaluated by the $F$ tests?

```r
# summary of aov object yields F tests:
summary(track.aov.01,split=list(itd=list(linear=1,quad=2,cubic=3)))
```

---

5. Use `lsmeans` to evaluate the trends.

```r
wL <- contr.poly(n=4)[,1]  # linear trend weights
wQ <- contr.poly(n=4)[,2]  # quadratic trend weights
wC <- contr.poly(n=4)[,3]  # cubic trend weights
trend.weights <- list(Lin=wL,Quad=wQ,Cubic=wC)  # store 3 sets in a single list
ls.track <- lsmeans(track.aov.01,"itd")
contrast(ls.track,trend.weights,adjust="none")  # no correction for multiple tests
```
## contrast estimate  SE  df  t.ratio  p.value
## Lin      7.87 1.47 16   5.356 0.0001
## Quad     -3.60 1.47 16  -2.449 0.0262
## Cubic    -1.25 1.47 16  -0.852 0.4068

# second method:
(tmp.results <- contrast(ls.track, method="poly") ) # t and p values are the same but psi differs

## contrast  estimate   SE  df  t.ratio  p.value
## linear     35.2 6.57 16   5.356 0.0001
## quadratic  -7.2 2.94 16  -2.449 0.0262
## cubic      -5.6 6.57 16  -0.852 0.4068

tmp.results

## itd  c.1  c.2  c.3
## d0      d0  -3   1  -1
## d20     d20  -1  -1   3
## d40     d40   1  -1  -3
## d60     d60   3   1   1

Answer: The statistical tests are the same as before. However, notices that the value of the quadratic trend is negative here but was positive with `split` and `aov`. The reason is that the values of the contrasts computed with `aov` are based on sums-of-squares, which means that the signs of the contrasts are lost. Finally, note that `contrast` can do polynomial contrasts (i.e., trends) that yield the same t and p values as the ones computed with our contrast weights, but the value of the comparisons differ. The reason for the difference is that the command uses contrast/trend weights (shown by the `coef(tmp.results)` command above) that consist of integers that differ from our weights.

6. Use `lm` to perform an ANOVA to evaluate the null hypothesis that performance is the same in all groups.

```
track.lm.01 <- lm(y~itd, data=tracklearn)
anova(track.lm.01) # the anova is the same as the one obtained with aov
```

```
## Analysis of Variance Table
##   ## Response: y
##   ## Df  Sum Sq  Mean Sq  F value  Pr(>F)
## itd   3  382.4 127.5 11.8   0.00025 ***
## Residuals 16 172.8 10.8
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

7. Take the result of your `lm` command and evaluate the linear, quadratic, and cubic trends using t tests. What null hypotheses are being evaluated by the t tests?

```
summary(track.lm.01)
```

```
## Call:
## lm(formula = y ~ itd, data = tracklearn)
## Residuals:
##     Min   1Q Median   3Q   Max
##  -5.00 -2.25  -0.60  2.25  5.00
##```
Answer: The hypotheses are the same as those being evaluated in question 4. Each t test is a two-tailed test that evaluates the null hypothesis that the trend (linear, quadratic, or cubic) is zero.

8. How are the F and t values in 4 and 7 related?

Answer: \( F = t^2 \)

9. Why is the t value for the quadratic trend in 7 less than zero?

Answer: If you plot the weights for the quadratic trend you will see that they are a U-function of itd (see Figure 3). The reason \( t \) is negative is because the quadratic trend of \( y \) across itd is an inverted-u function of itd (Figure 2), so the trend of the means is negatively correlated with the weights, and hence \( t < 0 \).

```r
# first plot group means:
xvals <- c(0, 20, 40, 60)
group.means <- with(tracklearn, tapply(y, itd, mean))
dev.new(height=4, width=4)
plot(xvals, group.means, "b", xlab="ITD", ylab="Tracking (s)", main="Group Means")

# then plot weights for each trend:
dev.new(height=3, width=9)
par(mfrow=c(1,3))
plot(xvals, lin.weights, "b", xlab="ITD", ylab="weight", main="Linear Trend")
plot(xvals, quad.weights, "b", xlab="ITD", ylab="weight", main="Quadratic Trend")
plot(xvals, cubic.weights, "b", xlab="ITD", ylab="weight", main="Cubic Trend")
```
Figure 2: Group means.
Figure 3: Weights for linear, quadratic, and cubic trends.