HUDM 5123 - Linear Models and Experimental Design
Lab 05 - Multiple Comparisons and Two-Way ANOVA (Part I)

1   The Data

The data for today's lab were the basis for a 2019 article in *PLOS ONE* entitled "The proximal experience of awe"; the data may be downloaded from a link to the Open Science Framework. Researchers recruited participants for two studies in which the researchers randomly assigned participants to either be exposed to an experimental manipulation meant to increase participants' sense of awe, or to a comparison condition. We will focus on the data generated for the second study. From the paper:

> Participants were randomly assigned to watch one of two short videos. Similar to Study 1, the first video depicted Earth zooming away (as if aboard a spaceship) while a narrator read selections from Carl Sagan's Pale Blue Dot (Pale Blue Dot condition n = 83). The second video depicted images of Earth from space and provided information about the Earth's rotation and orbit around the sun (Earth Facts condition n = 89). After watching their assigned video, participants completed measures of self-transcendent emotions, other-connectedness, self-relevant thoughts, small self, humility, and other positive and negative emotions.

Self-transcendent emotions were measured using a 15-item scale with Likert scoring on a 1 (low) to 7 (high) scale and three items for each of five emotions: awe, gratitude, love, compassion, and optimism. Average scores across the three items were taken to get a single score for each participant for each of the five self-transcendent emotions.

1.1   Cleaning Up

Begin by downloading the data (the one called STUDY 2) at the link given above. Then load the file with a call to `read.csv()` and call it `dat`. The authors used an online platform for data collection called Prolific Academic that is similar to Amazon's Mechanical Turk. In their online surveys they included an attention check that said, "Please select the option 'Somewhat disagree' below". Twenty-nine of their participants failed this check and, therefore, we will eliminate them from the analytic sample.

```
table(dat$distract_failed, useNA = "always")

   0 <NA>
 172    0
# Delete cases for participants that failed attention check
dat <- dat[which(dat$distract_failed == 0), ]
dim(dat)
[1] 172 182
```

Next, we need to create the average scores based on three items each for the five self-transcendent emotions.

```
##################
# AWE
grep(pattern = "awe",
     x = names(dat),
     ignore.case = TRUE,
     value = TRUE)
dat$awe_avg <- (dat$selfTrnsdEmo_awe_1+
                    dat$selfTrnsdEmo_awe_6 +
                    dat$selfTrnsdEmo_awe_11) / 3
# Compare means by condition
by(data = dat$awe_avg, INDICES = dat$Condition, FUN = mean)
by(data = dat$awe_avg, INDICES = dat$Condition, FUN = sd)


##################
# GRATITUDE
grep(pattern = "grat",
     x = names(dat),
     ignore.case = TRUE,
     value = TRUE)
dat$grat_avg <- (dat$selfTrnsdEmo_gratitude_2 +
                   dat$selfTrnsdEmo_gratitude_7 +
                   dat$selfTrnsdEmo_gratitude_12) / 3
# Compare means by condition
by(data = dat$grat_avg, INDICES = dat$Condition, FUN = mean)
by(data = dat$grat_avg, INDICES = dat$Condition, FUN = sd)


##################
# LOVE
grep(pattern = "love",
     x = names(dat),
     ignore.case = TRUE,
     value = TRUE)
dat$love_avg <- (dat$selfTrnsdEmo_love_3 +
                   dat$selfTrnsdEmo_love_8 +
                   dat$selfTrnsdEmo_love_13) / 3
# Compare means by condition
by(data = dat$love_avg, INDICES = dat$Condition, FUN = mean)
by(data = dat$love_avg, INDICES = dat$Condition, FUN = sd)


##################
# COMPASSION
grep(pattern = "comp",
     x = names(dat),
     ignore.case = TRUE,
     value = TRUE)
dat$comp_avg <- (dat$selfTrnsdEmo_compass_4 +
                   dat$selfTrnsdEmo_compass_9 +
                   dat$selfTrnsdEmo_compass_14) / 3
# Compare means by condition
by(data = dat$comp_avg, INDICES = dat$Condition, FUN = mean)
by(data = dat$comp_avg, INDICES = dat$Condition, FUN = sd)


##################
```

```
# OPTIMISM
grep(pattern = "opt",
     x = names(dat),
     ignore.case = TRUE,
     value = TRUE)
dat$optim_avg <- (dat$selfTrnsdEmo_optimsism_5 +
                  dat$selfTrnsdEmo_optimsism_10 +
                  dat$selfTrnsdEmo_optimsism_15) / 3
# Compare means by condition
by(data = dat$optim_avg, INDICES = dat$Condition, FUN = mean)
by(data = dat$optim_avg, INDICES = dat$Condition, FUN = sd)
```

