Notes on Maxwell & Delaney

PSYCH 710

November 14, 2023

9 Designs with Covariates

9.1 Blocking

Consider the following hypothetical experiment. We want to measure the effect of a drug on locomotor activity in hyperactive children. Children are assigned randomly to three groups that differ in the dosage of the drug (i.e., zero, low, and high), and the dependent measure is the amount of locomotor activity during some fixed period of time after administration of the drug. Now, children in our sample are likely to vary in the amount of locomotor activity that they produce *before* the administration of the drug: some children are more active than others. Should we worry about this inter-subject variability? Probably. These individual differences are likely to increase the variability of our dependent measure, and therefore make it more difficult to detect a significant effect of the drug. Is there some way of reducing the impact of these individual differences on the sensitivity of our experiment?

One way of addressing this issue would be to screen subjects and test only those children whose activity fell within a limited range. However, using this strategy would mean that our conclusions would be restricted to children exhibiting that particular level of activity used in your experiment. Another strategy would be to divide pre-drug activity into several levels (e.g.,"low", "medium", "high", and "very high"), select N subjects at each level, and then randomly assign subjects to a drug condition. This scheme of creating homogeneous sets of subjects is referred to as **blocking**; in this case our blocking variable would be pre-drug, or baseline, activity. Because subjects within each block are assigned randomly to treatment groups, this type of design is referred to as a **randomized blocks** design[1]. When analyzing the data, baseline activity is treated as a factor. Incorporating the blocking variable into our statistical model means that variability in the dependent variable that is related to baseline activity will be assigned to the the blocking factor instead of error. Hence, error will be reduced.

---

[1]Typically, randomized block designs have only a subjects per block, where a is the number of levels on the treatment variable. In other words, only one subject in each block would be assigned to each level of the treatment variable. The current design, which has 12 subjects per block assigned randomly to 3 treatments (i.e., 4 subjects per treatment) is an example of a generalized randomized block design (Kirk, 1995).

The following example illustrates a randomized blocks experiment. First, we create fake data. The variable `activity` represents basline locomotor activity in 48 subjects. The variable `block` is a factor that divides `activity` into four groups that differ in mean activity: in other words, `block` is our blocking variable. Next, we create the effects (i.e., the alpha's) associated with each level of the drug and store them in `drug.effect`. The variable `drug` is a factor that indicates the level of drug received by the subjects. Finally, we create a fake dependent variable `y`. Notice that `y` is related linearly to `activity`, and is affected by the dosage of the drug.

```r
# sum-to-zero definition of factor effects
options(contrasts=c("contr.sum","contr.poly"))
set.seed(437);
activity <- sort(rnorm(n=48,mean=10,sd=3));
block <- factor(rep(c("low","med","high","very"),each=12),
                levels=c("low","med","high","very"),
                ordered=FALSE);
drug <- factor(rep(c("zero","low","high","high","low","zero"),times=8),
                levels=c("zero","low","high"),
                ordered=FALSE)
drug.effect<-rep(c(-1,0,1,1,0,-1),times=8)
y <- 1 + 0.6*activity + 1.1*drug.effect + rnorm(activity,mean=0,sd=3);
theData <- data.frame(y, block,drug,activity,drug.effect)
```

First, we want to see if our blocking variable really is capturing variability among our subjects. The next several commands list the mean baseline activity within each block of subjects and then determines if the blocks differ significantly. The blocks *do* differ, so we succeeded in creating groups in which the between-block differences are large relative to the within-block differences.

```r
with(theData, tapply(activity, block,mean));


##    low     med    high    very
##   5.154   8.008 10.287 13.819


summary(aov(activity ~ block, data=theData) );


##              Df Sum Sq Mean Sq F value Pr(>F)
## block         3    483   161.0     106 <2e-16 ***
## Residuals    44     67     1.5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now we evaluate two nested linear models: The first model does not include our blocking factor, but the second one does.

