STA305/1004 - Class 18

March 9, 2016

Today's Class

- Coding qualitative predictors in regression models
- Estimating treatment effects using least squares
- Multiple comparisons
- Sample size for ANOVA

Coding Qualitative Predictors in Regression Models

- A dummy or indicator variable in a regression takes on a finite number of values so that different categories of a nominal variable can be identified.
- The term dummy reflects the fact that the values taken on by such variables (e.g., 0, 1, -1) do not indicate meaningful measurements but rather categories of interest. (Kleinbaum et al., 1998)

Coding Qualitative Predictors in Regression Models

Consider a regression model: $y = \beta_0 + \beta_1 X_i + \epsilon$

Examples of dummy variables are:

$$X_1 = \begin{cases} 1 & \text{if treatment A} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if subject is male} \\ -1 & \text{if subject is female} \end{cases}$$

The variables X_1, X_2 are nominal variables describing treatment group and sex respectively.

In HW#3 (q 2?)
need to find correct
reference category

The following rule should be applied to avoid collinearity in defining a dummy variable for regression analysis:

> *if the nominal independent variable of interest has k categories then exactly $k-1$ dummy variables should be defined to index the categories if the regression model contains an intercept term.*

4 colours of smarties
then use 3 dummy variables
to index colours.

$X_1 = \begin{cases} 1 & \text{Blue} \\ 0 & \text{o.w.} \end{cases}$ $X_2 = \begin{cases} \text{pink} \\ \text{o.w.} \end{cases}$ $X_3 = \begin{cases} \text{green} \\ \text{o.w.} \end{cases}$ yellow as reference category

Dummy Coding

- Dummy coding compares each level to the reference level.
- The intercept is the mean of the reference group.
- Suppose that we would like to compare the mean number of candy colours in each box. The data from 3 smarties boxes are below.

```
count <- c(4,3,4,3,1,4,2,5,1,1,2,4)
colour <- as.factor(c(rep("Yellow",3),rep("Purple",3),
                      rep("Green",3),rep("Pink",3)))
```

colour	count
Yellow	4
Yellow	3
Yellow	4
Purple	3
Purple	1
Purple	4
Green	2
Green	5
Green	1
Pink	1
Pink	2
Pink	4

Dummy Coding

The average and sd of each colour is:

need this for HW #3

```
#Get means for each flavour
sapply(split(count,colour),mean)
```

$\bar{x}_{Green}$ $\bar{x}_{pink}$ $\bar{x}_{..}$ $\bar{x}_{..}$

```
##    Green     Pink   Purple   Yellow
## 2.666667 2.333333 2.666667 3.666667
```

```
#Get means for each flavour
sapply(split(count,colour),sd)
```

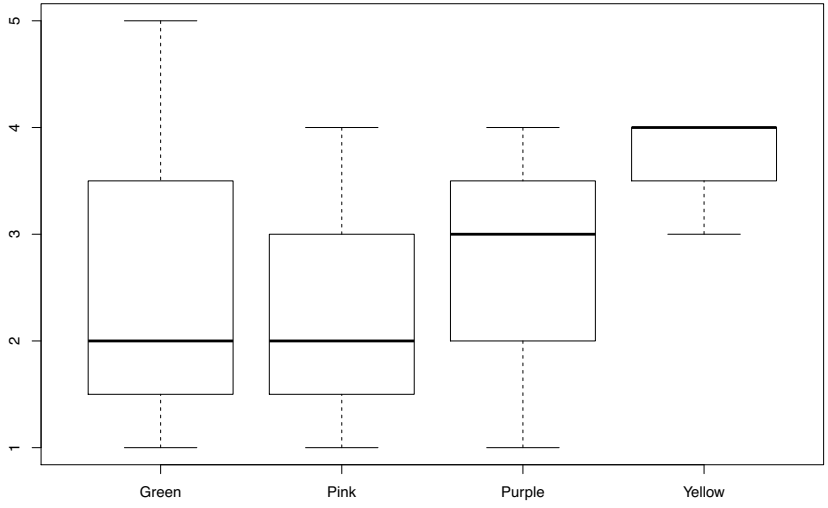
Sd_{Green} $\cdot -$

```
##    Green     Pink   Purple   Yellow
## 2.0816660 1.5275252 1.5275252 0.5773503
```

Dummy Coding

```
boxplot(count~colour)
```

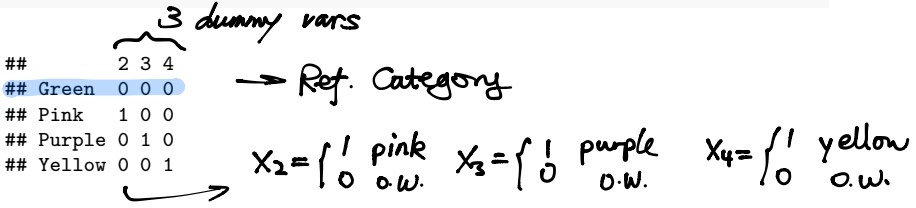
use also for HW#3 to compare dist'n



Dummy Coding

Dummy coding is the default in R and the most common coding scheme. It compares each level of the categorical variable to a fixed reference level.

```
contrasts(colour) <- contr.treatment(4)
contrasts(colour)  # print dummy coding - base is Green
```



