How can I obtain anova main-effects with dummy coding using the margin command in Stata?

Authored by

stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I obtain anova main-effects with dummy coding using the margin command in Stata?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=163554

The margin command in Stata allows users to calculate main-effects in ANOVA using dummy coding. This method involves creating dummy variables for each level of a categorical variable and using them in a regression model. The margin command then calculates the predicted means for each level of the categorical variable, taking into account the effects of other variables in the model. This approach is useful for analyzing the main effects of categorical variables in ANOVA, and the margin command simplifies the process by automatically generating the necessary dummy variables.

How can get anova main-effects with dummy coding using margin? Stata FAQ

Many researchers like to do their anova using regression with dummy coding but find it confusing when they don't get the same main-effects as in anova. This FAQ will show you how to get those main-effects.

Example 1

Let's begin by showing the normal anova using a dataset called crf24 to use as a comparison.

use https://stats.idre.ucla.edu/stat/stata/faq/crf24, clear

anova y a##b

Number of obs = 32 R-squared = 0.9214

Root MSE = .877971 Adj R-squared = 0.8985

Source | Partial SS df MS F Prob>F

-----------+--

Model | 217 7 31 40.22 0.0000

|

a | 3.125 1 3.125 4.05 0.0554

b | 194.5 3 64.833333 84.11 0.0000

a#b | 19.375 3 6.4583333 8.38 0.0006

|

Residual | 18.5 24 .77083333

-----------+--

Total | 235.5 31 7.5967742

Here is how the above analyses would look using Stata 15's factor variables with the regress command. The regression model will be followed by a test of the interaction, the margins command and the test of the two main effects using the testparm command.

regress y a##b

Source | SS df MS Number of obs = 32

-------------+------------------------------ F(7, 24) = 40.22

Model | 217 7 31 Prob > F = 0.0000

Residual | 18.5 24 .770833333 R-squared = 0.9214

-------------+------------------------------ Adj R-squared = 0.8985

Total | 235.5 31 7.59677419 Root MSE = .87797

--

y | Coef. Std. Err. t P>|t|

-------------+--

2.a | -2 .6208194 -3.22 0.004 -3.281308 -.7186918

|

b |

2 | .25 .6208194 0.40 0.691 -1.031308 1.531308

3 | 3.25 .6208194 5.24 0.000 1.968692 4.531308

4 | 4.25 .6208194 6.85 0.000 2.968692 5.531308

|

a#b |

2 2 | 1 .8779711 1.14 0.266 -.8120434 2.812043

2 3 | .5 .8779711 0.57 0.574 -1.312043 2.312043

2 4 | 4 .8779711 4.56 0.000 2.187957 5.812043

|

_cons | 3.75 .4389856 8.54 0.000 2.843978 4.656022

--

testparm a#b /* test of a#b interaction */

(1) 2.a#2.b = 0
(2) 2.a#3.b = 0
(3) 2.a#4.b = 0

F(3, 24) = 8.38
Prob > F = 0.0006

Even though the interaction is statistically significant we will go ahead and check out the
main effects. We will demonstrate two methods for computing the main effects for this
example. We need to make clear that there are more than two methods of obtaining the
main effects using the margins command. These are just two of the easier methods.

The first method uses testparm with the equal option.

estimates store m1 /* store regression results for later computations */

margins a b, asbalanced post /* margins command for main effects: method 1 */

Adjusted predictions Number of obs = 32

Model VCE : OLS

Expression : Linear prediction, predict()

at : a (asbalanced)

b (asbalanced)

--

| Delta-method

| Margin Std. Err. t P>|t|

-------------+--

a |

1 | 5.6875 .2194928 25.91 0.000 5.234489 6.140511

2 | 5.0625 .2194928 23.06 0.000 4.609489 5.515511

|

b |

1 | 2.75 .3104097 8.86 0.000 2.109346 3.390654

2 | 3.5 .3104097 11.28 0.000 2.859346 4.140654

3 | 6.25 .3104097 20.13 0.000 5.609346 6.890654

4 | 9 .3104097 28.99 0.000 8.359346 9.640654

--

testparm i.a, equal /* a main effect */

(1) - 1bn.a + 2.a = 0

F(1, 24) = 4.05

Prob > F = 0.0554

testparm i.b, equal /* b main effect */

(1) - 1bn.b + 2.b = 0
(2) - 1bn.b + 3.b = 0
(3) - 1bn.b + 4.b = 0

F(3, 24) = 84.11
Prob > F = 0.0000

Next, we demonstrate the second method for main effects using margins with the
dydx option.