```
aov.1<-aov(y~drug,data=theData)
aov.2<-aov(y~block+drug+block:drug,data=theData)
summary(aov.1)


##              Df Sum Sq Mean Sq F value Pr(>F)
## drug          2     37   18.54    2.17   0.13
## Residuals    45    384    8.53


summary(aov.2)


##              Df Sum Sq Mean Sq F value  Pr(>F)
## block         3  129.8    43.3    6.96 0.00082 ***
## drug          2   37.1    18.5    2.98 0.06329 .
## block:drug    6   30.4     5.1    0.81 0.56580
## Residuals    36  223.7     6.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice that SS_{total} is the same for both analyses, however $SS_{residuals}$ is much smaller in the analysis that includes `block`. Why? Because $SS_{residuals}$ in `aov.1` has been partitioned into two pieces: one due to error and another due to variation among blocks. You can verify this statement by summing the sums-of-squares for residuals, block, and block:drug in `aov.2`: the total should equal $SS_{residual}$ from `aov.1`. The reduction in $SS_{residuals}$ from 383.96 to 223.748 means that F_{drug} increases from 2.1727 to 2.9828, although it still is not significant ($df = (2, 36)$, $p = 0.063$). Hence, even after accounting for the effects of `block` (i.e., baseline activity), we fail to reject the null hypothesis of no effect of `drug`. The effect of `block` is significant, but that is not surprising.

9.2   ancova

I created the fake dependent variable, `y`, so that it was linearly related to baseline activity. Our blocking variable was a qualitative factor, however, and so did not take full advantage of the structure in the data. An alternative approach would be to include the numeric values themselves, rather than the level of a blocking factor, into our model. This is the approach used in the **analysis of covariance**. In the analysis of covariance, or ANCOVA, SS_{drug} is estimated by comparing the residuals from the following models:

$$Y \sim X + A$$
$$Y \sim X$$

3

where X and A are the covariate and grouping variables, respectively. In other words, SS_A represents the variation in Y that is due to A after controlling for the *linear* association between Y and X. The following commands perform an analysis of covariance on the current data:

```
lm.1<-lm(y~activity+drug,data=theData)
lm.2<-lm(y~activity,data=theData)
anova(lm.2,lm.1)


## Analysis of Variance Table
##
## Model 1: y ~ activity
## Model 2: y ~ activity + drug
##   Res.Df RSS Df Sum of Sq    F Pr(>F)
## 1     46 279
## 2     44 240  2      39.3 3.61  0.035 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

SS_{drug} equals 39.286, which is significant ($F = 3.6066$, $p = 0.035$). The same results can be obtained by using `drop1`, which also calculates $SS_{activity}$ after controlling for the effects of `drug`.

```
drop1(lm.1,.~.,test="F")


