power logistic onebin — Power analysis for logistic regression with one binary covariate+

+This command is part of StataNow.

Description

`power logistic` computes sample size, power, or effect size for a test of one coefficient in logistic regression. This entry describes how to use `power logistic` to plan a study that will be modeled using logistic regression with one binary covariate. For information about how to use `power logistic` with two binary covariates, see [PSS-2] **power logistic twobin**. To use `power logistic` with many covariates that can be continuous, discrete, or both, see [PSS-2] **power logistic general**.

By default, `power logistic` computes sample size for a given power and effect size, where the effect size may be specified as a coefficient or an odds ratio. Alternatively, it can compute power given sample size and effect size, or it can compute the effect size given power and sample size.

Quick start

Sample size for logistic regression of binary outcome Y on one binary covariate X, given an odds ratio of 1.5 for the effect of X on Y under the alternative hypothesis, population prevalences of X and Y of 0.22 and 0.13, and default power of 0.8 and significance level $\alpha = 0.05$

```
power logistic 1.5, px(0.22) py(0.13)
```

Same as above, but instead of using argument *oratio*$_X$ and the `py()` option to specify parameters for the logistic regression, we specify the respective `coefx()` and `intercept()` options

```
power logistic, px(0.22) coefx(0.4055) intercept(-2)
```

Same as above, but instead specify the odds that $X = 1$ and the outcome success probability conditional on $X = 1$ and $X = 0$

```
power logistic, oddsx(0.2821) pycondx1(0.1686) pycondx0(0.1191)
```

Sample-size calculation for all possible combinations of the specified values

```
power logistic (1.25 1.5 1.75 2), px(0.2 0.22 0.24) py(0.13)
```

Power given an odds ratio of 1.5, population prevalences of X and Y of 0.22 and 0.13, respectively, and a sample size of 2,000

```
power logistic 1.5, px(0.22) n(2000) py(0.13)
```

Same as above, but for different sample sizes and display results in a graph

```
power logistic 1.5, px(0.22) n(1500(250)2500) py(0.13) graph
```

Effect size given $\Pr(X = 1)$ of 0.22, a sample size of 2,000, power of 0.8, and $\Pr(Y = 1|X = 0)$ of 0.12

```
power logistic, px(0.22) n(2000) power(0.8) pycondx0(0.12)
```

Same as above, but display the effect size as a coefficient instead of an odds ratio

```
power logistic, px(0.22) n(2000) power(0.8) pycondx0(0.12) effect(coef)
```

Menu

Statistics > Power, precision, and sample size

Syntax

Compute sample size

power logistic *oratio*$_X$, { px(*numlist*) | oddsx(*numlist*) } [power(*numlist*) *onebinopts*]

Compute power

power logistic *oratio*$_X$, { px(*numlist*) | oddsx(*numlist*) } n(*numlist*) [*onebinopts*]

Compute effect size

power logistic, { px(*numlist*) | oddsx(*numlist*) } n(*numlist*) power(*numlist*) [*onebinopts*]

oratio$_X$ is the odds ratio for binary covariate of interest X under the alternative hypothesis H_a. Argument *oratio*$_X$ may be specified either as one number or as a list of values in parentheses (see **[U] 11.1.8 numlist**).

onebinopts	Description
Main	
* alpha(*numlist*)	significance level; default is alpha(0.05)
* power(*numlist*)	power; default is power(0.8)
* beta(*numlist*)	probability of type II error; default is beta(0.2)
* n(*numlist*)	sample size; required to compute power or effect size
nfractional	allow fractional sample size
* coefx(*numlist*)	coefficient for X in logistic regression; specify instead of odds ratio $oratio_X$
effect(oratio | coefficient)	specify the type of effect to display; default is effect(oratio)
†* px(*numlist*)	success probability of X, $\Pr(X = 1)$
†* oddsx(*numlist*)	odds of $X = 1$
* py(*numlist*)	success probability of Y in the population, $\Pr(Y = 1)$
* pycondx1(*numlist*)	success probability of Y given $X = 1$, $\Pr(Y = 1 \mid X = 1)$
* pycondx0(*numlist*)	success probability of Y given $X = 0$, $\Pr(Y = 1 \mid X = 0)$
* intercept(*numlist*)	intercept for logistic regression
direction(upper | lower)	direction of the effect for effect-size determination; default is direction(upper), which means that the postulated odds ratio for X is greater than 1 (thus, the coefficient is positive)
parallel	treat number lists in starred options or in command arguments as parallel when multiple values per option or argument are specified (do not enumerate all possible combinations of values)
Table	
[no]table[(*tablespec*)]	suppress table or display results as a table; see [PSS-2] **power, table**
saving(*filename* [, replace])	save the table data to *filename*; use replace to overwrite existing *filename*
Graph	
graph[(*graphopts*)]	graph results; see [PSS-2] **power, graph**
Iteration	
init(#)	initial odds ratio for effect-size calculation
iterate(#)	maximum number of iterations; default is iterate(500)
tolerance(#)	parameter tolerance; default is tolerance(1e-12)
ftolerance(#)	function tolerance; default is ftolerance(1e-12)
[no]log	suppress or display iteration log
[no]dots	suppress or display iterations as dots
notitle	suppress the title

†Either px() or oddsx() is required.

