Displays for Statistics 5401/8401

Lecture 15

October 10, 2005

Christopher Bingham, Instructor

612-625-1024, kb@umn.edu
372 Ford Hall

Class Web Page

http://www.stat.umn.edu/~kb/classes/5401

Choosing a test in profile analysis

Friday I looked at 4 sets of contrasts of variable means

$$\mathbf{C}_a\boldsymbol{\mu} = [\mu_2-\mu_1,\ \mu_3-\mu_2,\ldots,\mu_p-\mu_{p-1}]'$$

$$\mathbf{C}_b\boldsymbol{\mu} = [\mu_2-\mu_1,\ \mu_3-\mu_1,\ldots,\mu_p-\mu_1]'$$

$$\mathbf{C}_c\boldsymbol{\mu} = [\mu_1-\mu_2,\ \mu_1+\mu_2-2\mu_3,\ldots,\ \mu_1+\mu_2+\ \ldots\ +\mu_{p-1}-(p-1)\mu_p]'$$

$$\mathbf{C}_d\boldsymbol{\mu} = [\mu_2-\mu_1,\ \mu_3-\mu_1,\ \ldots,\mu_p-\mu_{p-1}],$$

where $\mathbf{C}_d\boldsymbol{\mu}$ has all distinct differences $\mu_i - \mu_j$ $i > j$

For these $\mathbf{C}$'s ($\mathbf{C}_a$, $\mathbf{C}_b$, $\mathbf{C}_c$, $\mathbf{C}_d$) and others, $\mu_1 = \mu_2 = \ldots = \mu_p$ if and only if $\mathbf{C}\boldsymbol{\mu} = \mathbf{0}$

This means you can test

$$H_0\text{: } \mu_1 = \mu_2 = \ldots = \mu_p$$

by <u>Bonferronizing t-tests</u> for the components any of these sets of contrasts or indeed components of other sets of contrasts as long as rank($\mathbf{C}$) = p-1.

How do you choose $\mathbf{C}$?

The question does not have a statistical answer. The contrasts you use should be tailored to your *particular research goals* so that you may answer specific questions of interest to you (or your client).

- When you are comparing p-1 treatments with a control you might Bonferronize the comparisons in $\mathbf{C}_b$
- When you are trying to identify a change point you might Bonferronize the comparisons in $\mathbf{C}_a$ or $\mathbf{C}_c$.
- When there is no structure of importance among the means, you may want all paired differences as defined by $\mathbf{C}_d$. This is repeated measures **multiple comparisons**.

To obtain a ***powerful test*** (high $P(\text{reject } H_0 \mid H_0 \text{ false})$), you may be able to use *prior or expert knowledge* to identify contrasts with large non-centrality $\sum c_i\mu_i/\{\sqrt{\mathbf{c}'\Sigma\mathbf{c}}\}$. They are likely to have large values of t. You would include such a $\mathbf{c}$ as a row of $\mathbf{C}$.

For instance, when the treatments are quantitative and you expect the profile might be linear with constant $\mu_{j+1}-\mu_j \neq 0$. Then a contrast with equally spaced c_j's is likely to be appropriate because it "matches" the pattern expected.

Example: When p = 7, this would be

$$\mathbf{c} = [-3, -2, -1, 0, 1, 2, 3]$$

When you have little idea how H_0 might be wrong and the data are highly correlated, T^2 is probably best.

MacAnova example using data in Table 6.2, p. 281 in the text.

```
Cmd> x <- read("","t06_02") # read JWData5.txt
T06_02    19    4 format
) Data from Table 6.2 p. 281 in
) Applied Mulivariate Statistical Analysis, 5th Edition
) by Richard A. Johnson and Dean W. Wichern, Prentice Hall, 2002
) These data were edited from file T6-2.DAT on disk from book
) Sleeping-dog data                    A            B
) Col. 1: Response for treatment 1 (High Co_2, pressure w/o H)
) Col. 2: Response for treatment 2 (Low Co_2, pressure w/o H)
) Col. 3: Response for treatment 3 (High Co_2, pressure with H)
) Col. 4: Response for treatment 4 (Low Co_2, pressure with H)
Read from file "TP1:Stat5401:Data:JWData5.txt"
```

The experiment has to do with testing the effect of the anesthetic halothane on 19 dogs. The treatments had a 2 by 2 factorial structure

- Factor A: High (A) and low (a) CO_2 pressure
- Factor B: Use (B) or non-use (b) of halothane.