## Single term deletions
##
## Model:
## y ~ activity + drug
##          Df Sum of Sq RSS   AIC F value  Pr(>F)
## <none>                240  85.2
## activity  1     144.3 384 105.8   26.50 5.9e-06 ***
## drug      2      39.3 279  88.5    3.61   0.035 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Finally, we could simply construct an ANOVA table from the full model:

```
anova(lm.1)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value  Pr(>F)
## activity   1  142.1   142.1   26.09 6.7e-06 ***
## drug       2   39.3    19.6    3.61   0.035 *
## Residuals 44  239.6     5.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

R prints sequential sums of squares, so SS_{drug} is calculated after controlling for the linear association between `activity` and the dependent variable.

So, the analysis of covariance indicates that the effect of drug is significant after controlling for the linear association between the covariate and independent variable. Note that the ANCOVA is more sensitive than the blocking strategy used in the previous section. This increased sensitivity reflects the fact that the relation between the covariate and dependent variable is *linear*. If the relation was nonlinear, then the blocking design may have been more sensitive. (N.B. It is possible, however, to alter the ANCOVA model to include a nonlinear term). Note also that the covariate in the ANCOVA model uses only a single degree of freedom, whereas the blocking factor used two degrees of freedom.

9.2.1 graphical representation

Figure 1 is a graphical illustration of an ancova model fit to data from two treatment groups. The two dashed lines are the best-fitting (least-squares) lines for the two sets of data, with the constraint that the lines have the same slopes. Because we defined our effects (i.e., the α's) using the sum-to-zero constraint, the intercepts for the two lines — i.e., the value of Y when the covariate, X, is zero — are $\mu + \alpha_1$ and $\mu + \alpha_2$. In this example there are only two groups, therefore $\alpha_2 = -\alpha_1$ and the two intercepts are $\mu + \alpha_1$ and $\mu - \alpha_1$.

9.2.2 adjusted means

The **adjusted means** for the two groups correspond to the height at which the regression lines intersect the vertical dotted line located at the overall mean of the covariate, $\bar{X}$ (see Figure 1). In other words, the adjusted means represent the means that we would expect if all of the groups were equivalent on the covariate. The adjusted means can be calculated for each group from the coefficients of the ancova model:

$$\bar{Y}_j' = \mu + \beta\bar{X} + \alpha_j$$

where β is the coefficient for the covariate (i.e., the slope of the lines in Figure 1), μ is the model's intercept, and α_j is the effect for group j. The coefficients for the model fit to the current data are
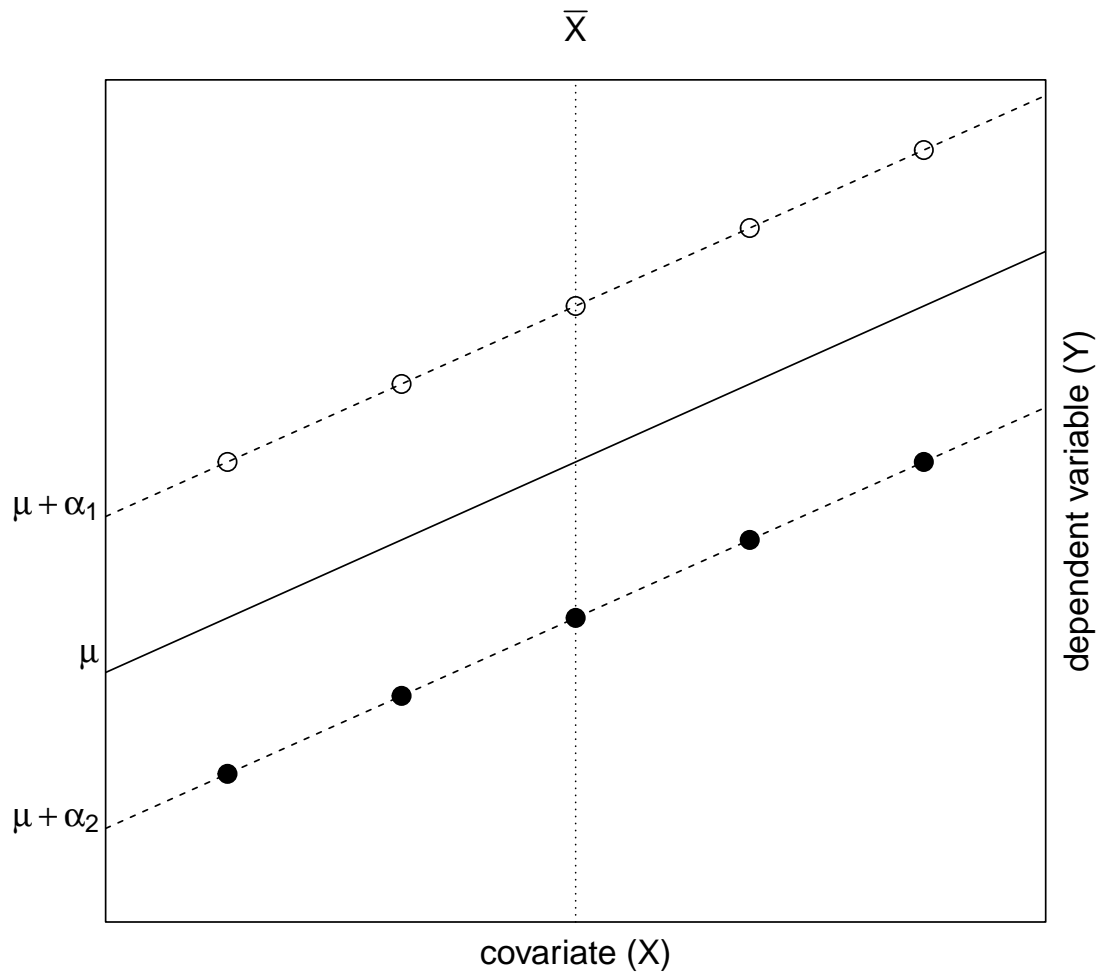
Figure 1: Graphical illustration of ancova model.

