7 Two-factor Experiments, Continued

In the last lecture and in lab we dealt with the parameters **Stata** (and most software packages) use to fit the additive and the interaction models for two-way ANOVA. The `lincom` command was one way we learned to deal with the parameters. That probably is not the easiest approach in many problems, however. If we learn a little more about the parameters, some information is fairly immediate.

Let's continue with the insecticide problem, where we have 4 poisons and 3 doses. Remember the pattern of population cell means as follows (ignoring marginal means for now):

	Dose		
Insecticide	1	2	3
1	μ_{11}	μ_{12}	μ_{13}
2	μ_{21}	μ_{22}	μ_{23}
3	μ_{31}	μ_{32}	μ_{33}
4	μ_{41}	μ_{42}	μ_{43}

Consider the full interaction model first. The parameterization for this is $\mu_{ij} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij},\ i = 1, \ldots, 4;\ j = 1, \ldots, 3$. We dodged the issue of constraints last time, but recall the problem: There are 12 real parameters (the μ_{ij}), but 20 new parameters ($1 + 4 + 3 + 12$). We need to put 8 constraints (restrictions) on these new parameters to bring us back down to 12. An old standard textbook solution to this, and one that makes the math look a lot simpler (for marginal means at least) is

$$0 = \sum_i \alpha_i = \sum_j \beta_j = \sum_i (\alpha\beta)_{ij} = \sum_j (\alpha\beta)_{ij}$$

(that looks like 9 constraints but one is redundant so it is 8). Software packages like **Stata** and **SAS** use an algorithm called the sweep algorithm that makes a completely different and much more useful set of constraints more natural, though. Effectively, they start adding parameters in the model and as soon as they hit a redundant one, they set the new parameter to 0. The new constraints become

$$0 = \alpha_4 = \beta_3 = (\alpha\beta)_{41} = (\alpha\beta)_{42} = (\alpha\beta)_{43} = (\alpha\beta)_{13} = (\alpha\beta)_{23} = (\alpha\beta)_{33}$$

i.e. if there are I levels of i and J levels of j then any time i reaches level I or j reaches level J then the parameter becomes 0. It is a little easier to see 8 constraints here.

If we plug in these constraints and rewrite all 12 cell means, we see the following pattern:

	Dose		
Insecticide	1	2	3
1	$\mu + \alpha_1 + \beta_1 + (\alpha\beta)_{11}$	$\mu + \alpha_1 + \beta_2 + (\alpha\beta)_{12}$	$\mu + \alpha_1$
2	$\mu + \alpha_2 + \beta_1 + (\alpha\beta)_{21}$	$\mu + \alpha_2 + \beta_2 + (\alpha\beta)_{22}$	$\mu + \alpha_2$
3	$\mu + \alpha_3 + \beta_1 + (\alpha\beta)_{31}$	$\mu + \alpha_3 + \beta_2 + (\alpha\beta)_{32}$	$\mu + \alpha_3$
4	$\mu + \beta_1$	$\mu + \beta_2$	μ

At first this may not seem like much simplification, but let's examine it a bit more carefully. μ is often refereed to as the *grand mean* (this comes from the old textbook parameterization) but here we see $\mu = \mu_{43}$. The last cell of the table has become the *reference group* with all other parameters being deviations from that reference group. β_2 is the difference between doses 2 and 3 for the 4^{th} poison, β_1 is the difference between doses 1 and 3 for the 4^{th} poison. α_3 is the difference between

poisons 3 and 4 for the 3^{rd} dose, α_2 is the difference between poisons 2 and 4 for the 3^{rd} dose, and α_1 is the difference between poisons 1 and 4 for the 3^{rd} dose. The difference between poisons 3 and 4 for the 2^{nd} dose is $\alpha_3 + (\alpha\beta)_{32}$ rather than just α_3 (if the difference does not depend on poison then there is no interaction).