The p = 4 treatments were Ab, ab, AB, aB.

You can often clarify output by adding labels. Command **setlabels()** is one way to do this:

```
Cmd> setlabels(x,structure("@",vector("Ab", "ab", "AB", "aB")))

Cmd> x[run(3),] # rows 1 - 3 of data
             Ab          ab          AB          aB
(1)         426         609         556         600
(2)         253         236         392         395
(3)         359         433         349         357
```

"@" specifies numerical labels for rows.

structure("@", "Trt ") would have created the less informative columns labels **Trt 1**. **Trt 2**, **Trt 3** and **Trt 4**.

```
Cmd> stats <- tabs(x,mean:T,covar:T)

Cmd> stats # three components
component: mean             x-bar  (column  vector)
(1)      368.21      404.63      479.26      502.89
component: covar            s_x
(1,1)      2819.3      3568.4      2943.5      2295.4
(2,1)      3568.4      7963.1        5304      4065.5
(3,1)      2943.5        5304      6851.3      4499.6
(4,1)      2295.4      4065.5      4499.6        4879
```

Because of the factorial structure, the following contrast matrix seems sensible

```
Cmd> c <- matrix(vector(1,-1,1,-1, -1,-1,1,1, 1,-1,-1,1),4)'

Cmd> setlabels(c,structure(vector("A","B","AB"),\
       getlabels(x,2)))
```

MacAnova: `getlabels(x,2)` retrieves the column labels of `x` so `setlabels()` sets row labels to `vector("A","B","AB")` and makes column labels the same as `x`.

```
Cmd> c
              Ab           ab           AB           aB
A              1           -1            1           -1
B             -1           -1            1            1
AB             1           -1           -1            1
```

- Row 1 compares A with a (main effect)
- Row 2 compares B with b (main effect)
- Row 3 is an AB interaction contrast.

```
Cmd> xbar <- stats$mean; xbar # sample mean vector
(1)       368.21       404.63       479.26       502.89

Cmd> s <- stats$covar # 4 by 4 sample variance matrix

Cmd> n <- nrows(x) # sample size
```

```
Cmd> vhat <- s/n # Vhat[xbar] = estimated var matrix of x-bar

Cmd> cxbar <- c %*% xbar; cxbar # = ybar = means of contrasts
            (1)
A       -60.053       Estimate of A effect
B        209.32       Estimate of B effect
AB      -12.789       Estimate of AB effect

Cmd> cvhatc <- c %*% vhat %*% c'; cvhatc # Vhat[ybar]
                A            B           AB
A          273.46       57.837       48.135
B          57.837       496.43       48.821
AB         48.135       48.821       397.76
```

- vhat is $\hat{V}[\bar{x}]$
- cxbar is $C\bar{x}$
- cvhatc is $C\hat{V}[\bar{x}]C' = \hat{V}[C\bar{x}]$

```
Cmd> tsq <- cxbar' %*% (cvhatc %\% cxbar); tsq
               (1)
(1)         116.02       Tests H0: μy = Cμx = 0
```

- tsq is $T^2 = (C\bar{x})'(C\hat{V}[\bar{x}]C')^{-1}(C\bar{x})$

MacAnova: `vhatc %\% cxbar` is the same as `solve(vhatc, cxbar)`.

```
Cmd> fe <- n - 1 # single sample error d.f.

Cmd> p <- ncols(x); q <- p - 1 # number of contrasts

Cmd> f <- (fe - q + 1)*tsq/(q*fe); f # f-stat for T^2
(1,1)       34.375

Cmd> 1 - cumF(f,q,fe-q+1) # P-value
(1,1)   3.3178e-07
```

You can also compute T^2 directly from the matrix `x %*% c'` of contrasts in the data.

```
Cmd> hotellval(x %*% c')
(1,1)      116.02
```

Conclusion: At least one of the contrasts is non-zero.