```
dummy.coef(lm.1);


## Full coefficients are
##
## (Intercept):        1.285
## activity:          0.5125
## drug:                 zero      low     high
##                    -0.8216  -0.4387   1.2603
```

The mean of the covariate is

```
mean(theData$activity)


## [1] 9.317
```

so the adjusted means for the three treatment groups are

```
1.28519 + 0.5124*9.3168-0.8216


## [1] 5.238


1.28519 + 0.5124*9.3168-0.4386


## [1] 5.621


1.28519 + 0.5124*9.3168+1.2603


## [1] 7.319
```

There are some advantages to using a centered covariate (i.e., a covariate that is centered on zero). A centered variable is obtained simply by subtracting the mean value from the original variable. The next few commands create a centered covariate `activity.c` and then refit the ancova model using the new covariate.

```
theData$activity.c <- theData$activity - mean(theData$activity)
lm.3 <- lm(y~activity.c + drug,data=theData)
drop1(lm.3,.~.,test="F")
```

```
## Single term deletions
##
## Model:
## y ~ activity.c + drug
##            Df Sum of Sq RSS   AIC F value  Pr(>F)
## <none>                  240   85.2
## activity.c  1     144.3 384 105.8   26.50 5.9e-06 ***
## drug        2      39.3 279  88.5    3.61   0.035 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


dummy.coef(lm.3)


## Full coefficients are
##
## (Intercept):      6.06
## activity.c:     0.5125
## drug:               zero     low    high
##                  -0.8216 -0.4387  1.2603
```

The main results are the same; however, the intercept now corresponds to the point at which the average regression line (i.e., the solid line in Figure 1) intersects the vertical dotted line (i.e., $X = \bar{X}$). Therefore, the adjusted means can be computed by simply adding the intercept and alpha's:

```
6.0596-0.8216


## [1] 5.238


6.0596-0.4386


## [1] 5.621


6.0596+1.2603


## [1] 7.32
```

The values are the same (to within rounding error) of those obtained previously.

The **effects** package can be used to calculate the adjusted means. If you have not yet installed the package on your computer, do so now using the following command:

```
install.packages("effects")
```

Next, you must load the package into memory with the command `library(effects)` command. Note that you only need to use this command once per R session. The adjusted means are calculated thusly:

```
library(effects)
effect(term="drug",lm.1)


##
##   drug effect
## drug
##   zero    low   high
## 5.238 5.621 7.320


effect(term="drug",lm.3)