With these constraints then $(\mu_{32} - \mu_{42}) - (\mu_{33} - \mu_{43}) = (\alpha\beta)_{32}$. Recall last lecture we said that **no** interaction (parallel profiles in an interaction plot) was this: If i, i', j, j' are legal indexes, then $\mu_{ij} - \mu_{ij'} = \mu_{i'j} - \mu_{i'j'}$, which is to say the difference between doses j and j' is the same for insecticide i as for insecticide i'; and $\mu_{ij} - \mu_{i'j} = \mu_{ij'} - \mu_{i'j'}$, which is to say the difference between insecticides i and i' is the same for dose j as it is for dose j'. Last week we saw that $[\mu_{ij} - \mu_{ij'}] - [\mu_{i'j} - \mu_{i'j'}] = [(\alpha\beta)_{ij} - (\alpha\beta)_{ij'}] - [(\alpha\beta)_{i'j} - (\alpha\beta)_{i'j'}]$. The constraints that $0 = (\alpha\beta)_{Ij} = (\alpha\beta)_{iJ}$ allow us to simplify greatly the `lincom` command for many such terms. Since $(\mu_{32} - \mu_{42}) - (\mu_{33} - \mu_{43}) = (\alpha\beta)_{32}$ *(only for these constraints, though!)* then a simple `lincom(_b[poison[3]*dose[2]])` gets that term. Better yet, we can have it automatically printed.

Stata Implementation

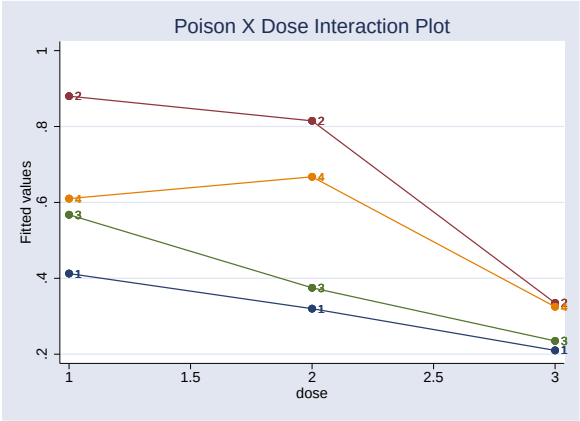
Anova problems actually are specialized regression problems (we will grapple with this idea later). What we want are regression estimates of all the effects $(\mu, \alpha_i, \beta_j, (\alpha\beta)_{ij})$. The `regress` option with `anova` gets that for us in a form that matches the `lincom` syntax. This can be treated as a post-estimation command, i.e. after issuing the `anova time poison dose poison*dose` command (and examining the the ANOVA table, interaction plots, etc.) just type another command `anova,regress` to get the following results

```
. anova, regress
      Source |       SS       df       MS              Number of obs =      48
-------------+------------------------------           F( 11,    36) =    9.01
       Model |  2.20435628    11  .200396025           Prob > F      =  0.0000
    Residual |   .800724989   36  .022242361           R-squared     =  0.7335
-------------+------------------------------           Adj R-squared =  0.6521
       Total |  3.00508126    47  .063937899           Root MSE      =  .14914
------------------------------------------------------------------------------
        time |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |        .325   .0745694     4.36   0.000     .1737663    .4762337
      poison |
          1  |       -.115    .105457    -1.09   0.283    -.3288767    .0988767
          2  |        .01    .105457     0.09   0.925    -.2038767    .2238767
          3  |       -.09    .105457    -0.85   0.399    -.3038767    .1238767
          4  |  (dropped)
      dose |
          1  |       .285    .105457     2.70   0.010     .0711233    .4988767
          2  |      .3425    .105457     3.25   0.003     .1286233    .5563767
          3  |  (dropped)
poison*dose |
        1  1 |     -.0825   .1491387    -0.55   0.584    -.3849674    .2199674
        1  2 |     -.2325   .1491387    -1.56   0.128    -.5349673    .0699674
        1  3 |  (dropped)
        2  1 |       .26   .1491387     1.74   0.090    -.0424673    .5624674
        2  2 |     .1375   .1491387     0.92   0.363    -.1649673    .4399674
        2  3 |  (dropped)
        3  1 |      .0475   .1491387     0.32   0.752    -.2549674    .3499674
        3  2 |     -.2025   .1491387    -1.36   0.183    -.5049673    .0999674
        3  3 |  (dropped)
        4  1 |  (dropped)
        4  2 |  (dropped)
        4  3 |  (dropped)
------------------------------------------------------------------------------
```