But which contrasts? That's where Bonferronized t is useful.

```
Cmd> stderrs <- sqrt(diag(cvhatc)) # standard errors of ybars

Cmd> tstats <- vector(cxbar/stderrs) # univariate t-stats

Cmd> tstats # t-statistics
(1)     -3.6315      9.3945    -0.64127

Cmd> q <- length(tstats) # Bonferronizing factor

Cmd> tcritval <- invstu(1 - .025/q, fe); tcritval
(1)      2.6391      Bonferronized 2-tail critical value

Cmd> q*twotailt(tstats,fe) #Bonferronized 2-tail p-values
(1)   0.0057264  6.9446e-08      1.5883
```

Or you could compute the t-statistics directly from `x %*% c'`:

```
Cmd> tstats <- tval(x %*% c'); tstats
(1)     -3.6315      9.3945    -0.64127
```

By identifying the significant contrasts, you can conclude

- the A main effect is significant
- the B main effect is significant
- there is no evidence the AB interaction contrast is non-zero.

Of course, any significant t implies that

$$H_0: \mu_1 = \mu_2 = \mu_3 = \mu_4 \text{ is false}$$

Since this follows a T^2, the analysis in terms of contrasts is sometimes called **post hoc** analysis.

Compare the Bonferronized t-critical value with the "ellipsoidal" critical value based on T^2.

```
Cmd> tsqcritval <- sqrt(fe*q*invF(1-.05,q,fe-q+1)/(fe-q+1))

Cmd> vector(q, fe-q+1)
(1)           3          16

Cmd> vector(tcritval,tsqcritval) # Bonferronized and ellipsoid
(1)      2.6391      3.3062

Cmd> tsqcritval/tcritval # ellipsoidal 25% larger than Bonf t
(1)      1.2528

Cmd> # Compute Bonferronized simultaneous confidence limits

Cmd> cxbar + tcritval*vector(-1,1)'*stderrs
(1,1)      150.51      268.12      Width =  117.6
(2,1)      -103.7      -16.41      Width =  87.286
(3,1)     -65.424      39.845      Width =  105.27

Cmd> # Compute Ellipsoidal limits

Cmd> cxbar + tsqcritval*vector(-1,1)'*stderrs
(1,1)      135.65      282.98      Width =  147.33
(2,1)     -114.73     -5.3782      Width =  109.35
(3,1)     -78.729       53.15      Width =  131.88
```

The "ellipsoidal" intervals based on the critical value for T^2 are much (25.3%) wider than Bonferronized Student's t intervals.

Since the three contrasts are sensible in view of the treatment structure and were selected before looking at the data, the Bonferronized t-limits are entirely appropriate.

Randomized Block Analysis

An informal check that univariate RCB ANOVA might be OK (equal σ_{ii}, equal ρ_{ij}):

```
Cmd> diag(s) # variances of the variables
(1)      2819.3      7963.1      6851.3        4879

Cmd> sqrt(diag(s))# standard deviations of the variables
(1)      53.097      89.236      82.773       69.85

Cmd> cor(x) # correlation matrix
              Ab          ab          AB          aB
Ab             1     0.75312     0.66974     0.61889
ab       0.75312           1     0.71808     0.65223
AB       0.66974     0.71808           1     0.77826
aB       0.61889     0.65223     0.77826           1
```

The standard deviations are not very different and neither are the correlations, so two-way univariate ANOVA may be OK. You need to restructure the data to do this.

```
Cmd> x1 <- vector(x') # unravel x by rows

Cmd> treatment <- factor(rep(run(4),nrows(x)))#1,2,3,4,1,2,3,4…

Cmd> dogs <- factor(rep(run(n),rep(4,n)))#1,1,1,1,2,2,2,2...

Cmd> anova("x1 = dogs + treatment",fstat:T) # dogs are blocks
Model used is x1 = dogs + treatment
              DF          SS          MS            F     P-value
CONSTANT       1   1.463e+07   1.463e+07   7913.35657     < 1e-08
dogs           1  3.0539e+05       16966      9.17702     < 1e-08
treatment      3  2.2602e+05       75340     40.75088     < 1e-08
ERROR1        54       99835      1848.8
```

The F-test for **treatment** is analogous to the T^2 test.