estimates restore m1 /* restore regression results */

margins, dydx(a b) asbalanced post /* margins command for main effects: method 2 */

Conditional marginal effects Number of obs = 32
Model VCE : OLS

Expression : Linear prediction, predict()
dy/dx w.r.t. : 2.a 2.b 3.b 4.b
at : a (asbalanced)

b (asbalanced)

| Delta-method
| dy/dx Std. Err. t P>|t|
-------------+--
2.a | -.625 .3104097 -2.01 0.055 -1.265654 .0156541
|
b |
2 | .75 .4389856 1.71 0.100 -.1560217 1.656022
3 | 3.5 .4389856 7.97 0.000 2.593978 4.406022
4 | 6.25 .4389856 14.24 0.000 5.343978 7.156022

Note: dy/dx for factor levels is the discrete change from the base level.

testparm i.a /* a main effect */

(1) 2.a = 0

F(1, 24) = 4.05
Prob > F = 0.0554

testparm i.b /* b main effect */

(1) 2.b = 0

(2) 3.b = 0

(3) 4.b = 0

F(3, 24) = 84.11

Prob > F = 0.0000

Example 2

This method generalizes to more complex designs with multiple factors, so let's consider a 3-factor completely crossed design.

use https://stats.idre.ucla.edu/stat/stata/faq/threeway, clear

anova y a##b##c

Number of obs = 24 R-squared = 0.9689

Root MSE = 1.1547 Adj R-squared = 0.9403

Source | Partial SS df MS F Prob>F

-----------+--

Model | 497.83333 11 45.257576 33.94 0.0000

|

a | 150 1 150 112.50 0.0000

b | .66666667 1 .66666667 0.50 0.4930

a#b | 160.16667 1 160.16667 120.13 0.0000

c | 127.58333 2 63.791667 47.84 0.0000

a#c | 18.25 2 9.125 6.84 0.0104

b#c | 22.583333 2 11.291667 8.47 0.0051

a#b#c | 18.583333 2 9.2916667 6.97 0.0098

|

Residual | 16 12 1.3333333

-----------+--

Total | 513.83333 23 22.34058

And here is the same model using the regress command.

regress y a##b##c

Source | SS df MS Number of obs = 24

-------------+---------------------------------- F(11, 12) = 33.94

Model | 497.833333 11 45.2575758 Prob > F = 0.0000

Residual | 16 12 1.33333333 R-squared = 0.9689

-------------+---------------------------------- Adj R-squared = 0.9403

Total | 513.833333 23 22.3405797 Root MSE = 1.1547

--

y | Coef. Std. Err. t P>|t|

-------------+--

2.a | -.5 1.154701 -0.43 0.673 -3.015876 2.015876

2.b | -.5 1.154701 -0.43 0.673 -3.015876 2.015876

|

a#b |

2 2 | 6.5 1.632993 3.98 0.002 2.942014 10.05799

|

c |

2 | 4 1.154701 3.46 0.005 1.484124 6.515876

3 | 8 1.154701 6.93 0.000 5.484124 10.51588

|

a#c |

2 2 | 1 1.632993 0.61 0.552 -2.557986 4.557986

2 3 | 3.09e-14 1.632993 0.00 1.000 -3.557986 3.557986

|

b#c |

2 2 | -4 1.632993 -2.45 0.031 -7.557986 -.4420135

2 3 | -9 1.632993 -5.51 0.000 -12.55799 -5.442014

|

a#b#c |

2 2 2 | 3 2.309401 1.30 0.218 -2.031753 8.031753

2 2 3 | 8.5 2.309401 3.68 0.003 3.468247 13.53175

|

_cons | 11 .8164966 13.47 0.000 9.221007 12.77899

--

testparm a#b#c /* test of the a#b#c interaction */

(1) 2.a#2.b#2.c = 0
(2) 2.a#2.b#3.c = 0

F(2, 12) = 6.97
Prob > F = 0.0098

Before we get to the main effects, we will test the three two-way interactions.

