PSYCH 710

Linear Contrasts, Trend Analysis, & Multiple Comparisons

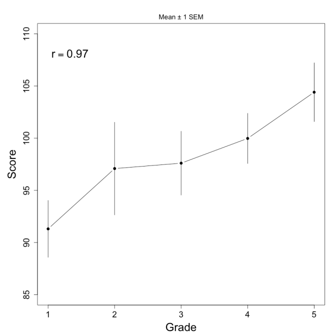
Sept 26, 2023

Prof. Patrick Bennett

1-Way ANOVA

Cognitive development study

- Cognitive test administered to grades 1-5
  - 15 children per grade
- average scores increase approx linearly
  - correlation between grade & $\bar{Y}$ = 0.97
- use ANOVA to evaluate group differences

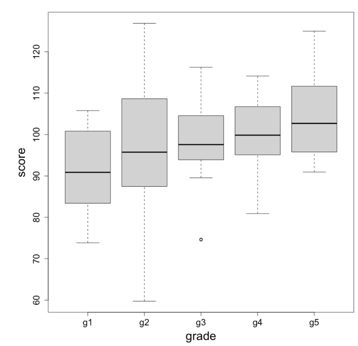


Cognitive development study

check constant variance assumption

```
> bartlett.test(score~grade,df3)
Bartlett test of homogeneity of variances
data:  score by grade
Bartlett's K-squared = 6.6227, df = 4, p-value = 0.1572
```

Do not reject null hypothesis that variances are equal

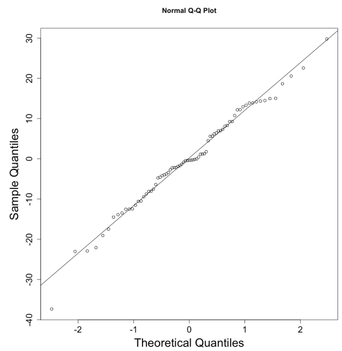


Cognitive development study

check normality assumption

> shapiro.test(residuals(aov.01))
Shapiro-Wilk normality test
data: residuals(aov.01)
W = 0.9871, p-value = 0.6482

> qqnorm(residuals(aov.01))
> qqline(residuals(aov.01))

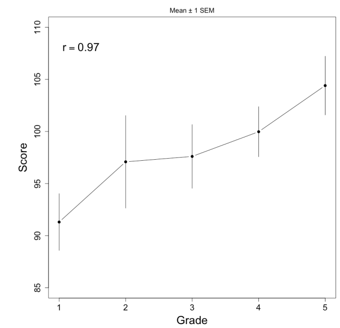


Do not reject null hypothesis that residuals are Normal

Cognitive development study

> load(file=url('http://pnb.mcmaster.ca/bennett/psy710/datasets/contrasts.rda'))
> df3$grade <- factor(df3$grade,ordered=FALSE)
> options(contrasts=c("contr.sum","contr.poly")) # IMPORTANT!!

> aov.01 <- aov(score~grade,df3)
> anova(aov.01)
Analysis of Variance Table
Response: score
Df Sum Sq Mean Sq F value Pr(>F)
grade 4 1361.2 340.31 2.2578 0.07152 .
Residuals 70 10550.9 150.73



- the effect of grade was not significant
- do not reject the null hypothesis of no difference among group means

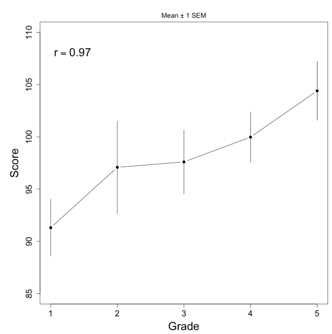
Cognitive development study

Note that 1-sided 95% CIs for effect size & association strength include zero

> library(effectsize)
> cohens_f(aov.01)
Effect Size for ANOVA
Parameter | Cohen's f | 95% CI
grade | 0.36 | [0.00, Inf]

> omega_squared(aov.01)
Effect Size for ANOVA
Parameter | Omega2 | 95% CI
grade | 0.06 | [0.00, 1.00]

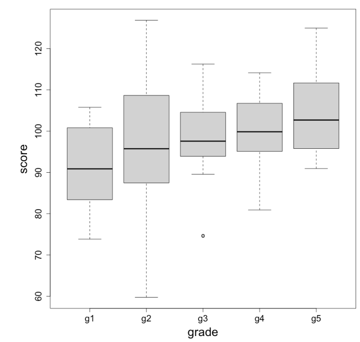
	Cohen's f	Omega-squared
small	0.1	0.01
medium	0.25	0.06
large	0.4	0.14



Cognitive development study

estimate power assuming medium effect size (f = 0.25)

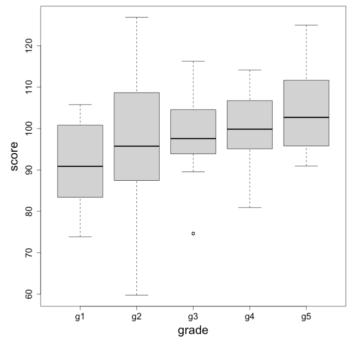
> library(pwr)
> pwr.anova.test(k=5,n=15,
+ f=0.25,sig.level=.05,
+ **power=NULL**)
Balanced 1-way anova power calculation
k = 5
n = 15
f = 0.36
sig.level = 0.05
power = 0.35
NOTE: n is number in each group



Cognitive development study

estimate power assuming f = 0.36

```
> library(pwr)
> pwr.anova.test(k=5,n=15,
+                  f=0.36,sig.level=.05,
+                  power=NULL)
Balanced 1-way anova power calculation
          k = 5
          n = 15
          f = 0.36
    sig.level = 0.05
       power = 0.67
NOTE: n is number in each group
```



Cognitive development study

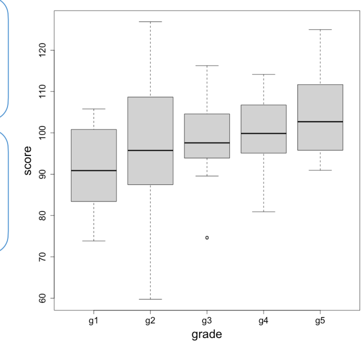
alternatives to ANOVA

```
> oneway.test(score~grade,data=df3)
1-way analysis of means (not assuming equal variances)
data:  score and grade
F = 3, num df = 4, denom df = 35, p-value = 0.04
```

```
> kruskal.test(score~grade,data=df3)
Kruskal-Wallis rank sum test
data:  score by grade
Kruskal-Wallis chi-squared = 9, df = 4, p-value = 0.07
```

K-W Null Hypothesis:

- groups were sampled from the same distribution
- if we assume distributions have same shape & scale:
  - then H0 is that group medians are equal



linear contrasts/comparisons

Omnibus vs. Focussed F tests

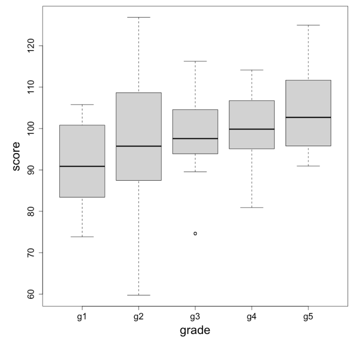
$$H0: \quad \alpha_1 = \alpha_2 = \cdots = \alpha_a = 0$$
$$H1: \quad \alpha_j \neq 0$$