##
##   drug effect
## drug
##   zero    low   high
## 5.238 5.621 7.320
```

9.2.3   homogeneity of slopes assumption

The ancova model fits regression lines to each group with the constraint that the slopes of the lines are equal. This constraint is equivalent to assuming that there is no interaction between the covariate and the treatment variable. If the assumption is valid, then the differences between groups — the vertical distances between the regression lines — are constant for all values of the covariate. However, if there is an interaction, then the differences between groups will depend on the value of the covariate. In this situation, it may not make sense to talk about the main effect of the treatment variable. Therefore, it is important to test the validity of the homogeneity of slopes assumption by comparing the goodness of fit obtained by models that do and do not have an interaction term:

```
lm.4 <- lm(y~activity.c + drug + activity.c:drug,data=theData)
anova(lm.3,lm.4)


## Analysis of Variance Table
##
```

```
## Model 1: y ~ activity.c + drug
## Model 2: y ~ activity.c + drug + activity.c:drug
##   Res.Df RSS Df Sum of Sq    F Pr(>F)
## 1     44 240
## 2     42 225  2      14.3 1.33   0.27
```

Alternatively, we can simply list the anova table for the full model:

```
anova(lm.4)
```

```
## Analysis of Variance Table
##
## Response: y
##               Df Sum Sq Mean Sq F value  Pr(>F)
## activity.c     1  142.1   142.1   26.49 6.6e-06 ***
## drug           2   39.3    19.6    3.66   0.034 *
## activity.c:drug 2   14.3     7.1    1.33   0.275
## Residuals     42  225.3     5.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$SS_{interaction} = 14.298$, which is not significant ($F = 1.332$, $df = (2, 42)$, $p = 0.274$). Therefore, the assumption appears to be valid for these data.

9.3   strength of association

The strength of association between the treatment variable and the dependent variable — omega squared — can be estimated with the following equation:

$$\omega^2 = \frac{df_{effect}(MS_{effect} - MS_{residuals})}{SS_{total} + MS_{residuals}}$$

I'll illustrate this calculation using the data presented in Table 9.7 in Maxwell and Delaney (2004).

```
# mw97<-read.table("chapter_9_table_7.dat")
fn <- "http://pnb.mcmaster.ca/bennett/psy710/datasets/mw/chapter_9_table_7.dat"
mw97 <- read.table(file=url(fn))
names(mw97)<-c("group","pre","post")
mw97$treatment<-factor(mw97$group,labels=c("ssri","placebo","waitList"),ordered=FALSE)
summary(mw97)
```

```
##      group        pre            post          treatment
##  Min.  :1   Min.  : 9.0   Min.  : 0.0   ssri    :10
```

```
##   1st Qu.:1    1st Qu.:14.0   1st Qu.: 7.5    placebo :10
##   Median :2    Median :16.0   Median :10.0    waitList:10
##   Mean   :2    Mean   :17.4   Mean   :11.2
##   3rd Qu.:3    3rd Qu.:20.8   3rd Qu.:15.0
##   Max.   :3    Max.   :29.0   Max.   :25.0
```

```
mw97[1:4,]
```

```
##   group pre post treatment
## 1     1  18   12       ssri
## 2     1  16    0       ssri
## 3     1  16   10       ssri
## 4     1  15    9       ssri
```

```
mw97.lm.1<-lm(post~pre+treatment,data=mw97)
drop1(mw97.lm.1,.~.,test="F")
```

```
## Single term deletions
##
## Model:
## post ~ pre + treatment
##          Df Sum of Sq  RSS AIC F value Pr(>F)
## <none>                 756 105
## pre       1       313 1070 113   10.77 0.0029 **
## treatment 2       217  973 108    3.73 0.0376 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(mw97.lm.1)
```

```
## Analysis of Variance Table
##
## Response: post
##           Df Sum Sq Mean Sq F value Pr(>F)
## pre        1    337     337   11.57 0.0022 **
## treatment  2    217     109    3.73 0.0376 *
## Residuals 26    756      29
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
df.effect <- 2;
SS.total <- 336.68+217.15+756.33;
(omega.2 <- df.effect*(108.57-29.09) / (SS.total+29.09) )
```

```
## [1] 0.1187
```

