```
> section<-factor(rep(1:2,c(21,24)))
```

These data are grades on an inclass exam given in Stat 5301. I had given two different exams, randomly intermixed through the students (though they looked pretty similar at a glance!), and I was interested in any possible differences between the average scores for the two exams. I was also interested in any possible effects of status (undergraduate versus graduate student) and lab section (I had two different TAs). This is not a randomized experiment, but it does illustrate some of the issues with ANOVA for unbalanced data.

```
> status<-factor(c(2,1,1,2,2,2,1,1,1,2,2,2,2,2,1,2,2,2,1,1,1,
1,1,2,1,1,2,1,1,2,2,2,2,2,1,2,2,1,2,2,2,1,1,2,2))

> exam<-factor(c(2,2,2,2,2,2,2,1,1,2,2,1,1,1,1,1,1,2,1,1,2,
2,1,2,2,1,2,1,2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1,1))

> grades<-c(62,34,75,80,89,63,42,36,40,91,
79,72,62,70,76,70,90,91,18,49,37,
65,48,66,59,63,75,51,50,75,62,88,84,70,33,60,54,
58,42,87,90,70,72,71,81)

> tapply(grades,list(exam,section,status),length)
```

We see that the counts are not the same in the different groups (not that we expected them to be the same). This means that we will need to consider the lack of balance when doing ANOVA.

```
, , 1

  1 2
1 5 6
2 4 4

, , 2

  1 2
1 5 6
2 7 8
```

```
> fit1 <- lm(grades˜status*exam*section);anova(fit1)
```

Here is a first ANOVA. By default, R rearranges into main effects, then second order interactions, then third order, and so on. Also by default, R gives you sequential sums of squares, that is, Type I sums of squares. In this output, all sums of squares are type I, exam:section is also type II, and status:exam:section is Type I, II, and III.

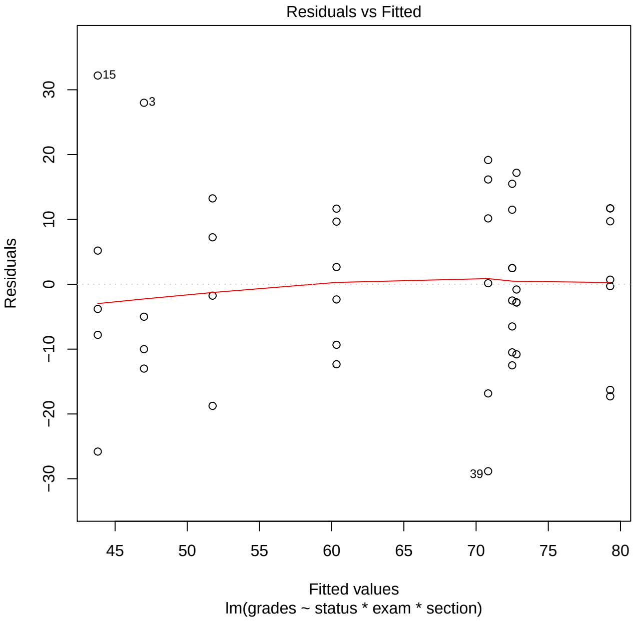
```
Analysis of Variance Table

Response: grades
                    Df Sum Sq Mean Sq F value    Pr(>F)
status               1 5622.7  5622.7 26.4818 8.976e-06 ***
exam                 1    7.5     7.5  0.0352   0.85229
section              1   53.9    53.9  0.2537   0.61746
status:exam          1  143.1   143.1  0.6739   0.41694
status:section       1  725.2   725.2  3.4158   0.07258 .
exam:section         1  164.5   164.5  0.7748   0.38443     Type II
status:exam:section  1   32.3    32.3  0.1522   0.69871     Type II,III
Residuals           37 7855.9   212.3
---
```

```
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

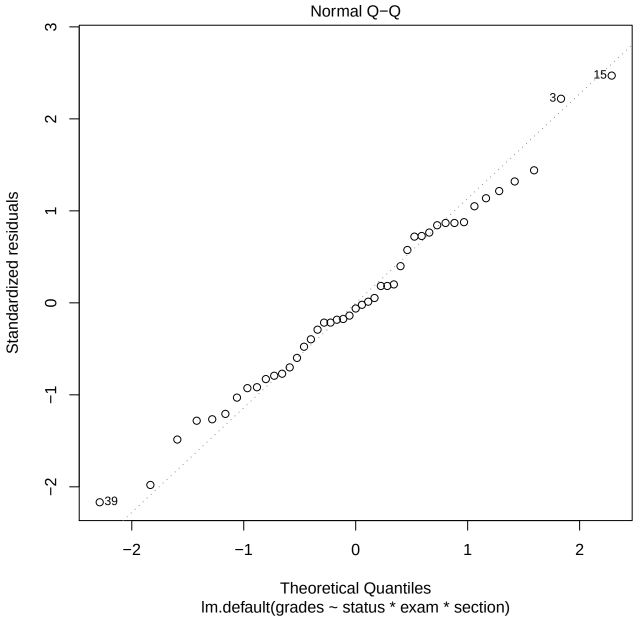
```
> plot(fit1,which=1)
```

There is a bit of decreasing variance. Box-Cox does not suggest a transformation. For simplicity, we will analyze on the original scale, however, consider the following. Look at a new response, which is `lost = 100 - grades` (ie, the number of points lost). With that response, we see increasing rather than decreasing variance, and the square root looks better than the original scale, although it is not quite significant via Box-Cox. Why do the transformations work on `lost` but not on `grades`?



