STA305/1004 Class Notes - Week 7

Nathan Taback

March 6, 2016

- 1 Coding Qualitative Predictors in Regression Models
    - 1.1 Dummy Coding
    - 1.2 Deviation Coding
- 2 Estimating treatment effects using least squares
    - 2.1 Using least squares to estimate the parameters
- 3 Multiple Comparisons
    - 3.1 The Bonferroni Method
    - 3.2 The Tukey Method
- 4 Sample size for ANOVA - Designing a study to compare more than two treatments
    - 4.1 Direct calculation of Power
    - 4.2 Example
    - 4.3 Calculating power and sample size using the pwr library
    - 4.4 Calculating power using simulation
- 5 Questions
- 6 Answers to questions

1 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)

Examples of dummy variables are:

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

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

The variables X and Y are nominal variables describing treatment group and sex respectively.

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.

1.1 Dummy Coding

Dummy coding compares each level to the reference level. The intercept is the mean of the reference group.

Smarties is a candy that comes in several different colours such as yellow, purple, green, and pink. 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

The average number of candies in each colour is:

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

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

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
```

```
##        2 3 4
## Green  0 0 0
## Pink   1 0 0
## Purple 0 1 0
## Yellow 0 0 1
```

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:

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

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

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

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.

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 = 3) # Now reference is purple
contrasts(colour)
```

```
##        1 2 4
## Green  1 0 0
## Pink   0 1 0
## Purple 0 0 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
```

1.2 Deviation Coding

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

The the three columns define three dummy variables:

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

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

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

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.

```
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
```

2 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:

$$y_{ij} = \mu + \tau_1 X_{i1} + \tau_2 X_{i2} + \tau_3 X_{i3} + \epsilon_{ij},$$

where,

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

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

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

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

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

The least squares estimates are:

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

This model can also be written in matrix notation $y = X\beta + \epsilon$, where $\beta = \left(\mu, \tau_1, \tau_2, \tau 3\right), X = (\mathbf{1}, X_{i1}, X_{i2}, X_{i3})$, 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.

2.1 Using least squares to estimate the parameters

Let's return to the blood coagulation 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

```
attach(tab0401)
contrasts(diets)
```

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

```
lm.diets <- lm(y~diets,data=tab0401)
summary(lm.diets)
```

```


Call:
lm(formula = y ~ diets, data = tab0401)

Residuals:
   Min     1Q Median     3Q    Max 
 -5.00  -1.25   0.00   1.25   5.00 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.100e+01  9.661e-01  63.141  < 2e-16 ***
dietsB       5.000e+00  1.366e+00   3.660  0.00156 ** 
dietsC       7.000e+00  1.366e+00   5.123 5.18e-05 ***
dietsD      -9.999e-15  1.366e+00   0.000  1.00000    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.366 on 20 degrees of freedom
Multiple R-squared:  0.6706,	Adjusted R-squared:  0.6212 
F-statistic: 13.57 on 3 and 20 DF,  p-value: 4.658e-05
```

The design matrix is

```
model.matrix(lm.diets)
```

```
##    (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
## 13           1      0      1      0
## 14           1      0      1      0
## 15           1      0      1      0
## 16           1      0      1      0
## 17           1      0      1      0
## 18           1      0      1      0
## 19           1      0      0      1
## 20           1      0      0      1
## 21           1      0      0      1
## 22           1      0      0      1
## 23           1      0      0      1
## 24           1      0      0      1
## attr(,"assign")
## [1] 0 1 1 1
## attr(,"contrasts")
## attr(,"contrasts")$diets
## [1] "contr.treatment"
```

The default dummy coding was used.

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 ($\bar{y}_{j.}$)	61	66	68	61

So we can verify that the least-squares estimates are differences of the treatment averages.

$$\begin{aligned}
\bar{y}_{1.} &= 61, \\
\hat{\tau}_1 &= \bar{y}_{2.} - \bar{y}_{1.} = 5 \\
\hat{\tau}_2 &= \bar{y}_{3.} - \bar{y}_{1.} = 7 \\
\hat{\tau}_3 &= \bar{y}_{3.} - \bar{y}_{1.} = -9.9 \times 10^{-15}.
\end{aligned}$$

