Simulation-based power analysis for linear and generalized linear models

Joerg Luedicke

Yale University & University of Florida

Outline

1. Introduction
2. The simulation-based approach
3. Stata module powersim
4. Example 1
5. Example 2
6. Outlook
Significance testing and statistical power

- Point null hypothesis significance testing
- Type I & Type II error
  - Type I: Reject $H_0$ when it is true
  - Type II: Failure to reject $H_0$ when it is false
  - Type I & Type II trade-off
- Statistical power
  - $\beta \Rightarrow$ probability of not rejecting $H_0$ when it is false
  - Power $\Rightarrow 1 - \beta$
  - i.e., the probability of rejecting $H_0$, given it is indeed false
- Importance of power analysis
  - Study planning
  - Reasonable resource allocation
  - Saving time and money
Analytical vs. simulation-based approaches

- **Analytical approach**
  - A number of formulas have been derived for some standard situations (e.g., difference in means between two groups).
  - Usually, these formulas are fairly restrictive with respect to the underlying assumptions,
  - and are not very flexible with regard to a user’s potential needs.

- **Simulation-based method**
  - A simulation-based approach is most flexible,
  - since it allows to perform power analyses for complex and/or highly specific scenarios.
  - Downside: computation time
The simulation procedure

Simulation procedure

1. Generate synthetic data, based on an assumed model, model parameters, and covariate distributions
2. Fit a model to the synthetic data
3. Do the significance test of interest and record the p-value
4. Repeat 1.-3. many times
5. The statistical power is the proportion of p-values that are lower than a specified $\alpha$-level
The **powersim** command

- Flexible power analysis for linear and generalized linear models
- Automated simulations, based on user input via command options
- **powersim** creates a do-file that is used for generating predictor data
- The do-file can be modified for more complex synthetic datasets and/or user defined link functions
- The analysis model can be specified using Stata’s **regress** or **glm** commands
- A summary of results is shown in the results pane
- Simulation results from each replication are stored in a dataset
- Power curves can be plotted using **powersimplot**
Specification of a data generating model

- Users can choose a distributional family,
- a link function,
- covariates with specified distributions,
- effect sizes for the respective regression parameters,
- correlated predictor variables (for Gaussian variables),
- interaction effects
Available distributional families

- Gaussian
- Inverse Gaussian
- Gamma
- Poisson
- Binomial
- Negative binomial
## Available link functions

<table>
<thead>
<tr>
<th>Link function</th>
</tr>
</thead>
<tbody>
<tr>
<td>identity</td>
</tr>
<tr>
<td>log</td>
</tr>
<tr>
<td>logit</td>
</tr>
<tr>
<td>probit</td>
</tr>
<tr>
<td>complementary log-log</td>
</tr>
<tr>
<td>odds power</td>
</tr>
<tr>
<td>power</td>
</tr>
<tr>
<td>negative binomial</td>
</tr>
<tr>
<td>log-log</td>
</tr>
<tr>
<td>log-complement</td>
</tr>
</tbody>
</table>
### Available covariate distributions

<table>
<thead>
<tr>
<th>Covariate distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal</td>
</tr>
<tr>
<td>Poisson</td>
</tr>
<tr>
<td>uniform</td>
</tr>
<tr>
<td>binomial</td>
</tr>
<tr>
<td>$\chi^2$</td>
</tr>
<tr>
<td>Student’s t</td>
</tr>
<tr>
<td>beta</td>
</tr>
<tr>
<td>gamma</td>
</tr>
<tr>
<td>negative binomial</td>
</tr>
<tr>
<td>equally sized groups</td>
</tr>
<tr>
<td>2x2 block design</td>
</tr>
</tbody>
</table>
Example 1: Simple comparison of means in a linear model

- Suppose we would like to compare two independent means and calculate power for varying mean differences, measured in standard deviation units, and a varying number of sample sizes.

- In Stata, we can calculate the statistical power for the different effect and sample size combinations with the `power` command:

```stata
power twomeans 0 (0.4 0.5 0.6), n(10(10)100) ///
graph(ylabel(0(.1)1) title("")) subtitle("")) ///
xval recast(line))
```
Example 1: mean differences (with Stata’s `power` command)

Parameters: \( \alpha = .05, \mu = 0, \sigma = 1 \)
Example 1: Simple comparison of means in a linear model

Now we can replicate these results using simulations (assuming a linear model with Gaussian error and two equally sized (fixed) groups):

**powersim code:**

```
powersim , ///
b(0.4 0.5 0.6) ///
alpha(0.05) ///
pos(1) ///
sample(10(10)100) ///
nreps(10000) ///
family(gaussian 1) ///
link(identity) ///
cov1(x1 _bp block 2) ///
dofile(ex1_dofile, replace) : reg y x1
```
Example 1: mean differences (*powersim* command)

![Graph showing power analysis for different sample sizes](image)

- **Power**
- **Sample size**

- **Graph Description:**
  - Legend:
    - Green line, \( b = 0.4 \)
    - Orange line, \( b = 0.5 \)
    - Blue line, \( b = 0.6 \)

- **Parameters:**
  - \( \alpha = 0.05 \)
  - Number of replications per sample and effect size: 10000

---

**Simulation-based power analysis**

Joerg Luedicke

**Introduction**

The simulation-based approach

**Stata module**

*powersim*

**Example 1**

**Example 2**

**Outlook**
Example 2: Poisson regression with an interaction effect and correlated predictors

- Now suppose that we would like to simulate the power for the test of an interaction effect of two correlated predictor variables in a Poisson model.

