How can I perform power and robustness analyses for factorial ANOVA in Stata?

Authored by

stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I perform power and robustness analyses for factorial ANOVA in Stata?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=164156

ARABPSYCHOLOGY.COM

Factorial ANOVA is a statistical technique used to analyze the effects of multiple independent variables on a continuous dependent variable. It allows for the examination of interactions between the independent variables, providing a more comprehensive understanding of the relationship between the variables. In Stata, there are two important analyses that can be performed to assess the quality and reliability of the results from factorial ANOVA: power analysis and robustness analysis.

Power analysis is a statistical tool that helps determine the sample size needed for a study to detect a certain effect size with a desired level of statistical power. This is important because a study with a small sample size may not have enough power to detect significant effects, while a large sample size may result in unnecessary costs and time. In Stata, power analysis for factorial ANOVA can be performed using the command "power anova factorial."

Robustness analysis, on the other hand, is used to assess the stability and reliability of the results obtained from the factorial ANOVA. It involves testing the robustness of the conclusions by varying the assumptions and conditions of the analysis. This helps to identify potential outliers or influential data points that may affect the results. In Stata, robustness analysis for factorial ANOVA can be performed using the command "rvfplot."

Overall, performing power and robustness analyses for factorial ANOVA in Stata is crucial in ensuring the accuracy and validity of the results obtained. These analyses allow researchers to make informed decisions about sample size, detect potential issues with the data, and strengthen the overall reliability of the findings.

ARABPSYCHOLOGY.COM

How can I do power and robustness analyses for factorial anova? | Stata FAQ

One way to assess the power of a factorial anova design is through the use of Monte-Carlo simulation. With this approach, one generate hundreds (or thousands) of randomly generated datasets. Analyze each one and retain the p-values from the analyses. Then it is a simple matter to count up the

proportion of p-values that are less than or equal to your nominal alpha level.

The user written program factorialsim (search factorialsim) will perform Monte-Carlo power analyses for two-way factorial anova designs. To use factorialsim you need to create a data file named simdat.dta. This data file will contain one row for each cell in your design. Here is an example:

```
list, sep(0)
```

```
+---------------------+
| a b n m s |
|---------------------|
1. | 1 1 8 20 9 |
2. | 1 2 8 27 9 |
3. | 1 3 8 19 9 |
4. | 2 1 12 20 10 |
5. | 2 2 12 25 10 |
6. | 2 3 12 30 10 |
+---------------------+
```

Variable a is the level of the first factor variable;

ARABPSYCHOLOGY.COM

variable b is the level
of the second factor variable; n is the number of observations in the cell; m
is the cell mean; and s is the cell standard deviation. The variables in the data file
must have the exact same names as shown above.

For each replication, factorialsim expands the simdat dataset on variable n.
Then, the program generates a random response variable with mean equal m and standard deviation equal s. The program makes use of Stata's
simulate command to collect and retain the Monte-Carlo results before
displaying the observed proportion of each of the p-values.

The examples below show how to use factorialsim for power and robustness analyses.

Example 1 - Prospective Power Analysis

Let's say that a researcher has decided that a 2×3 factorial design meets the need of his
research project. Through a literature review and a

ARABPSYCHOLOGY.COM

pilot-study he has, what he thinks,
are reasonable estimates for cell means and standard deviations. Further, he figures that
it will be fairly easy to obtain 60 subjects for his studies. Of course, he will need to
find more if the power analysis reveals low power especially for the important interaction hypothesis.

The researcher has also decided to put more subjects into the experimental groups than into the
control groups. Below is the code the researcher used to create the simdat data file.

```
clear
input a b n m s
1 1 8 20 9
1 2 8 27 9
1 3 8 19 9
2 1 12 20 10
2 2 12 25 10
2 3 12 30 10
end
save simdat, replace
```

Now that the simdat data file has been created we will run the most basic factorialsim

ARABPSYCHOLOGY.COM

command.

`factorialsim`

command: factorial_sim, obs(0)
pab: r(pab)
pa: r(pa)
pb: r(pb)

Simulations (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
.. 50
.. 100

alpha = .05 replications = 100

simulated power for a*b = .54
simulated power for a = .25
simulated power for b = .48

We won't discuss these results because too few replications (only 100) were used.
We will rerun the command setting the reps option to 1,000. We will also
use the nodots option to suppress the display of the dots for each simulated

ARABPSYCHOLOGY.COM

dataset.

```
factorialsim, reps(1000) nodots
```

command: factorial_sim, obs(0)

pab: r(pab)

pa: r(pa)

pb: r(pb)

alpha = .05 replications = 1000

simulated power for a*b = .486

simulated power for a = .229

simulated power for b = .415

This proposed design seems to be under powered. We would like to get the power up to at least 0.8. We can see what would happen if we were to increase each of the cell sizes to 20 observations using the obs command.

```
factorialsim, reps(1000) obs(20) nodots
```

command: factorial_sim, obs(20)

pab: r(pab)

pa: r(pa)

ARABPSYCHOLOGY.COM

pb: r(pb)

alpha = .05 replications = 1000

simulated power for a*b = .834
simulated power for a = .41
simulated power for b = .75

Now the power looks good for the interaction and is much better for the b main effect.
However, the a main effect still lacks power. Since the interaction has good power the
researcher must decide whether to stick with this sample size or increase it until the
main effect for a has a simulated power of 0.8.

It is also possible to run the analysis with the robust option. The robust
option causes factorialsim to run the model using regress with Huber-White
robust standard errors. Below we will rerun the analysis with obs(20) as in the
previous example.

```
factorialsim, reps(1000) nodots robust obs(20)
```

ARABPSYCHOLOGY.COM

command: factorial_simr, obs(20)
pab: r(pab)
pa: r(pa)
pb: r(pb)

alpha = .05 replications = 1000

simulated power for a*b = .838
simulated power for a = .056
simulated power for b = .748

The power for a*b and b held up very well. However, the power for a
has dropped dramatically due to heteroscedasticity between the two levels of a.

Example 2 - Observed Power Analysis

Observed (post-hoc) power analysis is one that is done after the data have been collected and analyzed.

We begin by loading a dataset (hsbdemo) and running an anova.

```
use https://stats.idre.ucla.edu/stat/data/hsbdemo, clearanova write female##prog
```

Number of obs = 200 R-squared = 0.2590

ARABPSYCHOLOGY.COM

Root MSE = 8.26386 Adj R-squared = 0.2399

Source | Partial SS df MS F Prob > F

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

Model | 4630.36091 5 926.072182 13.56 0.0000

|

female | 1261.85329 1 1261.85329 18.48 0.0000

prog | 3274.35082 2 1637.17541 23.97 0.0000

female#prog | 325.958189 2 162.979094 2.39 0.0946

|

Residual | 13248.5141 194 68.2913097

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

Total | 17878.875 199 89.843593

Next, we create a dataset that contains the frequency, mean and standard deviation for each cell using the collapse command.

```
collapse collapse (mean) m=write (sd) s=write (count) n=write, by(female
prog)rename female a
rename prog b
```

list, clean nolabel noobs

a b m s n

0 1 49.14286 10.36478 21

ARABPSYCHOLOGY.COM

0 2 54.61702 8.656622 47

0 3 41.82609 8.003705 23

1 1 53.25 8.205248 24

1 2 57.58621 7.115672 58

1 3 50.96296 8.341193 27

```
save simdat, replace
```

Now we can run factorialsim to get the Monte-Carlo post-hoc power estimates.

```
factorialsim, reps(1000) nodots
```

command: factorial_sim, obs(0)
pab: r(pab)
pa: r(pa)
pb: r(pb)

alpha = .05 replications = 1000

simulated power for a*b = .49
simulated power for a = .98
simulated power for b = 1

The power for the a and b main effects are more than adequate while the power for the

ARABPSYCHOLOGY.COM

interaction is a bit on the low side. If we play around with the obs option
we will see that it requires around 65 observation per cell to obtain a power of
.8 for the interaction this model.

```
factorialsim, reps(1000) obs(65) nodots
```

command: factorial_sim, obs(65)
pab: r(pab)
pa: r(pa)
pb: r(pb)

alpha = .05 replications = 1000

simulated power for a*b = .814
simulated power for a = 1
simulated power for b = 1

Example 3 - Robustness Analysis

To assess robustness of a model, you just set all the mean values to be equal. factorialsim
accomplishes this by setting all of the means to zero. Next you run the Monte-Carlo simulation.
If the portion of p-values is close to the nominal alpha

ARABPSYCHOLOGY.COM

level then the model displays good robustness. Depending on cell sizes and variances the observed p-values can be much smaller then alpha or much larger.

Just to show that robustness works well under optimal conditions, we will begin with an example in which all the cells are the same size and have the same standard deviations.

```
clear
input a b n m s
```

1 1 20 20 10

1 2 20 27 10

1 3 20 19 10

2 1 20 20 10

2 2 20 25 10

2 3 20 30 10

```
end
save simdat, replacefactorialsim, reps(1000) nodots zero
```

command: factorial_sim, obs(0) zero

pab: r(pab)

pa: r(pa)

pb: r(pb)

alpha = .05 replications = 1000

ARABPSYCHOLOGY.COM

simulated robustness for a*b = .047

simulated robustness for a = .05

simulated robustness for b = .056

Note that all of the Monte-Carlo simulated alpha values are very close to the nominal 0.05 alpha levels.

Next, we will run an example in which cell size and variability differ. In this case the smaller cells in the design have the larger variability.

```
clear
input a b n m s
```

1 1 10 20 20

1 2 10 27 20

1 3 10 19 20

2 1 30 20 10

2 2 30 25 10

2 3 30 30 10

end

save simdat, replace

```
factorialsim, reps(1000) nodots zero
```

ARABPSYCHOLOGY.COM

command: factorial_sim, obs(0) zero

pab: r(pab)

pa: r(pa)

pb: r(pb)

alpha = .05 replications = 1000

simulated robustness for a*b = .214

simulated robustness for a = .148

simulated robustness for b = .207

The observed proportions across our 1,000 samples is much larger that the nominal 0.05 level that is assumed.

Here's what this analysis with the robust option looks like.

```
factorialsim, reps(1000) nodots zero robust
```

command: factorial_simr, obs(0) zero

pab: r(pab)

pa: r(pa)

pb: r(pb)

alpha = .05 replications = 1000

ARABPSYCHOLOGY.COM

simulated robustness for a*b = .088

simulated robustness for a = .099

simulated robustness for b = .087

The values with robust option are much better.

Now let's run it again, this time with the smaller cells in the design having the lower variability.

```
clear
input a b n m s
```

1 1 10 20 10

1 2 10 27 10

1 3 10 19 10

2 1 30 20 20

2 2 30 25 20

2 3 30 30 20

```
end
save simdat, replacefactorialsim, reps(1000) nodots zero
```

command: factorial_sim, obs(0) zero

pab: r(pab)

pa: r(pa)

pb: r(pb)

alpha = .05 replications = 1000

ARABPSYCHOLOGY.COM

simulated robustness for a*b = .003
simulated robustness for a = .005
simulated robustness for b = .002

In this case the observed proportions are all much lower than the nominal alpha level of 0.05.

Once again, we will rerun the analysis with the robust option.

```
factorialsim, reps(1000) nodots zero robust
```

command: factorial_simr, obs(0) zero
pab: r(pab)
pa: r(pa)
pb: r(pb)

alpha = .05 replications = 1000

simulated robustness for a*b = .055
simulated robustness for a = .064
simulated robustness for b = .088

The p-values, once again, are much closer to the nominal 0.05 levels than without the

ARABPSYCHOLOGY.COM

robust option.

Thus, factorialsim can be a useful tool in exploring robustness in two-way factorial designs.

ARABPSYCHOLOGY.COM