- A significant omnibus F test tests a very general hypothesis
  - H0: all group means are equal; H1: not all means are equal
  - H0: all group effects are zero; H1: not all group effects are zero
- Significant F doesn't tell us how group means differ
- Generality of omnibus F often comes at cost of reduced power

Omnibus vs. Focussed F tests

Omnibus F test is not significant:

```
> lm.01 <- lm(score~grade,data=df3)
> anova(lm.01)
Analysis of Variance Table
Response: score
          Df  Sum Sq Mean Sq F value  Pr(>F)
grade      4  1361.2  340.31  2.2578 0.07152
Residuals 70 10550.9  150.73
```



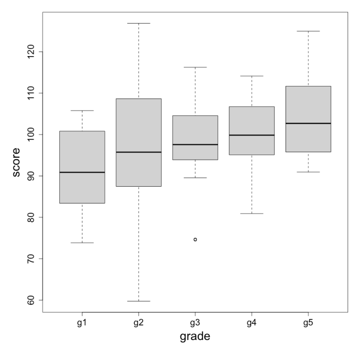
focussed tests provide more power

linear contrasts often more appropriate & more powerful

Linear Contrast Example

H0: (means of grades 3,4,5) = (means of grades 1,2)
H1: (means of grades 3,4,5) ≠ (means of grades 1,2)

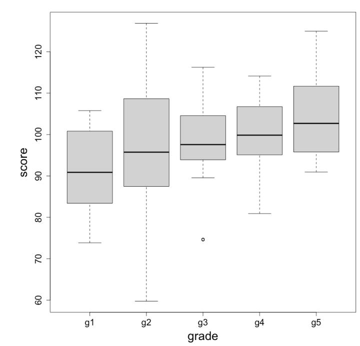
```
> c1 <- c(-1/2,-1/2,1/3,1/3,1/3) # (g1 & g2) vs (g3 & g4 & g5)
> c2 <- c(-1,1,0,0,0) # g1 vs g2
> c3 <- c(0,0,-1,1/2,1/2) # g3 vs (g4 & g5)
> c4 <- c(0,0,0,-1,1) # (g4 vs g5)
> contrasts(df3$grade) <- cbind(c1,c2,c3,c4)

> fractions( contrasts(df3$grade) )
     c1  c2  c3  c4
g1 -1/2  -1   0   0
g2 -1/2   1   0   0
g3  1/3   0  -1   0
g4  1/3   0 1/2  -1
g5  1/3   0 1/2   1
```



Linear Contrast Example

H0: (means of grades 3,4,5) = (means of grades 1,2)
H1: (means of grades 3,4,5) ≠ (means of grades 1,2)

```
> aov.02 <- aov(score~grade,data=df3)

> summary(aov.02,
+       split=list(grade=list(c1=1,c2=2,c3=3,c4=4)))
               Df   SS    MS     F    Pr(>F)
grade           4  1361  340.3  2.258 0.0715 .
  grade: c1     1   753  752.9  4.995 0.029 *
  grade: c2-c4  3   608  202.7  1.345 0.269
  Residuals    70 10551  150.7
```

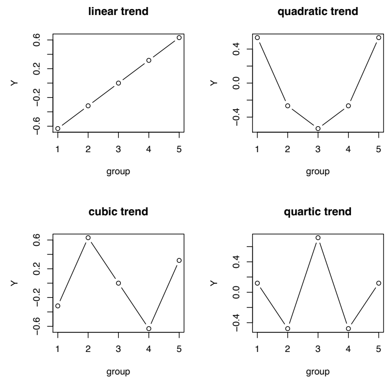


Trend Analysis Example

linear contrasts can be more powerful & more appropriate tests of null hypothesis

H0: linear trend of score across grade = 0
H1: linear trend of score across grade ≠ 0

	Lin	Quad	Cubic	Quartic
[g1]	-0.632	0.535	-3.16e-01	0.120
[g2]	-0.316	-0.267	6.32e-01	-0.478
[g3]	0.000	-0.535	-4.10e-16	0.717
[g4]	0.316	-0.267	-6.32e-01	-0.478
[g3]	0.632	0.535	3.16e-01	0.120

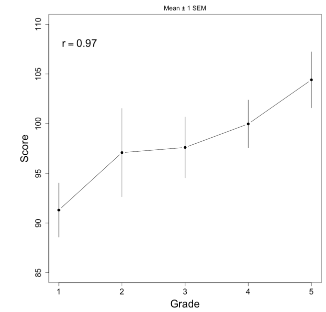


Trend Analysis Example

linear contrasts can be more powerful & more appropriate tests of null hypothesis

H0: linear trend of score across grade = 0
H1: linear trend of score across grade ≠ 0

```
> summary(aov.trends,
+        split=list(grade=list(Lin=1,NonLin=2:4)))
               Df    SS     MS     F    Pr(>F)
grade           4  1361  340.3 2.258  0.07152
  grade: Lin    1  1270 1269.9 8.425  0.00495 **
  grade: NonLin 3    91   30.5 0.202  0.89460
Residuals      70 10551  150.7
```



Linear Contrasts (Comparisons)

- Contrasts allow us to evaluate focussed hypotheses
  - evaluate specific pattern of differences among group means
- Each contrast is defined by a set of contrast weights
  - weights (c_1, c_2, … c_a) specify a pattern of group means
  - value of contrast, ψ, is a weighted combination of group means
    - $\psi = c_1 \bar{Y}_1 + c_2 \bar{Y}_2 + c_3 \bar{Y}_3 + c_4 \bar{Y}_4 + \ldots c_a \bar{Y}_a$

Hypotheses tested with Linear Contrasts

- Linear contrasts are defined by weights
  - must sum to zero
  - sum(1/2, 1/2,-1/3, -1/3,-1/3) = 0
- Multiplying weights by constant produces an equivalent linear contrast
  - w_1 = (1/2, 1/2,-1/3, -1/3,-1/3)
  - w_2 = 6 x w_1 = (3,3,-2,-2,-2)
  - w_1 is equivalent to w_2

contrast weights

$$H0: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 = 0$$

$$H1: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 \neq 0$$

contrast weights

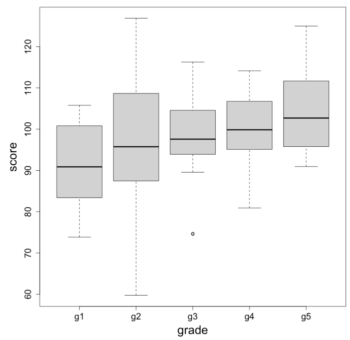
$$H0: 3\mu_1 + 3\mu_2 - 2\mu_3 - 2\mu_4 - 2\mu_5 = 0$$

$$H1: 3\mu_1 + 3\mu_2 - 2\mu_3 - 2\mu_4 - 2\mu_5 \neq 0$$

Contrasts are defined by weights

each contrast sums to zero!

```
> myC1 <- c(-1/2,-1/2,1/3,1/3,1/3) # (grades 1,2) vs (grades 3,4,5)
> myC2 <- c(-1,1,0,0,0) # (grade 1) vs (grade 2)
> myC3 <- c(0,0,-1,1/2,1/2) # (grade 3) vs (grades 4,5)
> myC4 <- c(0,0,0,-1,1) # (grade 4) vs (grade 5)
> cMat <- cbind(myC1, myC2, myC3, myC4)
> fractions(cMat) # fractions() in MASS library
     myC1 myC2 myC3 myC4
[1,] -1/2  -1    0    0
[2,] -1/2   1    0    0
[3,]  1/3   0   -1    0
[4,]  1/3   0   1/2  -1
[5,]  1/3   0   1/2   1
```