```
> plot(fit1,which=2)
```

Normality is pretty good.



```
> trms <- terms(grades˜exam*section+exam*status++status*exam*section,
   keep.order=TRUE)
```

If we want to put our model terms in a specific order, then we need to use a terms() command to put them in that order. Otherwise, R will just reorder them as above.

```
> anova(lm(trms))
Analysis of Variance Table

Response: grades
                    Df Sum Sq Mean Sq F value    Pr(>F)
exam                 1  203.0   203.0  0.9561    0.3345
section              1   72.4    72.4  0.3410    0.5628
exam:section         1  239.0   239.0  1.1259    0.2955
status               1 5445.2  5445.2 25.6458 1.157e-05 ***    Type II
exam:status          1  149.5   149.5  0.7042    0.4068
section:status       1  607.7   607.7  2.8622    0.0991 .       Type II
exam:section:status  1   32.3    32.3  0.1522    0.6987         Type II,III
Residuals           37 7855.9   212.3
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> trms <- terms(grades˜section*status+section*exam++status*exam*section,
   keep.order=TRUE)
```

Different orders give us different sums of squares.

```
> anova(lm(trms))
Analysis of Variance Table

Response: grades
                    Df Sum Sq Mean Sq F value    Pr(>F)
section              1   66.7    66.7  0.3142   0.57851
status               1 5608.8  5608.8 26.4163 9.154e-06 ***
```

```
section:status       1  741.9   741.9  3.4942   0.06952 .
exam                 1   11.3    11.3  0.0533   0.81867     Type II
section:exam         1  158.5   158.5  0.7466   0.39312
status:exam          1  129.6   129.6  0.6105   0.43956     Type II
section:status:exam  1   32.3    32.3  0.1522   0.69871     Type II,III
Residuals           37 7855.9   212.3
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
> trms <- terms(grades~status*exam+status*section++status*exam*section,
   keep.order=TRUE)
> anova(lm(trms))
Analysis of Variance Table

Response: grades
                    Df Sum Sq Mean Sq F value    Pr(>F)
status               1 5622.7  5622.7 26.4818 8.976e-06 ***
exam                 1    7.5     7.5  0.0352   0.85229
status:exam          1  145.9   145.9  0.6873   0.41240
section              1   51.0    51.0  0.2403   0.62685     Type II
status:section       1  725.2   725.2  3.4158   0.07258 .
exam:section         1  164.5   164.5  0.7748   0.38443     Type II
status:exam:section  1   32.3    32.3  0.1522   0.69871     Type II,III
Residuals           37 7855.9   212.3
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

> #
```

We can now construct a Type II ANOVA by assembling the lines we need from the Type I ANOVAs done in various orders.

	DF	SS	MS	P-value
section	1	51.03	51.03	0.6269
exam	1	11.32	11.32	0.8187
status	1	5445	5445	1.157e-05
exam.section	1	164.5	164.5	0.3844
section.status	1	607.7	607.7	0.09909
status.exam	1	129.6	129.6	0.4396
section.status.exam	1	32.31	32.31	0.6987
Error	37	7856	212.3	

The three-factor interaction is not significant, so we can check the two-ways. None of them is significant, so we can check the main effects. Only status is significant.

```
> Anova(fit1,type=2)
```

It's important to know what Type II is and does: it compares hierarchical models where the base model is the largest hierarchical model not including the term of interest. However, it gets old really fast doing it by hand. This function (from the car package) gets the Type II tests for you.

```
Anova Table (Type II tests)

Response: grades
                    Sum Sq Df F value    Pr(>F)
status              5445.2  1 25.6458 1.157e-05 ***
exam                  11.3  1  0.0533    0.8187
section               51.0  1  0.2403    0.6269
status:exam          129.6  1  0.6105    0.4396
```

```
status:section       607.7  1  2.8622    0.0991 .
exam:section         164.5  1  0.7748    0.3844
status:exam:section   32.3  1  0.1522    0.6987
Residuals           7855.9 37
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> model.effects(fit1,"status")
```

Although I haven't shown it, the estimated effects depend on which terms are in the model, but not on the order in which the terms are entered.

```
        1         2
-11.56696  11.56696
```

```
> linear.contrast(fit1,status,c(-1,1))
```

Let's look at a contrast for status.

```
  estimates       se  t-value      p-value lower-ci upper-ci
1  23.13393 4.463302 5.183142 7.999982e-06 14.09042 32.17744
```

```
> 5.183142^2
```

The square of the t is an F. Neither this F nor the p-value from the contrast match any of the F's or p-values we had in any of the ANOVAs.

The SS for a contrast is the increase in error SS that we would obtain if we removed the contrast df from the model, leaving all the other terms in the model. Note that since we are using a contrast in a main effect, we are taking the main effect of status out of the model, but leaving in all of the interactions that include status. This is OK if you are really interested in testing the corresponding equally weighted hypotheses about those parameters, but it does violate model hierarchy. This SS is a Type III SS.

```
[1] 26.86496
```

```
> Anova(fit1,type=3)
```

lm() likes hierarchy so much that you need extreme measures to get something that is not hierarchical. The Anova() function with type=3 will do that for you. These are sometimes called marginal or full parametric tests.

```
Anova Table (Type III tests)

Response: grades
                    Sum Sq Df  F value Pr(>F)
(Intercept)         165406  1 779.0286 <2e-16 ***
status                5704  1  26.8650  8e-06 ***
exam                     5  1   0.0241 0.8776
section                105  1   0.4926 0.4871
status:exam            122  1   0.5748 0.4532
status:section         601  1   2.8304 0.1009
exam:section           184  1   0.8648 0.3584
status:exam:section     32  1   0.1522 0.6987
Residuals             7856 37
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> fit2 <- lm(grades~status);summary(fit2)
```

For unbalanced data, the coefficients can change when you change the terms in the model. Once you have determined which terms should be retained in your reduced model, refit the model using just your selected terms to get the appropriate estimated effects.

```
Call:
lm.default(formula = grades ~ status)

Residuals:
    Min      1Q  Median      3Q     Max 
-33.368 -11.368  -1.368  13.000  24.632 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   62.684      2.181  28.740  < 2e-16 ***
status1      -11.316      2.181  -5.188 5.45e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1 

Residual standard error: 14.45 on 43 degrees of freedom
Multiple R-squared: 0.385,Adjusted R-squared: 0.3707 
F-statistic: 26.92 on 1 and 43 DF,  p-value: 5.455e-06 

> step(fit1,direction="backward")
```

Here is a more automated way to find a model. step() with direction backward does backward elimination starting with the full model and keeping to hierarchical models. It finds the model with the minimum AIC (Akaike Information Criterion). In general, AIC is a bit more willing to keep variables than are F-tests at the usual significance levels. In this case, the selected model has section*status, not just status.

```
Start:  AIC=248.31
grades ~ status * exam * section

                      Df Sum of Sq    RSS    AIC
- status:exam:section  1    32.309 7888.3 246.49
<none>                             7855.9 248.31

Step:  AIC=246.49
grades ~ status + exam + section + status:exam + status:section + 
    exam:section

                 Df Sum of Sq    RSS    AIC
- status:exam     1    129.63 8017.9 245.22
- exam:section    1    164.50 8052.8 245.42
<none>                        7888.3 246.49
- status:section  1    607.70 8496.0 247.83

Step:  AIC=245.22
grades ~ status + exam + section + status:section + exam:section

                 Df Sum of Sq    RSS    AIC
- exam:section    1    158.52 8176.4 244.11
<none>                        8017.9 245.22
- status:section  1    627.58 8645.5 246.62

Step:  AIC=244.11
grades ~ status + exam + section + status:section
```

```
                 Df Sum of Sq    RSS    AIC
- exam            1     11.32 8187.7 242.17
<none>                        8176.4 244.11
- status:section  1    744.68 8921.1 246.03

Step:  AIC=242.17
grades ~ status + section + status:section

                 Df Sum of Sq    RSS    AIC
<none>                        8187.7 242.17
- status:section  1    741.89 8929.6 244.07

Call:
lm.default(formula = grades ~ status + section + status:section)

Coefficients:
     (Intercept)           status1          section1  status1:section1
          62.623           -11.562            -1.720            -4.119
```

```
> muij<-rep(1:4,c(20,1,1,25))
```

Here are some cell means for a 2x2 factorial with unequal replication. The means are 1 larger in the second row than the first, and 2 larger in the second column than the first. The means are additive.

```
> a<-factor(rep(c(1,2,1,2),c(20,1,1,25)))

> b<-factor(rep(c(1,1,2,2),c(20,1,1,25)))

> y<-muij+1.5*rnorm(47)
```

Our data will be the cell means plus random normal errors (the 47 because we need 47 data values).

```
> fit3 <- lm(y~a*b)

> Anova(fit3,type=3)
```

Neither a, nor b, nor a:b is significant using type II or Type III. What gives?

```
Anova Table (Type III tests)

Response: y
             Sum Sq Df F value    Pr(>F)
(Intercept)  39.501  1 15.5629 0.0002901 ***
a             7.938  1  3.1275 0.0840743 .
b             0.981  1  0.3864 0.5374917
a:b           0.505  1  0.1988 0.6579326
Residuals   109.140 43
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

> Anova(fit3,type=2)
Anova Table (Type II tests)
```

```
Response: y
           Sum Sq Df F value  Pr(>F)
a           7.957  1  3.1351 0.08371 .
b           0.987  1  0.3890 0.53610
a:b         0.505  1  0.1988 0.65793
Residuals 109.140 43
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> anova(lm(y~a:b))
```

Here we lump all the df for the model together into one term; it is highly significant! The reason this happens is that the data are structured so that either a or b could explain the variation, but a is not needed if b is present, and b is not needed if a is present. Thus neither a nor b looks significant when tested, but they are part of significant model.

```
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq F value    Pr(>F)
a:b        3  85.898 28.6325  11.281 1.377e-05 ***
Residuals 43 109.140  2.5381
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> y<-c(1,2,2,3,3,4,4,4,4)+rnorm(9)
```

Now we are going to build some proportionally balanced data. This is a 2x2 with replication 1,2,2,4 and an additive mean structure.

```
> a<-factor(c(1,2,2,1,1,2,2,2,2))

> b<-factor(c(1,1,1,2,2,2,2,2,2))

> anova(lm(y~a*b))
```

These data are not balanced, but they are proportionally balanced. It turns out that that is good enough to make order not matter.

```
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq F value  Pr(>F)
a          1 10.3507 10.3507  6.4621 0.05176 .
b          1  2.0149  2.0149  1.2579 0.31300
a:b        1  1.8662  1.8662  1.1651 0.32970
Residuals  5  8.0088  1.6018
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

> Anova(lm(y~a*b),type=2)
Anova Table (Type II tests)

Response: y
           Sum Sq Df F value  Pr(>F)
a         10.3507  1  6.4621 0.05176 .
b          2.0149  1  1.2579 0.31300
a:b        1.8662  1  1.1651 0.32970
```

```
Residuals  8.0088  5
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

> Anova(lm(y~a*b),type=3)
Anova Table (Type III tests)

Response: y
            Sum Sq Df F value   Pr(>F)
(Intercept) 46.789  1 29.2109 0.002931 **
a            6.646  1  4.1489 0.097254 .
b            0.780  1  0.4867 0.516510
a:b          1.866  1  1.1651 0.329704
Residuals    8.009  5
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```