Let's try to make sense of these coefficients by relating them both to the table of sample cell means and to an interaction plot from the last lecture. First, try to reconstruct the sample cell means from the coefficients. Recall that the full interaction model imposes no restrictions.

Various questions arise. Why is poison highly significant yet none of the poison coefficients is significant? – Get your answer from the interaction plot and what these coefficients are estimating. One of the large differences appears to be between poisons 1 and 2 at dose level 2. How would you estimate that difference? How would you test for specific interactions (i.e. different slopes in the above plot?). We will spend some time examining all this.

	Dose			
Insecticide	1	2	3	Insect marg
1	.413	.320	.210	.314
2	.880	.815	.335	.677
3	.568	.375	.235	.393
4	.610	.668	.325	.534
Dose marg	.618	.544	.277	.480



The Additive Model

Since interaction was not significant (there is not much data so this *might* just be poor power) we should see how all this looks when we fit the no-interaction (additive) model. This is a highly restricted model and we will not reproduce all the sample cell means from this model. Now we fit $\mu_{ij} = \mu + \alpha_i + \beta_j$ using the command, with ensuing results given below:

```
. anova time poison dose
                          Number of obs =      48     R-squared     =  0.6503
                          Root MSE      = .158179     Adj R-squared =  0.6087
                  Source |  Partial SS    df       MS           F     Prob > F
              -----------+----------------------------------------------------
                   Model |  1.95421877     5  .390843755      15.62     0.0000
                         |
                  poison |   .921206282     3  .307068761      12.27     0.0000
                    dose |  1.03301249     2  .516506246      20.64     0.0000
                         |
                Residual |  1.05086249    42  .025020536
              -----------+----------------------------------------------------
                   Total |  3.00508126    47  .063937899
```

All constraints are as previously described, and easily seen from the following:

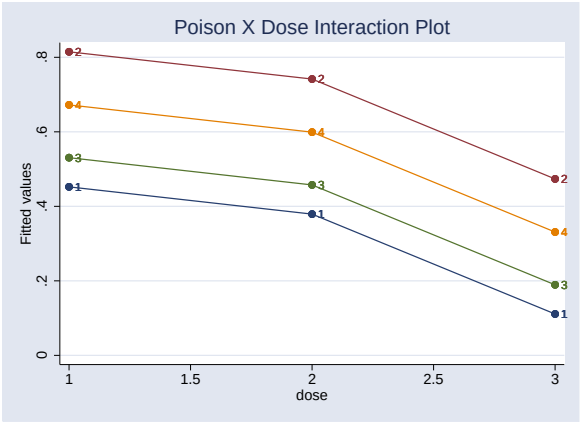
```
. anova,regress
      Source |       SS       df       MS              Number of obs =      48
-------------+------------------------------           F(  5,    42) =   15.62
       Model |  1.95421877     5  .390843755           Prob > F      =  0.0000
    Residual |  1.05086249    42  .025020536           R-squared     =  0.6503
-------------+------------------------------           Adj R-squared =  0.6087
```

```
       Total |  3.00508126    47  .063937899           Root MSE      =  .15818
------------------------------------------------------------------------------
        time |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |   .3310417   .0559247     5.92   0.000     .2181811    .4439022
      poison |
          1 |        -.22   .0645762    -3.41   0.001    -.3503201   -.0896799
          2 |       .1425   .0645762     2.21   0.033     .0121799    .2728201
          3 |  -.1416667   .0645762    -2.19   0.034    -.2719868   -.0113466
          4 |  (dropped)
       dose |
          1 |     .34125   .0559247     6.10   0.000     .2283895    .4541105
          2 |    .268125   .0559247     4.79   0.000     .1552645    .3809855
          3 |  (dropped)
------------------------------------------------------------------------------
```