Hypotheses Evaluated by a Contrast

```
> my.weights0 <- c(3,3,-2,-2,-2)
```

N.B. my.weights0 is equivalent to c(1/2, 1/2, -1/3,-1/3,-1/3)

H0: $3\mu_1 + 3\mu_2 - 2\mu_3 - 2\mu_4 - 2\mu_5 = 0$

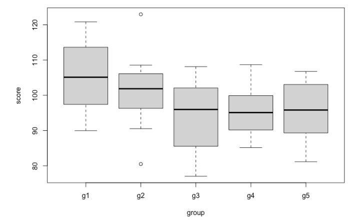
$$3(\mu_1 + \mu_2) - 2(\mu_3 + \mu_4 + \mu_5) = 0$$

$$3(\mu_1 + \mu_2) = 2(\mu_3 + \mu_4 + \mu_5)$$

$$3\frac{(\mu_1 + \mu_2)}{2} = (\mu_3 + \mu_4 + \mu_5)$$

$$\frac{(\mu_1 + \mu_2)}{2} = \frac{(\mu_3 + \mu_4 + \mu_5)}{3}$$

H1: $\frac{(\mu_1 + \mu_2)}{2} \neq \frac{(\mu_3 + \mu_4 + \mu_5)}{3}$



Hypotheses Evaluated by a Contrast

```
> my.weights <- c(-1, -1, -1, -1, -1, -1, 6)
```

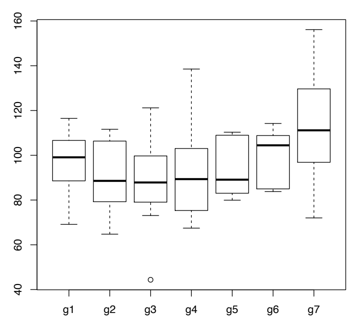
H0:

$$-1(\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6) + 6\mu_7 = 0$$

$$\mu_7 = \frac{1}{6}(\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6)$$

H1:

$$\mu_7 \neq \frac{1}{6}(\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6)$$



Hypotheses Evaluated by a Contrast

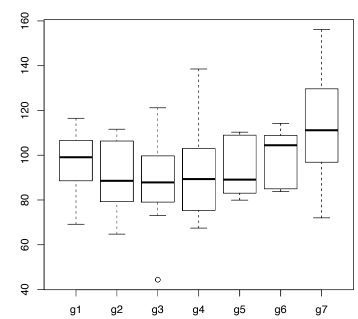
```
> w1 <- c(-2,-2,0,1,1,1,1)
> (w2 <- w1/4)
[1] -1/2 -1/2  0  1/4  1/4  1/4  1/4
> # sum(w1) = sum(w2) = 0
```

H0: $\frac{-1}{2}(\mu_1 + \mu_2) + 0 \times \mu_3 + \frac{1}{4}(\mu_4 + \mu_5 + \mu_6 + \mu_7) = 0$

$$\frac{(\mu_4 + \mu_5 + \mu_6 + \mu_7)}{4} = \frac{(\mu_1 + \mu_2)}{2}$$

H1: $\frac{(\mu_4 + \mu_5 + \mu_6 + \mu_7)}{4} - \frac{(\mu_1 + \mu_2)}{2} \neq 0$

$$\frac{(\mu_4 + \mu_5 + \mu_6 + \mu_7)}{4} \neq \frac{(\mu_1 + \mu_2)}{2}$$



Hypotheses Evaluated by a Contrast

> my.weights0 <- c(3,0,-1,-1,-1)

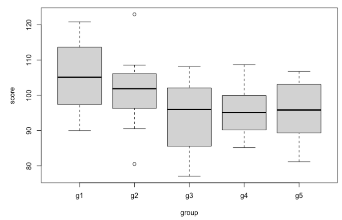
HO: $3\mu_1 - 0\mu_2 - 1\mu_3 - 1\mu_4 - 1\mu_5 = 0$

$3\mu_1 - 1(\mu_3 + \mu_4 + \mu_5) = 0$

$3\mu_1 = 1(\mu_3 + \mu_4 + \mu_5)$

$\mu_1 = \frac{1}{3}(\mu_3 + \mu_4 + \mu_5)$

H1: $\mu_1 \neq \frac{1}{3}(\mu_3 + \mu_4 + \mu_5)$



> w2 <- c(1,0,-1/3,-1/3,-1/3) # my.weights0 / 3

General Form of Linear Contrast

$\text{H0}: \; c_1\mu_1 + c_2\mu_2 + \cdots + c_a\mu_a = \Psi = 0$ weighted sum of population means equals zero

$\sum_{j=1}^{a} c_j = 0$ sum of weights must equal zero

$\hat{\Psi} = \sum_{j=1}^{a} (c_j \bar{Y}_j)$ value of contrast equals weighted sum of group means

$SS_{contrast} = MS_{contrast}$

Evaluate comparison with F:

$F = \frac{(\Psi^2)/\sum_{j=1}^{a}(c_j^2/n_j)}{\text{MS}_W}$

With equal n per group:

$F = \frac{(n\Psi^2)/\sum_{j=1}^{a}(c_j^2)}{\text{MS}_W}$

contrast df = 1

df = (1, N-a)

Hypotheses tested with Linear Contrasts

2-tailed tests

$H0: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 = 0$

$H1: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 \neq 0$

$H0: \frac{\mu_1 + \mu_2}{2} = \frac{\mu_3 + \mu_4 + \mu_5}{3}$

$H1: \frac{\mu_1 + \mu_2}{2} \neq \frac{\mu_3 + \mu_4 + \mu_5}{3}$

1-tailed tests

$H0: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 \geq 0$

$H1: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 < 0$

$H0: \frac{\mu_1 + \mu_2}{2} \geq \frac{\mu_3 + \mu_4 + \mu_5}{3}$

$H1: \frac{\mu_1 + \mu_2}{2} < \frac{\mu_3 + \mu_4 + \mu_5}{3}$

General Form of Linear Contrast

(directional tests evaluated with t statistic)