Green is the reference category. The first column compares Pink to Green, the second column compares Purple to Green, and the third column compares Yellow to Green. The the three columns define three dummy variables:

Dummy Coding

$$X_1 = \begin{cases} 1 & \text{if smartie is pink} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if smartie is purple} \\ 0 & \text{otherwise} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

If $X_1 = X_2 = X_3 = 0$ then the colour of the smartie is green - the reference category. This shows that we only require 3 dummy variables to define a nominal variable with 4 categories.

Dummy Coding

To change the reference level change the value of base in `contr.treatment()`.

```
contrasts(colour) <- contr.treatment(4,base = 2) # Now reference is pink
contrasts(colour)
```

```
##        1 3 4
## Green  1 0 0
## Pink   0 0 0
## Purple 0 1 0
## Yellow 0 0 1
```

```
contrasts(colour) <- contr.treatment(4,base = 4) # Now reference is yellow
contrasts(colour)
```

```
##        1 2 3
## Green  1 0 0
## Pink   0 1 0
## Purple 0 0 1
## Yellow 0 0 0
```

Deviation Coding

- This coding system compares the mean of the dependent variable for a given level to the overall mean of the dependent variable.
-

$$X_1 = \begin{cases} 1 & \text{if smartie is green} \\ -1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if smartie is pink} \\ -1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{if smartie is purple} \\ -1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

set to -1

- 1 is used to compare a level to all other levels and -1 is assigned to yellow because it's the level that will never be compared to the other levels.
- In R the variables can be created using the `contr.sum()` function.
- The argument of 4 in `contr.sum(4)` indicates the number of levels of the factor.

Deviation Coding

```
contrasts(colour) <- contr.sum(4)
contrasts(colour)
```

```
##        [,1] [,2] [,3]
## Green     1    0    0
## Pink      0    1    0
## Purple    0    0    1
## Yellow   -1   -1   -1
```

X_1 X_2 X_3

defined on the previous slide

Estimating treatment effects using least squares

y_{ij} is the j^{th} observation under the i^{th} treatment. Let μ be the overall mean. The model for diet $y_{ij} = \mu + \tau_i + \epsilon_{ij}$, $\epsilon_{ij} \sim N(0, \sigma^2)$ can be written in terms of the dummy variables X_1, X_2, X_3 as:

$$y_{ij} = \mu + \tau_1 X_{1j} + \tau_2 X_{2j} + \tau_3 X_{3j} + \epsilon_{ij},$$

where,

$$X_{1j} = \begin{cases} 1 & \text{if jth unit recieves diet 2} \\ 0 & \text{otherwise} \end{cases}$$

$$X_{2j} = \begin{cases} 1 & \text{if jth unit recieves diet 3} \\ 0 & \text{otherwise} \end{cases}$$

$$X_{3j} = \begin{cases} 1 & \text{if jth unit recieves diet 4} \\ 0 & \text{otherwise} \end{cases}$$

Estimating treatment effects using least squares

It follows that $E(y_{Aj}) = \mu_A = \mu$ is the mean of diet A so

$$E(y_{Bj}) = \mu_B = \mu_A + \tau_1 \Rightarrow \tau_1 = \mu_B - \mu_A$$
$$E(y_{Cj}) = \mu_C = \mu_A + \tau_2 \Rightarrow \tau_2 = \mu_C - \mu_A$$
$$E(y_{Dj}) = \mu_D = \mu_A + \tau_3 \Rightarrow \tau_3 = \mu_D - \mu_A$$

treatment deviation from diet A

The least squares estimates are:

$$\hat{\mu} = \bar{y}_{1\cdot},$$
$$\hat{\tau}_1 = \bar{y}_{2\cdot} - \bar{y}_{1\cdot},$$
$$\hat{\tau}_2 = \bar{y}_{3\cdot} - \bar{y}_{1\cdot},$$
$$\hat{\tau}_3 = \bar{y}_{3\cdot} - \bar{y}_{1\cdot}.$$

Estimating treatment effects using least squares

- This model can also be written in matrix notation

$$y = X\beta + \epsilon$$

column corresponding to X_{i1}

column of 1s

,

where $\beta = (\mu, \tau_1, \tau_2, \tau 3)$, $X = (\mathbf{1}, X_{i1}, X_{i2}, X_{i3})$, and $\epsilon = (\epsilon_{ij})$.

- ~~X is an 30×4 design matrix with $\mathbf{1}$ is a 30×1 column vector of 1s, and ϵ is an 30×1 column vector.~~
- ~~Note that τ_4 corresponding to the 4th treatment is implicitly set to 0. It is used as a constraint so that that $(X'X)^{-1}$ exists.~~

Example - blood coagualtion study

The table below gives coagulation times for blood samples drawn from 24 animals receiving four different diets A, B, C, and D.

