2/4/2010 36-402/608 ADA-II H. Seltman
Breakout #8 Comments

Remember the iridium levels at different depths. Here are a set of planned contrasts designed to answer specific questions beyond the "overall null hypothesis" of $H_0 : \mu_A = \cdots = \mu_F$ for the ANOVA F test.

```
e0 = aov(iridium ~ depth, extinct)
summary(e0)
#            Df Sum Sq Mean Sq F value    Pr(>F)
#depth        5 735267  147053  7.7058 0.0002568 ***
#Residuals   22 419834   19083

# Make a set of planned contrasts:
oc = rbind(CvsOthers=c(-1/5,-1/5,1,-1/5,-1/5,-1/5),
           DEFvsAB=c(-1/2,-1/2,0,1/3,1/3,1/3),
           BvsA=c(-1,1,0,0,0,0),
           DvsEF=c(0,0,0,1,-1/2,-1/2),
           EvsF=c(0,0,0,0,1,-1))

# Check: Are the contrasts orthogonal, i.e., all dot products=0?
round(t(oc)%*%oc,5)
#          CvsOthers DEFvsAB BvsA DvsEF EvsF
#CvsOthers       1.2 0.00000    0   0.0    0
#DEFvsAB         0.0 0.83333    0   0.0    0
#BvsA            0.0 0.00000    2   0.0    0
#DvsEF           0.0 0.00000    0   1.5    0
#EvsF            0.0 0.00000    0   0.0    2
```

Zeros on the off diagonals are equivalent to zero dot products using the definition of matrix multiplication and thinking about the form of the transposed matrix.

Question 1: What is being tested with the "oc" contrasts?

Each is a comparison of the population mean of one set of treatment groups to the population mean of another set. To keep the resulting estimate on the right scale, we should use 1/b as the coefficient in front of any group of size b.

Here is a function to do the standard contrasts (matching SAS, SPSS, etc) in R: You need to install package "gmodels". The function `fit.contrast()` wants each contrast in a row. For I levels of the factor, you can have up to $I - 1$ contrasts. Confidence intervals are an option:

```
round(fit.contrast(e0, "depth", t(oc), conf.int=0.95), 3)
#                Estimate Std. Error t value Pr(>|t|) lower CI upper CI
# depthCvsOthers  390.026     68.662   5.680    0.000  247.630  532.423
# depthDEFvsAB    -59.905     63.562  -0.942    0.356 -191.723   71.914
# depthBvsA        26.833    105.508   0.254    0.802 -191.977  245.644
# depthDvsEF      125.607     80.041   1.569    0.131  -40.388  291.602
# depthEvsF        42.286     80.888   0.523    0.606 -125.466  210.037
```

Question 3: Explain the meaning of the first two contrast estimates. What would be different and what would be the same if we used `CvsOthers=c(-1,-1,5,-1,-1,-1)`?

Our best estimate of how much higher the population mean of iridium levels in depth C is compared to the average of all other levels is 390 ppt (95% CI = [248, 532]).

Our best estimate of how much higher the population mean of iridium levels in depths A and B is compared to depths D, E, and F is 60 ppt (95% CI = [72, 192]).

Important technical note: You may run across the `C()` or `contrasts()` functions in R. These give the correct t and p-value, but ignore your scaling, so the estimate and SE are meaningless, and a CI cannot be constructed.

```
# Another set of contrasts (which are not mutually orthogonal):
noc = rbind(CvsOthers=c(-1/5,-1/5,  1, -1/5,-1/5,-1/5),
            DvsOthers=c(-1/5,-1/5,-1/5,  1, -1/5,-1/5),
            EvsOthers=c(-1/5,-1/5,-1/5,-1/5,  1, -1/5),
            BCvsOthers=c(-1/4, 1/2, 1/2,-1/4,-1/4,-1/4),
            CDvsOthers=c(-1/4,-1/4, 1/2, 1/2,-1/4,-1/4))
round(fit.contrast(e0, "depth", noc), 3)
# Error in make.contrasts(coeff, ncol(coeff)) : singular contrast matrix
```

E.g., the dot product of the first two is $1/25+1/25-5/25-5/25+1/25+1/25 = -6/25 \neq 0$.

```
## (Technical note: Although mutual orthogonality is sufficient for
##  a valid set of planned contrasts, it is not necessary.  The
##  necessary condition is that $C'C$ matrix has as many non-zero
##  eigenvalues as the number of specified contrasts.)
round(fit.contrast(e0, "depth", noc[-4,]), 3)
# Error in make.contrasts(coeff, ncol(coeff)) : singular contrast matrix
```