Create a smaller data set with only the relevant variables called `rct`.

```
vars <- c("Age.x", "Marital.status", "sex",
          "Condition", "Political.Affiliation..US.",
          "comp_avg", "optim_avg", "grat_avg",
          "awe_avg", "love_avg")
rct <- subset(x = dat, select = vars)
names(rct)[2] <- "mar_status"
names(rct)[5] <- "pol_affil"
dim(rct)
[1] 172  10
head(rct)
  Age.x          mar_status    sex Condition   pol_affil comp_avg optim_avg grat_avg  awe_avg love_avg
1    24              Single   Male   Control  Republican 2.333333  3.333333 3.000000 4.333333 2.333333
2    33              Single Female       Awe    Democrat 3.000000  5.000000 4.000000 3.666667 3.666667
3    25              Single Female   Control    Democrat 1.000000  1.333333 1.000000 1.333333 1.000000
4    35              Single   Male       Awe        None 3.666667  2.666667 1.333333 4.333333 2.333333
5    28 In a relationship Female       Awe             4.000000  5.000000 5.000000 5.000000 4.666667
6    32 In a relationship Female       Awe        None 3.666667  3.666667 3.666667 4.000000 3.666667
```

2   Visually Explore Two-Way Interactions with Compassion Outcome

2.1   Sex

Create an interaction plot to look at the interaction between participant sex and condition for the compassion outcome variable.

```
interaction.plot(x.factor = rct$Condition,
                 trace.factor = rct$sex,
                 response = rct$comp_avg,
                 trace.label = "Sex",
                 xlab = "Condition",
                 ylab = "Compassion")
```

Note that (1) the treatment condition is first and the control is second and (2) there is an extra dotted line in the legend. Both these issues can be fixed by applying the `factor()` function to the variables in the `rct` data frame. For the first issue, we will need to relevel the condition variable, placing control as the first level.
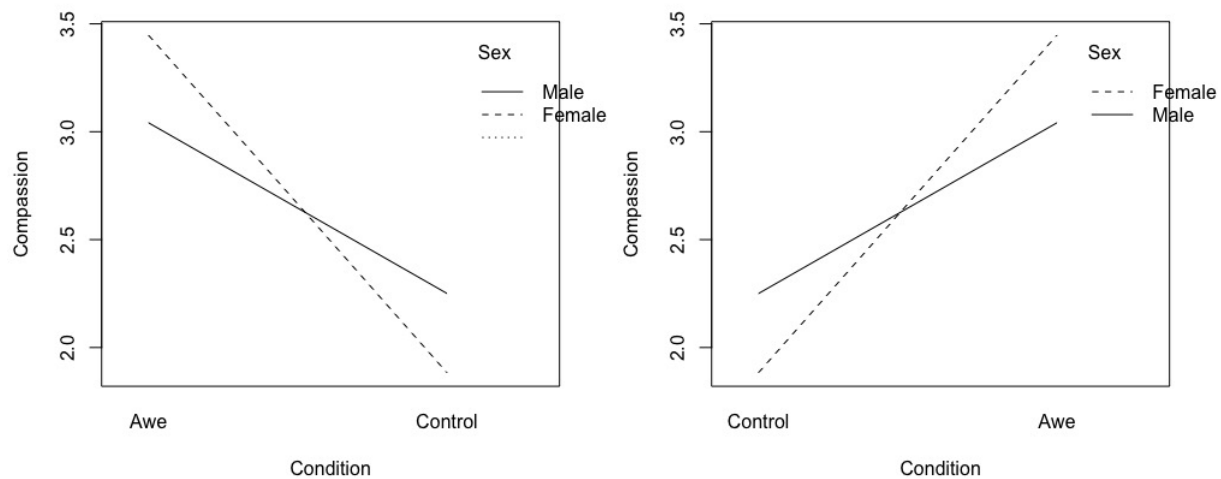
Figure 1: Interaction plots of participant sex by treatment condition for the compassion outcome variable

```
rct$Condition <- relevel(x = rct$Condition,
                         ref = "Control")
levels(rct$Condition)
[1] "Control" "Awe"
```