	A	B	C	D
	60	65	71	62
	63	66	66	60
	59	67	68	61
	63	63	68	64
	62	64	67	63
	59	71	68	56
Treatment Average	61	66	68	61
Grand Average	64	64	64	64
Difference	-3	2	4	-3

Example - blood coagualtion study (treatment coding)

treatment coding scheme

```
contrasts(tab0401$diets)
```

	B	C	D
A	0	0	0
B	1	0	0
C	0	1	0
D	0	0	1

diet A is ref. category

- If a different coding scheme is chosen, then different H_0, H_1

coagulation time

```
lm.diets <- lm(y~diets,data=tab0401);round(summary(lm.diets)$coefficients,2)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	61	0.97	63.14	0
dietsB	5	1.37	3.66	0
dietsC	7	1.37	5.12	0
dietsD	0	1.37	0.00	1

p-value of the test

→ $H_0: \mu_B - \mu_A = 0$ vs. $H_1: \mu_B - \mu_A \neq 0$

Example - blood coagualtion study (treatment coding)

The averages for each of the four diets are in the table below.

Diet	A ($j = 1$)	B ($j = 2$)	C ($j = 3$)	D ($j = 4$)
Average ($\bar{y}_{j\cdot}$)	61	66	68	61

$\bar{y}_{1\cdot} = 61,$

$\hat{\tau}_1 = \bar{y}_{2\cdot} - \bar{y}_{1\cdot} = 5$ $66 - 61 = 5$

$\hat{\tau}_2 = \bar{y}_{3\cdot} - \bar{y}_{1\cdot} = 7$ $68 - 61 = 7$

$\hat{\tau}_3 = \bar{y}_{3\cdot} - \bar{y}_{1\cdot} = -9.9 \times 10^{-15}.$ $61 - 61 \doteq 0$

Example - blood coagualtion study (treatment coding)

The design matrix (first 12 observations) is

```
model.matrix(lm.diets)[1:12,]
```

```
##    (Intercept) dietsB dietsC dietsD
## 1            1      0      0      0
## 2            1      0      0      0
## 3            1      0      0      0
## 4            1      0      0      0
## 5            1      0      0      0
## 6            1      0      0      0
## 7            1      1      0      0
## 8            1      1      0      0
## 9            1      1      0      0
## 10           1      1      0      0
## 11           1      1      0      0
## 12           1      1      0      0
```

$$\underset{\sim}{Y} = \underset{\sim}{X}\underset{\sim}{\beta} + \underset{\sim}{\varepsilon}$$

Example - blood coagualtion study (treatment coding)

The design matrix (first 12 observations) with the observations y and treatment variable `diets` (first 12 observations) is

```
cbind(tab0401$y,tab0401$diets,model.matrix(lm.diets))[1:12,]
```

```
##    y diet (Intercept) dietsB dietsC dietsD
## 1 62    1           1      0      0      0
## 2 60    1           1      0      0      0
## 3 63    1           1      0      0      0
## 4 59    1           1      0      0      0
## 5 63    1           1      0      0      0
## 6 59    1           1      0      0      0
## 7 63    2           1      1      0      0
## 8 67    2           1      1      0      0
## 9 71    2           1      1      0      0
## 10 64   2           1      1      0      0
## 11 65   2           1      1      0      0
## 12 66   2           1      1      0      0
```

Example - blood coagualtion study (deviation coding)

If deviation coding was used then the parameter estimates would represent different treatment effects. In the regression model the dummy variables would be defined as

$$X_1 = \begin{cases} 1 & \text{if diet is A} \\ -1 & \text{if diet is D} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if diet is B} \\ -1 & \text{if diet is D} \\ 0 & \text{otherwise} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{if diet is C} \\ -1 & \text{if diet is D} \\ 0 & \text{otherwise} \end{cases}$$

Example - blood coagualtion study (deviation coding)

It follows that

$$
\begin{aligned}
E(y_{Aj}) &= \mu_A = \tau_0 + \tau_1 \\
E(y_{Bj}) &= \mu_B = \tau_0 + \tau_2 \\
E(y_{Cj}) &= \mu_C = \tau_0 + \tau_3 \\
E(y_{Dj}) &= \mu_D = \tau_0 - \tau_1 - \tau_2 - \tau_3
\end{aligned}
$$

So,

$$
\begin{aligned}
\tau_0 &= \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4} \\
\tau_1 &= \mu_A - \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4} \\
\tau_2 &= \mu_B - \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4} \\
\tau_3 &= \mu_C - \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4}
\end{aligned}
$$

Example - blood coagualtion study (deviation coding)