Compute contrasts in treatment means:

```
Cmd> con1 <- contrast(treatment,vector(c[1,]))

Cmd> con2 <- contrast(treatment,vector(c[2,]))

Cmd> con3 <- contrast(treatment,vector(c[3,]))

Cmd> compnames(con1)
(1) "estimate"
(2) "ss"
(3) "se"

Cmd> vector(con1$estimate,con2$estimate,con3$estimate)
(1)    -60.053      209.32     -12.789

Cmd> cxbar' # repeat of previously computed contrast means
(1,1)    -60.053      209.32     -12.789    Same values

Cmd> vector(con1$se,con2$se,con3$se) # ANOVA standard errors
(1)      19.729      19.729      19.729

Cmd> stderrs # repeat of previously computed contrast Std errs
(1)      16.537      22.281      19.944
```

The standard errors are in the same ball park but not identical.

```
Cmd> 3*twotailt(vector(con1[1],con2[1],con3[1])/\
            vector(con1[3],con2[3],con3[3]),54)
(1)    0.010811  2.4212e-14      1.5587
```

Find Bonferronized confidence limits based on univariate analysis:

```
Cmd> con1$estimate + vector(-1,1)*invstu(1 - .025/3,54)*con1$se
(1)      -108.8     -11.306    vs  -103.7        -16.41 before

Cmd> con2$estimate + vector(-1,1)*invstu(1 - .025/3,54)*con2$se
(1)      160.57      258.06    vs  150.51        268.12 before

Cmd> con3$estimate + vector(-1,1)*invstu(1 - .025/3,54)*con3$se
(1)     -61.536      35.957    vs -65.424        39.845 before
```

The univariate limits are shorter in each case.

It would be probably be simpler just to introduce factors for CO_2 and halothane.

```
Cmd> co2 <- factor(1+(treatment == 1 || treatment == 3))

Cmd> halo <- factor(1+(treatment == 3 || treatment == 4))

Cmd> head(hconcat(co2,halo), 8) # 2 dogs worth of co2 & halo
(1,1)          2           1  Dog 1 hi Co2, no halothane
(2,1)          1           1  Dog 1 low Co2, no halothane
(3,1)          2           2  Dog 1 hi Co2, with halothane
(4,1)          1           2  Dog 1 low Co2, with halothane
(5,1)          2           1  Dog 2 hi Co2, no halothane
(6,1)          1           1  Dog 2 low Co2, no halothane
(7,1)          2           2  Dog 2 hi Co2, with halothane
(8,1)          1           2  Dog 2 low Co2, with halothane

Cmd> anova("x1 = dogs + co2 + halo + co2.halo",fstat:T)
Model used is x1 = dogs + co2 + halo + co2.halo
                DF          SS          MS           F     P-value
CONSTANT         1   1.463e+07   1.463e+07  7913.35657  2.9806e-60
dogs            18  3.0539e+05       16966     9.17702  1.0083e-10
co2              1       17130       17130     9.26554   0.0036036
halo             1  2.0811e+05  2.0811e+05   112.56684  8.0708e-15
co2.halo         1      776.96      776.96     0.42025     0.51956
ERROR1          54       99835      1848.8

Cmd> SS # computed by anova
    CONSTANT        dogs         co2        halo    co2.halo
      ERROR1
   1.463e+07  3.0539e+05       17130  2.0811e+05      776.96
       99835

Cmd> DF # computed by anova
    CONSTANT        dogs         co2        halo    co2.halo
      ERROR1
           1          18           1           1           1
          54

Cmd> MS <- SS/DF # mean squares

Cmd> fstats <- MS[run(3,5)]/MS[6]; fstats # F-statistics
         co2        halo    co2.halo
      9.2655      112.57     0.42025

Cmd> 3*cumF(fstats,DF[run(3,5)],DF[6],upper:T) # Bonf. P-values
(1)    0.010811  2.4212e-14      1.5587
```

Univariate Linear Models

There are at least three standard types of univariate linear models.