For the second issue, first let's explore why this is happening. We can get a clue by looking at the table of `sex`. Notice the blank category on the left with zero frequency? That is causing the problem. It is simple to drop levels that have zero frequency: just apply the `factor()` function. After applying these changes, rerun the interaction plot to get a more sensible picture.

```
table(rct$sex, useNA = "always")

     Female    Male   <NA>
  0      72     100      0

levels(rct$sex)
[1] ""       "Female" "Male"
rct$sex <- factor(rct$sex)
levels(rct$sex)
[1] "Female" "Male"
```

2.2 Political Affiliation

```
table(rct$pol_affil, useNA = "always")

        Democrat         N/A        None     Other Republican         <NA>
     6          73           8          23        24         38            0
```

Note that there are 8 missing values labeled "N/A", and 6 missing values with an empty label. R's system missing is `NA`, so these other categories are not counted as missing by R until they are correctly labeled. We will do that next. Notice that after the missing data are correctly labeled as `NA`, there are 14 `NA` values and 0 in the two categories where they were previously housed. To drop the levels associated with zero frequency, run `factor()`.

```
which(rct$pol_affil == "N/A" | rct$pol_affil == "")
 [1]   5  74  91  99 100 111 119 127 131 137 141 143 144 167
miss_ind <- which(rct$pol_affil == "N/A" | rct$pol_affil == "")
rct$pol_affil[miss_ind] <- NA
table(rct$pol_affil, useNA = "always")


          Democrat        N/A       None   Other Republican       <NA>
        0        73          0         23     24         38         14


rct$pol_affil <- factor(rct$pol_affil)
table(rct$pol_affil, useNA = "always")

  Democrat       None   Other Republican       <NA>
        73         23     24         38         14
```

Examine the two-way plot of group means via an interaction plot with party affiliation as x-factor and group assignment as trace factor and vice versa.

```
interaction.plot(x.factor = rct$Condition,
                 trace.factor = rct$pol_affil,
                 response = rct$comp_avg,
                 trace.label = "Marital\nStatus",
                 xlab = "Condition",
                 ylab = "Compassion")
```

2.3 Marital Status

```
table(rct$mar_status, useNA = "always")
```

	Divorced	In a relationship	Married	N/A	Never married	Separated	Single	Widowed
11	10	31	49	8	6	2	54	1

Create a new variable with three categories: married, in a relationship, single.

```
rct$mar_status2 <- NA
> rct$mar_status2[which(rct$mar_status == "Married")] <- "Married"
> rct$mar_status2[which(rct$mar_status == "Single" |
+                       rct$mar_status == "Never married" |
+                       rct$mar_status == "Widowed" |
+                       rct$mar_status == "Separated" |
+                       rct$mar_status == "Divorced")] <- "Single"
> rct$mar_status2[which(rct$mar_status == "In a relationship")] <- "Relationship"
```
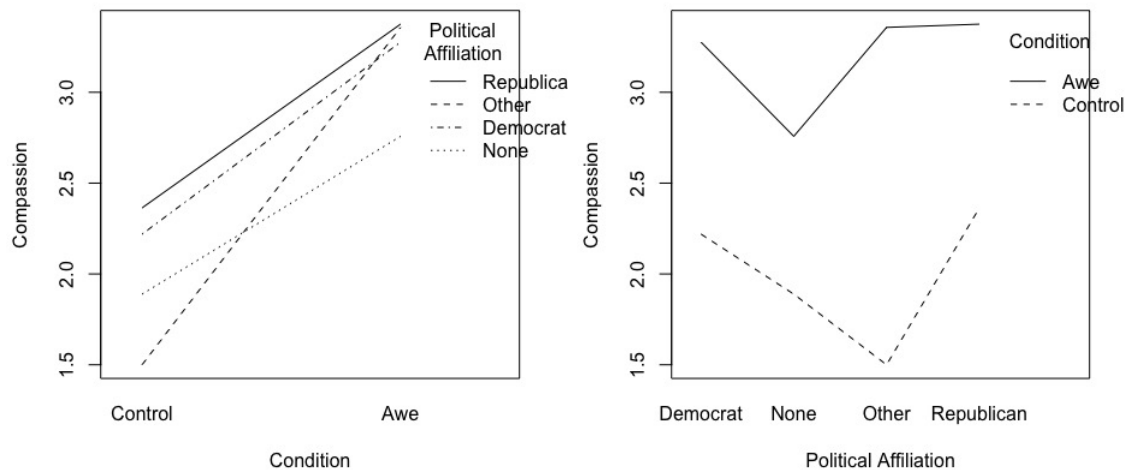
Figure 2: Interaction plots of participant political affiliation by treatment condition for the compassion outcome variable

```
> table(rct$mar_status2, useNA = "always")

    Married Relationship          Single           <NA>
         49             31              73             19

interaction.plot(x.factor = rct$Condition,
                 trace.factor = rct$mar_status2,
                 response = rct$comp_avg,
                 trace.label = "Marital\nStatus",
                 xlab = "Condition",
                 ylab = "Compassion")
```

3   Contrast Testing

Looking at the plot for political affiliation, it appears as though there is no evidence of interaction for all groups except for the "other" group. In open-ended surveys, the "other" group is often enigmatic because it is a catch-all. Therefore, the other group may contain participants with vastly different views who *shouldn't* be considered as part of the same group based on any reasonable substantive definitions. For example, in this case, someone who responded "tea party" and someone who responded "green party" would be placed in the same group. For these reasons, let's drop the "other" group for the remaining analyses here.

```
rct2 <- rct[-which(rct$pol_affil == "Other"),]
dim(rct2)
[1] 148  11
```

Figure 3: Interaction plots of participant marital status by treatment condition for the compassion outcome variable

Furthermore, for the sake of having a categorical predictor with three (instead of two) levels, we will test for a main effect of political affiliation on compassion, ignoring treatment condition. Again, the rationale for doing that is that there appears to be no evidence of a two-way interaction with condition, so it appears to be safe to average across condition to calculate the main effects of political affiliation. (Note that we would typically test the two-way interaction first with an F test, but we haven't covered that yet; we will do so next class).

```
lm1 <- lm(formula = comp_avg ~ pol_affil,
          data = rct2)
library(car)
Anova(lm1, type = 3)
emm1 <- emmeans::emmeans(object = lm1,
              specs = ~ pol_affil)
pairs(emm1, adjust = "none")
 contrast               estimate    SE  df t.ratio p.value
 Democrat - None          0.4217 0.279 131   1.513  0.1326
 Democrat - Republican   -0.0634 0.233 131  -0.272  0.7859
 None - Republican       -0.4851 0.308 131  -1.576  0.1175


pairs(emm1, adjust = "Tukey")
 contrast               estimate    SE  df t.ratio p.value
 Democrat - None          0.4217 0.279 131   1.513  0.2881
 Democrat - Republican   -0.0634 0.233 131  -0.272  0.9600
 None - Republican       -0.4851 0.308 131  -1.576  0.2598


P value adjustment: tukey method for comparing a family of 3 estimates
```

```
pairs(emm1, adjust = "Bonferroni")
 contrast                estimate     SE  df t.ratio p.value
 Democrat - None          0.4217 0.279 131   1.513  0.3979
 Democrat - Republican   -0.0634 0.233 131  -0.272  1.0000
 None - Republican       -0.4851 0.308 131  -1.576  0.3526

P value adjustment: bonferroni method for 3 tests
pairs(emm1, adjust = "Holm")
 contrast                estimate     SE  df t.ratio p.value
 Democrat - None          0.4217 0.279 131   1.513  0.3526
 Democrat - Republican   -0.0634 0.233 131  -0.272  0.7859
 None - Republican       -0.4851 0.308 131  -1.576  0.3526

P value adjustment: holm method for 3 tests
pairs(emm1, adjust = "Scheffe")
 contrast                estimate     SE  df t.ratio p.value
 Democrat - None          0.4217 0.279 131   1.513  0.3214
 Democrat - Republican   -0.0634 0.233 131  -0.272  0.9637
 None - Republican       -0.4851 0.308 131  -1.576  0.2924

P value adjustment: scheffe method with dimensionality 2

pairs(emm1, adjust = "FDR")
 contrast                estimate     SE  df t.ratio p.value
 Democrat - None          0.4217 0.279 131   1.513  0.1990
 Democrat - Republican   -0.0634 0.233 131  -0.272  0.7859
 None - Republican       -0.4851 0.308 131  -1.576  0.1990

P value adjustment: fdr method for 3 tests
```

Note that each method reports what are referred to as *multiplicity-adjusted p-values*. The idea with an adjusted p-value is that it represents the smallest level of significance that leads to rejection of the null hypothesis (Wright, *Biometrics*, 1992, p. 1006). See R help on the `p.adjust` function for further references, and see the Wright paper for a general definition and extensions and see Aickin and Gensler, *Public Health Breifs*, 1996 for an application to Bonferroni and Holm-Bonferroni.