- The assumed model can be expressed as:
  \[ y \sim \text{Poisson}(\exp(0.5 - 0.25 \times x_1 + 0.4 \times x_2 + _bp \times x_1 \times x_2)), \]

- where \(_bp\) is a placeholder for the various effect sizes for which we simulate the power,

- and \(x_1, x_2 \sim N(\mu, \Sigma)\) with zero means, unit variances, and \(\rho = 0.5\)

- Now, before we fire up the simulations we create a single synthetic dataset (using \texttt{powersim}'s \texttt{gendata()}\ option) in order to check whether the assumed model is consistent with our hypotheses:
Example 2: Poisson regression with an interaction effect and correlated predictors

powersim command:

```stata
powersim , b(0.1) alpha(0.05) pos(3) ///
sample(300) nreps(500) ///
family(poisson) link(log) ///
cov1(x1 -0.25 normal 0 1) ///
cov2(x2 0.4 normal 0 1) ///
inter1(_bp x1*x2) ///
cons(0.5) ///
corr12(0.5) ///
inside ///
gendra /// // <-- creating a single realization
dofile(ex2_dofile, replace) : ///
glm y c.x1##c.x2, family(poisson) link(log)
```
Example 2: Poisson regression with an interaction effect and correlated predictors

Now we could fit the analysis model to the fabricated data:

```
.glm y c.x1##c.x2, family(poisson) link(log) nolog
```

| OIM            |   Coef. |   Std. Err. |     z  |    P>|z| | [95% Conf. Interval] |
|----------------|---------|-------------|--------|-----------|----------------------|
| y              |         |             |        |           |                      |
| x1             | -.2475788 | .0087262 | -28.37 | 0.000     | -.2646817 .2304758  |
| x2             | .3971381  | .0085716  | 46.33  | 0.000     | .380338 .4139383    |
| c.x1#c.x2      | .0994309  | .0056088  | 17.73  | 0.000     | .0884378 .110424    |
| _cons          | .5009491  | .008613   | 58.16  | 0.000     | .484068 .5178303    |
Example 2: Poisson regression, inspecting synthetic data

... and can do some checking WRT to our hypotheses, for example:

Visualization of the interaction effect $0.1 \times x_1 \times x_2$

(Predicted values of outcome $y$ as a function of $x_1$, with $x_2$ fixed at some representative values)
Example 2: Running the simulations

Now we run the simulations by removing the `gendata()` option. We also add a few more sample sizes and add an additional effect size:

**powersim command:**

```
powersim, ///
b(0.07 0.1) alpha(0.05) pos(3) ///
sample(200(50)400) nreps(1000) ///
family(poisson) link(log) ///
cov1(x1 -0.25 normal 0 1) ///
cov2(x2 0.4 normal 0 1) ///
inter1(_bp x1*x2) ///
cons(0.5) corr12(0.5) inside ///
dofile(example2_dofile, replace): ///
glm y c.x1##c.x2, family(poisson) link(log)
```
Power analysis simulations

Effect sizes $b$: .07 .1

H0: $b = 0$

Sample sizes: 200 250 300 350 400

alpha: .05

N of simulations: 1000

do-file used for data generation: example2_dofile

Model command: \texttt{glm y c.x1##c.x2, family(poisson) link(log)}

Power by sample and effect sizes:

<table>
<thead>
<tr>
<th>Sample size</th>
<th>Effect size .07</th>
<th>Effect size .1</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>0.363</td>
<td>0.608</td>
</tr>
<tr>
<td>250</td>
<td>0.416</td>
<td>0.733</td>
</tr>
<tr>
<td>300</td>
<td>0.479</td>
<td>0.792</td>
</tr>
<tr>
<td>350</td>
<td>0.537</td>
<td>0.853</td>
</tr>
<tr>
<td>400</td>
<td>0.607</td>
<td>0.888</td>
</tr>
</tbody>
</table>
Example 2: Power curves

Now we can simply type: powersimplot

alpha = .05; N of replications per sample and effect size: 1000
Example 2: Post-simulation - results dataset

Contains data from /tmp/St01559.00000d
obs: 10,000
vars: 9
size: 510,000

```
. des

Contains data from /tmp/St01559.00000d
obs: 10,000
vars: 9
size: 510,000

variable name    storage    display   value label
variable label

    nd    double    %10.0g    Iteration ID
    b     double    %10.0g    Effect b
    se    double    %10.0g    Standard error of b
    p     double    %10.0g    p-value
    n     double    %10.0g    Sample size
    c95   byte      %8.0g    95% coverage (1=covered)
    power byte      %8.0g    1 = p < .05
    esize double    %10.0g    Effect size
    esize_id byte    %8.0g    eid    Effect size ID
```

Sorted by:  n  esize_id
Example 2: Post-simulation - inspecting simulation results

Example: 95% CI coverage

```
. tabstat c95 if esize_id==2, by(n)
```

Summary for variables: c95
by categories of: n (Sample size)

<table>
<thead>
<tr>
<th>n</th>
<th>mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>.95</td>
</tr>
<tr>
<td>250</td>
<td>.959</td>
</tr>
<tr>
<td>300</td>
<td>.949</td>
</tr>
<tr>
<td>350</td>
<td>.945</td>
</tr>
<tr>
<td>400</td>
<td>.951</td>
</tr>
<tr>
<td>Total</td>
<td>.9508</td>
</tr>
</tbody>
</table>

User-written commands for analyzing simulation results:
- `simsum` from Ian White (SSC)
- `simpplot` from Maarten Buis (SSC)
Implementing additional features:

- More models:
  - (un)ordered categorical
  - zero-inflated count models
  - beta regression
  - random effects models
  - meglm

- Correlated predictor data:
  - binary-binary
  - binary-normal

- Dialog box (?)
Thank you!

Contact:
joerg.luedicke@ufl.edu