Contrasts 1 through 4 (plus a fifth one chosen by the function to be orthogonal to all the others) do form a valid decomposition of the SS for depth.

```
round(fit.contrast(e0, "depth", noc[-5,]), 3)
#                 Estimate Std. Error t value Pr(>|t|)
# depthCvsOthers   390.026     68.662   5.680    0.000
# depthDvsOthers    -6.274     75.037  -0.084    0.934
# depthEvsOthers  -131.631     60.561  -2.174    0.041
# depthBCvsOthers  232.033     59.580   3.894    0.001
```

Contrasts 1 through 3 plus 5 (plus a fifth one chosen by the function to be orthogonal to all the others) do not form a valid decomposition of the SS for depth.

```
# Note: With non-orthogonal (singular) contrasts, type 1 error
# is not preserved, so you probably don't want to cheat
# and enter them with two calls to fit.contrast().
```

Here is the experiment from Sleuth chapter 13 on seaweed regrowth, The outcome is regrowth of seaweed (% coverage) on rocks that were scraped clean of seaweed at 8 different locations on the sea floor. The locations are treated as blocks because conditions differ in important non-quantifiable ways at the different locations.

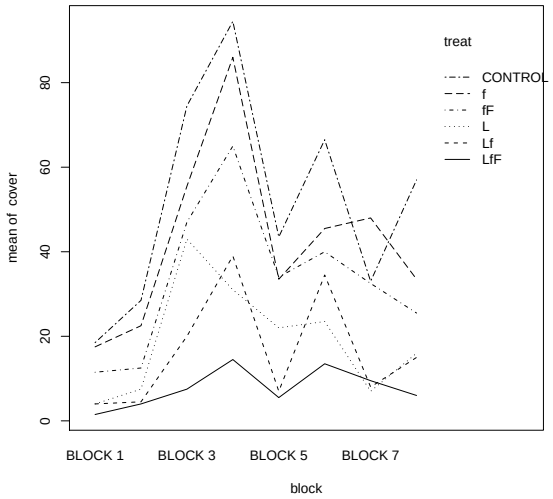
The treatment conditions are different kinds of cages and barriers to keep out different kinds of sea life: limpets (a kind of snail), small fish, and large fish. Each treatment was used twice at each location. For technical reasons not all eight combinations of organism presence vs. absence can be tested, so treatment is considered as a single factor with 6 levels.

The treatments are labeled according to the organisms that can access the rocks: L=limpet snails, f=small fish, F=large fish. (The data are available on the Sleuth CD.)

```
sea = read.csv("case1301.csv")
dim(sea) # [1] 96  3
sapply(sea,class)
#    COVER     BLOCK     TREAT
#"integer"  "factor"  "factor"
names(sea) = casefold(names(sea))

# Check if we have  a balanced design:
with(sea, table(block, treat))
#          treat
# block     CONTROL f fF L Lf LfF
#   BLOCK 1       2 2  2 2  2   2
#   BLOCK 2       2 2  2 2  2   2
#   BLOCK 3       2 2  2 2  2   2
#   BLOCK 4       2 2  2 2  2   2
#   BLOCK 5       2 2  2 2  2   2
#   BLOCK 6       2 2  2 2  2   2
#   BLOCK 7       2 2  2 2  2   2
#   BLOCK 8       2 2  2 2  2   2

# A nice plot for 2 way ANOVA:
with(sea, interaction.plot(block, treat, cover))
```

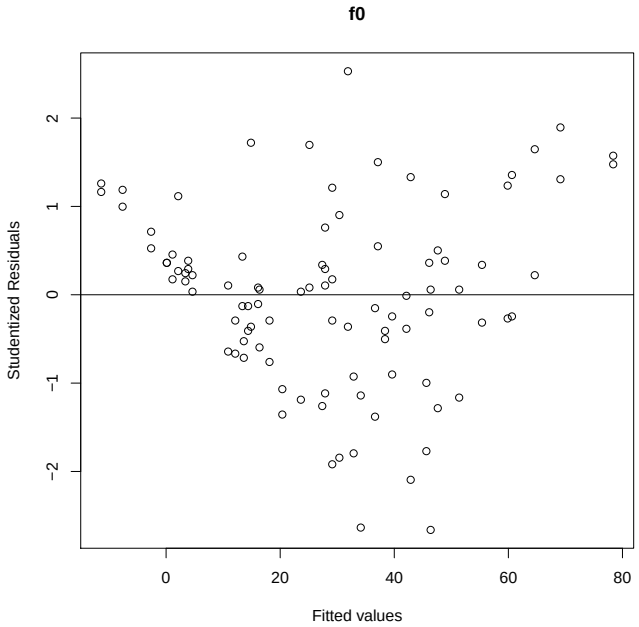


Question 4: What pattern do you see?

Blocks are in meaningless order on the x-axis. Mean percent cover (of two trials in one block with one treatment) form each point. The different lines are different treatments.

The top three lines are for treatments without limpets, and these have the highest regrowth of algae, suggesting that limpets limit regrowth. The fish also appear to reduce regrowth. The lines appear non-parallel suggesting an interaction between treatment and block in their effects on cover, i.e., the difference in cover between some pairs of treatments differs in magnitude across at least some blocks.

```
f0 = aov(cover ~ block + treat, sea)
rp(f0, fname="BO8MEresfit")
```

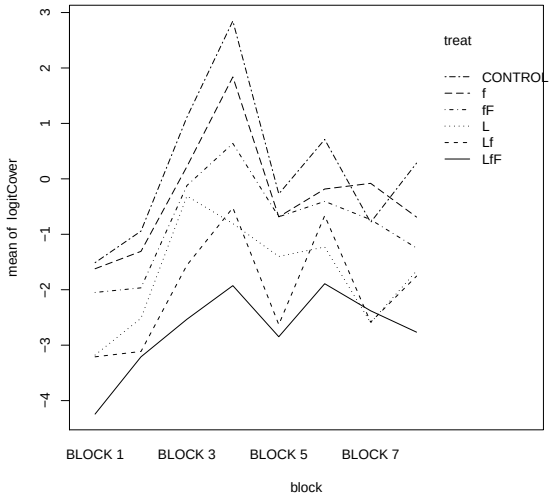


Question 5: What problem do you see?

There is non-linearity and non-constant variance with less variance at the high and low percent cover than in the middle.

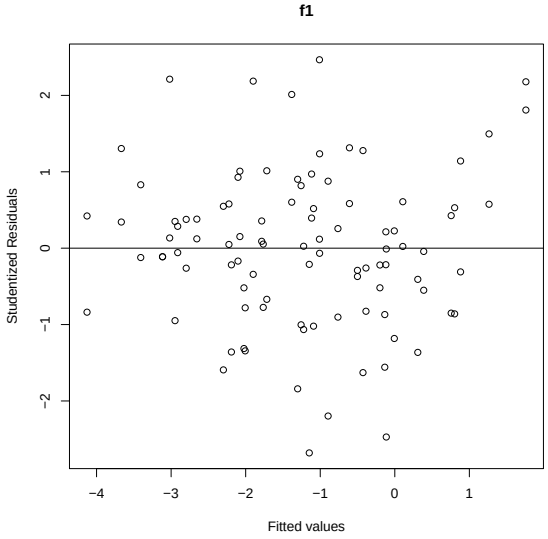
This problem is common for percents. The solution is either the arcsin(sqrt(percent)) transformation or the logit transformation: log(percent / (100-percent)).

```
sea$logitCover = with(sea, log(cover/(100-cover)))
with(sea, interaction.plot(block, treat, logitCover))
```




```
f1 = aov(logitCover ~ block + treat, sea)
```

```
rp(f1, fname="BO8MELogitResFit")
```



This is much better: Now let's check the interaction model:

```
f3 = aov(logitCover ~ block * treat, sea)
rp(f3, fname="BO8IAresFit")
summary(f3)
#            Df Sum Sq Mean Sq F value Pr(>F)
#block        7 76.239 10.8912 35.9634 <2e-16 ***
#treat        5 96.993 19.3986 64.0553 <2e-16 ***
#block:treat 35 15.230  0.4352  1.4369 0.1209
#Residuals   48 14.536  0.3028
```

The interaction is not statistically significant. In other words we retain the null hypothesis that an additive (parallel) model is sufficient. In other words, we do not have good evidence that the pattern of treatment effects (across the six different treatments) varies across the eight block locations. So we will continue our analysis with the additive model (although we might be making a type 2 error regarding the interaction).

```
summary(f1)
#             Df Sum Sq Mean Sq F value    Pr(>F)
# block        7 76.239 10.8912  30.368 < 2.2e-16 ***
# treat        5 96.993 19.3986  54.090 < 2.2e-16 ***
# Residuals   83 29.767  0.3586

summary(lm(logitCover ~ block + treat, sea))
# Coefficients:
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept)   -1.2226     0.2204  -5.548 3.37e-07 ***
```

```
# blockBLOCK 2   0.4600     0.2445   1.881   0.0634 .
# blockBLOCK 3   2.1046     0.2445   8.608 3.97e-13 ***
# blockBLOCK 4   2.9807     0.2445  12.192  < 2e-16 ***
# blockBLOCK 5   1.2160     0.2445   4.974 3.49e-06 ***
# blockBLOCK 6   2.0251     0.2445   8.283 1.77e-12 ***
# blockBLOCK 7   1.1085     0.2445   4.534 1.93e-05 ***
# blockBLOCK 8   1.3300     0.2445   5.440 5.27e-07 ***
# treatf        -0.4941     0.2117  -2.334   0.0220 *
# treatfF       -1.0019     0.2117  -4.732 9.03e-06 ***
# treatL        -1.8925     0.2117  -8.938 8.68e-14 ***
# treatLf       -2.1849     0.2117 -10.319  < 2e-16 ***
# treatLfF      -2.9052     0.2117 -13.721  < 2e-16 ***
```

Question 6: How is ANOVA better able to answer the key questions here than regression?

With multi-level categorical variables, regression does not directly answer the overall question: does a particular factor have equal population means of the outcome or does it have means with some differences? This can also be answered with two regressions, with and without the factor, then by applying the `anova()` function to do the added-sum-of-squares F test.

For both regression and anova you can use the same `fit.contrasts()` function to do the followup contrast testing.

Now for some quick contrast tests:

```
levels(sea$treat)
# [1] "CONTROL" "f"       "fF"      "L"       "Lf"      "LfF"
with(sea, aggregate(cover, list(treat=treat), mean)$x)
# [1] 52.00 42.75 33.50 19.25 16.50  7.75

with(sea, aggregate(logitCover, list(treat=treat), mean)$x)
# [1]  0.1804836 -0.3136515 -0.8214197 -1.7119924 -2.0043847 -2.7246679

tcont = rbind(CvsOthers = c(1, rep(-1/5,5)),
              FfvsFfL = c(0, 1/2, 1/2, -1/3,-1/3, -1/3),
              fvsfF = c(0,1,-1,0,0,0),
              LFfvsL = c(0,0,0,1,-1/2,-1/2),
              LfvsLfF = c(0,0,0,0,1,-1))

You can see that these are orthogonal in the way they divide up
the information about the means without re-using the same
information for two different purposes: compare control to others,
```

```
then break up others into with and without limpets, then break
up without limpets into small and large fish, then break up
with limpets into with and without fish, and finally break
up limpets with fish into limpets with small fish and
limpets with large fish.

crslt = fit.contrast(f1, "treat", tcont, conf.int=0.95)
round(crslt, 1)
#                Estimate Std. Error t value Pr(>|t|) lower CI upper CI
# treatCvsOthers      1.7        0.2    10.3        0      1.4      2.0
# treatFfvsFfL        1.6        0.1    11.6        0      1.3      1.9
# treatfvsfF          0.5        0.2     2.4        0      0.1      0.9
# treatLFfvsL         0.7        0.2     3.6        0      0.3      1.0
# treatLfvsLfF        0.7        0.2     3.4        0      0.3      1.1

round( exp(crslt[,c(5,1,6)]), 3)
#                lower CI Estimate upper CI
# treatCvsOthers    3.933    5.450    7.553
# treatFfvsFfL      3.697    4.852    6.368
# treatfvsfF        1.091    1.662    2.532
# treatLFfvsL       1.334    1.920    2.766
# treatLfvsLfF      1.349    2.055    3.131
```

Interpretation: A difference between groups in a log transformation is a ratio on the scale of what was logged, which is the ratio of fraction covered with algae to fraction not covered. So the covered to not covered ratio is 3.9 to 7.6 times bigger for the control vs. the average of the other groups (95% CI).