They all model a dependent or *response* variable y in the form

$$\boxed{y = \textit{predictable part} + \textit{unpredictable part}}$$

where the predictable part is described using parameters that enter *linearly*.

The "+" is important -- the unpredictable part enters *additively.*

The *unpredictable part* may itself be the *sum* of several independent pieces, say a block effect and a plot effect.

Notation: At least in today's examples the predictable part is in (...) and the unpredictable part in {...}

Examples

- $y = (\beta_1 + \beta_2 x^{\beta_3}) + \{\varepsilon\}$ There are 2 linear parameters (β_1 and β_2) and 1 nonlinear one (β_3), so this is not a linear model
- *Multiple Linear Regression*

  $$y_i = (Z_{i0}\beta_0 + Z_{i1}\beta_1 + \ldots + Z_{ik}\beta_k) + \{\varepsilon_i\}$$

  where $E[\varepsilon_i] = 0$ & (usually) $Z_{i0} \equiv 1$

  There are k + 1 linear parameters.

  I use $Z_{ij}\beta_j$ rather than $\beta_j Z_{ij}$ to make it easier to generalize the notation to a multivariate dependent variable.

  The Z's are predictor or independent variables, usually quantitative (except for Z_{i0}).

- *ANOVA* (additive linear model)

One way ANOVA with g groups

$$y_{ij} = (\mu + \alpha_i) + \{\varepsilon_{ij}\}$$

$$i = 1,\ldots,g,\ j = 1,\ldots,n_i$$

Usually $\sum_{1\le i\le g}\alpha_i = 0$

The α's are *fixed* group *effects*

Randomized blocks (two-way ANOVA)

$$y_{ij} = (\mu + \alpha_i) + \{B_j + \varepsilon_{ij}\}$$

Usually $\sum_{1\le i\le g}\alpha_i = 0$.

Always $E[B_j] = E[\varepsilon_{ij}] = 0$

The α's are *fixed* group or treatment effects.

The B's are *random* block effects.

Split Plot with 1 whole plot factor (A) and 1 subplot factor (N) with whole plots arranged in RCB design

$$y_{ijk} = (\mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}) + \{B_k + \varepsilon_{ik}^{w} + \varepsilon_{ijk}^{s}\}$$

The α_i's are **fixed** main effects for the whole plot factor, $\sum_i\alpha_i = 0$.

The β_i's are **fixed** main effects for the subplot factor, $\sum_j\beta_j = 0$.

The $(\alpha\beta)$'s are **fixed** interaction effects, $\sum_i(\alpha\beta)_{ij} = \sum_j(\alpha\beta)_{ij} = 0$

The B's are **random** block effects.

The ε^ws are **random** whole plot errors within blocks

The ε^ss are **random** subplot errors within whole plots

More generally, in an ANOVA type model, y may have *multiple* subscripts and the model is of the form

$$y_{ijk\ldots} = \mu + (T_1 + T_2 + \ldots) + \{E_1 + E_2 + \ldots\}$$

where

- Each term T_k is a subscripted parameter such as α_i, β_j, γ_ℓ, $(\alpha\beta)_{ij}$, or $(\alpha\beta\gamma)_{ij\ell}$, usually satisfying restrictions like $\sum_i(\alpha\beta)_{ij} = \sum_j(\alpha\beta)_{ij} = 0$.
- Each term E_m is a random effect such as B_ℓ and $\varepsilon_{ij\ell}$, a subscripted part of the *unpredictable* part. They satisfy $E[E_m] = 0$, and are all independent of one another.

ANACOVA (analysis of covariance)

This combines ANOVA and regression.

One-way ANACOVA (or **ANCOVA**)

$$y_{ij} = Z_{ij0}\beta_0 + Z_{ij1}\beta_1 + \ldots + Z_{ijk}\beta_k + \alpha_i + \varepsilon_{ij}$$
$$E[\varepsilon_{ij}] = 0,\ \text{usually } \textstyle\sum_i\alpha_i = 0,\ i = 1,\ldots,g$$

Except for Z_{ij0}, *covariates* are the Z's which are quantitative variables.

When $Z_{ij0} \equiv 1$, for each group this is a multiple regression with

- intercept $\beta_0 + \alpha_i$ which may differ among groups
- the <u>same</u> slopes $\beta_1, \ldots, \beta_k$ in each group.

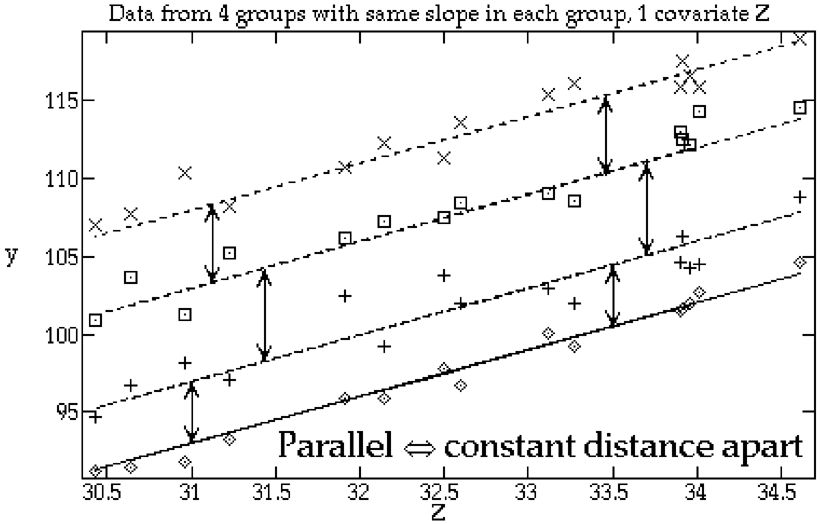
More generally, there can be other terms:

$$y_{ijk\ldots} = (\beta_0 Z_{ij\ell\ldots0} + \beta_1 Z_{ij\ell..1} + \ldots + \beta_k Z_{ij\ell..k} + T_1 + T_2 + \ldots) + \{E_1 + E_2 + \ldots\},$$
$$E[E_m] = 0$$

With k = 1 covariate Z, the model is

$$y_{ij} = \mu + Z_{ij}\beta + \alpha_i + \varepsilon_{ij},\ \mu = \beta_0,\ \beta = \beta_1$$

Here is a plot of data that might come from a one way ANACOVA model when the number of groups = g = 4 and k = 1.



The mean of the group i data for given Z is $\mu_i(Z) = \mu + \alpha_i + \beta Z_1$, *parallel* lines.

The difference in means between groups i_1 and i_2 is $\alpha_{i_1} - \alpha_{i_2}$ and is the same for any value of Z_1,

The groups differ in the intercepts $\mu + \alpha_i$ but not the slopes. More general models allow the slopes to differ among groups.

Because the slopes do not differ, the difference between mean responses for two groups, at a specific value z of the covariate does not depend on z:

$$\mu_i(z) - \mu_j(z) =$$

$$(\mu + \alpha_i + \beta z) - (\mu + \alpha_j + \beta z) = \alpha_i - \alpha_j$$

When slopes do differ between groups, no single number which summarizes the difference between two groups:

$$\mu_i(z)-\mu_j(z) = (\mu + \alpha_i + \beta_i z) - (\mu + \alpha_j + \beta_j z)$$
$$= \alpha_i - \alpha_j + (\beta_i - \beta_j)z$$

where β_j is the slope for group j.

This depends on z.