```
attach(tab0401)
contrasts(tab0401$diets) <- contr.sum(4)
lm.diets <- lm(y~diets,data=tab0401)
round(summary(lm.diets)$coefficients,2)
```

$H_0: \mu_A - \frac{\mu_A+\mu_B+\mu_C+\mu_D}{4} = 0$

$\bar{y}_{..}$

$\bar{\mu}$

$H_1: \mu_A - \bar{\mu} \neq 0$

$p\text{-value} \approx 0.00$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	64	0.48	132.49	0.00
diets1	-3	0.84	-3.59	0.00
diets2	2	0.84	2.39	0.03
diets3	4	0.84	4.78	0.00

$61 - 64 = -3$

$\bar{y}_{A.} - \bar{y}_{..}$

- The estimate of the intercept $\hat{\tau}_0$ is the grand average.
- The slope estimates $\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3$ are the differences between the treatment averages and grand average of diets A, B, C, D.

Multiple Comparisons

Suppose that experimental units were randomly assigned to three treatment groups. The hypothesis of intrest is:

$$H_0 : \mu_1 = \mu_2 = \mu_3 \text{ vs. } H_1 : \mu_i \neq \mu_j.$$

Now, suppose that we reject H_0 at level α. Which pairs of means are significantly different from each other at level α? There are $\binom{3}{2} = 3$ possibilites.

1. $\mu_1 \neq \mu_2$
2. $\mu_1 \neq \mu_3$
3. $\mu_2 \neq \mu_3$

Multiple Comparisons

Suppose that $k = 3$ separate (independent) hypothesis level α tests are conducted

$$H_{0_k} : \mu_i = \mu_j \text{ vs. } H_{1_k} : \mu_i \neq \mu_j,$$

All pairwise comparisons

When H_0 is true, $P(\text{reject } H_0) = \alpha \Rightarrow P(\text{do not reject } H_0) = 1 - \alpha$.

So, if H_0 is true then

P(Reject at least one hypothesis when H_0 is true)

$$P\left(\text{reject at least one } H_{0_k}\right) = 1 - P\left(\text{do not reject any } H_{0_k}\right)$$

This is the same as

$$1 - P(\text{do not reject } H_{0_1} \text{and do not reject } H_{0_2} \text{and do not reject } H_{0_3})$$

or since the hypotheses are independent

P(reject at least one H_0)

$$1 - P(\text{do not reject } H_{0_1})\, P(\text{do not reject } H_{0_2})\, P(\text{do not reject } H_{0_3}) = 1 - (1 - \alpha)^3$$

If $\alpha = 0.05$ then the probability that at least one H_0 will be falsely rejected is $1 - (1 - .05)^3 = 0.14$, which is almost three times the type I error rate.

Multiple Comparisons

In general if

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_k \text{ vs. } H_1 : \mu_i \neq \mu_j.$$

If c independent hypotheses are conducted then the probability

$$P\left(\text{reject at least one } H_{0_k}\right) = 1 - (1 - \alpha)^c$$

is called the **family-wise error rate**.

The **pairwise error rate** is $P\left(\text{reject } H_{0_k}\right) = \alpha$ for any c.

$X_1, X_2, \cdots, X_n$ independent Bern (p). $P(X_i=1)=p$, $P(X_i=0)=1-p$

$X_i = \begin{cases} 1 \\ 0 \end{cases}$ independent

Pr(at least one $X_i=1$) $= P(X_1=1 \text{ or } X_2=1 \text{ or} \cdots \text{or} X_n=1) = 1-P(X_1=0 \text{ and } X_2=0 \text{ and} \cdots X_n=0)$

$= 1-(1-p)^n$

$X_i = \begin{cases} 1 & \text{Reject } H_{0_i} \\ 0 & \text{do not reject } H_{0_i} \end{cases}$

if H_0 is true then, ...

---

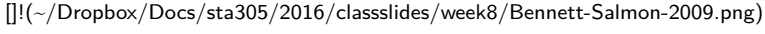
4 treatments, 6 comparisons

$1-(1-0.05)^6 = .26 > .05$

Multiple Comparisons

- The multiple comparison problem is that multiple hypotheses are tested level α which increases the probability that at least one of the hypotheses will be falsely rejected (family-wise error rate).
- When groups are significantly different from ANOVA researchers often wish to explore where the differences lie. Is it appropriate to test for differences looking at all pairwise comparisons?
- Testing all possible pairs increases the type I error rate.
- This means the chance that there is a higher probability, beyond the pre-stated type I error rate (e.g. 0.05), that that a significant difference is detected when the truth is that no difference exists.

Example


[]!(~/Dropbox/Docs/sta305/2016/classslides/week8/Bennett-Salmon-2009.png)

The Bonferroni Method

To test for the difference between the *ith* and *jth* treatments, it is common to use the two-sample t test. The two-sample t statistic is

$$t_{ij} = \frac{\bar{y_j}. - \bar{y_i}.}{\hat{\sigma}\sqrt{1/n_j + 1/n_i}},$$

where $\bar{y_j}.$ is the average of the n_i observations for treatment j and $\hat{\sigma}$ is $\sqrt{MS_E}$ from the ANOVA table.

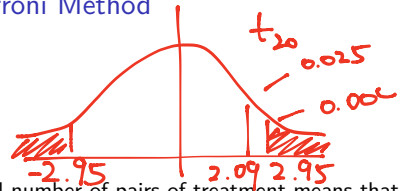
Treatments i and j are declared significantly different at level α if

$$|t_{ij}| > t_{N-k,\alpha/2},$$

– t-critical value

where $t_{N-k,\alpha/2}$ is the upper $\alpha/2$ percentile of a t_{N-k}.

The Bonferroni Method



The total number of pairs of treatment means that can be tested is

$$c = \binom{k}{2} = \frac{k(k-1)}{2}.$$

The Bonferroni method for testing $H_0 : \mu_i = \mu_j$ vs. $H_0 : \mu_i \neq \mu_j$ rejects H_0 at level α if

$$|t_{ij}| > t_{N-k,\alpha/2c},$$

where c denotes the number of pairs being tested.

$\alpha/2 \cdot c$

$C = \#$ of comparisons

$N-K=20 \quad K=4$

$\alpha = 0.05$

$0.05/2 = 0.025$

$C = 6 = \binom{4}{2} \Rightarrow \frac{.05}{2.6} = .004$

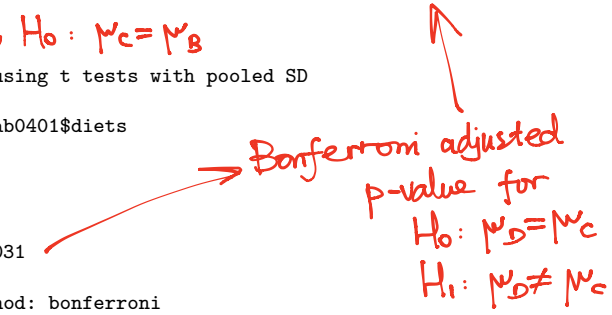
The Bonferroni Method

In R the function `pairwise.t.test()` can be used to compute Bonferroni adjusted p-values.

This is illustrated below for the blood coagualtion study.