$\hat{\Psi} = \sum_{j=1}^{a} (c_j \bar{Y}_j)$

$F = \frac{(\Psi^2)/\sum_{j=1}^{a}(c_j^2/n_j)}{\text{MS}_W}$

$t = \frac{\Psi / \sqrt{\sum_{j=1}^{a}(c_j^2/n_j)}}{\sqrt{\text{MS}_W}}$ df = N-a

$t^2 = F$

t statistic more useful for 1-tailed tests

$H0: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 \geq 0$

$H1: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 < 0$

$H0: \frac{\mu_1 + \mu_2}{2} \geq \frac{\mu_3 + \mu_4 + \mu_5}{3}$

$H1: \frac{\mu_1 + \mu_2}{2} < \frac{\mu_3 + \mu_4 + \mu_5}{3}$

General Form of Linear Contrast

(the sign of the weights determines the direction of the test)

t statistic: the sign of weights matters!
w = [1/2, 1/2, -1/3, -1/3, -1/3]

$$H0: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 \geq 0$$

$$H1: \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 - \frac{1}{3}\mu_3 - \frac{1}{3}\mu_4 - \frac{1}{3}\mu_5 < 0$$

$$H0: \frac{\mu_1 + \mu_2}{2} \geq \frac{\mu_3 + \mu_4 + \mu_5}{3}$$

$$H1: \frac{\mu_1 + \mu_2}{2} < \frac{\mu_3 + \mu_4 + \mu_5}{3}$$

equivalent!

t statistic: the sign of weights matters!
w = [-1/2, -1/2, 1/3, 1/3, 1/3]

$$H0: \frac{-1}{2}\mu_1 + \frac{-1}{2}\mu_2 + \frac{1}{3}\mu_3 + \frac{1}{3}\mu_4 + \frac{1}{3}\mu_5 \leq 0$$

$$H1: \frac{-1}{2}\mu_1 + \frac{-1}{2}\mu_2 + \frac{1}{3}\mu_3 + \frac{1}{3}\mu_4 + \frac{1}{3}\mu_5 > 0$$

$$H0: \frac{\mu_3 + \mu_4 + \mu_5}{3} \leq \frac{\mu_1 + \mu_2}{2}$$

$$H1: \frac{\mu_3 + \mu_4 + \mu_5}{3} > \frac{\mu_1 + \mu_2}{2}$$

calculating contrasts with aov & emmeans

Conducting Contrasts with R aov()

```
> contrasts(df3$grade) <- cMat
> fractions( contrasts(df3$grade) )
    myC1 myC2  myC3  myC4
g1  -1/2   -1     0     0
g2  -1/2    1     0     0
g3   1/3    0    -1     0
g4   1/3    0   1/2    -1
g5   1/3    0   1/2     1
> aov.02 <- aov(score~grade,data=df3)
```

Store contrast weights as columns in a matrix & then assign contrast weights to grouping variable

Perform ANOVA with aov

Conducting Contrasts with R aov()

```
> aov.02 <- aov(score~grade,data=df3)
> summary(aov.02,
+       split=list(grade=list(c1=1,c2=2,c3=3,c4=4)))

              Df Sum Sq Mean Sq F value Pr(>F)
grade          4   1361   340.3   2.258 0.0715 .
  grade: c1    1    753   752.9   4.995 0.0286 *
  grade: c2    1    251   250.6   1.662 0.2015
  grade: c3    1    210   210.5   1.396 0.2413
  grade: c4    1    147   147.3   0.977 0.3262
Residuals     70  10551   150.7
```

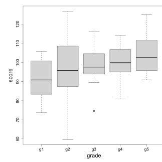
Write ANOVA table with summary(), but split results for grouping variable into separate lines for different contrasts

split = list(factor.name=list(contrast.name.1=1, contrast.name.2=2,…))

Linear contrasts assessed with F tests

Conducting Contrasts with emmeans

emmeans = estimated marginal means [Very statisticious: Getting started with emmeans]



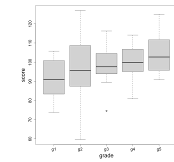
```
> # create emmeans object
> library(emmeans)
> aov.01 <- aov(score~grade,data=df3)
> aov.em <- emmeans(aov.01,specs="grade")
> aov.em
 grade emmean   SE df lower.CL upper.CL
 g1      91.3 3.17 70     85.0     97.6
 g2      97.1 3.17 70     90.8    103.4
 g3      97.6 3.17 70     91.3    103.9
 g4     100.0 3.17 70     93.7    106.3
 g5     104.4 3.17 70     98.1    110.7
Confidence level used: 0.95
```

(specs is the factor being analyzed)

(Estimated Marginal Means)

Conducting Contrasts with emmeans

emmeans = estimated marginal means [Very statisticious: Getting started with emmeans]



```
> myContrasts <- list(c1=myC1,
+                     c2=myC2,
+                     c3=myC3,
+                     c4=myC4)
> contrast(aov.em,
+          method=myContrasts,
+          adjust="none")
 contrast estimate   SE df t.ratio p.value
 c1           6.47 2.89 70   2.235  0.0286
 c2           5.78 4.48 70   1.289  0.2015
 c3           4.59 3.88 70   1.182  0.2413
 c4           4.43 4.48 70   0.989  0.3262
```

(p-values are the same as those obtained previously)

Conducting Contrasts with linear.comparison

linear.comparison() & emmeans() yield same results

```
> source(url("http://pnb.mcmaster.ca/bennett/psy710/Rscripts/linear_contrast_v2.R"))
[1] "loading function linear.comparison"
> y <- df3$score
> g <- df3$grade
> myContrast1 <- linear.comparison(y,g,c.weights = myContrasts,var.equal=T)
[1] "computing linear comparisons assuming equal variances among groups"
[1] "C 1: F=4.995, t=2.235, p=0.029, psi=6.467, CI=(0.367,12.568), adj.CI= (-0.952,13.887)"
[1] "C 2: F=1.662, t=1.289, p=0.202, psi=5.780, CI=(-4.615,16.175), adj.CI= (-5.714,17.274)"
[1] "C 3: F=1.396, t=1.182, p=0.241, psi=4.588, CI=(-2.544,11.719), adj.CI= (-5.366,14.542)"
[1] "C 4: F=0.977, t=0.989, p=0.326, psi=4.432, CI=(-2.955,11.819), adj.CI= (-7.062,15.926)"
```

trend analysis

Trend Analysis

trends are linear contrasts

- the analysis of trends uses the <u>same</u> methods as linear contrasts
- weights are designed to evaluate specific differences across groups:
  - linear, quadratic, cubic, etc.
- weights must sum to zero
- weights can be calculated using R's contr.poly function
  - useful when differences between levels on group variable are not constant

```
> contr.poly(n=5,scores=c(8,9,10,11,12))
             .L         .Q            .C         ^4
[1,] -0.6324555  0.5345225 -3.162278e-01  0.1195229
[2,] -0.3162278 -0.2672612  6.324555e-01 -0.4780914
[3,]  0.0000000 -0.5345225 -4.095972e-16  0.7171372
[4,]  0.3162278 -0.2672612 -6.324555e-01 -0.4780914
[5,]  0.6324555  0.5345225  3.162278e-01  0.1195229
```

```
> contr.poly(n=5,scores=c(8,9,10,12,15))
             .L          .Q         .C          ^4
[1,] -0.5045250  0.54194676 -0.4466312  0.22862383
[2,] -0.3243375 -0.01290349  0.4344281 -0.71127414
[3,] -0.1441500 -0.38710483  0.4685966  0.64014672
[4,]  0.2162250 -0.59356074 -0.6077113 -0.17781853
[5,]  0.7567875  0.45162230  0.1513177  0.02032212
```

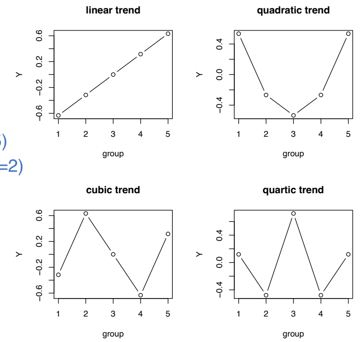
Trend Analysis Example

trends are linear contrasts

```
>  # set polynomial contrasts as default for ordered factors:
> options(contrasts=c("contr.sum","contr.poly")
> load(file=url('http://pnb.mcmaster.ca/bennett/psy710/labs/L3/hw3-2021.rda'))
> sapply(df3,class)
$grade
[1] "ordered" "factor"
$score
[1] "numeric"
> contrasts(df3$grade)
     .L    .Q    .C    ^4
g1 -0.63  0.53 -0.32  0.12
g2 -0.32 -0.27  0.63 -0.48
g3  0.00 -0.53  0.00  0.72
g4  0.32 -0.27 -0.63 -0.48
g5  0.63  0.53  0.32  0.12
```

Trend weights are orthogonal

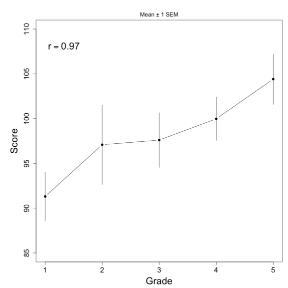
```
> polyWeights <- contr.poly(n=5)
> round(cor(polyWeights),digits=2)
      .L    .Q    .C    ^4
.L  1.00  0.00  0.00  0.00
.Q  0.00  1.00  0.00  0.00
.C  0.00  0.00  1.00  0.00
^4  0.00  0.00  0.00  1.00
```



Trend Analysis Example

trends are linear contrasts

```
> contrasts(df3$grade) <- contr.poly(n=5,scores=1:5)
> aov.trends <- aov(score~grade,data=df3)
> summary(aov.trends,
+         split=list(grade=list(Lin=1,Quad=2,Cube=3,Quart=4)))
               Df    SS     MS      F     Pr(>F)
grade           4  1361  340.3  2.258   0.07152
  grade: Lin    1  1270 1269.9  8.425   0.00495 **
  grade: Quad   1     1    0.8  0.005   0.94354
  grade: Cube   1    80   80.4  0.533   0.46760
  grade: Quart  1    10   10.2  0.068   0.79530
Residuals      70 10551  150.7
```



H0 & H1 defined by trend weights:

$$\text{H0}: \; -0.63\mu_1 - 0.32\mu_2 + 0.32\mu_4 + 0.63\mu_5 = 0$$

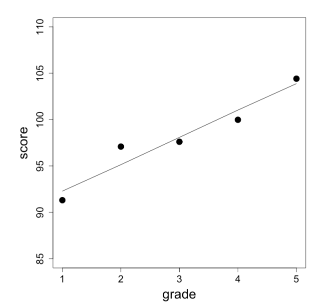
$$\text{H1}: \; -0.63\mu_1 - 0.32\mu_2 + 0.32\mu_4 + 0.63\mu_5 \neq 0$$

Trend Analysis Example

trends are linear contrasts

```
> coef(aov.trends)
(Intercept)   grade.L   grade.Q   grade.C   grade^4
   98.08        9.20     -0.23      2.32     -0.83
> wLin <- c(-0.63,-0.32,0,0.32,0.63) # linear trend weights
> gradeMean <- with(df3,tapply(score,grade,mean)) # group means
> gNumber <- seq(1:5)
> plot(x=gNumber,
     y=98.08+9.20*wLin,       # line defined by intercept &
     type="l",                # linear trend coefficient
     ylim=c(85,110),
     xlab="grade",
     ylab="score")
> points(x=gNumber,
       y=gradeMean,
       pch=19, cex=2)
```

intercept determines height
grade.L coefficient & trend weights determine slope

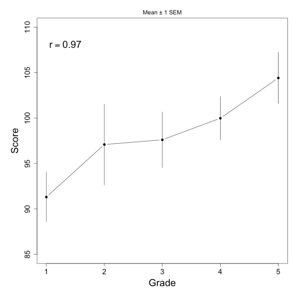


Trend Analysis Example

trends are linear contrasts

```
> # emmeans poly method uses polynomial contrasts
> # and assumes equally-spaced levels on grouping factor
> # ?poly.emmc for details
> aov.01.em <- emmeans(aov.01,specs="grade")
> contrast(aov.01.em,method="poly")
```

contrast	estimate	SE	df	t.ratio	p.value
linear	29.096	10.0	70	2.903	0.0049
quadratic	-0.843	11.9	70	-0.071	0.9435
cubic	7.321	10.0	70	0.730	0.4676
quartic	-6.907	26.5	70	-0.260	0.7953



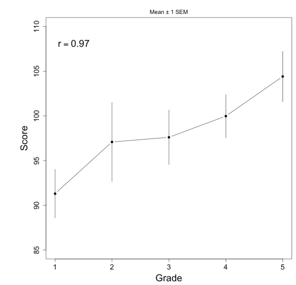
Trend Analysis Example

trends are linear contrasts

Can evaluate all higher-order, nonlinear trends with a single F test

```
> summary(aov.trends,
+         split=list(grade=list(Lin=1,NonLin=2:4)))
```

	Df	SS	MS	F	Pr(>F)
grade	4	1361	340.3	2.258	0.07152
grade: Lin	1	1270	1269.9	8.425	0.00495 **
grade: NonLin	3	91	30.5	0.202	0.89460
Residuals	70	10551	150.7		



effect size & association strength

Effect Size for a Linear Comparison

linear contrasts are used to compare two weighted means, so Cohen's d is approprate

Cohen's d (for a contrast)

$$d = 2\Psi / \left(\sigma_e \left[\sum_{j=1}^{a} |c_j| \right] \right)$$

$$d = 2\hat{\Psi} / \left(\sqrt{\mathrm{MS}_W} \left[\sum_{j=1}^{a} |c_j| \right] \right)$$

Expresses Ψ in terms of the number of standard deviations of population error distribution