estimates store m1 /* store regression results for later computations */

margins, dydx(a) over(b) asbal post noatlegend /* margins for a#b interaction */

Conditional marginal effects Number of obs = 24
Model VCE : OLS

Expression : Linear prediction, predict()
dy/dx w.r.t. : 2.a

over : b

| Delta-method

| dy/dx Std. Err. t P>|t|

---------+---

1.a | (base outcome)

---------+---

2.a |

b |

1 | -.1666667 .6666667 -0.25 0.807 -1.619209 1.285875

2 | 10.16667 .6666667 15.25 0.000 8.714125 11.61921

Note: dy/dx for factor levels is the discrete change from the base level.

test 1.b=2.b /* test of a#b interaction */

(1) 1bn.b - 2.b = 0

F(1, 12) = 120.12

Prob > F = 0.0000

estimates restore m1

margins, dydx(a) over(c) asbal post noatlegend /*

margins for a#c interaction */

Conditional marginal effects Number of obs = 24
Model VCE : OLS

Expression : Linear prediction, predict()
dy/dx w.r.t. : 2.a
over : c

```
------------------------------------------------------
| Delta-method
| dy/dx Std. Err. t P>|t|
---------+----------------------------------------
1.a | (base outcome)
---------+----------------------------------------
2.a |
c |
1 | 2.75 .8164966 3.37 0.006 .9710068 4.528993
2 | 5.25 .8164966 6.43 0.000 3.471007 7.028993
3 | 7 .8164966 8.57 0.000 5.221007 8.778993
------------------------------------------------------
```

Note: dy/dx for factor levels is the discrete change from the base level.

test (1.c=2.c)(1.c=3.c) /* test of a#c interaction */

(1) 1bn.c - 2.c = 0

(2) 1bn.c - 3.c = 0

F(2, 12) = 6.84

Prob > F = 0.0104

estimates restore m1

margins, dydx(b) over(c) asbal post noatlegend /* margins for b#c interaction */

Conditional marginal effects Number of obs = 24

Model VCE : OLS

Expression : Linear prediction, predict()

dy/dx w.r.t. : 2.b

over : c

| Delta-method

| dy/dx Std. Err. t P>|t|

---------+--

1.b | (base outcome)

---------+--

2.b |

c |

1 | 2.75 .8164966 3.37 0.006 .9710068 4.528993

2 | .25 .8164966 0.31 0.765 -1.528993 2.028993

3 | -2 .8164966 -2.45 0.031 -3.778993 -.2210068

Note: dy/dx for factor levels is the discrete change from the base level.

test (1.c=2.c)(1.c=3.c) /* test of b#c interaction */

(1) 1bn.c - 2.c = 0

(2) 1bn.c - 3.c = 0

F(2, 12) = 8.47

Prob > F = 0.0051

Finally, we will compute the main effects using testparm with method 2 as shown above.

estimates restore m1

margins, dydx(a b c) asbalanced post /* margins command for main effects */

Conditional marginal effects Number of obs = 24

Model VCE : OLS

Expression : Linear prediction, predict()

dy/dx w.r.t. : 2.a 2.b 2.c 3.c

at : a (asbalanced)

b (asbalanced)

c (asbalanced)

| Delta-method

| dy/dx Std. Err. t P>|t|

---------+---

2.a | 5 .4714045 10.61 0.000 3.972898 6.027102

2.b | .3333333 .4714045 0.71 0.493 -.6937689 1.360436

|

c |

2 | 3.25 .5773503 5.63 0.000 1.992062 4.507938

3 | 5.625 .5773503 9.74 0.000 4.367062 6.882938

Note: dy/dx for factor levels is the discrete change from the base level.

testparm i.a /* a main-effect */

(1) 2.a = 0

F(1, 12) = 112.50

Prob > F = 0.0000

testparm i.b /* b main-effect */

(1) 2.b = 0

F(1, 12) = 0.50
Prob > F = 0.4930

testparm i.c /* c main-effect */

(1) 2.c = 0
(2) 3.c = 0

F(2, 12) = 47.84
Prob > F = 0.0000