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

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

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

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

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,

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

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

```


Call:
lm(formula = y ~ diets, data = tab0401)

Residuals:
   Min     1Q Median     3Q    Max 
 -5.00  -1.25   0.00   1.25   5.00 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  64.0000     0.4830 132.493  < 2e-16 ***
diets1       -3.0000     0.8367  -3.586 0.001849 ** 
diets2        2.0000     0.8367   2.390 0.026781 *  
diets3        4.0000     0.8367   4.781 0.000114 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.366 on 20 degrees of freedom
Multiple R-squared:  0.6706,	Adjusted R-squared:  0.6212 
F-statistic: 13.57 on 3 and 20 DF,  p-value: 4.658e-05
```

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

3 Multiple Comparisons

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

$$H_0 : \mu_1 = \mu_2 = \mu_3 \, vs. \; H_1 : \mu_i \neq \mu_j.$$

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. $\mu_2 \neq \mu_3$

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

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

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

$$\begin{aligned} P\left(\text{reject at least one } H_{0_k}\right) &= 1 - P\left(\text{do not reject any } H_{0_k}\right) \\ &= 1 - P\left(\text{do not reject } H_{0_1} \text{and do not reject } H_{0_2} \text{and do not reject } H_{0_3}\right) \\ &= 1 - P\left(\text{do not reject } H_{0_1}\right) P\left(\text{do not reject } H_{0_2}\right) P\left(\text{do not reject } H_{0_3}\right) \quad \text{since the hypotheses are independent} \\ &= 1 - (1 - \alpha)^3 \end{aligned}$$

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.

In general if

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_k \, vs. \; H_1 : \mu_i \neq \mu_j.$$

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.

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.

3.1 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

$$t_{ij} = \frac{\bar{y}_{j\cdot} - \bar{y}_{i\cdot}}{\hat{\sigma}\sqrt{1/n_j + 1/n_i}},$$

where $\bar{y}_{j\cdot}$ 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

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

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

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.

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.

For comparison the unadjusted p-values are also calculated.

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

```
## 
##  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 then the p-values adjusted using the Bonferroni method.

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

$$\bar{y}_{j\cdot} - \bar{y}_{i\cdot} \pm t_{N-k,\alpha/2c}\hat{\sigma}\sqrt{1/n_j + 1/n_i}.$$

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

The 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

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

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

```
## 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)
```

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

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).

3.2 The Tukey Method

The only difference between the Tukey and Bonferroni methods is in the choice of the ctitical value.

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

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

where t_{ij} is the observed value of the two-sample t-statistic and $q_{k,N-k,\alpha}$ 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.

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

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

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

In the coagualtion study $N = 24, k = 4$ so the 5% critical value of the Studentizee 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(.05,4,16,lower.tail = FALSE)
```

```
## [1] 4.046093
```

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

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

```
## [1] 1.175925
```

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

```
## [1] 8.824075
```

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

```
(1/sqrt(2))*qtukey(p = .05,nmeans = 4,df = 20,lower.tail = FALSE)*sqrt(5.6)*sqrt(1/6+1/6)
```

```
## [1] 3.824075
```

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

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

```
## [1] 4.024113
```

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.

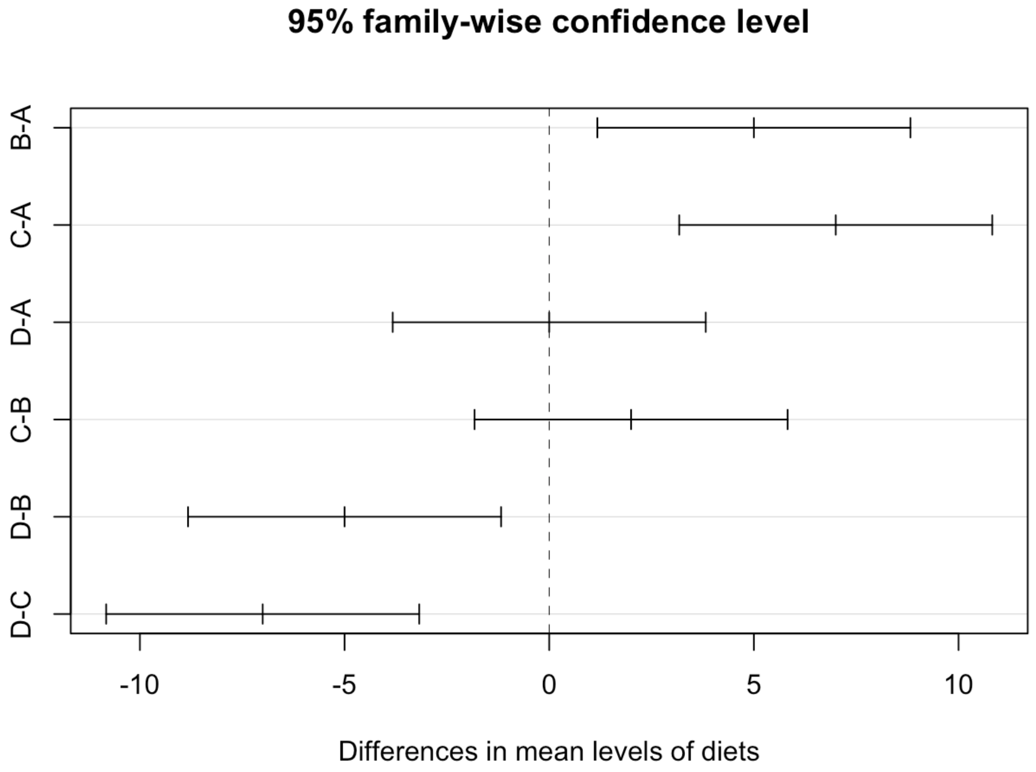
Continuing with the blood coagualtion study all of the 95% Tukey confidence intervals for the diets are