*Specifying a list of values in at least two starred options, or in argument $oratio_X$ and at least one starred option, results in computations for all possible combinations of the values; see [U] **11.1.8 numlist**. Also see parallel.

collect is allowed; see [U] **11.1.10 Prefix commands**.

notitle does not appear in the dialog box.

where *tablespec* is

column[:*label*] [*column*[:*label*] [. . .]] [, *tableopts*]

column is one of the columns defined below, and *label* is a column label (may contain quotes and compound quotes).

column	Description	Symbol
`alpha`	significance level	α
`power`	power	$1 - \beta$
`beta`	type-II-error probability	β
`N`	number of subjects	N
`delta`	effect size	δ
`oratiox`	odds ratio for X	OR_X
`coefx`	coefficient for X	β_X
`px`	success probability of X, $\Pr(X = 1)$	p_X
`oddsx`	odds of $X = 1$	$p_X/(1 - p_X)$
`py`	success probability of Y, $\Pr(Y = 1)$	p_Y
`pycondx1`	success probability of Y given X is 1, $\Pr(Y = 1\|X = 1)$	$p_{Y\|X=1}$
`pycondx0`	success probability of Y given X is 0, $\Pr(Y = 1\|X = 0)$	$p_{Y\|X=0}$
`intercept`	intercept	ζ_0
`target`	target parameter; odds ratio or coefficient for X	
`_all`	display all supported columns	

Options

Main

`alpha()`, `power()`, `beta()`, `n()`, `nfractional`; see [PSS-2] **power**. The `nfractional` option is allowed only for sample-size determination.

`coefx(`*numlist*`)` specifies the coefficient for binary covariate X in the logistic regression, β_X, where $\beta_X \neq 0$. The `coefx()` option may be specified instead of argument *oratio*$_X$. This option is not allowed with effect-size determination.

`effect(oratio | coefficient)` specifies how to report the effect size in the output. If `coefx()` is specified, then by default the effect size is β_X, the coefficient for X. Otherwise, the default effect size is the odds ratio for X: $\Pr(Y = 1|X = 1)\Pr(Y = 0|X = 0)/\{\Pr(Y = 0|X = 1)\Pr(Y = 1|X = 0)\}$. The `effect()` option is used to override the default.

`px(`*numlist*`)` specifies the success probability of binary covariate X in the target population, $\Pr(X = 1)$, where $0 < \Pr(X = 1) < 1$. Either the `px()` or `oddsx()` option must be specified, but not both.

`oddsx(`*numlist*`)` specifies the odds of binary covariate X in the target population, $\Pr(X = 1)/\Pr(X = 0)$, where $\Pr(X = 1)/\Pr(X = 0) > 0$. Either the `px()` or `oddsx()` option must be specified, but not both.

`py(`*numlist*`)` specifies the marginal success probability of Y in the target population, $\Pr(Y = 1)$, where $0 < \Pr(Y = 1) < 1$. This option is not allowed with effect-size determination.

`pycondx1(`*numlist*`)` specifies the conditional success probability of Y given X equals 1, $\Pr(Y = 1|X = 1)$, where $0 < \Pr(Y = 1|X = 1) < 1$. This option is not allowed with effect-size determination.

`pycondx0(`*numlist*`)` specifies the conditional success probability of Y given X equals 0, $\Pr(Y = 1|X = 0)$, where $0 < \Pr(Y = 1|X = 0) < 1$. Only one of the `intercept()` or `pycondx0()` option may be specified.

`intercept(`*numlist*`)` specifies the intercept for the logistic regression, ζ_0. Only one of the `intercept()` or `pycondx0()` option may be specified.

`direction()`, `parallel`; see [PSS-2] **power**.

Table

`table`, `table()`, `notable`; see [PSS-2] **power, table**.

`saving()`; see [PSS-2] **power**.

Graph

`graph`, `graph()`; see [PSS-2] **power, graph**. Also see the *column* table for a list of symbols used by the graphs.

Iteration

`init(#)` specifies the initial value of the odds ratio for binary covariate X during effect-size determination.

`iterate()`, `tolerance()`, `ftolerance()`, `log`, `nolog`, `dots`, `nodots`; see [PSS-2] **power**.

The following option is available with `power logistic` but is not shown in the dialog box:

`notitle`; see [PSS-2] **power**.

Remarks and examples

Remarks are presented under the following headings:

This entry describes the `power logistic` command and the methodology for power and sample-size analysis for logistic regression with one binary covariate. See [PSS-2] **power logistic** for power and sample-size analysis for logistic regression in other settings; [PSS-2] **Intro (power)** for a general introduction to power and sample-size analysis; and [PSS-2] **power** for a general introduction to using the `power` command for hypothesis tests.