What we have fit now is the much simpler structure for population cell means (all the parameters have very easy interpretations – what are they?):

	Dose		
Insecticide	1	2	3
1	$\mu+\alpha_1+\beta_1$	$\mu+\alpha_1+\beta_2$	$\mu+\alpha_1$
2	$\mu+\alpha_2+\beta_1$	$\mu+\alpha_2+\beta_2$	$\mu+\alpha_2$
3	$\mu+\alpha_3+\beta_1$	$\mu+\alpha_3+\beta_2$	$\mu+\alpha_3$
4	$\mu+\beta_1$	$\mu+\beta_2$	μ

You should confirm that you no longer reproduce the sample cell means $\bar{y}_{ij.}$ from the estimated regression coefficients, but you do reproduce the sample marginal means $\bar{y}i..$ and $\bar{y}.j.$. We can look at an interaction plot of predicted cell means, but note that we have *forced* it to look this way.



Estimable Functions

What we have been covering is expanded upon at considerable length in the **SAS** manual and many textbooks under the topic of *estimable functions*. This is a fairly advanced topic, but the gist of it is that only linear combinations of *population cell means* can legally be estimated (things of the form $\sum_i \sum_j c_{ij}\mu_{ij}$ for some constants c_{ij} – we have been using 0, 1 and -1 as constants). Anything we estimate or test has to be a linear combination of population cell means. In particular, μ and α_i, for instance, are not estimable since there is ambiguity about what they are until constraints are put on them. Different constraints give different interpretations. What we have been doing is relating everything back to the μ_{ij} in order to keep it all "legal". I can run an analysis on two different packages (or even the same package with different options) and get considerably different estimates of α_1 reported. As long as I stick to estimable functions, though, I always get the same estimate.

Parameters in the One-Way Problem

A look back at the one-way ANOVA problem shows the constraints can make some things simpler there too. In that case we have a model $y_{ij} = \mu_i + \epsilon_{ij};\ i = 1, \ldots, I;\ j = 1, \ldots, n_i$. Let's do the same type of decomposition of μ_i, i.e. $\mu_i = \mu + \alpha_i$, with the same problem that we have I "real" group means and now $I + 1$ new parameters. We need a constraint, and the one **Stata** and **SAS** impose is $\alpha_I = 0$ (set the first redundant parameter to 0).

What is the implication? Now $\mu = \mu_I$ and $\alpha_i = \mu_i - \mu_I$, i.e. the *last* group has become a reference group and all the parameter estimates (the α_i) are deviations from this reference group. In the CHDS example this means we could get mostly for free the t-tests comparing non-smokers to heavy smokers and light smokers to heavy smokers. It might be more convenient to reorder so that nonsmokers were last, so that the easy tests would compare the two smoking groups to nonsmokers. The `lincom` command can fill in the last comparison in either case ($\mu_1 - \mu_2 = \alpha_1 - \alpha_2$). The idea of estimable functions still applies; we need to make sure we are looking at linear combinations of the means.

Unbalanced Data

Returning to the two-way problem, we wrote a model

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk};\ \ i = 1, 2, \ldots, I;\ \ j = 1, 2, \ldots, J;\ \ k = 1, 2, \ldots, K$$

where the sample size in each cell (i, j) was the same number K. Unbalanced data allow the sample size to vary with cell, so now

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk};\ \ i = 1, 2, \ldots, I;\ \ j = 1, 2, \ldots, J;\ \ k = 1, 2, \ldots, K_{ij}$$

What difference does it make? — In practice, not much. The reason the topic gets mentioned is that there are disagreeable aspects to unbalanced designs. There are multiple ways to formulate SSs and F-tests. **SAS** provides 4, Types I-IV, and **Stata** provides the same as **SAS** Types I and III. With balanced data all the types agree, but for unbalanced data they do not all agree. The only reason this is not much of a practical problem is that most analysts use Type III (**Stata**'s default) and don't anguish much over it. The t-tests on coefficients are not obviously affected (i.e. `lincom` results), although comparing main effects in the presence of interaction is a subtle business in unbalanced designs (the preferred approach being least squares means, and Stata makes those awkward to get).