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

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = y ~ diets, data = tab0401)
##
## $diets
##              diff        lwr       upr     p adj
## B-A  5.000000e+00   1.175925  8.824075 0.0077978
## C-A  7.000000e+00   3.175925 10.824075 0.0002804
## D-A -1.421085e-14  -3.824075  3.824075 1.0000000
## C-B  2.000000e+00  -1.824075  5.824075 0.4766005
## D-B -5.000000e+00  -8.824075 -1.175925 0.0077978
## D-C -7.000000e+00 -10.824075 -3.175925 0.0002804
```

A plot of the 95% confidence intervals can be obtained by using the `plot()` function.

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



4 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 = $1 - P(\text{Type II error})$ is this probability.

The null and alternative hypotheses are:

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_k \, vs. \; H_1 : \mu_i \neq \mu_j.$$

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.

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

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

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 .

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

4.1 Direct calculation of Power

- 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.
- The degrees of freedom used in the power function are $k - 1$ for the numerator degrees of freedom (k is the number of groups) and $N - k$ for the denominator degrees of freedom, where $N = n_i k$ (n_i is the total number of observations in group i) the total number of observations in all the groups.

4.2 Example

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

```
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
```

So, we will use $\mu_1 =$ 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 estimnated values are the true values of the parameters, the non-centrality parameter of the F distribution is

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

This was calculated using R

```
(3*((61-64)^2+(66-64)^2+(68-64)^2+(61-64)^2))/5.6
```

```
## [1] 20.35714
```

If we choose $\alpha = 0.05$ as the significance level then $F_{3,8,0.05} =$ 4.0661806. The power of the test is then

$$P\left(F_{3,8}(20.36) > 4.07\right) = 0.85.$$

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

```
1-pf(q = 4.07,df1 = 3,df2 = 8,ncp = 20.36)
```

```
## [1] 0.8496248
```

4.3 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 standard deviation of the population means divided by the common within-population standard deviation.

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

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

The relationship between effect size f for ANOVA and the non-centrality parameter δ is

$$\delta = kn_if^2,$$

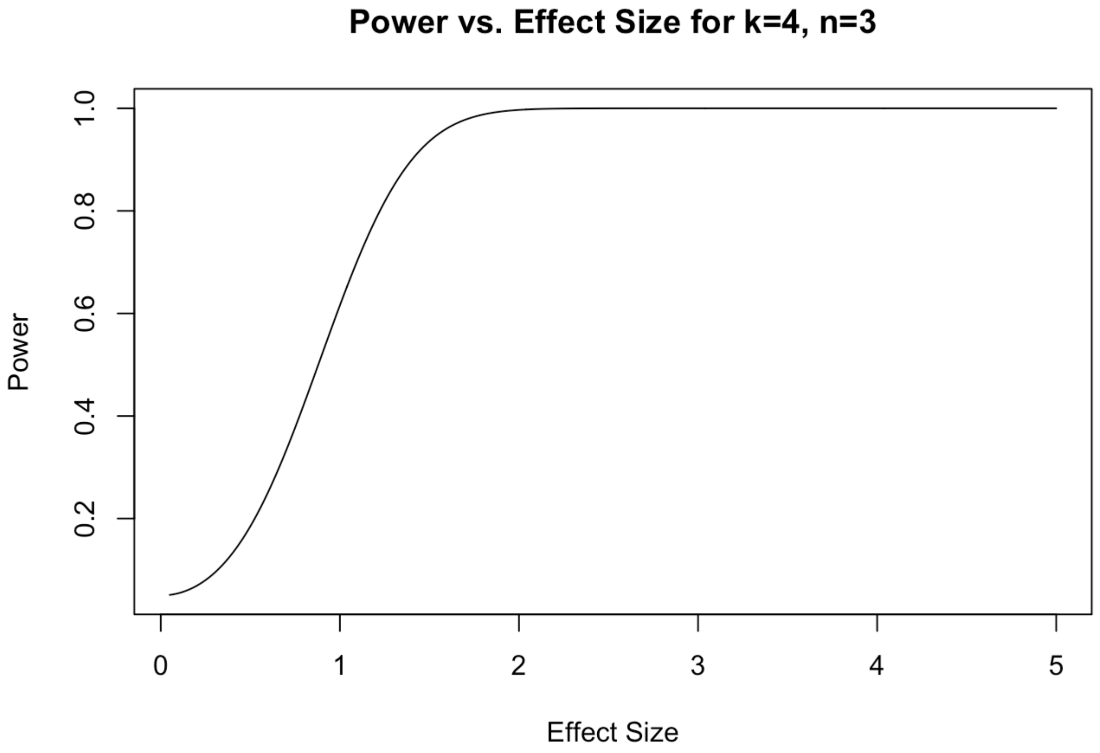
where n_i is the number of observations in group $i = 1, \ldots, k$.

