How can I get anova type III results using mixed in Stata?

Authored by

stats writer

July 1, 2024

The process of obtaining ANOVA Type III results using mixed in Stata involves using the mixed command in Stata to fit a linear mixed effects model. This model takes into account both fixed and random effects, allowing for the analysis of data with both within-group and between-group variation. By specifying the appropriate syntax and variables, the mixed command can generate ANOVA Type III results, which provide information on the overall significance of each variable in the model, while controlling for other variables. This method is commonly used in the analysis of complex data sets, such as longitudinal studies or clustered data.

How can I get anova type III results using mixed? | Stata FAQ

Consider an anova design with three between-subject factors; t (2 levels), c (3 levels) and p (4 levels) and one within-subject factor s with 2 levels. The within-subject factor is a pre/post variable for 350 subjects. If we were to run an anova the Stata code would look like this:

anova y t##c##p / id|t#c#p s##t##c##p

With 350 subjects this design is huge. It will not run in Stata/IC and will only run in Stata/SE when given lots of memory, variables, matsize and time. One alternative is to run the model using mixed something like this:

mixed y t##c##p##s || id:

This works fine but Stata's factor variables use indicator (dummy) coding so it is not easy to produce anova type results with main effects and interactions.
In fact the only term that is equivalent to anova is the s#t#c#p interaction term. It is
possible to compute the other interactions and the main effects using linear
combinations of
coefficients but with a design this large it would almost be prohibitively difficult.

There is another way we could get the results that we want and that is to use a different coding scheme,
say an orthogonal scheme such as Helmert coding. We can do this using a program written
by John Hendrickx called desmat (search desmat). To find out more about desmat
go to the Desmat home page.
Here is the desmat command for our model and its output.

desmat t*c*p*s, defcon(hel)

Desmat generated the following design matrix:

nr Variables Term Parameterization
First Last

1 _x_1 t helm(F)
2 _x_2 _x_3 c helm(F)
3 _x_4 _x_5 t.c helm(F).helm(F)
4 _x_6 _x_8 p helm(F)
5 _x_9 _x_11 t.p helm(F).helm(F)
6 _x_12 _x_17 c.p helm(F).helm(F)
7 _x_18 _x_23 t.c.p helm(F).helm(F).helm(F)
8 _x_24 s helm(F)
9 _x_25 t.s helm(F).helm(F)
10 _x_26 _x_27 c.s helm(F).helm(F)
11 _x_28 _x_29 t.c.s helm(F).helm(F).helm(F)
12 _x_30 _x_32 p.s helm(F).helm(F)
13 _x_33 _x_35 t.p.s helm(F).helm(F).helm(F)
14 _x_36 _x_41 c.p.s helm(F).helm(F).helm(F)
15 _x_42 _x_47 t.c.p.s helm(F).helm(F).helm(F).helm(F)

The desmat output shows us that there are 15 terms in the model with 47 parameters to be estimated. Now we can go ahead and run mixed with these new variables.

mixed y _x* || id:

Performing EM optimization:

Performing gradient-based optimization:

Iteration 0: log restricted-likelihood = 870.89244
Iteration 1: log restricted-likelihood = 870.89251

Computing standard errors:

Mixed-effects REML regression Number of obs = 644
Group variable: id Number of groups = 322

Obs per group: min = 2
avg = 2.0
max = 2

Wald chi2(47) = 185.80
Log restricted-likelihood = 870.89251 Prob > chi2 = 0.0000

--
y | Coef. Std. Err. z P>|z|
-------------+--
_x_1 | .025292 .0057073 4.43 0.000 .014106 .036478
_x_2 | -.0446694 .0060619 -7.37 0.000 -.0565505 -.0327883

_x_3 | -.0217396 .0069802 -3.11 0.002 -.0354205 -.0080588

_x_4 | .0057967 .0121238 0.48 0.633 -.0179655 .029559

_x_5 | -.0017563 .0139603 -0.13 0.900 -.0291181 .0256054

_x_6 | .0149388 .0062873 2.38 0.017 .002616 .0272616

_x_7 | .008656 .0071202 1.22 0.224 -.0052993 .0226114

_x_8 | .0027882 .0082792 0.34 0.736 -.0134388 .0190152

_x_9 | -.0026387 .0125745 -0.21 0.834 -.0272843 .0220069

_x_10 | .0084589 .0142404 0.59 0.553 -.0194519 .0363696

_x_11 | .0024302 .0165585 0.15 0.883 -.0300238 .0348842

_x_12 | .0012552 .0136317 0.09 0.927 -.0254625 .0279728

_x_13 | .0378146 .0150399 2.51 0.012 .0083369 .0672922

_x_14 | -.0144293 .0173664 -0.83 0.406 -.0484668 .0196082

_x_15 | .0026577 .015053 0.18 0.860 -.0268455 .032161

_x_16 | .024938 .0175149 1.42 0.155 -.0093906 .0592666

_x_17 | -.0291296 .0205043 -1.42 0.155 -.0693172 .011058

_x_18 | .0225859 .0272634 0.83 0.407 -.0308494 .0760211

_x_19 | -.0221501 .0300798 -0.74 0.462 -.0811053 .0368052

_x_20 | .0183917 .0347328 0.53 0.596 -.0496834 .0864668

_x_21 | .0079867 .0301059 0.27 0.791 -.0510199 .0669932

_x_22 | .0471881 .0350298 1.35 0.178 -.0214691 .1158453

_x_23 | -.0065515 .0410085 -0.16 0.873 -.0869267

.0738237

_x_24 | .01577 .0029548 5.34 0.000 .0099786 .0215613

_x_25 | -.0178688 .0059096 -3.02 0.002 -.0294515 -.0062861

_x_26 | -.0090746 .0062769 -1.45 0.148 -.0213771 .0032278

_x_27 | -.0075168 .0072277 -1.04 0.298 -.0216828 .0066492

_x_28 | .0105155 .0125538 0.84 0.402 -.0140894 .0351205

_x_29 | -.0079447 .0144554 -0.55 0.583 -.0362767 .0203873

_x_30 | .0078044 .0065102 1.20 0.231 -.0049554 .0205642

_x_31 | -.0186991 .0073727 -2.54 0.011 -.0331494 -.0042489

_x_32 | .010444 .0085728 1.22 0.223 -.0063585 .0272464

_x_33 | .0325985 .0130204 2.50 0.012 .0070789 .0581181

_x_34 | .0001508 .0147454 0.01 0.992 -.0287497 .0290513

_x_35 | .0054314 .0171457 0.32 0.751 -.0281735 .0390362

_x_36 | .0088749 .0141151 0.63 0.530 -.0187902 .03654

_x_37 | .0133038 .0155732 0.85 0.393 -.0172192 .0438267

_x_38 | -.0356037 .0179823 -1.98 0.048 -.0708483 -.0003592

_x_39 | -.0075208 .0155868 -0.48 0.629 -.0380703 .0230287

_x_40 | -.0120109 .018136 -0.66 0.508 -.0475569 .0235351
_x_41 | -.004214 .0212314 -0.20 0.843 -.0458268 .0373987
_x_42 | -.0390237 .0282302 -1.38 0.167 -.0943538 .0163065
_x_43 | -.0165987 .0311465 -0.53 0.594 -.0776446 .0444473
_x_44 | -.0121021 .0359645 -0.34 0.736 -.0825913 .058387
_x_45 | -.0069893 .0311735 -0.22 0.823 -.0680883 .0541097
_x_46 | -.0095715 .0362721 -0.26 0.792 -.0806634 .0615204
_x_47 | -.0971743 .0424627 -2.29 0.022 -.1803998 -.0139489
_cons | -.3523807 .0028536 -123.49 0.000 -.3579737 -.3467877

Random-effects Parameters | Estimate Std. Err.
-----------------------------+--
id: Identity |
sd(_cons) | .0425494 .0024652 .037982 .0476662
-----------------------------+--
sd(Residual) | .0364142 .0014916 .033605 .0394582

LR test vs. linear regression: chibar2(01) = 120.77 Prob >= chibar2 = 0.0000

**In addition to creating the new variables desmat also creates global macros
that contain the variables for each term. Say we were interested in term 6, the
c#p interaction. We could test it by typing, testparm $term6. Since
the results are given in terms of chi-square we can convert it to an F-ratio
by dividing the chi-square value by its degrees of freedom.**

**That's fine, but what we really want is the anova type III F-ratios for all of the terms in the
model. To do that we will write a short forvalues loop.**

```
forvalues i=1/15 {
local term = "$"+"term"+"`i'"
testparm `term'
display "F = " r(chi2)/r(df)
}
```

(1) _x_1 = 0

chi2(1) = 19.64
Prob > chi2 = 0.0000
F = 19.638667

(1) _x_2 = 0
(2) _x_3 = 0

chi2(2) = 62.33
Prob > chi2 = 0.0000
F = 31.166002

(1) _x_4 = 0
(2) _x_5 = 0

chi2(2) = 0.25
Prob > chi2 = 0.8828
F = .12470646

(1) _x_6 = 0
(2) _x_7 = 0
(3) _x_8 = 0

chi2(3) = 7.55
Prob > chi2 = 0.0563

F = 2.5163986

(1) _x_9 = 0
(2) _x_10 = 0
(3) _x_11 = 0

chi2(3) = 0.44
Prob > chi2 = 0.9325
F = .14565519

(1) _x_12 = 0
(2) _x_13 = 0
(3) _x_14 = 0
(4) _x_15 = 0
(5) _x_16 = 0
(6) _x_17 = 0

chi2(6) = 9.73
Prob > chi2 = 0.1366
F = 1.621271

(1) _x_18 = 0
(2) _x_19 = 0
(3) _x_20 = 0
(4) _x_21 = 0

(5) _x_22 = 0

(6) _x_23 = 0

chi2(6) = 3.52

Prob > chi2 = 0.7416

F = .58630685

(1) _x_24 = 0

chi2(1) = 28.48

Prob > chi2 = 0.0000

F = 28.483828

(1) _x_25 = 0

chi2(1) = 9.14

Prob > chi2 = 0.0025

F = 9.1425429

(1) _x_26 = 0

(2) _x_27 = 0

chi2(2) = 3.06

Prob > chi2 = 0.2164

F = 1.530474

(1) _x_28 = 0

(2) _x_29 = 0

chi2(2) = 1.04

Prob > chi2 = 0.5944

F = .52026247

(1) _x_30 = 0

(2) _x_31 = 0

(3) _x_32 = 0

chi2(3) = 9.00

Prob > chi2 = 0.0293

F = 2.999999

(1) _x_33 = 0

(2) _x_34 = 0

(3) _x_35 = 0

chi2(3) = 6.55

Prob > chi2 = 0.0877

F = 2.1835276

(1) _x_36 = 0

(2) _x_37 = 0

(3) _x_38 = 0

(4) _x_39 = 0
(5) _x_40 = 0
(6) _x_41 = 0

chi2(6) = 5.37
Prob > chi2 = 0.4977
F = .89444705

(1) _x_42 = 0
(2) _x_43 = 0
(3) _x_44 = 0
(4) _x_45 = 0
(5) _x_46 = 0
(6) _x_47 = 0

chi2(6) = 8.19
Prob > chi2 = 0.2248
F = 1.3643889

If you don't mind the having chi-square as the test statistic there is an easier way to display your results using destest (included with desmat).

destest

Term Wald chi2 df P > chi2

---

t 19.639 1 0.000**

c 62.332 2 0.000**

p 7.549 3 0.056

t.c 0.249 2 0.883

t.p 0.437 3 0.933

c.p 9.728 6 0.137

t.c.p 3.518 6 0.742

s 28.484 1 0.000**

c.s 3.061 2 0.216

t.c.s 1.041 2 0.594

p.s 9.000* 3 0.029

t.p.s 6.551 3 0.088

c.p.s 5.367 6 0.498

t.c.p.s 8.186 6 0.225

*** p < .05**

**** p < .01**

So that's how to compute anova type III results using mixed.