```
pairwise.t.test(tab0401$y,tab0401$diets,p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  tab0401$y and tab0401$diets
##
##   A       B       C
## B 0.00934 -       -
## C 0.00031 0.95266 -
## D 1.00000 0.00934 0.00031
##
## P value adjustment method: bonferroni
```



There are signifciant differences at the 5% level between diets A and B, A and C, B and D, and C and D using the Bonferroni method.

The Bonferroni Method

For comparison the unadjusted p-values are also calculated.

```
pairwise.t.test(tab0401$y,tab0401$diets,p.adjust.method = "none")
```

means unadjusted t-tests

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  tab0401$y and tab0401$diets
##
##   A       B      C
## B 0.0016  -      -
## C 5.2e-05 0.1588 -
## D 1.0000  0.0016 5.2e-05
##
## P value adjustment method: none
```

The significant differences are the same using the unadjusted p-values but the p-values are ~~larger~~ smaller then the p-values adjusted using the Bonferroni method.

The Bonferroni Method

A $100(1-\alpha)\%$ simultaneous confidence interval for c pairs $\mu_i - \mu_j$ is

$$\bar{y_j}. - \bar{y_i}. \pm t_{N-k,\alpha/2c}\hat{\sigma}\sqrt{1/n_j + 1/n_i}.$$

After identifying which pairs are different, the confidence interval quantifies the range of plausible values for the differences.

The Bonferroni Method - coagulation study

The treatment means can be obtained from the table below.

	A	B	C	D
	60	65	71	62
	63	66	66	60
	59	67	68	61
	63	63	68	64
	62	64	67	63
	59	71	68	56
Treatment Average	61	66	68	61
Grand Average	64	64	64	64
Difference	-3	2	4	-3

The Bonferroni Method - coagulation study

$\hat{\sigma} = \sqrt{MS_E}$ can be obtained from the ANOVA table.

```
anova(lm(y~diets,data=tab0401))
```

ANOVA Command gets ANOVA table from regression model

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## diets      3    228    76.0  13.571 4.658e-05 ***
## Residuals 20    112     5.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The upper $.05/(2 \cdot 6) = 0.004$ percentile of the t_{24-4} can be obtained with the t quantile function in R `qt()`.

```
qt(p = 1-0.004,df = 20)
```

quantile function for t_{20}

```
## [1] 2.945349
```

inverse CDF

Plugging in these values to the confidence interval formula we can obtain a Bonferroni adjusted 95% confidence interval for $\mu_B - \mu_A$:

$$66 - 61 \pm 2.95\sqrt{5.6}\sqrt{1/6 + 1/6}$$

The lower and upper limits can be calculated in R.

```
66-61 - qt(p = 1-0.004,df = 20)*sqrt(5.6)*sqrt(1/6+1/6) # lower limit
```

```
## [1] 0.9758869
```

```
66-61 + qt(p = 1-0.004,df = 20)*sqrt(5.6)*sqrt(1/6+1/6) # upper limit
```

```
## [1] 9.024113
```