```
eff.size <- sqrt(((61-64)^2+(66-64)^2+(68-64)^2+(61-64)^2)/4/5.6)
library(pwr)
pwr.anova.test(k = 4,n = 3,f = eff.size)
```

```
##
##      Balanced one-way analysis of variance power calculation
##
##               k = 4
##               n = 3
##               f = 1.30247
##       sig.level = 0.05
##           power = 0.8499001
##
## NOTE: n is number in each group
```

Recall that 1.3 is a very large effect size. A plot of effect size versus power when $k = 4, n = 3$ is shown below.

```
library(pwr)
x <- seq(.05,5,by=0.01)
plot(x,pwr.anova.test(k = 4,n = 3,f = x)$power,type="l",xlab="Effect Size",ylab="Power",main
="Power vs. Effect Size for k=4, n=3")
```



The plot shows that power is an increasing function of effect size. The power is 1 for effect sizes at least 1.9.

4.4 Calculating power using simulation

The power of an ANOVA design can be simulated using R.

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.

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.

A simple R program that implements 1-4 above, for three treatment groups, is given below.

```
#Simulate power of ANOVA for three groups

NSIM <- 10000 # 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
  {
y1 <- rnorm(n = n1,mean = mu1,sd = sigma1) # generate a random sample of size n1 from N(mu1,
sigma1^2)
y2 <- rnorm(n = n2,mean = mu2,sd = sigma2) # generate a random sample of size n2 from N(mu2,
sigma2^2)
y3 <- rnorm(n = n3,mean = mu3,sd = sigma3) # generate a random sample of size n3 from N(mu3,
sigma3^2)
y <- c(y1,y2,y3) # store all the values from the groups
trt <- as.factor(c(rep(1,n1),rep(2,n2),rep(3,n3))) # generate the treatment assignment for e
ach group
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] 0.6199
```

We can check to make sure that our program works by verifying two scenarios where we know the answers.

1. Calculate the power of the test when $\mu_1 = 2, \mu_2 = 2.5, \mu_3 = 2, \sigma = 1, n_1 = n_2 = n_3 = 40$, and

$$f = \sqrt{\frac{\Sigma_{i=1}^{3}\left(\mu_i - 2.17\right)^2/3}{1^2}} = 0.2357023,$$

using `pwr.anova.test()`

```
mug <- sum(mu1,mu2,mu3)/3
mui <- c(mu1,mu2,mu3)
f1 <- sqrt(((1/3)*sum((mui-mug)^2))/sigma1)
pwr.anova.test(k = 3,f = f1,n = 40,sig.level = 0.05)
```

```
##
##      Balanced one-way analysis of variance power calculation
##
##               k = 3
##               n = 40
##               f = 0.2357023
##       sig.level = 0.05
##           power = 0.6207319
##
## NOTE: n is number in each group
```

The simulation program and `pwr.anova.test()` both calculate power equal to 62%.

2. Calculate the power directly.