Introduction

Logistic regression is a commonly used statistical method for analyzing binary outcome variables. Researchers often need to determine the appropriate sample size to ensure sufficient power for detecting an association between a binary covariate of interest and a binary outcome variable. For example, consider a study that examines whether migratory birds return to the same nesting site from one year to the next. Binary outcome Y is an indicator of whether the nesting site was reused, where the observed $y_i = 1$ if bird i returns to the nesting site it used last year and $y_i = 0$ if bird i does not. We will use logistic regression to test whether birds of one sex are more likely to return to the same nesting site than birds of the other sex. We define binary covariate of interest X as an indicator for female birds, where the observed $x_i = 1$ if bird i is female and $x_i = 0$ if bird i is male.

The logistic regression can be written as

$$\Pr(y_i = 1|x_i) = H(\beta_X x_i + \zeta_0) \qquad i = 1, 2, \ldots, n$$

where β_X is the coefficient quantifying the effect of covariate X, a bird's sex, on nesting-site reuse; ζ_0 is the logistic intercept; and n is the sample size. Function $H(\eta) = \{1 + \exp(-\eta)\}^{-1}$ is the logistic distribution function. The effect of X can also be expressed in terms of an odds ratio,

$$\text{OR}_X = \exp(\beta_X) = \Pr(Y = 1|X = 1)\Pr(Y = 0|X = 0)/\{\Pr(Y = 0|X = 1)\Pr(Y = 1|X = 0)\}$$

The null hypothesis is $H_0\colon \beta_X = 0$, which can also be expressed as $H_0\colon \text{OR}_X = 1$, and the alternative hypothesis is $H_a\colon \beta_X \neq 0$ or, equivalently, $H_a\colon \text{OR}_X \neq 1$.

Logistic regression is commonly used to analyze binary outcomes in observational studies, such as the study of nesting-site reuse. But it can also be used to analyze data from a randomized controlled trial, where trial participants are randomly assigned to a treatment. For instance, a public health study might investigate whether attending a support group helps smokers quit smoking. In this example, participation in the support group is the binary covariate of interest: Some participants will be randomly assigned to attend support group meetings for, say, three months, whereas other participants will be randomized to the control group, which does not attend support group meetings. At the end of three months, smoking status is recorded; this is the binary outcome.

Here the observed $x_i = 1$ if participant i attends the support group, and $x_i = 0$ if participant i does not attend the support group. The observed $y_i = 1$ if participant i successfully quits smoking by the end of the three-month study, and $y_i = 0$ if participant i continues to smoke. The null and alternative hypotheses are $H_0\colon \beta_X = 0$ and $H_a\colon \beta_X \neq 0$, respectively.

The `power logistic` command provides power and sample-size analysis for the test of $\beta_X = 0$ in logistic regression. The formula for power, sample-size, and effect-size calculations is based on the likelihood-ratio test of $\beta_X = 0$. However, Bush (2015) demonstrates that sample-size requirements for Wald and score tests are generally equivalent to the sample-size requirement for the likelihood-ratio test. Thus, these calculations may be used to plan studies that will be analyzed using the `logit` command, which conducts a Wald test of $\beta_X = 0$, or the `logistic` command, which conducts an equivalent Wald test of $\text{OR}_X = 1$. If you prefer a likelihood-ratio test, you can use the `lrtest` command.

Using power logistic with one binary covariate

`power logistic` computes sample size, power, or effect size for a test of one coefficient in logistic regression. This entry describes how to use `power logistic` when the logistic regression has a single binary covariate. All computations are performed for a two-sided hypothesis test where, by default, the significance level is set to 0.05. You may change the significance level by specifying the `alpha()` option.

You must specify either the prevalence or the odds of a binary covariate X in the target population in the respective `px()` or `oddsx()` option. The `px()` option specifies $\Pr(X = 1)$, whereas the `oddsx()` option specifies $\Pr(X = 1)/\Pr(X = 0)$. When you collect observational data, $\Pr(X = 1)$ indicates the prevalence of X in the target population. In the observational study about nesting-site reuse, $\Pr(X = 1)$ is the proportion of female birds in the population. But when you collect data from a randomized controlled trial, the "target population" is restricted to study participants, and the meaning of $\Pr(X = 1)$ changes accordingly. In the smoking-cessation trial, $\Pr(X = 1)$ is the proportion of participants who are randomly assigned to the support group. The ratio of participants in the experimental arm to the control arm is known as the allocation ratio.

Power and sample-size calculations require that you specify information about two parameters for logistic regression: the X coefficient (β_X) and the intercept (ζ_0). The information about β_X can be specified directly in the `coefx()` option or indirectly as an argument *oratio*$_X$, in the `py()` option, or in the `pycondx1()` option. The information about ζ_0 can be specified directly in the `intercept()` option or indirectly in the `pycondx0()`, `pycondx1()`, or `py()` option. See figure 1 for a graphical depiction of various ways to provide information about β_X and ζ_0.