The 95% confidence interval for $\mu_B - \mu_A$ is (0.98, 9.02).

does not contain 0

$H_0: \mu_A - \mu_B = 0$

$H_1: \mu_A - \mu_B \neq 0$

$\therefore$ reject at $\alpha = 0.05$

The Tukey Method

- The only difference between the Tukey and Bonferroni methods is in the choice of the critical value.
- Treatments i and j are declared significantly different at level α if

$$|t_{ij}| > \frac{1}{\sqrt{2}} q_{k,N-k,\alpha},$$

- t_{ij} is the observed value of the two-sample t-statistic
- $q_{k,N-k,\alpha}$ is the upper α percentile of the Studentized range distribution with parameters k and $N-k$ degrees of freedom.
- The CDF and inverse CDF of the Studentized Range Distribution is available in R via the functions `ptukey()` and `qtukey()` respectively.

it's the distribution of $\frac{\bar{y}_{max} - \bar{y}_{min}}{\hat{\sigma}}$ depends on df in and # treatments

The Tukey Method

A $100(1-\alpha)\%$ simultaneous confidence interval for c pairs $\mu_i - \mu_j$ is

$$\bar{y_{j\cdot}} - \bar{y_{i\cdot}} \pm \frac{1}{\sqrt{2}} q_{k,N-k,\alpha} \hat{\sigma} \sqrt{1/n_j + 1/n_i}.$$

$\sqrt{MS_E}$

# of means

df for MS_E

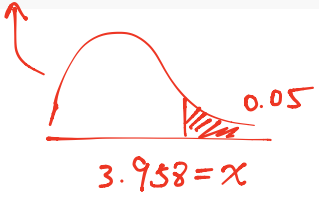
The Bonferroni method is more conservative than Tukey's method. In other words, the simutaneous confidence intervals based on the Tukey method are shorter.

The Tukey Method

- In the coagualtion study $N = 24, k = 4$ so the 5% critical value of the Studentized range distribution is obtained using the the inverse CDF function `qtukey()` for this distribution.
- The argument `lower.tail=FALSE` is used so we obtain the upper percentile of the distribution (i.e., the value of x such that $P(X > x) = 0.05$).

```
qtukey(p = .05,nmeans = 4,df = 20,lower.tail = FALSE)
```

```
## [1] 3.958293
```



The Tukey Method

- Let's obtain the Tukey p-value and confidence interval for $\mu_B - \mu_A$.
- The observed value of the test statistic is

$$q^{obs} = \sqrt{2}|t_{AB}|,$$

where

$$t_{AB} = \frac{\bar{y_{A\cdot}} - \bar{y_{B\cdot}}}{\hat{\sigma}\sqrt{1/n_A + 1/n_B}}.$$

```
(sqrt(2)*(66-61))/(sqrt(5.6)*sqrt(1/6+1/6))
```

```
## [1] 5.175492
```

The Tukey Method

The p-value

$$P\left(q_{4,20} > q^{obs}\right)$$

is then obtained using the CDF of the Studentized range distribution

```
1-ptukey(q = sqrt(2)*5/sqrt(2*5.6/6),nmeans = 4,df = 20)
```

```
## [1] 0.007797788
```

The Tukey Method

The 95% limits of the Tukey confidence interval for $\mu_B - \mu_A$ is

critical value

```
tuk.crit <- qtukey(p=.05,nmeans=4,df=20,lower.tail=FALSE)
#lower limit
round(5-(1/sqrt(2))*tuk.crit*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 1.18
```

lower

```
#upper limit
round(5+(1/sqrt(2))*tuk.crit*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 8.82
```

upper

The Tukey Method

The width of the Tukey confidence interval for $\mu_B - \mu_A$ is

```
round((1/sqrt(2))*tuk.crit*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 3.82
```

margin of error

The width of Bonferroni $\mu_B - \mu_A$ is

```
round(qt(p = 1-0.004,df = 20)*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 4.02
```

The Tukey Method

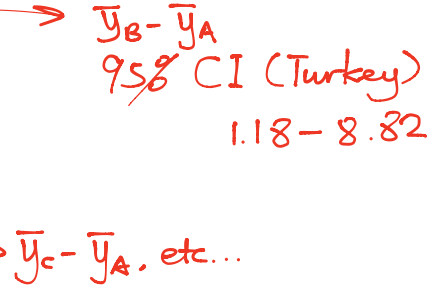
- This shows that the Tukey confidence interval is shorter than Bonferroni confidence intervals.
- The command `TukeyHSD()` can be used to obtain all the Tukey confidence intervals and p-values for an ANOVA.