```
k <- 3
n <- 40
delta <- n*sum((mui-mug)^2)/sigma1
qf(p = .95,df1 = 3,df2 = 117)
```

```
## [1] 2.682132
```

```
1-pf(q = qf(p = .95,df1 = k-1,df2 = n*k-k), df1 = k-1,df2 = n*k-k,ncp = delta,lower.tail = T)
```

```
## [1] 0.6207319
```

3. When $\mu_1 = \mu_2 = \mu 3$ (i.e., when H_0 is true) the power of the test is α.

When $\delta = 0$

$$P\left(F_{k-1,N-k}(0) > F_{k-1,N-K,\alpha}\right) = \alpha.$$

```
#Simulate power of ANOVA for three groups

NSIM <- 10000 # number of simulations
res <- numeric(NSIM) # store p-values in res

mu1 <- 2; mu2 <- 2;mu3 <- 2  # true mean values of treatment groups
sigma1 <- 1; sigma2 <- 1; sigma3 <- 1 #variances in each group
n1 <- 10; n2 <- 10; n3 <- 10 #sample size in each group

for (i in 1:NSIM) # do the calculations below N times
  {
y1 <- rnorm(n = n1,mean = mu1,sd = sigma1) # generate a random sample of size n1 from N(mu1, sigma1^2)
y2 <- rnorm(n = n2,mean = mu2,sd = sigma2) # generate a random sample of size n2 from N(mu2, sigma2^2)
y3 <- rnorm(n = n3,mean = mu3,sd = sigma3) # generate a random sample of size n3 from N(mu3, sigma3^2)
y <- c(y1,y2,y3) # store all the values from the groups
trt <- as.factor(c(rep(1,n1),rep(2,n2),rep(3,n3))) # generate the treatment assignment for each group
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] 0.0512
```

5 Questions

Question A

Interpret the parameters in the additive model for ANOVA

$$y_{ti} = \mu + \tau_t + \epsilon_{ti},$$

where, y_{ti} is the i^{th} observation in the t^{th} treatment group, μ is the overall mean, and τ_t is the deviation produced by treatment t, and ϵ_{ti} is the error.

6 Answers to questions

Question A

Interpret the parameters in the additive model for ANOVA

$$y_{ti} = \mu + \tau_t + \epsilon_{ti},$$

where, y_{ti} is the i^{th} observation in the t^{th} treatment group, μ is the overall mean, and τ_t is the deviation produced by treatment t, and ϵ_{ti} is the error.

> $\hat{\mu} = \bar{y}_1$ and $\hat{\mu} + \hat{\tau}_t = \bar{y}_t, t = 2, 3, 4,$ where $\bar{y}_t = \Sigma_{k=1}^{n_t} y_{tk}/n_t$ is the mean for for the n_t units in treatment t. So, the least squares estimates $\hat{\tau}_t = \bar{y}_t - \bar{y}_1, t = 2, 3, 4.$
>
> ϵ_{ti} has a $N(0, \sigma^2)$ distribution.

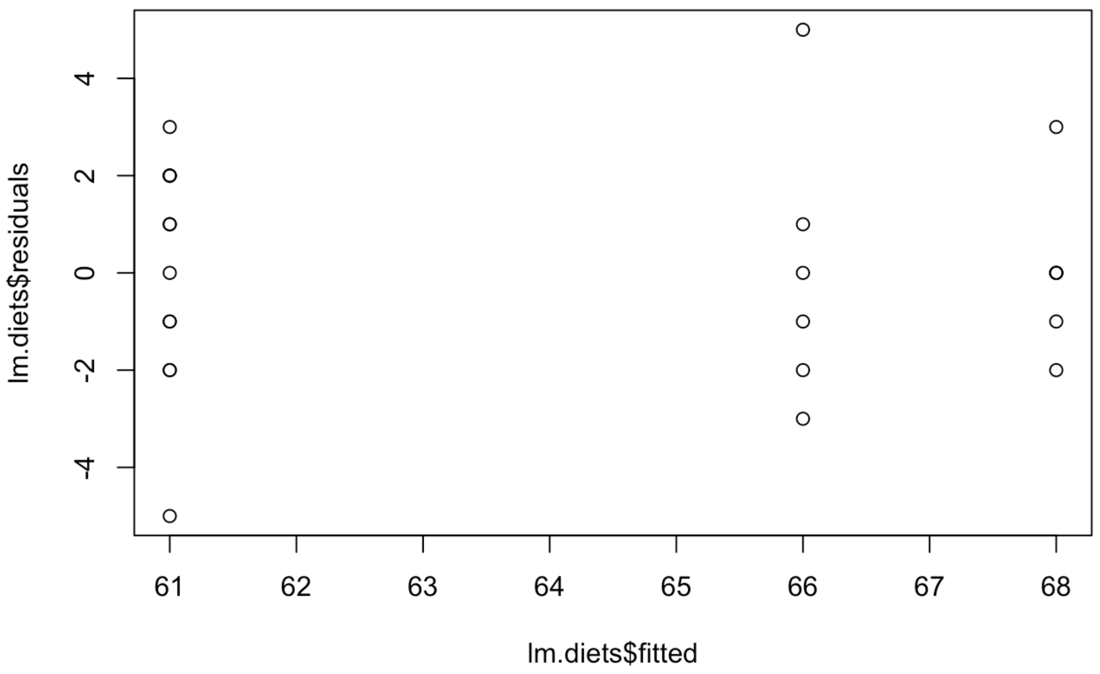
```
lm.diets <- lm(y~diets,data=tab0401)
anova(lm.diets)
```