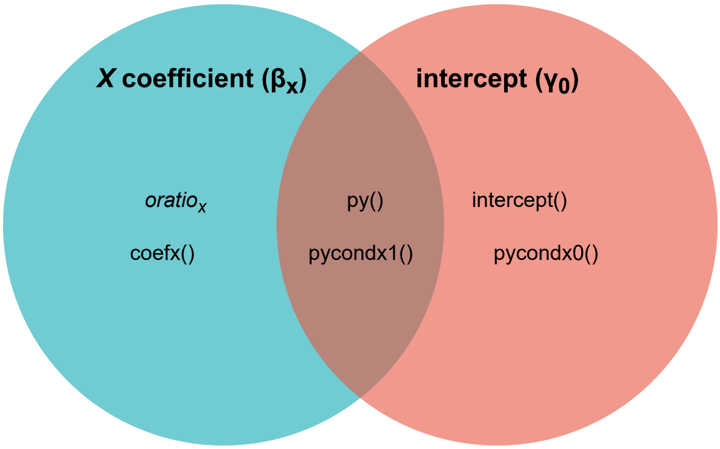

Figure 1. Specifying information about β_X and ζ_0

Valid ways of specifying the X coefficient and intercept include, for example, `coefx()` and `intercept()`, `py()` and `pycondx0()`, or `py()` and `pycondx1()`. However, the combination of `intercept()` and `pycondx0()` is invalid because no information is provided about the X coefficient. When you compute sample size, the power of the test may be specified using the `power()` option, which has a default of 0.8. When you compute power, the sample size must be specified using the `n()` option.

When `power logistic` is used to calculate effect size β_X, the command specification cannot include *oratio*$_X$ or options that provide information about the X coefficient, such as the `py()` and `pycondx1()` options. In this case, the information about the intercept must be specified using `intercept()` or `pycondx0()`. To calculate effect size, you must specify sample size using the `n()` option, power using the `power()` option, and, optionally, the direction of the effect using the `direction()` option. The default is `direction(upper)`, which means that coefficient β_X is assumed to be positive. This is equivalent to assuming that the odds ratio OR_X is greater than 1. You can change the direction to `lower`, which means that $\beta_X < 0$ or, equivalently, $\text{OR}_X < 1$.

The `effect()` option can be used to specify the type of the effect size to be reported in the output. Valid choices are `effect(oratio)` and `effect(coefficient)`. By default, the effect size is output as the odds ratio for X unless `coefx()` is specified, in which case it defaults to the coefficient for X. The `effect()` option is used to override the default.

By default, the computed sample size is rounded up. You can specify the `nfractional` option to see the corresponding fractional sample size; see *Fractional sample sizes* in [PSS-4] **Unbalanced designs** for an example. The `nfractional` option is allowed only for sample-size determination.

Some of the computations of `power logistic` require iteration, specifically, the computations used in effect-size determination. The default initial value of the estimated effect size is obtained using a bisection algorithm. This may be changed by specifying the `init()` option. See [PSS-2] **power** for descriptions of other options that control the iteration procedure.

In the following sections, we describe the use of `power logistic` with one binary covariate accompanied by examples for computing sample size, power, and effect size.

Computing sample size

To compute sample size, you must specify the prevalence or odds of X in the target population in the respective `px()` or `oddsx()` option; the information necessary to determine parameters β_X and ζ_0 (as described in *Using power logistic with one binary covariate* above); and, optionally, the power of the test using the `power()` option or the type-II-error probability using the `beta()` option. A default power of 0.8 is assumed if `power()` is not specified.

▷ Example 1: Sample size when OR_X and $\Pr(Y|X=0)$ are known

Consider an example from the seminal work on sample-size calculation for logistic regression by Whittemore (1981, 31) that describes the design of a study testing “the null hypothesis that risk of coronary heart disease (CHD) among white males aged 39–59 is unaffected by serum cholesterol levels”. Here the X variable is an indicator for elevated serum cholesterol, where the observed $x_i = 1$ if participant i has elevated serum cholesterol and $x_i = 0$ otherwise. Based on data from Hulley et al. (1980), we assume a probability of 0.07 that an individual in the target population will develop CHD during an 18-month study period if he does not have elevated serum cholesterol; this will be entered using the `pycondx0()` option. We assume that 13% of our target population has elevated serum cholesterol, corresponding to `px(0.13)`. Like Whittemore, we will calculate the sample size required to detect an odds ratio of 1.65 at the $\alpha = 0.05$ level, but we will use the default power of 80%.

```
. power logistic 1.65, px(0.13) pycondx0(0.07)

Estimated sample size for logistic regression odds-ratio test
Likelihood-ratio test
H0: OR_X = 1  versus  Ha: OR_X != 1

Study parameters:

        alpha =    0.0500
        power =    0.8000
        delta =    1.6500  (odds ratio)
      oratiox =    1.6500
           px =    0.1300
     pycondx0 =    0.0700

Estimated sample size:

            N =     3,256
```

The output of `power logistic` begins by displaying information about the test to be conducted and its null and alternative hypotheses. The study parameters we specified are listed next, followed by the estimated sample size. We find that a sample of 3,256 subjects is required to detect an odds ratio of 1.65 with 80% power using a 5% level test.

◁

▷ Example 2: Sample size when β_X and $\Pr(Y|X = 0)$ are known

Instead of the odds ratio for X, as in example 1, we specify the effect as $\beta_X = \log(1.65) = 0.50077$. We leave the rest of the command specification unchanged.

```
. power logistic, px(0.13) coefx(0.50077) pycondx0(0.07)
Estimated sample size for logistic regression coefficient test
Likelihood-ratio test
H0: beta_X = 0  versus  Ha: beta_X != 0
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    0.5008  (coefficient)
        coefx =    0.5008
           px =    0.1300
     pycondx0 =    0.0700
Estimated sample size:
            N =     3,256
```

The output of this command is identical to the output displayed in example 1, with the exception that the coefficient for X, `coefx`, is displayed instead of the odds ratio. If we wanted to see the odds ratio in the output despite inputting the coefficient for X, we could add the `effect(oratio)` option.

◁

▷ Example 3: Sample size when ζ_0 and $\Pr(Y)$ are known

Now we assume that we know $\Pr(Y = 1)$, the probability of observing CHD in a randomly selected individual from the study population. Recall from example 1 that the probability of observing CHD in an individual without elevated serum cholesterol was 7%; once we include individuals with elevated cholesterol, the overall probability of CHD rises to 7.5261%. This is calculated as $\Pr(Y = 1) = \text{invlogit}(\text{logit}(0.07) + 0.50077) \times 0.13 + 0.07 \times 0.87 = 0.075261$. For demonstration purposes, instead of specifying `pycondx0()`, we specify the intercept directly as $\zeta_0 = \texttt{logit}(0.07) = -2.5867$.

```
. power logistic, px(0.13) py(0.075261) intercept(-2.5867)
Estimated sample size for logistic regression odds-ratio test
Likelihood-ratio test
H0: OR_X = 1  versus  Ha: OR_X != 1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.6500  (odds ratio)
      oratiox =    1.6500
           px =    0.1300
           py =    0.0753
    intercept =   -2.5867
Estimated sample size:
            N =     3,256
```

The output of this command is identical to the output displayed in example 1, with the exception that study parameters `intercept` and `py` are displayed instead of `pycondx0`. If we wanted to see `coefx` in the output instead of `oratiox`, we could add the `effect(coefficient)` option.

◁

Computing power

To compute power, you must specify the sample size using the `n()` option, the prevalence or odds of X in the target population using the respective `px()` or `oddsx()` option, and the information necessary to determine parameters β_X and ζ_0 (as described in *Using power logistic with one binary covariate*).

▷ Example 4: Power of a logistic regression odds-ratio test

Continuing with example 1, we will suppose that we are designing a new study and anticipate a sample of 3,000 subjects. To compute the power corresponding to this sample size given the study parameters from example 1, we specify the sample size of 3,000 in the `n()` option:

```
. power logistic 1.65, px(0.13) n(3000) pycondx0(0.07)
Estimated power for logistic regression odds-ratio test
Likelihood-ratio test
H0: OR_X = 1  versus  Ha: OR_X != 1
Study parameters:
        alpha =    0.0500
            N =     3,000
        delta =    1.6500  (odds ratio)
      oratiox =    1.6500
           px =    0.1300
     pycondx0 =    0.0700
Estimated power:
        power =    0.7672
```

If the study recruits only 3,000 participants, the power to detect an odds ratio of 1.65 drops to 76.72%.

◁

▷ Example 5: Multiple values of study parameters

To investigate the effect of sample size on power, we can specify a list of sample sizes in the `n()` option:

```
. power logistic 1.65, px(0.13) n(3000 4000 5000) pycondx0(0.07)
Estimated power for logistic regression odds-ratio test
Likelihood-ratio test
H0: OR_X = 1  versus  Ha: OR_X != 1
```

alpha	power	N	delta	oratiox	px	pycondx0
.05	.7672	3,000	1.65	1.65	.13	.07
.05	.874	4,000	1.65	1.65	.13	.07
.05	.9348	5,000	1.65	1.65	.13	.07

As expected, when the sample size increases, the power increases toward 1.

For multiple values of parameters, the results are automatically displayed in a table, as we see above. For more examples of tables, see [PSS-2] **power, table**. If you wish to produce a power plot, see [PSS-2] **power, graph**.

◁

Computing effect size

By default, effect size δ for a logistic regression odds-ratio test is defined as the odds ratio for X: $\delta = \text{OR}_X = \exp(\beta_X)$. Sometimes, we want to know the smallest effect that can be detected with a level α test at a prespecified power and sample size.

To compute the minimum detectable effect size, you must specify sample size and power, as well as the intercept parameter ζ_0. This may be done via the `intercept()` or `pycondx0()` option; for details about specifying logistic regression parameters, see *Using power logistic with one binary covariate*. In addition, you must pick the level of the test and the direction of the effect. The level of the test is specified using the `alpha()` option, with a default of `alpha(0.05)`. The direction of the effect is specified using the `direction()` option; the default is `direction(upper)`, which means that $\text{OR}_X > 1$ or, equivalently, $\beta_X > 0$. Specifying `direction(lower)` means that $\text{OR}_X < 1$ and $\beta_X < 0$. The estimated minimum detectable effect size is reported as an odds ratio by default. To display it as a coefficient, specify `effect(coefficient)`.

▷ Example 6: Minimum detectable odds ratio

We continue with example 4, where we learned that a size 0.05 test with 3,000 subjects would have 76.72% power to detect an odds ratio of 1.65. How much larger would the odds ratio need to be to detect it with 80% power? We use `power logistic` to find out.

```
. power logistic, px(0.13) n(3000) power(0.8) pycondx0(0.07)
Performing iteration ...
Estimated odds ratio for logistic regression odds-ratio test
Likelihood-ratio test
H0: OR_X = 1  versus  Ha: OR_X != 1
Study parameters:
        alpha =    0.0500
        power =    0.8000
            N =     3,000
           px =    0.1300
     pycondx0 =    0.0700
Estimated effect size and odds ratio:
        delta =    1.6806  (odds ratio)
      oratiox =    1.6806
```

We see that a slightly larger odds ratio of 1.68 can be detected with 80% power.

In the above, we assumed the effect to be in the upper direction. There exists an effect size in the lower direction that can also be detected with 80% power. We specify `direction(lower)` to find it, and we add the `effect(coefficient)` option to display it as a coefficient instead of an odds ratio.

```
. power logistic, px(0.13) n(3000) power(0.8) pycondx0(0.07)
> direction(lower) effect(coefficient)

Performing iteration ...

Estimated coefficient for logistic regression coefficient test
Likelihood-ratio test
H0: beta_X = 0  versus  Ha: beta_X != 0

Study parameters:

        alpha =    0.0500
        power =    0.8000
            N =     3,000
           px =    0.1300
     pycondx0 =    0.0700

Estimated effect size and coefficient:

        delta =   -0.7213  (coefficient)
        coefx =   -0.7213
```

By specifying `direction(lower)`, we anticipate coefficient $\beta_X < 0$, which is what we see. Had we omitted the `effect(coefficient)` option, the odds ratio $\mathrm{OR}_X = \exp(\beta_X)$ would have been displayed as $\exp(-0.7213) = 0.486$.

◁

Performing hypothesis tests with logistic regression

In this section, we briefly demonstrate the use of the `logistic` command for testing logistic regression coefficients; see [R] **logistic** for details. Alternatively, we could use the `logit` command to perform logistic regression because `logit` performs the same calculations as `logistic` but reports coefficients instead of odds ratios; see [R] **logit** for details and example 7 of [PSS-2] **power logistic twobin** for a demonstration of how `logit` can be used to analyze the results of a pilot study.

▷ Example 7: Analyzing a pilot study

`nlsw88.dta` contains employment data from the 1988 extract of the National Longitudinal Study of Young Women. We will treat this dataset as if it came from a pilot study investigating the relationship between college graduation and marital status and use it to plan a follow-up study. Our population of interest is American young women who are union members, so we will drop nonmembers from the dataset and then perform logistic regression of `collgrad` on `married`.

```
. use https://www.stata-press.com/data/r19/nlsw88
(NLSW, 1988 extract)

. keep if union == 1
(1,785 observations deleted)

. logistic collgrad married

Logistic regression                                     Number of obs =    461
                                                        LR chi2(1)    =   2.77
                                                        Prob > chi2   = 0.0959
Log likelihood = -287.96102                             Pseudo R2     = 0.0048

------------------------------------------------------------------------------
    collgrad | Odds ratio   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
     married |   1.410769   .2936711     1.65   0.098     .9381378    2.121511
       _cons |   .3816794   .0634479    -5.79   0.000       .27555     .528685
------------------------------------------------------------------------------
Note: _cons estimates baseline odds.
```

The odds ratio OR_X of 1.41 suggests that married women are more likely to be college graduates than unmarried women, but the evidence is not strong enough to reject $H_0 : \text{OR}_X = 1$ at the 0.05 level. We use the parameter estimates from the pilot study to calculate the sample size for a follow-up study that has 80% power to detect an odds ratio of 1.41 with a 0.05-level test.

We will use `power logistic` with an effect size OR_X of 1.41, but what about the other parameters from the logistic regression? We also need to specify information about the intercept and $\Pr(X = 1)$.

The `logistic` command displays parameter estimates as odds ratios, so we take the natural logarithm of baseline odds parameter `_cons` to get the intercept from the logistic regression: $\log(0.38) \approx -0.96$. (Or we could have used `logistic, coef` to display coefficients instead of odds ratios.) To find $\Pr(X = 1)$, we use the `tabulate` command to calculate the prevalence of marriage in the target population; see **[R] tabulate oneway** for details.

```
. tabulate married

    Married |      Freq.     Percent        Cum.
------------+-----------------------------------
     Single |        181       39.26       39.26
    Married |        280       60.74      100.00
------------+-----------------------------------
      Total |        461      100.00
```

Approximately 61% of sampled women were married, so we will specify `px(0.61)`. We omit the `power()` option because we are designing our follow-up study to have 80% power, which is the default.

```
. power logistic 1.41, px(0.61) intercept(-0.96)

Estimated sample size for logistic regression odds-ratio test
Likelihood-ratio test
H0: OR_X = 1  versus  Ha: OR_X != 1

Study parameters:

        alpha =    0.0500
        power =    0.8000
        delta =    1.4100  (odds ratio)
      oratiox =    1.4100
           px =    0.6100
    intercept =   -0.9600

Estimated sample size:

            N =     1,310
```

We find that the sample size required to detect an odds ratio of 1.41 with 80% power using a 5% level test is 1,310. One detail that bears mentioning is that this sample-size calculation is for a likelihood-ratio test, but the `logistic` and `logit` commands report Wald tests of coefficients. Fortunately, an extensive simulation study by Bush (2015) demonstrates that sample-size requirements for Wald and likelihood-ratio tests of logistic regression coefficients are nearly identical. If you prefer a likelihood-ratio test, you can use the `lrtest` command; see [R] **lrtest** for details.