The Tukey Method

```
TukeyHSD(aov(y~diets,data=tab0401))
```

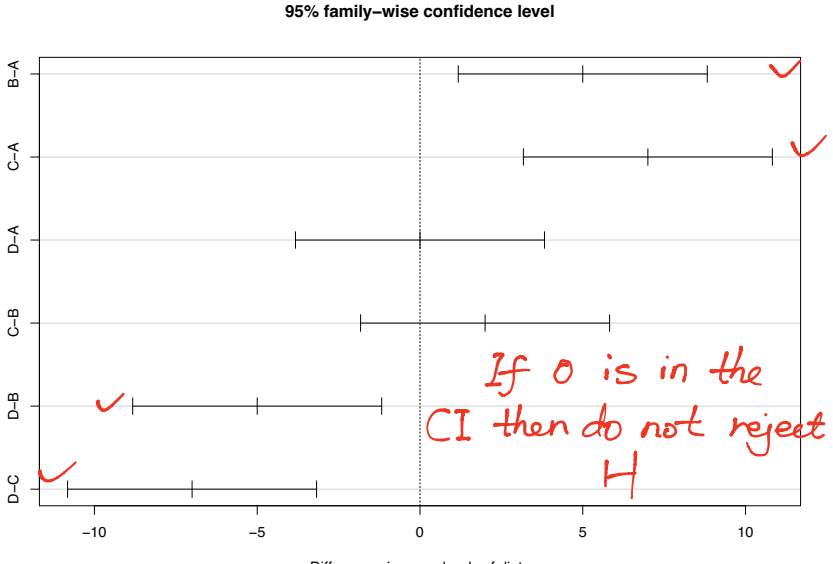
```
round(TukeyHSD(aov(y~diets,data=tab0401))$diets,2)
```

```
##      diff   lwr   upr p adj
## B-A     5  1.18  8.82  0.01
## C-A     7  3.18 10.82  0.00
## D-A     0 -3.82  3.82  1.00
## C-B     2 -1.82  5.82  0.48
## D-B    -5 -8.82 -1.18  0.01
## D-C    -7 -10.82 -3.18  0.00
```



The Tukey Method

```
plot(TukeyHSD(aov(y~diets,data=tab0401)))
```



Sample size for ANOVA - Designing a study to compare more than two treatments

- Consider the hypothesis that k means are equal vs. the alternative that at least two differ.
- What is the probability that the test rejects if at least two means differ?
- Power $= 1 - P$(Type II error) is this probability.

Sample size for ANOVA - Designing a study to compare more than two treatments

The null and alternative hypotheses are:

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_k \text{ vs. } H_1 : \mu_i \neq \mu_j.$$

The test rejects at level α if

$$MS_{Treat}/MS_E \geq F_{k-1,N-K,\alpha}.$$

The power of the test is

$$1 - \beta = P\left(MS_{Treat}/MS_E \geq F_{k-1,N-K,\alpha}\right),$$

when H_0 is false.

Sample size for ANOVA - Designing a study to compare more than two treatments

When H_0 is false it can be shown that:

- MS_{Treat}/σ^2 has a non-central Chi-square distribution with $k-1$ degrees of freedom and non-centrality parameter δ.
- MS_{Treat}/MS_E has a non-central F distribution with the numerator and denominator degrees of freedom $k-1$ and $N-k$ respectively, and non-centrality parameter

$$\delta = \frac{\sum_{i=1}^{k} n_i (\mu_i - \bar{\mu})^2}{\sigma^2},$$

where n_i is the number of observations in group i, $\bar{\mu} = \sum_{i=1}^{k} \mu_i/k$, and σ^2 is the within group error variance .

This is dentoted by $F_{k-1,N-k}(\delta)$.

underlying data is normally distributed

Direct calculation of Power

To calc. power regular..?

- The power of the test is

$$P\left(F_{k-1,N-k}(\delta) > F_{k-1,N-K,\alpha}\right).$$

- The power is an increasing function δ
- The power depends on the true values of the treatment means μ_i, the error variance σ^2, and sample size n_i.
- If the experimentor has some prior idea about the treament means and error variance the sample size (number of replications) that will guaruntee a pre-assigned power of the test.

Blood coagulation example - sample size

Suppose that an investigator would like to replcate the blood coagulation study with only 3 animals per diet. In this case $k = 4, n_i = 3$. The treatment means from the initial study are:

Diet	A	B	C	D
Average	61	66	68	61

μ_1 μ_2 μ_3 μ_4

```
anova(lm.diets)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## diets      3    228    76.0  13.571 4.658e-05 ***
## Residuals 20    112     5.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

END

Blood coagulation example - sample size

- $\mu_1 = 61$, $\mu_2 = 66$, $\mu_3 = 68$, $\mu_4 = 61$.
- The error variance σ^2 was estimated as $MS_E = 5.6$.
- Assuming that the estimated values are the true values of the parameters, the non-centrality parameter of the F distribution is

$$\delta = 3 \times \left((61-64)^2 + (66-64)^2 + (68-64)^2 + (61-64)^2\right)/5.6 = 20.35714$$

Blood coagulation example - sample size

If we choose $\alpha = 0.05$ as the significance level then $F_{3,20,0.05} = 3.0983912$. The power of the test is then

$$P\left(F_{3,20}(20.36) > 3.10\right) = 0.94.$$

This was calculated using the CDF for the F distribution in R `pf()`.

```
1-pf(q = 3.10,df1 = 3,df2 = 20,ncp = 20.36)
```

```
## [1] 0.9435208
```

Calculating power and sample size using the pwr library

- There are several libraries in R which can calculate power and sample size for statistical tests. The library `pwr()` has a function
- `pwr.anova.test(k = NULL, n = NULL, f = NULL, sig.level = 0.05, power = NULL)`

for computing power and sample size.

- k Number of groups
- n Number of observations (per group)
- f Effect size
- The effect size is the square root of the non-centrality parameter of the non-central F distribution.

$$f = \sqrt{\frac{\sum_{i=1}^{k} n_i \left(\mu_i - \bar{\mu}\right)^2}{\sigma^2}}.$$

where n_i is the number of observations in group i, $\bar{\mu} = \sum_{i=1}^{k} \mu_i / k$, and σ^2 is the within group error variance.

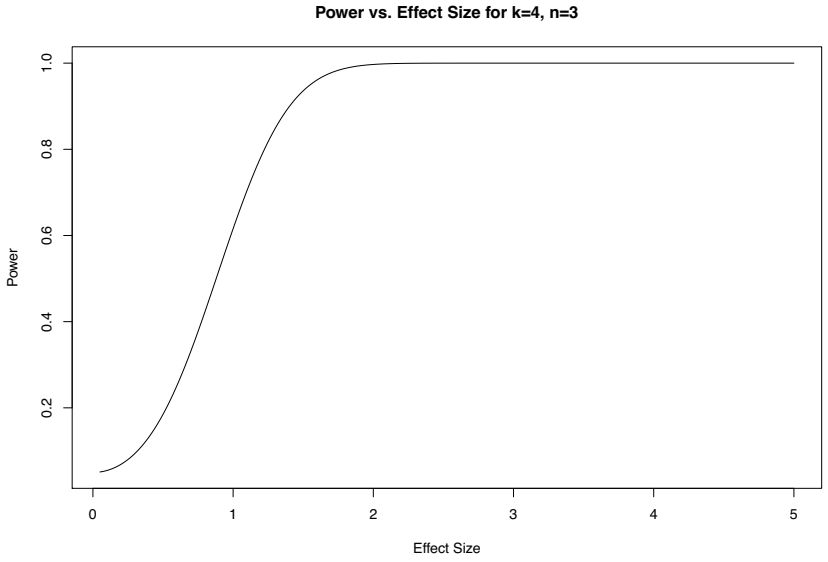
Calculating power and sample size using the pwr library

In the previous example $\delta = 20.35714$ so $f = \sqrt{20.35714} = 4.5118887$.

```
library(pwr)
pwr.anova.test(k = 4,n = 3,f = 4.5)
```

```
##
##      Balanced one-way analysis of variance power calculation
##
##               k = 4
##               n = 3
##               f = 4.5
##       sig.level = 0.05
##           power = 1
##
## NOTE: n is number in each group
```

Calculating power and sample size using the pwr library



Calculating power using simulation

The general procedure for simulating power is:

1. Use the underlying model to generate random data with (a) specified sample sizes, (b) parameter values that one is trying to detect with the hypothesis test, and (c) nuisance parameters such as variances.
2. Run the estimation program (e.g., `t.test()`,`lm()`) on these randomly generated data.
3. Calculate the test statistic and p-value.
4. Do Steps 1–3 many times, say, N, and save the p-values. The estimated power for a level alpha test is the proportion of observations (out of N) for which the p-value is less than alpha.

Calculating power using simulation

One of the advantages of calculating power via simulation is that we can investigate what happens to power if, say, some of the assumptions behind one-way ANOVA are violated.

Calculating power using simulation - R program

```r
#Simulate power of ANOVA for three groups

NSIM <- 1000 # number of simulations
res <- numeric(NSIM) # store p-values in res

mu1 <- 2; mu2 <- 2.5;mu3 <- 2  # true mean values of treatment groups
sigma1 <- 1; sigma2 <- 1; sigma3 <- 1 #variances in each group
n1 <- 40; n2 <- 40; n3 <- 40 #sample size in each group

for (i in 1:NSIM) # do the calculations below N times
  {
# generate sample of size n1 from N(mu1,sigma1^2)
y1 <- rnorm(n = n1,mean = mu1,sd = sigma1)
# generate sample of size n2 from N(mu2,sigma2^2)
y2 <- rnorm(n = n2,mean = mu2,sd = sigma2)
# generate sample of size n3 from N(mu3,sigma3^2)
y3 <- rnorm(n = n3,mean = mu3,sd = sigma3)
y <- c(y1,y2,y3) # store all the values from the groups
# generate the treatment assignment for each group
trt <- as.factor(c(rep(1,n1),rep(2,n2),rep(3,n3)))
m <- lm(y~trt) # calculate the ANOVA
res[i] <- anova(m)[1,5] # p-value of F test
}
sum(res<=0.05)/NSIM # calculate p-value
```

```
## [1] 0.607
```