```
## 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
```

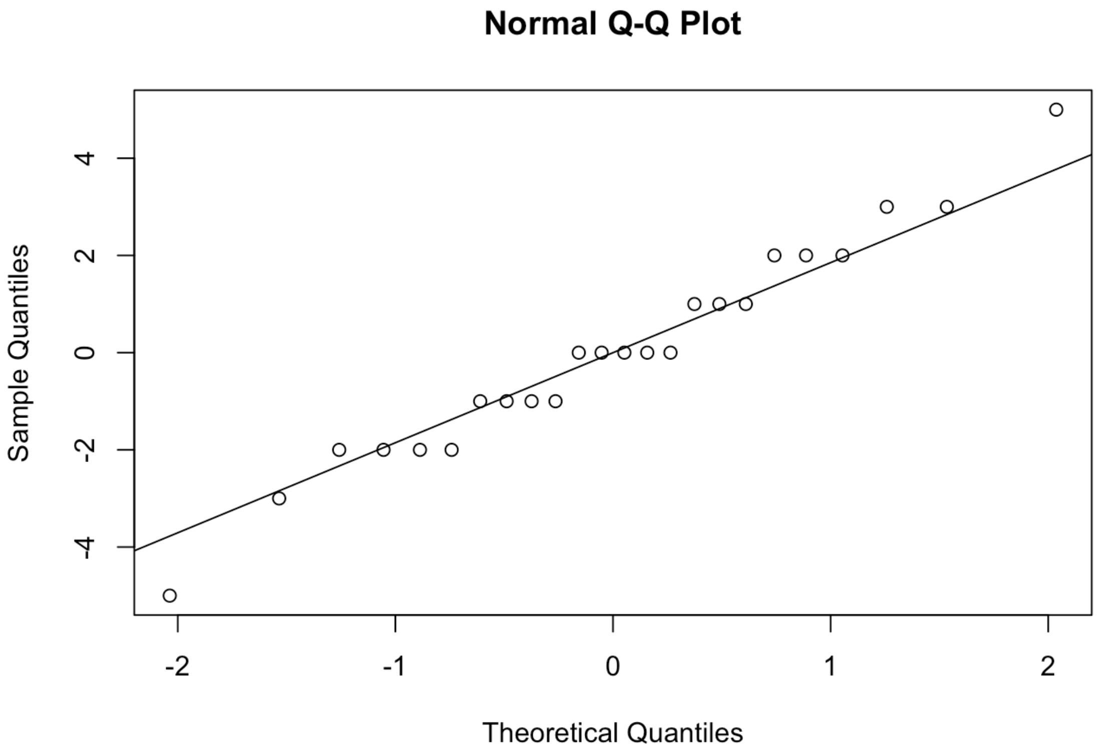
```
summary(lm.diets)
```

```
##
## Call:
## lm(formula = y ~ diets, data = tab0401)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -5.00  -1.25   0.00   1.25   5.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  64.0000     0.4830 132.493  < 2e-16 ***
## diets1       -3.0000     0.8367  -3.586 0.001849 **
## diets2        2.0000     0.8367   2.390 0.026781 *
## diets3        4.0000     0.8367   4.781 0.000114 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.366 on 20 degrees of freedom
## Multiple R-squared:  0.6706, Adjusted R-squared:  0.6212
## F-statistic: 13.57 on 3 and 20 DF,  p-value: 4.658e-05
```

```
plot(lm.diets$fitted,lm.diets$residuals)
```



```
qqnorm(lm.diets$residuals);qqline(lm.diets$residuals)
```


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles