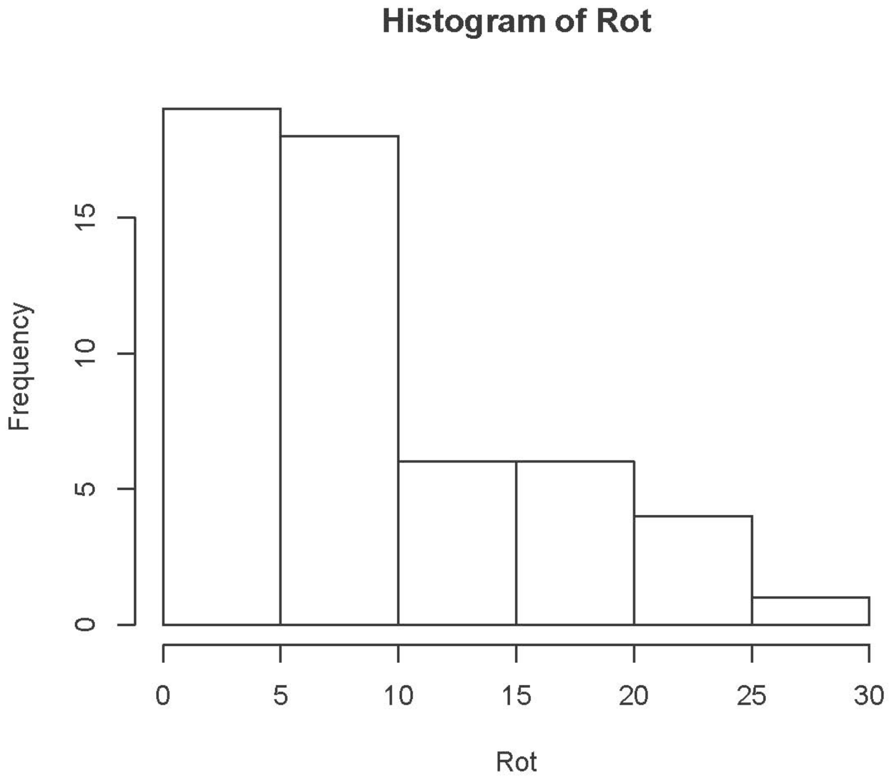
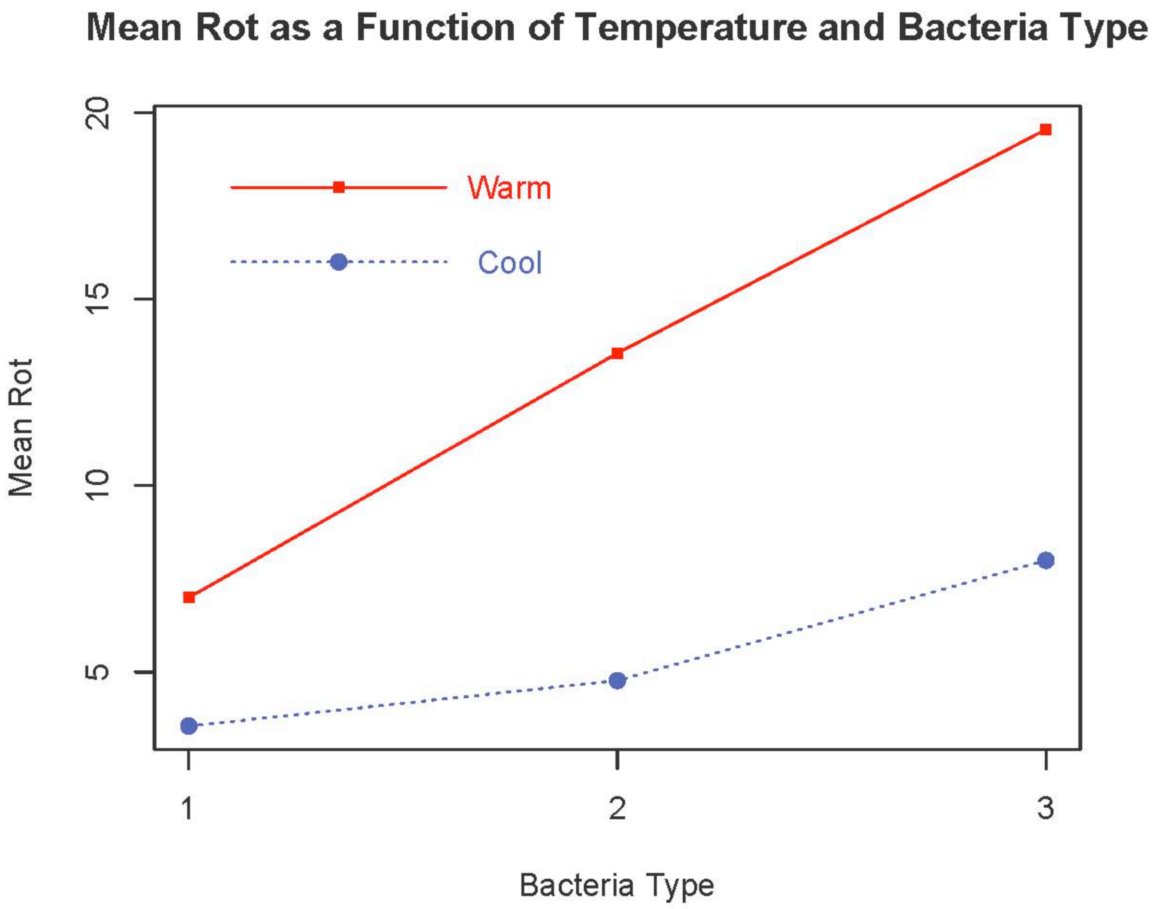
# Factorial Analysis of Variance with R

```
> # Potato Data with R
> potato =
read.table("http://www.utstat.toronto.edu/~brunner/appliedf14/code_n_data/lecture/p
otato2.data")
> potato
   Bact Temp Rot
1     1    1   7
2     1    1   7
3     1    1   9
4     1    1   0

.     .    .   .
.     .    .   .
.     .    .   .

53    3    2  24
54    3    2   8
> attach(potato)
> hist(Rot)
```



```
> table(Rot)
Rot
 0  2  3  4  5  6  7  8  9 10 11 13 14 15 17 18 19 20 22 23 24 26 
 7  2  2  5  3  1  5  3  3  6  1  1  1  3  2  2  1  1  1  1  2  1 
> table(Temp,Bact)
    Bact
Temp 1 2 3
   1 9 9 9
   2 9 9 9
> meanz = aggregate(Rot~Temp+Bact, FUN=mean); meanz
  Temp Bact       Rot
1    1    1  3.555556
2    2    1  7.000000
3    1    2  4.777778
4    2    2 13.555556
5    1    3  8.000000
6    2    3 19.555556
> 
> meantable = meanz[,3]; dim(meantable) = c(2,3); meantable
         [,1]      [,2]     [,3]
[1,] 3.555556  4.777778  8.00000
[2,] 7.000000 13.555556 19.55556
> dimnames(meantable) = list(c("Cool","Warm"),c("Bact1","Bact2","Bact3"))
> meantable
        Bact1     Bact2    Bact3
Cool 3.555556  4.777778  8.00000
Warm 7.000000 13.555556 19.55556
> 
> # Plot the means
> cool = meantable[1,]; warm = meantable[2,]
> Bacteria = c(1:3,1:3); MeanRot = c(cool,warm)
> # Invisible points at first, x axis points at 1,2,3; see help(plot)
> plot(Bacteria,MeanRot,pch=" ",xaxp=c(1,3,2), xlab="Bacteria Type",
+ ylab="Mean Rot")
> title("Mean Rot as a Function of Temperature and Bacteria Type")
> points(1:3,warm,col='red',pch=15) # Red squares
> points(1:3,cool,col='blue',pch=19) # Blue circles
> lines(1:3,warm,lty=1,col='red'); lines(1:3,cool,lty=3,col='blue')
> # Annotate the plot
> x1 = c(1.1,1.6); y1 = c(18,18); lines(x1,y1,lty=1,col='red')
> points(1.35,18,col='red',pch=15)
> text(1.75,18,'Warm',col='red')
> x2 = c(1.1,1.6); y2 = c(16,16); lines(x2,y2,lty=3,col='blue')
> points(1.35,16,col='blue',pch=19)
> text(1.75,16,'Cool',col='blue')
> 
```



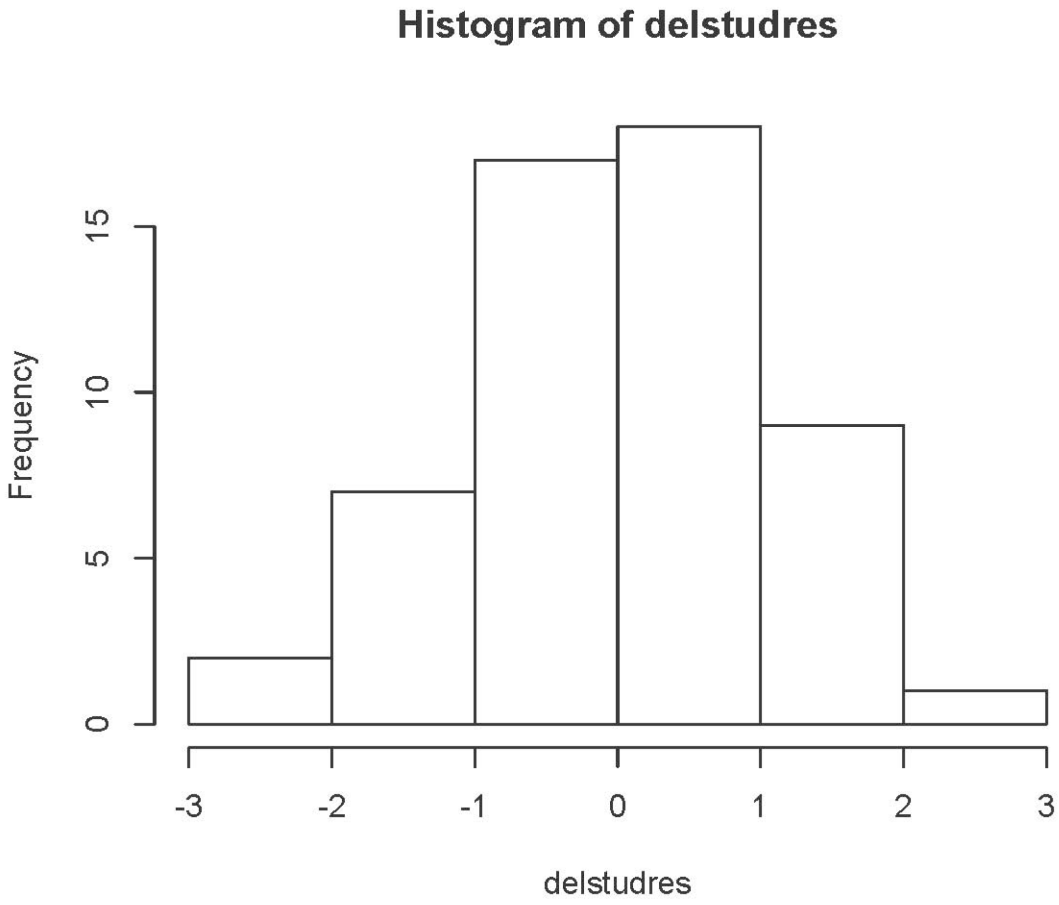
```
> # Two-factor ANOVA, several different ways
>
> # First with Dummy variable regression, cell means coding and contrasts.
> # Make 6 indicator dummy variables.
> n = length(Rot)
> p11 = p12 = p13    =
+ p21 = p22 = p23    = numeric(n)
> TB = 10*Temp + Bact # Yields 11 12 13  21 22 23
> table(TB)
TB
11 12 13 21 22 23
 9  9  9  9  9  9
> p11[TB==11] = 1; p12[TB==12] = 1; p13[TB==13] = 1
> p21[TB==21] = 1; p22[TB==22] = 1; p23[TB==23] = 1
> # data.frame(Temp,Bact,p11,p12,p13,p21,p22,p23) # To check
> cellmeans_model = lm(Rot ~ 0 + p11+p12+p13 + p21+p22+p23)
> summary(cellmeans_model)

Call:
lm(formula = Rot ~ 0 + p11 + p12 + p13 + p21 + p22 + p23)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.5556  -3.5556   0.2222   3.4444   9.4444 

Coefficients:
    Estimate Std. Error t value Pr(>|t|)    
p11    3.556      1.562   2.276  0.02734 *  
p12    4.778      1.562   3.058  0.00363 ** 
p13    8.000      1.562   5.121 5.33e-06 ***
p21    7.000      1.562   4.481 4.60e-05 ***
p22   13.556      1.562   8.677 2.13e-11 ***
p23   19.556      1.562  12.518  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.686 on 48 degrees of freedom
Multiple R-squared: 0.8592, Adjusted R-squared: 0.8416 
F-statistic: 48.81 on 6 and 48 DF,  p-value: < 2.2e-16

> # Define ftest = function(model,L,h=0)
> source("http://www.utstat.utoronto.ca/~brunner/Rfunctions/ftest.txt")
> # For any factorial design, THE BASICS are an overall test for equality of
> # treatment means, and also tests for all the main effects and interactions.
> # For the potato data, first the overall test for equality of the 6
> #  treatment means
> L1 = rbind(c(1,-1, 0, 0, 0, 0),
+            c(0, 1,-1, 0, 0, 0),
+            c(0, 0, 1,-1, 0, 0),
+            c(0, 0, 0, 1,-1, 0),
+            c(0, 0, 0, 0, 1,-1))
> round(ftest(cellmeans_model,L1),5)
       F      df1      df2  p-value 
15.05093  5.00000 48.00000  0.00000 
> # Test for differences between marginal means (Main effects)
> # Better LOOK at the marginal means
> addmargins(meantable,c(1,2),FUN=mean)
Margins computed over dimensions
in the following order:
1: 
2: 
        Bact1     Bact2    Bact3      mean
Cool 3.555556  4.777778  8.00000  5.444444
Warm 7.000000 13.555556 19.55556 13.370370
mean 5.277778  9.166667 13.77778  9.407407
> # Averaging across Bacteria Types, does Storage Temperature affect
> # the amount of rot?
> L2 = rbind(c(1,1,1,-1,-1,-1))
> round(ftest(cellmeans_model,L2),5)
       F      df1      df2  p-value 
38.61383  1.00000 48.00000  0.00000 
> # Averaging across Storage Temperatures, does Bacteria Type affect
> # the amount of rot?
> L3 = rbind(c(1,-1, 0, 1,-1, 0),
+            c(0, 1,-1, 0, 1,-1))
> round(ftest(cellmeans_model,L3),5)
       F      df1      df2  p-value 
14.83895  2.00000 48.00000  0.00001 
> # Now the Interaction: Does the effect of Temperature depend on type of bacteria?
> # H0: mu21-mu11 = mu22-mu12 = mu23-mu13
> # <=> (-1)*mu11 + (1)*mu12  + (0)*mu13 + (1)*mu21 + (-1)*mu22 +  (0)*mu23 = 0 and
> #      (0)*mu11 + (-1)*mu12 + (1)*mu13 + (0)*mu21 +  (1)*mu22 + (-1)*mu23 = 0
> L4 =  rbind(c(-1, 1, 0, 1,-1, 0),
+             c( 0,-1, 1, 0, 1,-1) )
> round(ftest(cellmeans_model,L4),5)
       F      df1      df2  p-value 
 3.48145  2.00000 48.00000  0.03874 
> 
> # Dummy variable regression with effect coding
> t1 = numeric(n); t1[Temp==1] = 1; t1[Temp==2] = -1
> b1 = numeric(n); b1[Bact==1] = 1; b1[Bact==3] = -1
> b2 = numeric(n); b2[Bact==2] = 1; b2[Bact==3] = -1
> tb1 = t1*b1; tb2 = t1*b2
> effect_coding_model = lm(Rot ~ t1 + b1 + b2 + tb1 + tb2)
> summary(effect_coding_model)
> summary(effect_coding_model)
Call:
lm(formula = Rot ~ t1 + b1 + b2 + tb1 + tb2)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.5556  -3.5556   0.2222   3.4444   9.4444 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   9.4074     0.6377  14.751  < 2e-16 ***
t1           -3.9630     0.6377  -6.214 1.18e-07 ***
b1           -4.1296     0.9019  -4.579 3.33e-05 ***
b2           -0.2407     0.9019  -0.267   0.7907    
tb1           2.2407     0.9019   2.484   0.0165 *  
tb2          -0.4259     0.9019  -0.472   0.6389    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.686 on 48 degrees of freedom
Multiple R-squared: 0.6106, Adjusted R-squared:  0.57 
F-statistic: 15.05 on 5 and 48 DF,  p-value: 7.003e-09

> # Test main effects for temperature with full vs. reduced
> notemp = update(effect_coding_model, . ~ . - t1)
> anova(notemp,effect_coding_model)
Analysis of Variance Table

Model 1: Rot ~ b1 + b2 + tb1 + tb2
Model 2: Rot ~ t1 + b1 + b2 + tb1 + tb2
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)    
1     49 1902.3                                 
2     48 1054.2  1    848.07 38.614 1.18e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # Main effects for Bacteria Type
> nobact = update(effect_coding_model, . ~ . - b1 - b2)
> anova(nobact,effect_coding_model)
Analysis of Variance Table

Model 1: Rot ~ t1 + tb1 + tb2
Model 2: Rot ~ t1 + b1 + b2 + tb1 + tb2
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)    
1     50 1706.0                                  
2     48 1054.2  2    651.81 14.839 9.608e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> # Test interaction
> nointer = update(effect_coding_model, . ~ . - tb1 - tb2)
> anova(nointer,effect_coding_model)
Analysis of Variance Table

Model 1: Rot ~ t1 + b1 + b2
Model 2: Rot ~ t1 + b1 + b2 + tb1 + tb2
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1     50 1207.2
2     48 1054.2  2    152.93 3.4815 0.03874 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # Could have used general linear test for all this
> # For example, test main effects for Bacteria
> L5 = rbind(c(0,0,1,0,0,0),
+            c(0,0,0,1,0,0)  )
> round(ftest(effect_coding_model,L5),5)
       F      df1      df2  p-value
14.83895  2.00000 48.00000  0.00001
>
```

```
> # Now finally do it the easy way
> 
> Bacteria=factor(Bact); Temperature = factor(Temp, labels = c("Cool","Warm"))
> easy = lm(Rot~Temperature+Bacteria+Temperature:Bacteria)
> summary(easy)

Call:
lm(formula = Rot ~ Temperature + Bacteria + Temperature:Bacteria)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.5556  -3.5556   0.2222   3.4444   9.4444 

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)  
(Intercept)                  3.556      1.562   2.276   0.0273 *
TemperatureWarm              3.444      2.209   1.559   0.1255  
Bacteria2                    1.222      2.209   0.553   0.5827  
Bacteria3                    4.444      2.209   2.012   0.0499 *
TemperatureWarm:Bacteria2    5.333      3.124   1.707   0.0943 .
TemperatureWarm:Bacteria3    8.111      3.124   2.596   0.0125 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 4.686 on 48 degrees of freedom
Multiple R-squared: 0.6106,	Adjusted R-squared:  0.57 
F-statistic: 15.05 on 5 and 48 DF,  p-value: 7.003e-09 

> # What happened with the t-test for TemperatureWarm?
> anova(easy)
Analysis of Variance Table

Response: Rot
                     Df  Sum Sq Mean Sq F value    Pr(>F)    
Temperature           1  848.07  848.07 38.6138 1.180e-07 ***
Bacteria              2  651.81  325.91 14.8390 9.608e-06 ***
Temperature:Bacteria  2  152.93   76.46  3.4815   0.03874 *  
Residuals            48 1054.22   21.96                      
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 
> # It worked because sample sizes are equal, and R saved us from the
> # dummy variable coding method.
> # WARNING: If factors are related, test effects one at a time with
> # full-reduced approach.