The value of ω^2 is 0.118, so the treatment accounts for approximately 12% of the total variance in post-test scores in the population. Note that this definition of association strength treats the covariate as an intrinsic factor. In other words, the value of ω^2 is the proportion of variance in the dependent measure that is explained by the treatment while *ignoring* the effects of the covariate. An alternative measure of association strength, partial omega squared, is the the proportion of variance that is accounted for after *controlling* for the effects of the covariate and is defined by the equation

$$\omega^2_{partial(group)} = \frac{df_{group}(F_{group} - 1)}{df_{group}(F_{group} - 1) + N}$$

where N is the number of subjects in the experiment (Keppel and Wickens, 2004). For the data in Table 9.7, partial omega squared is 0.154.

9.4   comparisons among group means

Tests of specific contrasts are done on *adjusted* means. The value of a linear contrast is obtained in the usual way:

$$\hat{\psi} = \sum_{j=1}^{a} c_j \bar{Y}'_j$$

However, calculation of the variance of the contrast differs from before:

$$s^2_{\hat{\psi}} = MS_{residuals} \left[\sum_{j=1}^{a} \frac{c_j^2}{n_j} + \frac{\left(\sum_{j=1}^{a} c_j \bar{X}_j\right)^2}{\sum_{j=1}^{a} \sum_{i=1}^{n} (X_{ij} - \bar{X}_j)^2} \right]$$

where $MS_{residuals}$ is taken from the full model (e.g., `lm.1`). The test statistic is

$$F = \frac{\hat{\psi}^2}{s^2_{\hat{\psi}}}$$

which is distributed as an F variable with 1 and $N - a - 1$ degrees of freedom. If the comparisons are post-hoc, you should use the Scheffe method to control Type I error rate. The critical value of F for a study that had a levels on the treatment variable would be $(a-1)F_\alpha(df_1 = (a-1), df_2 = N - a - 1)$, where α is the desired Type I error rate.

These calculations are tedious, so I have written a function for R that do them for you.

```
psi.adj<-function(y.adj,x,group,c.weights,ms.error){
    groups <- levels(group);
    ng <- length(groups); # number of group levels
```

```
    x.means <- tapply(x,group,mean); # group means on X
    n<-tapply(x,group,length); # group n
    psi <- sum(c.weights*y.adj); # value of comparison
    # calculate variance of comparison
    term.1 <- sum( c.weights*c.weights/n);
    term.2 <- (sum(c.weights*x.means))^2;
    x.bar.j <-tmp<-rep(x=x.means,times=n);
    ss.x <-  sum( (x-x.bar.j)^2 );
    psi.var <- ms.error * (term.1 + (term.2 / ss.x) );
    F.psi <- (psi*psi)/psi.var; # F value
    p <- 1-pf(F.psi,df1=1,df2=(length(x)-ng-1)); # p value
    the.results <- list(psi=psi,psi.var=psi.var,F=F.psi,p.value=p)
    the.results # show results
    }
```

The `psi.adj` function can be loaded into R with the following command:

```
source("http://pnb.mcmaster.ca/bennett/psy710/Rscripts/psi_adj.R")
```