Effect Size

Cohen's d calculation with emmeans & linear.comparison

```
> library(emmeans)
> aov.01 <- aov(score~grade,data=df3)

> sigma <- sigma(aov.01) # sqrt(MS.resid)
> edf <- df.residual(aov.01) # residual df
> aov.em <- emmeans(aov.01,specs="grade")
> myContrasts <- list(c1=myC1,c2=myC2,c3=myC3,c4=myC4)
> # calculate Cohen's d for each contrast:
> eff_size(aov.em,sigma,edf,method=myContrasts)
 contrast effect.size  SE  df  lower.CL  upper.CL
 c1           0.53   0.24  70    0.05     1.01
 c2           0.47   0.37  70   -0.26     1.20
 c3           0.37   0.32  70   -0.26     1.01
 c4           0.36   0.37  70   -0.37     1.09
```

Effect Size

Cohen's d calculation with emmeans & linear.comparison

```
> y <- df3$score
> g <- df3$grade
> myContrast1 <- linear.comparison(y,g,c.weights = myContrasts,var.equal=T)
[1] "computing linear comparisons assuming equal variances among groups"
[1] "C 1: F=4.995, t=2.235, p=0.029, psi=6.467, CI=(0.367,12.568), adj.CI= (-0.952,13.887)"
[1] "C 2: F=1.662, t=1.289, p=0.202, psi=5.780, CI=(-4.615,16.175), adj.CI= (-5.714,17.274)"
[1] "C 3: F=1.396, t=1.182, p=0.241, psi=4.588, CI=(-2.544,11.719), adj.CI= (-5.366,14.542)"
[1] "C 4: F=0.977, t=0.989, p=0.326, psi=4.432, CI=(-2.955,11.819), adj.CI= (-7.062,15.926)"
> myContrast1[[1]]$d.effect.size
[1] 0.53
> myContrast1[[2]]$d.effect.size
[1] 0.47
> myContrast1[[3]]$d.effect.size
[1] 0.37
> myContrast1[[4]]$d.effect.size
[1] 0.36
```

Note double brackets [[x]]!

Association Strength for a Linear Comparison

$$R^2_{alerting} = SS_{contrast}/SS_B$$

- Proportion of Between-Groups variation accounted for by contrast
- With equal n, equals squared correlation between contrast weights & group means

$$R^2_{effectsize} = SS_{contrast}/SS_{Total}$$

- Proportion of total variation accounted for by contrast

$$R^2_{contrast} = SS_{contrast}/(SS_{contrast} + SS_W)$$

- Variation accounted for by contrast relative to the sum of contrast-variation and within-group (error) variation
- Not affected by groups that are weighted zero
- More resistant to changes in experimental design (e.g., adding or removing groups).

Association Strength

linear.comparison

```
> str(myContrast1[[1]])
List of 15
 $ contrast     : num [1:5] -0.5 -0.5 0.333 0.333 0.333
 $ F            : num 4.99
 $ t            : num 2.23
 $ df1          : num 1
 $ df2          : int 70
 $ p.2tailed    : num 0.0286
 $ psi          : num 6.47
 $ confinterval : num [1:2] 0.367 12.568
 $ adj.confint  : num [1:2] -0.952 13.887
 $ alpha        : num 0.05
 $ SS.contrast  : num 753
 $ d.effect.size : num 0.527
 $ R2.alerting  : num 0.553
 $ R2.effect.size: num 0.0632
 $ R2.contrast  : num 0.0666
```

$$R^2_{alerting} = SS_{contrast}/SS_B$$

$$R^2_{effectsize} = SS_{contrast}/SS_{Total}$$

$$R^2_{contrast} = SS_{contrast}/(SS_{contrast} + SS_W)$$

unequal variances

Unequal Group Variances

- So far our tests assume equal variance in different groups
- F/t tests for contrasts are <u>not</u> robust to violation of equal variance assumption
- When groups have unequal variances, use a different method to calculate F/t denominator, which is an estimate of population error variance
- Correcting for unequal var reduces denominator df (and, hence, power)

$$F = \frac{(\Psi^2)/\sum_{j=1}^{a}(c_j/n_j)}{\sum_{j=1}^{a}\left[(c_j^2/n_j)s_j^2\right] / \sum_{j=1}^{a}(c_j^2/n_j)}$$

$$df = \frac{\left[\sum_{j=1}^{a}(c_j^2 s_j^2/n_j)\right]^2}{\sum_{j=1}^{a}\left[(c_j^2 s_j^2/n_j)^2/(n_j-1)\right]}$$

Contrasts with unequal variances

linear.comparison() can correct for unequal variances

$$F = \frac{(\Psi^2)/\sum_{j=1}^{a}(c_j/n_j)}{\sum_{j=1}^{a}\left[(c_j^2/n_j)s_j^2\right] / \sum_{j=1}^{a}(c_j^2/n_j)}$$

$$df = \frac{\left[\sum_{j=1}^{a}(c_j^2 s_j^2/n_j)\right]^2}{\sum_{j=1}^{a}\left[(c_j^2 s_j^2/n_j)^2/(n_j-1)\right]}$$

```
> myContrast2 <- linear.comparison(y,g,c.weights = myContrasts,var.equal=F)
[1] "computing linear comparisons assuming unequal variances among groups"
[1] "C 1: F=4.471, t=2.114, p=0.041, psi=6.467, CI=(0.289,12.646), adj.CI= (-1.526,14.461)"
[1] "C 2: F=1.230, t=1.109, p=0.279, psi=5.780, CI=(-4.996,16.556), adj.CI= (-8.331,19.892)"
[1] "C 3: F=1.646, t=1.283, p=0.211, psi=4.588, CI=(-2.785,11.960), adj.CI= (-5.053,14.228)"
[1] "C 4: F=1.432, t=1.197, p=0.242, psi=4.432, CI=(-3.164,12.028), adj.CI= (-5.473,14.337)"
```

orthogonal contrasts

Orthogonal Contrasts

Equal n:

$$\sum_{j=1}^{a}(c_{1j}c_{2j}) = 0$$

Unequal n:

$$\sum_{j=1}^{a}(c_{1j}c_{2j}/n_j) = 0$$

A set of contrasts is mutually orthogonal if all pairs of contrasts are orthogonal

Orthogonal contrasts evaluate independent questions about group means

Complete Set of Mutually Orthogonal Contrasts

If there are *a* groups, then the largest set of mutually orthogonal contrasts will have *(a-1)* contrasts, and:

$$\sum_{j=1}^{a-1}\mathrm{SS}_{contrast,j} = \mathrm{SS}_B$$

- A complete set of orthogonal contrasts divides SS_B into independent pieces of variation,
- the sum of the *(a-1)* $SS_{contrasts}$ will equal SS_B,
- and the average of the contrast F values will equal the omnibus F.

Complete set of orthogonal contrasts

breaks SS_{group} into separate pieces

```
> cMat <- contrasts(df3$grade)
> fractions(cMat)
  myC1 myC2 myC3 myC4
g1 -1/2   -1    0    0
g2 -1/2    1    0    0
g3  1/3    0   -1    0
g4  1/3    0  1/2   -1
g5  1/3    0  1/2    1
> # these contrasts/columns are mutually orthogonal:
> round(t(cMat) %*% cMat,digits=2)
     myC1 myC2 myC3 myC4
myC1 0.83    0  0.0    0
myC2 0.00    2  0.0    0
myC3 0.00    0  1.5    0
myC4 0.00    0  0.0    2
```

N.B. Each element in this matrix is the sum of cross-products.

Complete set of orthogonal contrasts

breaks SS_{group} into separate pieces

```
> cMat <- contrasts(df3$grade)
> fractions(cMat)
  myC1 myC2 myC3 myC4
g1 -1/2   -1    0    0
g2 -1/2    1    0    0
g3  1/3    0   -1    0
g4  1/3    0  1/2   -1
g5  1/3    0  1/2    1
> # these contrasts/columns are mutually orthogonal:
> round(t(cMat) %*% cMat,digits=2)
     myC1 myC2 myC3 myC4
myC1 0.83    0  0.0    0
myC2 0.00    2  0.0    0
myC3 0.00    0  1.5    0
myC4 0.00    0  0.0    2
```

```
> aov.10 <- aov(score~grade,data=df3)
> summary(aov.10,
+
split=list(grade=list(myC1=1,myC2=2,myC3=3,myC4=4)))
               Df    SS    MS     F Pr(>F)
grade           4  1361 340.3 2.258 0.0715
  grade: myC1   1   753 752.9 4.995 0.0286 *
  grade: myC2   1   251 250.6 1.662 0.2015
  grade: myC3   1   210 210.5 1.396 0.2413
  grade: myC4   1   147 147.3 0.977 0.3262
Residuals      70 10551 150.7
```

SS_{grade} = 1361 = 753+251+210+147

F_{grade} = 2.258 = (4.995+1.662+1.396+0.977) ÷ 4 multiple comparisons

Multiple Comparisons of Group Means

- Multiple comparisons inflate Type I error rate
- Generally want to control family-wise Type I error rate by adjusting the per-comparison Type I error rate
- for C = 100 comparisons
  - if α_{PC}=.00051, then $\alpha_{FW} \le .05$
- there are several methods for adjusting α_{PC}

$$P(\text{at least one Type I error}) = \alpha_{FW} = 1 - (1 - \alpha_{PC})^C$$

if α_{PC} = 0.05 and C = 100, then α_{FW} = 0.994

$$\alpha_{PC} = 1 - (1 - \alpha_{FW})^{1/C}$$

Controlling False Discovery Rate

- Instead of controlling α_{FW}, control **False Discovery Rate** (FDR):
  - Q = (# of false H0 rejections) / (total # H0 rejections)
  - FDR = Expected Value[Q]
- When all H0 are true, controlling α_{FW} and FDR are equivalent
- When some H0 are false, FDR-based methods are more powerful

Corrections for Multiple Comparisons

- Controlling α_{FW} by adjusting α_{PC}:
  - Bonferroni Adjustment (aka Dunn's Procedure)
  - Holm's Sequential Bonferroni Test
- Controlling False Discovery Rate (FDR):
  - Benjamini & Hochberg's (1995) Linear Step-Up Procedure (FDR)
- Relative Power: FDR > Holm's > Bonferroni

Multiple Comparisons in R

adjust p values with p.adjust()

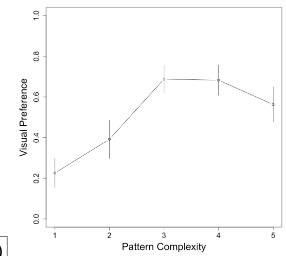
```
> my.p.values <- c(.127,.08,.03,.032,.02,.001,.01,.005,.025)
> sort(my.p.values)
[1] 0.001 0.005 0.010 0.020 0.025 0.030 0.032 0.080 0.127
> p.adjust(sort(my.p.values),method='bonferroni')
[1] 0.009 0.045 0.090 0.180 0.225 0.270 0.288 0.720 1.000
> p.adjust(sort(my.p.values),method='holm')
[1] 0.009 0.040 0.070 0.120 0.125 0.125 0.125 0.160 0.160
> p.adjust(sort(my.p.values),method='fdr')
[1] 0.009 0.0225 0.0300 0.04114 0.04114 0.04114 0.04114 0.090 0.127
```

Significant tests (alpha/FDR = .05) are highlighted in orange font.
N.B. Sorting p-values is not required.

Controlling Type I error rate

p.adjust()

```
> summary(aov.vp,split=list(complexity=list(L=1,Q=2,C=3,q4=4)))
                Df   SS      MS      F      Pr(>F)
complexity       4   1.2709  0.3177  6.214  0.000691 ***
  complexity: L  1   0.7441  0.7441  14.552 0.000532 ***
  complexity: Q  1   0.4357  0.4357  8.521  0.006100 **
  complexity: C  1   0.0477  0.0477  0.933  0.340714
  complexity: q4 1   0.0434  0.0434  0.848  0.363286
Residuals       35   1.7897  0.0511
```

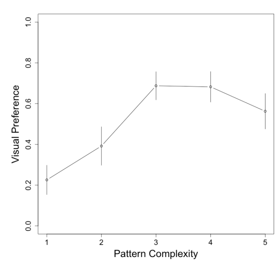



```
> p.adjust(p=c(0.000532,0.006100,0.340714,0.363286),method="bonferroni" )
[1] 0.002128 0.024400 1.000000 1.000000
> p.adjust(p=c(0.000532,0.006100,0.340714,0.363286),method="holm" )
[1] 0.002128 0.018300 0.681428 0.681428
> p.adjust(p=c(0.000532,0.006100,0.340714,0.363286),method="fdr" )
[1] 0.002128 0.012200 0.363286 0.363286
```

Controlling Type I error rate

emmeans

```
> aov.vp <- aov(visPref~complexity,data=df4)
> vp.em <- emmeans(aov.vp,specs="complexity")

> contrast(vp.em,method="poly",adjust="fdr")
 contrast  estimate  SE    df  t      p
 linear     0.964   0.253 35  3.815  0.0021
 quadratic -0.873   0.299 35 -2.919  0.0122
 cubic     -0.244   0.253 35 -0.966  0.3633
 quartic    0.616   0.669 35  0.921  0.3633

P value adjustment: fdr method for 4 tests
```



Setting family-wise alpha and FDR

- Generally, α_{FW} and FDR are set to 0.01 or 0.05
- larger α_{FW} may be justified for small number of orthogonal comparisons
  - Bonferroni & Holm tests may reduce power too much
  - perhaps set α_{PC} to 0.05 or 0.01
    - family-wise Type I error will increase but Type II error will decrease
  - Note: we do this with factorial ANOVA already…

All pairwise tests (Tukey HSD)

- Tukey HSD evaluates all pairwise differences between groups
- Is more powerful than Bonferroni method (for between-subj designs)
- Tukey HSD:
  - **NOT** necessary to evaluate omnibus F prior to Tukey test
  - assumes equal n per group & equal variances
  - Tukey-Kramer is valid with sample sizes are unequal
  - Dunnett's T3 test is better with unequal n & unequal variances
    [see Kirk (1995, pp. 146-50) for more details]

Tukey HSD (all pairwise differences)

optimal method for evaluating all pairwise differences

assumes equal variances

```
> TukeyHSD(aov.vp,which="complexity")
 Tukey multiple comparisons of means
  95% family-wise confidence level
Fit: aov(formula = visPref ~ complexity, data = df4)
$complexity
         diff    lwr   upr  p adj
p2-p1  0.1663 -0.159 0.49  0.59
p3-p1  0.4620  0.137 0.79  0.00
p4-p1  0.4569  0.132 0.78  0.00
p5-p1  0.3369  0.012 0.66  0.04
p3-p2  0.2957 -0.029 0.62  0.09
p4-p2  0.2906 -0.035 0.62  0.10
p5-p2  0.1706 -0.154 0.50  0.56
p4-p3 -0.0051 -0.330 0.32  1.00
p5-p3 -0.1250 -0.450 0.20  0.80
p5-p4 -0.1199 -0.445 0.21  0.83
```

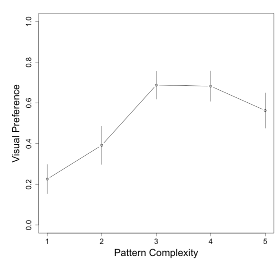
does not assume equal variances

```
> library(PMCMRplus)
> dunnettT3Test(x=df4$visPref,g=df4$complexity)

Pairwise comparisons using Dunnett's T3 test for
multiple comparisons with unequal variances

data: df4$visPref and df4$complexity

   p1     p2     p3     p4
p2 0.8081 -      -      -
p3 0.0037 0.1934 -      -
p4 0.0057 0.2340 1.0000 -
p5 0.0834 0.8475 0.9321 0.9563

P value adjustment method: single-step
alternative hypothesis: two.sided
```

Tukey HSD (all pairwise differences)

emmeans (assumes equal variances)

```
> vp.em <- emmeans(aov.vp,specs="complexity")
> contrast(vp.em,method="pairwise",adjust="tukey")
 contrast estimate   SE df t.ratio p.value
 p1 - p2    -0.17 0.113 35 -1.500  0.5900
 p1 - p3    -0.46 0.113 35 -4.100  <.0001
 p1 - p4    -0.46 0.113 35 -4.000  <.0001
 p1 - p5    -0.34 0.113 35 -3.000  0.0400
 p2 - p3    -0.30 0.113 35 -2.600  0.0900
 p2 - p4    -0.29 0.113 35 -2.600  0.1000
 p2 - p5    -0.17 0.113 35 -1.500  0.5600
 p3 - p4     0.01 0.113 35  0.000  1.0000
 p3 - p5     0.13 0.113 35  1.100  0.8000
 p4 - p5     0.12 0.113 35  1.100  0.8300

P value adjustment: tukey method for comparing a family of 5 estimates
```



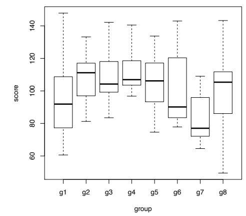
post-hoc comparisons

Scheffe method

Performing a Single Comparison

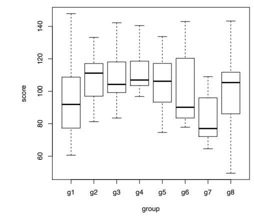
After plotting data, I decide to compare means of groups 4 & 7 using a t-test:

```
      Two Sample t-test

data:  y.4 and y.7
t = 4.165, df = 18, p-value = 0.0005813
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 13.84 42.01
sample estimates:
mean of x mean of y
   111.33     83.41
```



Performing a Single Comparison

Next I use a linear contrast which uses <u>all</u> groups to derive estimate of error variance:




```
> my.contrast<-list(c(0,0,0,1,0,0,-1,0) );
> c.4vs7 <- linear.comparison(y,g,c.weights=my.contrast )

[1] "computing linear comparisons assuming equal variances among groups"
[1] "C 1: F=9.915, t=3.149, p=0.002, psi=27.924, CI=(14.560,41.287), adj.CI= (10.245,45.602)"
```

What was wrong with the preceding analyses?

Answer: I performed the analyses after inspecting the data and choosing to compare groups 4 & 7 <u>because they looked different</u> which, obviously, inflates Type I error

Planned vs. Post-hoc Comparisons

- Previous comparisons were <u>planned</u>
- Last 2 comparisons, made after looking at data, were <u>post-hoc</u>
- Scheffe method is preferred for post-hoc linear contrasts
  - compute contrast with normal procedures
  - evaluate observed F with new critical value:
    - $F_{Scheffe}$ = (a-1) x $F_{\alpha(FW)}$ (df1= a-1; df2 = N-a)
    - a = number of groups
    - $F_{\alpha(FW)}$ is the F value required for desired alpha
  - $F_{Scheffe}$ is "normal" omnibus F x (a-1)
  - alternatively, keep standard F & adjust p values using Scheffe adjustment
- Scheffe method and omnibus F test are <u>**mutually consistent**</u>

Scheffe test

for post-hoc comparisons

```
> (con.poly <- contrast(vp.em,method="poly",adjust="none"))
 contrast  estimate    SE df t.ratio p.value
 linear       0.964 0.253 35   3.815  0.0005
 quadratic   -0.873 0.299 35  -2.919  0.0061
 cubic       -0.244 0.253 35  -0.966  0.3407
 quartic      0.616 0.669 35   0.921  0.3633

> summary(con.poly,adjust="scheffe",scheffe.rank=4)
 contrast  estimate    SE df t.ratio p.value
 linear       0.964 0.253 35   3.815  0.0140
 quadratic   -0.873 0.299 35  -2.919  0.0977
 cubic       -0.244 0.253 35  -0.966  0.9178
 quartic      0.616 0.669 35   0.921  0.9300

P value adjustment: scheffe method with rank 4
```

These methods compute normal F and adjust the p value to be consistent with Scheffe method

Scheffe.rank should be set to degrees of freedom for grouping factor (i.e., a-1)

Scheffe test

for post-hoc comparisons

```
> c1 <- c(-3, -3, 2, 2, 2)
> c2 <- c(-1,1,0,0,0)
> (con1 <- contrast(vp.em,method=list(c1,c2),adjust="none"))
 contrast            estimate    SE df t.ratio p.value
 c(-3, -3, 2, 2, 2)     2.013 0.438 35   4.596  0.0001
 c(-1, 1, 0, 0, 0)      0.166 0.113 35   1.471  0.1503

> summary(con1,adjust="scheffe",scheffe.rank=4)
 contrast            estimate    SE df t.ratio p.value
 c(-3, -3, 2, 2, 2)     2.013 0.438 35   4.596  0.0019
 c(-1, 1, 0, 0, 0)      0.166 0.113 35   1.471  0.7068

P value adjustment: scheffe method with rank 4
```

These methods compute normal F and adjust the p value to be consistent with Scheffe method

Scheffe.rank should be set to degrees of freedom for grouping factor (i.e., a-1)