◁

Stored results

`power logistic` with one binary covariate stores the following in `r()`:

Scalars

`r(alpha)`	significance level
`r(power)`	power
`r(beta)`	probability of a type II error
`r(delta)`	effect size
`r(N)`	sample size
`r(nfractional)`	`1` if `nfractional` is specified, `0` otherwise
`r(px)`	success probability of X
`r(oddsx)`	odds that $X = 1$
`r(oratiox)`	odds ratio for X
`r(coefx)`	coefficient for X
`r(intercept)`	intercept from logistic regression
`r(py)`	success probability of Y in the target population
`r(pycondx1)`	success probability of Y given $X = 1$
`r(pycondx0)`	success probability of Y given $X = 0$
`r(separator)`	number of lines between separator lines in the table
`r(divider)`	`1` if `divider` is requested in the table, `0` otherwise
`r(init)`	initial value for odds ratio (if specified)
`r(maxiter)`	maximum number of iterations (for effect-size calculation)
`r(iter)`	number of iterations performed (for effect-size calculation)
`r(tolerance)`	requested parameter tolerance (for effect-size calculation)
`r(deltax)`	final parameter tolerance achieved (for effect-size calculation)
`r(ftolerance)`	requested distance of the objective function from zero (for effect-size calculation)
`r(function)`	final distance of the objective function from zero (for effect-size calculation)
`r(converged)`	`1` if iteration algorithm converged, `0` otherwise (for effect-size calculation)

Macros

`r(type)`	`test`
`r(method)`	`logistic`
`r(direction)`	`upper` or `lower` (for effect-size calculation)
`r(columns)`	displayed table columns
`r(labels)`	table column labels
`r(widths)`	table column widths
`r(formats)`	table column formats

Matrices

`r(pss_table)`	table of results

Methods and formulas

Methods and formulas are presented under the following headings:

Coefficient tests in logistic regression

Shieh (2000a) used simulation to compare the performance of two sample-size formulas for coefficient tests in logistic regression: the method of Whittemore (1981) and that of Self, Mauritsen, and Ohara (1992). Shieh generalized the superior of those two methods, that of Self, Mauritsen, and Ohara, in Shieh (2000b), which provides the formulas implemented in `power logistic` for power, sample-size, and effect-size calculations for likelihood-ratio tests in logistic regression.

In practice, it is more common to use the Wald test of logistic regression coefficients than the likelihood-ratio test, so there has been some concern about whether these calculations are appropriate for use with a Wald test (Demidenko 2007). Demidenko notes that the Wald and likelihood-ratio tests have asymptotically equivalent type I errors and that they are "*locally* equivalent, so that the power functions are close when the alternative approaches the null" (Demidenko 2007, 3385). Nevertheless, Demidenko raises the point that the two tests are not globally equivalent, so there is no theoretical guarantee that the power functions will be similar under the alternative hypothesis. Thankfully, an extensive simulation study by Bush (2015) found little difference between the power curves of the Wald, likelihood-ratio, and score tests over a range of scenarios. Additionally, Bush compared the performance of seven sample-size formulas for logistic regression and determined that the method of Shieh (2000b) was consistently accurate, regardless of the test that was used.

Logistic regression

The logistic regression with one binary covariate can be written as

$$\Pr(y_i = 1|x_i) = H(\beta_X x_i + \zeta_0) \qquad i = 1, 2, \ldots, n$$

where β_X is the coefficient for covariate X, ζ_0 is the logistic intercept, and n is the sample size. Function $H(\eta) = \{1 + \exp(-\eta)\}^{-1}$ is the logistic distribution function.

The effect of binary covariate X can also be expressed in terms of an odds ratio,

$$\text{OR}_X = \exp(\beta_X) = \Pr(Y = 1|X = 1)\Pr(Y = 0|X = 0)/\{\Pr(Y = 0|X = 1)\Pr(Y = 1|X = 0)\}$$

The null hypothesis is $H_0\colon \beta_X = 0$, which can also be expressed as $H_0\colon \text{OR}_X = 1$, and the alternative hypothesis is $H_a\colon \beta_X \neq 0$ or, equivalently, $H_a\colon \text{OR}_X \neq 1$.

Parameters β_X and ζ_0 may be directly specified by using the `coefx()` and `intercept()` options, respectively. But when the `py()` option is used to specify $\Pr(Y = 1)$, we need the equation

$$\Pr(Y = 1) = \Pr(X = 1) \times H(\beta_X + \zeta_0) + \Pr(X = 0) \times H(\zeta_0)$$

to solve for the unknown parameter, either β_X or ζ_0.