Alternatively, you can type the `psi.adj` commands into a word processor, save the com-
mands as an ascii text file placed in your working directory, and then read it into R using
the command `source("filename.txt")` command. The following commands show how
to use it evaluate all pairwise comparisons between groups in the activity example:

```
anova(lm.1)


## Analysis of Variance Table
##
## Response: y
##            Df Sum Sq Mean Sq F value  Pr(>F)
## activity    1  142.1   142.1   26.09 6.7e-06 ***
## drug        2   39.3    19.6    3.61   0.035 *
## Residuals  44  239.6     5.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


MS.resid <- 5.446
df.resid <- 44
effect(term="drug",mod=lm.1)


##
##  drug effect
## drug
##  zero   low  high
## 5.238 5.621 7.320
```

```
y.adj<-c(5.234,5.62,7.32)
X <- theData$activity
G <- theData$drug
psi.adj(y.adj,x=X,group=G,c.weights=c(-1,1,0),MS.resid)


##  psi
## [1] 0.386
##
##  psi.var
## [1] 0.6808
##
##  F
## [1] 0.2189
##
##  p.value
## [1] 0.6422


psi.adj(y.adj,x=X,group=G,c.weights=c(-1,0,1),MS.resid)


##  psi
## [1] 2.086
##
##  psi.var
## [1] 0.6809
##
##  F
## [1] 6.391
##
##  p.value
## [1] 0.01514


psi.adj(y.adj,x=X,group=G,c.weights=c(0,-1,1),MS.resid)


##  psi
## [1] 1.7
##
##  psi.var
## [1] 0.6808
##
##  F
## [1] 4.245
##
##  p.value
## [1] 0.04532
```

The p-values are not adjusted for multiple comparisons. To use Tukey's HSD, you need to compare the observed F's to Tukey's critical value of F that is derived from the formula in Table 5.16 of Maxwell and Delaney (2004, page 262):

```
alpha.fw <- .05
F.tukey.critical <- ( qtukey(1-alpha.fw, nmeans=3, df=df.resid)^2 ) / 2
F.tukey.critical
```

```
## [1] 5.883
```

Only the second comparison, between the `zero` and `high` groups, is significant ($p < .05$).

9.4.1   comparisons using `emmeans`

Comparisons among adjusted means can be done using the R package `emmeans`. The following code chunk uses `emmeans` to compute the three contrasts described in the previous section. Note that the linear model object must be transformed into an `emmGrid` object before performing the contrasts.

```
library(emmeans)
( emm.obj <- emmeans(lm.1,specs=~activity+drug) ) # estimated marginal means

##  activity drug emmean    SE df lower.CL upper.CL
##      9.32 zero   5.24 0.583 44     4.06     6.41
##      9.32 low    5.62 0.583 44     4.45     6.80
##      9.32 high   7.32 0.584 44     6.14     8.50
##
## Confidence level used: 0.95


# contrast weights:
c1 <- c(-1,1,0)
c2 <- c(-1,0,1)
c3 <- c(0,-1,1)
con.res <- contrast(emm.obj,method=list(c1=c1,c2=c2,c3=c3))
summary(con.res,adjust="none") # no p-value adjustment

##  contrast estimate    SE df t.ratio p.value
##  c1          0.383 0.825 44   0.464  0.6449
##  c2          2.082 0.825 44   2.523  0.0153
##  c3          1.699 0.825 44   2.059  0.0454
```

Note that the p values are essentially identical to the ones provided by `psi.adj`: `emmeans`
performs the contrasts on the *adjusted* group means. It is a simple matter to adjust the p
values using the Tukey HSD method:

```
summary(con.res,adjust="tukey") # tukey  adjustment


##  contrast estimate    SE df t.ratio p.value
##  c1           0.383 0.825 44   0.464  0.9552
##  c2           2.082 0.825 44   2.523  0.0453
##  c3           1.699 0.825 44   2.059  0.1302
##
## P value adjustment: sidak method for 3 tests
```

As we found in the previous section, only the second pairwise comparison is significant.