Let's return to the unbalanced example of rat insulin levels from the last lecture. The ANOVA table indicates that the vein and time effects are significant, with p-values of .0001 and .0014, respectively, but that the interaction is not significant (p-value=.575). Recall that the jugular and portal profiles are reasonably parallel, which is consistent with a lack of interaction. Looking at the estimates below, let us figure out as an in-class exercise how to interpret the various coefficient estimates, and how to test for significance of the important effects. We really ought to see if we can transform to a scale where the residuals look better, too.

```
. anova,regress
      Source |       SS       df       MS              Number of obs =      48
-------------+------------------------------           F(  5,    42) =    8.12
       Model |  99478.4833     5  19895.6967           Prob > F      =  0.0000
    Residual |  102914.183    42   2450.3377           R-squared     =  0.4915
-------------+------------------------------           Adj R-squared =  0.4310
       Total |  202392.667    47  4306.22695           Root MSE      =  49.501
------------------------------------------------------------------------------
     insulin |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |       128.5   14.28967     8.99   0.000     99.66227    157.3377
vein         |
          1  |   -67.16667   31.95268    -2.10   0.042    -131.6498   -2.68354
          2  |   (dropped)
             |
time         |
          1  |   -46.58333   20.20865    -2.31   0.026    -87.36604   -5.800623
          2  |        44.4   21.19501     2.09   0.042     1.626732    87.17327
          3  |   (dropped)
             |
vein*time    |
        1  1 |       11.85   41.41541     0.29   0.776    -71.72969    95.42969
        1  2 |   -26.23333    40.9194    -0.64   0.525     -108.812    56.34536
        1  3 |   (dropped)
        2  1 |   (dropped)
        2  2 |   (dropped)
        2  3 |   (dropped)
------------------------------------------------------------------------------

. anova insulin vein time
                           Number of obs =      48     R-squared     =  0.4779
                           Root MSE      = 49.0037     Adj R-squared =  0.4424
                  Source |  Partial SS    df       MS           F     Prob > F
              -----------+----------------------------------------------------
                   Model |  96732.5694     3  32244.1898      13.43     0.0000
                         |
                    vein |  51594.4685     1  51594.4685      21.49     0.0000
                    time |  50332.2893     2  25166.1447      10.48     0.0002
                         |
                Residual |  105660.097    44  2401.36585
              -----------+----------------------------------------------------
                   Total |  202392.667    47  4306.22695

. anova,regress
      Source |       SS       df       MS              Number of obs =      48
-------------+------------------------------           F(  3,    44) =   13.43
       Model |  96732.5694     3  32244.1898           Prob > F      =  0.0000
    Residual |  105660.097    44  2401.36585           R-squared     =  0.4779
-------------+------------------------------           Adj R-squared =  0.4424
       Total |  202392.667    47  4306.22695           Root MSE      =  49.004
------------------------------------------------------------------------------
     insulin |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |    129.6685   13.03897     9.94   0.000     103.3902    155.9468
vein         |
          1  |   -73.00912   15.75087    -4.64   0.000    -104.7529   -41.26531
          2  |   (dropped)
             |
time         |
          1  |   -42.54816   17.42256    -2.44   0.019    -77.66102   -7.435306
          2  |    35.58493   17.82622     2.00   0.052    -.3414601    71.51132
          3  |   (dropped)
------------------------------------------------------------------------------
```