> 
```

```
> # Can make R use effect coding for factors
> contrasts(Temperature) = contr.sum(2)
> contrasts(Bacteria) = contr.sum(3); contrasts(Bacteria)
  [,1] [,2]
1    1    0
2    0    1
3   -1   -1
> easy2 = lm(Rot~Temperature+Bacteria+Temperature:Bacteria)
> summary(easy2)

Call:
lm(formula = Rot ~ Temperature + Bacteria + Temperature:Bacteria)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.5556  -3.5556   0.2222   3.4444   9.4444 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)              9.4074     0.6377  14.751  < 2e-16 ***
Temperature1            -3.9630     0.6377  -6.214 1.18e-07 ***
Bacteria1               -4.1296     0.9019  -4.579 3.33e-05 ***
Bacteria2               -0.2407     0.9019  -0.267   0.7907    
Temperature1:Bacteria1   2.2407     0.9019   2.484   0.0165 *  
Temperature1:Bacteria2  -0.4259     0.9019  -0.472   0.6389    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 4.686 on 48 degrees of freedom
Multiple R-squared: 0.6106,	Adjusted R-squared:  0.57 
F-statistic: 15.05 on 5 and 48 DF,  p-value: 7.003e-09 

> anova(easy2)
Analysis of Variance Table

Response: Rot
                     Df  Sum Sq Mean Sq F value    Pr(>F)    
Temperature           1  848.07  848.07 38.6138 1.180e-07 ***
Bacteria              2  651.81  325.91 14.8390 9.608e-06 ***
Temperature:Bacteria  2  152.93   76.46  3.4815   0.03874 *  
Residuals            48 1054.22   21.96                      
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

> 
```

```
> # Residuals for all the models are the same
> delstudres = rstudent(easy)
> hist(delstudres) # Very nice lesson
```



```
>
> # To test non-standard hypotheses, it is easiest to use the cell means model
> # Just a few examples out of MANY possible
>
> # Is there an effect of bacteria type at cool temperatures?
> # H0: mu11=mu12=mu13
> L6 = rbind(c(1,-1, 0, 0, 0, 0),
+            c(0, 1,-1, 0, 0, 0) )
> round(ftest(cellmeans_model,L6),5)
       F      df1      df2  p-value
 2.16020  2.00000 48.00000  0.12638
>
```

```
> # Is there an effect of bacteria type at warm temperatures?
> # H0: mu21=mu22=mu23
> L7 = rbind(c(0, 0, 0, 1,-1, 0),
+            c(0, 0, 0, 0, 1,-1) )
> round(ftest(cellmeans_model,L7),5)
      F     df1     df2 p-value 
16.1602  2.0000 48.0000  0.0000 
> 
> # Is the effect of temperature different for Bacteria types 1 and 3?
> # H0: mu11-mu21 = mu13-mu23
> L8 = rbind(c(1, 0,-1,-1, 0, 1))
> round(ftest(cellmeans_model,L8),5)
       F      df1      df2  p-value 
 6.73988  1.00000 48.00000  0.01247 
> # Could look at all pairwise comparisons of differences
> # between Cool and Warm

> # Any of these F- tests could be Scheffe follow-ups to the test with null
> # hypothesis no main effects and no interactions, which is the same as
> # H0: all expected values equal.

> # Make a table of fa = r/s * f.
> f = qf(0.95,5,48); f # Critical value of initial test
[1] 2.408514
> r=5; s = 1:5; fa = r/s * f
> cbind(s,fa)
     s        fa
[1,] 1 12.042571
[2,] 2  6.021285
[3,] 3  4.014190
[4,] 4  3.010643
[5,] 5  2.408514
> # For example look,
> anova(easy2)


                     Df  Sum Sq Mean Sq F value    Pr(>F)    
Temperature           1  848.07  848.07 38.6138 1.180e-07 ***
Bacteria              2  651.81  325.91 14.8390 9.608e-06 ***
Temperature:Bacteria  2  152.93   76.46  3.4815   0.03874 *  
Residuals            48 1054.22   21.96                      
```

This handout was prepared by Jerry Brunner, Department of Statistical Sciences, University of Toronto. It is licensed under a Creative Commons Attribution - ShareAlike 3.0 Unported License. Use any part of it as you like and share the result freely. The OpenOffice.org document is available from the course website:

http://www.utstat.toronto.edu/~brunner/oldclass/appliedf14