Power, sample-size, and effect-size calculations

Shieh (2000b) builds on the work of Self, Mauritsen, and Ohara (1992) and Self and Mauritsen (1988) to estimate the distribution of the likelihood-ratio statistic: $2\{l(\widehat{\beta}_X, \widehat{\mathbf{c}}) - l(0, \widehat{\mathbf{c}}_0)\}$. Here $l(\cdot)$ is the log-likelihood function for the logistic regression, and $\mathbf{c}$ is a vector of nuisance parameters; for logistic regression with one binary covariate, the only nuisance term is the logistic intercept, so $\mathbf{c} = \zeta_0$. $\widehat{\beta}_X$ and $\widehat{\mathbf{c}}$ are the maximum likelihood estimates of β_X and $\mathbf{c}$ under the alternative hypothesis, and $\widehat{\mathbf{c}}_0$ is the maximum likelihood estimate of $\mathbf{c}$ under the null hypothesis. If the null hypothesis is not true, $\widehat{\mathbf{c}}_0$ is not a consistent estimate of $\mathbf{c}$ but instead converges to $\mathbf{c}_0^* = \zeta_0^* = \zeta_0 + \beta_X E(X)$, as described in Self and Mauritsen (1988, eq. 2.2). For a full decomposition of the likelihood-ratio statistic, see Shieh (2000b, 1193).

To calculate sample size, we begin by specifying the desired size of the test (also known as the type I error, α) and the desired power (which equals $1-\beta$, where β is the desired type II error of the test). We equate the $(1-\alpha)100$th percentile of a central χ^2 distribution with 1 degree of freedom to the $\beta100$th percentile of a noncentral χ^2 distribution with noncentrality parameter $\lambda = n\Delta^*$. We define Δ^* as

$$\Delta^* = 2E[H(\eta)(\eta - \eta^*) - \log\{1 + \exp(\eta)\} + \log\{1 + \exp(\eta^*)\}]$$

where $\eta = \beta_X X + \mathbf{c}$ and $\eta^* = \mathbf{c}_0^*$. Sample size is calculated as $n = \lambda/\Delta^*$. Power is computed similarly by starting with a known value of n and solving for the power required to yield the desired value of λ.

There is no closed-form expression that can be used to calculate effect size δ, either coefficient β_X or odds ratio OR_X. Effect size is estimated iteratively, and the default starting value for δ is calculated using a bisection algorithm, which is refined using a nonlinear solver; see [M-5] **solvenl()** for details. To skip the bisection algorithm, you can use the `init()` option to specify a starting odds ratio for the nonlinear solver. You can control the iteration process with the `iterate()`, `tolerance()`, `ftolerance()`, `log`, `nolog`, `dots`, and `nodots` options.

References

Bush, S. 2015. Sample size determination for logistic regression: A simulation study. *Communications in Statistics—Simulation and Computation* 44: 360–373. https://doi.org/10.1080/03610918.2013.777458.

Demidenko, E. 2007. Sample size determination for logistic regression revisited. *Statistics in Medicine* 26: 3385–3397. https://doi.org/10.1002/sim.2771.

Hulley, S. B., R. H. Rosenman, R. D. Bawol, and R. J. Brand. 1980. Epidemiology as a guide to clinical decisions—the association between triglyceride and coronary heart disease. *New England Journal of Medicine* 302: 1383–1389. https://doi.org/10.1056/nejm198006193022503.

Self, S. G., and R. H. Mauritsen. 1988. Power/sample size calculations for generalized linear models. *Biometrika* 44: 79–86. https://doi.org/10.2307/2531897.

Self, S. G., R. H. Mauritsen, and J. Ohara. 1992. Power calculations for likelihood ratio tests in generalized linear models. *Biometrika* 48: 31–39. https://doi.org/10.2307/2532736.

Shieh, G. 2000a. A comparison of two approaches for power and sample size calculations in logistic regression models. *Communications in Statistics—Simulation and Computation* 29: 763–791. https://doi.org/10.1080/03610910008813639.

———. 2000b. On power and sample size calculations for likelihood ratio tests in generalized linear models. *Biometrics* 56: 1192–1196. https://doi.org/10.1111/j.0006-341x.2000.01192.x.

Whittemore, A. S. 1981. Sample size for logistic regression with small response probability. *Journal of the American Statistical Association* 76: 27–32. https://doi.org/10.2307/2287036.

Also see

[PSS-2] **power logistic** — Power analysis for logistic regression+

[PSS-2] **power logistic general** — Power analysis for logistic regression: General case+

[PSS-2] **power logistic twobin** — Power analysis for logistic regression with two binary covariates+

[PSS-2] **power** — Power and sample-size analysis for hypothesis tests

[PSS-2] **power, graph** — Graph results from the power command

[PSS-2] **power, table** — Produce table of results from the power command

[PSS-5] **Glossary**

[R] **logistic** — Logistic regression, reporting odds ratios

[R] **logit** — Logistic regression, reporting coefficients

[R] **lrtest** — Likelihood-ratio test after estimation