9.4.2   Tukey tests with `TukeyHSD` & `multcomp`

Tukey tests can be performed with `TukeyHSD` and the `glht` command in the `multcomp`
package. However, these procedures perform the comparisons on the marginal means and
ignore the covariate. In other words, the comparisons are performed on the *unadjusted*
group means.

```
aov.1 <- aov(y~activity+drug,data=theData)
TukeyHSD(aov.1,which="drug") # pairwise tests on unadjusted group means


##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = y ~ activity + drug, data = theData)
##
## $drug
##             diff      lwr   upr  p adj
## low-zero  0.3829 -1.61840 2.384 0.8884
## high-zero 2.0815  0.08021 4.083 0.0399
## high-low  1.6986 -0.30265 3.700 0.1104


# install.packages("multcomp")
library(multcomp)
drug.tukey <- glht(lm.1,linfct=mcp(drug="Tukey") ) # pairwise Tukey tests
summary(drug.tukey) # comparisons on unadjusted means
```

```
##
##        Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lm(formula = y ~ activity + drug, data = theData)
##
## Linear Hypotheses:
##                  Estimate Std. Error t value Pr(>|t|)
## low - zero == 0     0.383      0.825    0.46     0.89
## high - zero == 0    2.082      0.825    2.52     0.04 *
## high - low == 0     1.699      0.825    2.06     0.11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```

9.5   Generalization of the ANCOVA model

9.5.1   nonlinear relations

The basic ANCOVA model assumes that the relation between the dependent variable and covariate is linear. However, this is not a necessary restriction, and the model can be generalized to include nonlinear relations. For example, comparing the following models would estimate SS_{group} after controlling for the linear and quadratic effects of the covariate, X:

$$\begin{aligned} Y_{ij} &= \mu + \alpha_j + \beta_1 X_{ij} + \beta_2 X_{ij}^2 + \epsilon_{ij} \\ Y_{ij} &= \mu + \beta_1 X_{ij} + \beta_2 X_{ij}^2 + \epsilon_{ij} \end{aligned}$$

Adjusted means are calculated by computing Y when $X = \bar{X}$. The model assumes that both the linear and quadratic components of X do not interact with the treatment variable.

9.5.2   multiple factors

It is also possible to include more than one factor.

$$Y_{ij} = \mu + \alpha_j + \gamma_k + \beta X_{ij} + (\alpha\gamma)_{jk} + \epsilon_{ij}$$

Once again, the model assumes that the main effects for each factor, and the interaction, do not interact with the covariate.

When analyzing designs that include two or more factors, we often will want to do comparisons among adjusted row and column marginal means, as well as adjusted cell means. Again, computational details can be found in Kirk (1995).

9.6   An alternative to ANCOVA

You might have wondered why an ANCOVA was necessary for the data presented here. Why not just conduct an ANOVA on the difference scores computed subtracting pre- and post-treatment activity levels? I conduct that analysis here:

```
theData$diff <- theData$y - theData$activity;
diff.lm.1 <- lm(diff~drug,data=theData)
anova(diff.lm.1);


## Analysis of Variance Table
##
## Response: diff
##           Df Sum Sq Mean Sq F value Pr(>F)
## drug       2     41   20.73    2.52   0.092 .
## Residuals 45    370    8.23
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The effect of group is not significant. Obviously, this analysis was less sensitive than the ANCOVA. Why? The reason is that the difference score approach is based on a more restrictive model than the one used in ANCOVA. The model is

$$Y_{ij} - X_{ij} = \mu + \alpha_j + \epsilon_{ij}$$

which can be re-written as

$$Y_{ij} = \mu + \alpha_j + X_{ij} + \epsilon_{ij}$$

This model is similar to the one used in ANCOVA, except that the slope of the linear relation between Y and X is assumed to be one. When the linear relation *does* have a slope of one (i.e., $\beta = 1$), then an ANOVA on difference scroes will yield the same results as an ANCOVA. When $\beta \neq 1$, ANCOVA will have greater sensitivity.

References

Keppel, G. and Wickens, T. (2004). *Design and analysis: A researcher's handbook*. Pearson Education, Inc., 4th edition.

Kirk, R. E. (1995). *Experimental design: Procedures for the behavioral sciences*. Brooks/-Cole, 3rd edition.

Maxwell, S. E. and Delaney, H. D. (2004). *Designing experiments and analyzing data: a model comparison perspective*. Lawrence Erlbaum Associates, Mahwah, N.J., 2nd ed edition.