To see one difference with the unbalanced design, consider the following two ANOVA tables; the first is the usual one, the second is an optional one. Note the differences for SS of main effects. If the data were balanced, the default (SAS Type III SS) and the sequential (SAS Type I SS) would be the same. The second form even depends upon the order terms are entered into the model.

```
. anova insulin vein time vein*time
                          Number of obs =      48     R-squared     =  0.4915
                          Root MSE      = 49.5009     Adj R-squared =  0.4310
                  Source |  Partial SS    df       MS           F     Prob > F
              -----------+----------------------------------------------------
                   Model |  99478.4833     5  19895.6967       8.12     0.0000
                         |
                    vein |  48212.7037     1  48212.7037      19.68     0.0001
                    time |   37734.188     2   18867.094       7.70     0.0014
                vein*time |   2745.9139     2  1372.95695       0.56     0.5752
                         |
                Residual |  102914.183    42   2450.3377
              -----------+----------------------------------------------------
                   Total |  202392.667    47  4306.22695
. anova insulin vein time vein*time,seq
                          Number of obs =      48     R-squared     =  0.4915
                          Root MSE      = 49.5009     Adj R-squared =  0.4310
                  Source |    Seq. SS     df       MS           F     Prob > F
              -----------+----------------------------------------------------
                   Model |  99478.4833     5  19895.6967       8.12     0.0000
                         |
                    vein |  46400.2801     1  46400.2801      18.94     0.0001
                    time |  50332.2893     2  25166.1447      10.27     0.0002
                vein*time |   2745.9139     2  1372.95695       0.56     0.5752
                         |
                Residual |  102914.183    42   2450.3377
              -----------+----------------------------------------------------
                   Total |  202392.667    47  4306.22695
```

Regression on Dummy Variables: Stata's `xi`

The way `anova` works is that it creates special variables called indicator or *dummy* variables for categorical variables and performs regression on them. One dummy variable is created for each level of a categorical variable and has a value of 1 if the observation has that level of the category else the value is 0. We do not need to worry much about this if we can use the `anova` command, but if we want to do all this in logistic regression we have to get explicit about it. All the big statistics packages do this, and if we were using SAS I could hide it all from you, but Stata requires you to learn about it. The `xi` facility in Stata is one we will need for many problems.

Following is the insecticide data analyzed as a regression problem using xi:

```
. xi: regress time i.poison i.dose i.poison*i.dose
i.poison          _Ipoison_1-4        (naturally coded; _Ipoison_1 omitted)
i.dose            _Idose_1-3          (naturally coded; _Idose_1 omitted)
i.poi~n*i.dose    _IpoiXdos_#_#       (coded as above)
      Source |       SS       df       MS              Number of obs =      48
-------------+------------------------------           F( 11,    36) =    9.01
       Model |  2.20435628    11  .200396025           Prob > F      =  0.0000
    Residual |  .800724989    36  .022242361           R-squared     =  0.7335
-------------+------------------------------           Adj R-squared =  0.6521
       Total |  3.00508126    47  .063937899           Root MSE      =  .14914
------------------------------------------------------------------------------
        time |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  _Ipoison_2 |      .4675    .105457     4.43   0.000     .2536233    .6813767
  _Ipoison_3 |       .155    .105457     1.47   0.150    -.0588767    .3688767
  _Ipoison_4 |      .1975    .105457     1.87   0.069    -.0163767    .4113767
    _Idose_2 |     -.0925    .105457    -0.88   0.386    -.3063767    .1213767
    _Idose_3 |     -.2025    .105457    -1.92   0.063    -.4163767    .0113767
  _Ipoison_2 |  (dropped)
  _Ipoison_3 |  (dropped)
  _Ipoison_4 |  (dropped)
```

```
    _Idose_2 |  (dropped)
    _Idose_3 |  (dropped)
_IpoiXdo~2_2 |      .0275   .1491387     0.18   0.855    -.2749674    .3299674
_IpoiXdo~2_3 |     -.3425   .1491387    -2.30   0.028    -.6449674   -.0400326
_IpoiXdo~3_2 |        -.1   .1491387    -0.67   0.507    -.4024674    .2024674
_IpoiXdo~3_3 |       -.13   .1491387    -0.87   0.389    -.4324674    .1724674
_IpoiXdo~4_2 |        .15   .1491387     1.01   0.321    -.1524674    .4524674
_IpoiXdo~4_3 |     -.0825   .1491387    -0.55   0.584    -.3849674    .2199674
       _cons |      .4125   .0745694     5.53   0.000     .2612663    .5637337
------------------------------------------------------------------------------
```

You should do this and then examine the variables Stata has placed in the data set. This does not give us the tests from the ANOVA table. The test on interaction is easy:

```
. testparm _IpoiXdo*
 ( 1)  _IpoiXdos_2_2 = 0
 ( 2)  _IpoiXdos_2_3 = 0
 ( 3)  _IpoiXdos_3_2 = 0
 ( 4)  _IpoiXdos_3_3 = 0
 ( 5)  _IpoiXdos_4_2 = 0
 ( 6)  _IpoiXdos_4_3 = 0
       F(  6,    36) =    1.87
            Prob > F =    0.1123
```

but the tests on main effects (equality of marginal means) are a lot less obvious (don't worry, you won't have to do it this way for anova problems). Because this is such a mess, we probably would not test for main effects if interaction were present if we had to go through these steps. With no interaction the procedure is just like the test above.

```
 . test _Ipoison_2 + (_IpoiXdos_2_2+_IpoiXdos_2_3)/3 = _Ipoison_3 + (_IpoiXdos_3_2+_IpoiXdo
> s_3_3)/3 = _Ipoison_4 + (_IpoiXdos_4_2+_IpoiXdos_4_3)/3 =0
 ( 1)  _Ipoison_2 - _Ipoison_3 + .3333333 _IpoiXdos_2_2 + .3333333 _IpoiXdos_2_3 - .333333
> 3 _IpoiXdos_3_2 - .3333333 _IpoiXdos_3_3 = 0
 ( 2)  _Ipoison_2 - _Ipoison_4 + .3333333 _IpoiXdos_2_2 + .3333333 _IpoiXdos_2_3 - .333333
> 3 _IpoiXdos_4_2 - .3333333 _IpoiXdos_4_3 = 0
 ( 3)  _Ipoison_2 + .3333333 _IpoiXdos_2_2 + .3333333 _IpoiXdos_2_3 = 0
       F(  3,    36) =   13.81
            Prob > F =    0.0000

 . test _Idose_2+(_IpoiXdos_2_2+_IpoiXdos_3_2+_IpoiXdos_4_2)/4 =_Idose_3+(_IpoiXdos_2_3+_I
> poiXdos_3_3+_IpoiXdos_4_3)/4=0
 ( 1)  _Idose_2 - _Idose_3 + .25 _IpoiXdos_2_2 - .25 _IpoiXdos_2_3 + .25 _IpoiXdos_3_2 - .
> 25 _IpoiXdos_3_3 + .25 _IpoiXdos_4_2 - .25 _IpoiXdos_4_3 = 0
 ( 2)  _Idose_2 + .25 _IpoiXdos_2_2 + .25 _IpoiXdos_3_2 + .25 _IpoiXdos_4_2 = 0
       F(  2,    36) =   23.22
            Prob > F =    0.0000
```

Annoyingly, Stata has changed constraints on us (now the first level of a categorical variable gets zeroed out). We can set what level gets zeroed out (and thus becomes the reference level) with the `char` command

```
. char poison[omit] 4
. char dose[omit] 3
. xi: regress time i.poison i.dose i.poison*i.dose
```

Execute these commands and confirm the original parameters from `anova,regress` are reproduced. We rarely use `xi` and explicit regression on dummy variables in anova problems. We will need it with logistic regression, though.