

power onemean, cluster — Power analysis for a one-sample mean test, CRD

Description	Quick start	Menu	Syntax
Options	Remarks and examples	Stored results	Methods and formulas
References	Also see		

Description

`power onemean, cluster` computes the number of clusters, cluster size, power, or target mean for a one-sample mean test in a cluster randomized design (CRD). It computes the number of clusters given cluster size, power, and the values of the null and alternative means. It also computes cluster size given the number of clusters, power, and the values of the null and alternative means. Alternatively, it computes power given the number of clusters, cluster size, and the values of the null and alternative means, or it computes the target mean given the number of clusters, cluster size, power, and the null mean. See [PSS-2] [power onemean](#) for a general discussion of power and sample-size analysis for a one-sample mean test. Also see [PSS-2] [power](#) for a general introduction to the `power` command using hypothesis tests.

Quick start

Compute number of clusters for two-sided test of $H_0: \mu = 10$ versus $H_a: \mu \neq 10$ with null mean $m_0 = 10$, alternative mean $m_a = 15$, standard deviation of 12, and cluster size of 5, using default intraclass correlation of 0.5, power of 0.8, and significance level $\alpha = 0.05$

```
power onemean 10 15, m(5) sd(12)
```

Same as above, but with an intraclass correlation of 0.2

```
power onemean 10 15, m(5) sd(12) rho(0.2)
```

Same as above, but the cluster size varies with a coefficient of variation of 0.6

```
power onemean 10 15, m(5) sd(12) rho(0.2) cvcluster(0.6)
```

Compute cluster size when 30 clusters are sampled

```
power onemean 10 15, k(30) sd(12)
```

Power for 30 clusters with cluster size of 5

```
power onemean 10 15, k(30) m(5) sd(12)
```

Same as above, but for 30, 35, 40, 45, and 50 clusters and display results in a graph of power versus number of clusters

```
power onemean 10 15, k(30(5)50) m(5) sd(12) graph
```

Effect size and target mean for $m_0 = 10$ with standard deviation of 4, for 8 clusters with cluster size of 5, power of 0.9, and $\alpha = 0.01$

```
power onemean 10, k(8) m(5) power(0.9) sd(4) alpha(0.01)
```

Menu

Statistics > Power, precision, and sample size

Syntax

Compute number of clusters

```
power onemean  $m_0$   $m_a$  , { m(numlist) | n(numlist) cluster } [options]
```

Compute cluster size

```
power onemean  $m_0$   $m_a$  , k(numlist) [options]
```

Compute power

```
power onemean  $m_0$   $m_a$  , k(numlist) { m(numlist) | n(numlist) } [options]
```

Compute effect size and target mean

```
power onemean  $m_0$  , k(numlist) { m(numlist) | n(numlist) } power(numlist) [options]
```

where m_0 is the null (hypothesized) mean or the value of the mean under the null hypothesis and m_a is the alternative (target) mean or the value of the mean under the alternative hypothesis. m_0 and m_a may each be specified either as one number or as a list of values in parentheses (see [U] 11.1.8 **numlist**).

<i>options</i>	Description
Main	
<code>cluster</code>	perform computations for a CRD; implied by <code>k()</code> or <code>m()</code>
* <code>alpha(numlist)</code>	significance level; default is <code>alpha(0.05)</code>
* <code>power(numlist)</code>	power; default is <code>power(0.8)</code>
* <code>beta(numlist)</code>	probability of type II error; default is <code>beta(0.2)</code>
* <code>k(numlist)</code>	number of clusters
* <code>m(numlist)</code>	cluster size
* <code>n(numlist)</code>	number of observations
<code>nfractional</code>	allow fractional number of clusters, cluster size, and sample size
* <code>diff(numlist)</code>	difference between the alternative mean and the null mean, $m_a - m_0$; specify instead of the alternative mean m_a
* <code>sd(numlist)</code>	standard deviation; default is <code>sd(1)</code>
* <code>rho(numlist)</code>	intraclass correlation; default is <code>rho(0.5)</code>
* <code>cvcluster(numlist)</code>	coefficient of variation for cluster sizes
<code>direction(upper lower)</code>	direction of the effect for effect-size determination; default is <code>direction(upper)</code> , which means that the postulated value of the parameter is larger than the hypothesized value
<code>onesided</code>	one-sided test; default is two sided
<code>parallel</code>	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	
<code>[no]table[(tablespec)]</code>	suppress table or display results as a table; see [PSS-2] power, table
<code>saving(filename [, replace])</code>	save the table data to <i>filename</i> ; use <code>replace</code> to overwrite existing <i>filename</i>
Graph	
<code>graph[(graphopts)]</code>	graph results; see [PSS-2] power, graph
Iteration	
<code>init(#)</code>	initial value for number of clusters, cluster size, or mean
<code>iterate(#)</code>	maximum number of iterations; default is <code>iterate(500)</code>
<code>tolerance(#)</code>	parameter tolerance; default is <code>tolerance(1e-12)</code>
<code>ftolerance(#)</code>	function tolerance; default is <code>ftolerance(1e-12)</code>
<code>[no]log</code>	suppress or display iteration log
<code>[no]dots</code>	suppress or display iterations as dots
<code>notitle</code>	suppress the title

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

`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).

<i>column</i>	Description	Symbol
alpha	significance level	α
power	power	$1 - \beta$
beta	type II error probability	β
K	number of clusters	K
M	cluster size	M
N	number of observations	N
delta	effect size	δ
m0	null mean	μ_0
ma	alternative mean	μ_a
diff	difference between the alternative and null means	$\mu_a - \mu_0$
sd	standard deviation	σ
rho	intraclass correlation	ρ
CV_cluster	coefficient of variation for cluster sizes	CV_{cl}
target	target parameter; synonym for ma	
_all	display all supported columns	

Column **beta** is shown in the default table in place of column **power** if specified.

Columns **diff** and **CV_cluster** are shown in the default table if specified.

Options

Main

cluster specifies that computations should be performed for a CRD. This option is implied when either the **k()** or **m()** option is specified. It is required if the **n()** option is used to compute the number of clusters.

alpha(), **power()**, **beta()**; see [PSS-2] **power**.

k(numlist) specifies the number of clusters. This option is required to compute the cluster size, power, or effect size.

m(numlist) specifies the cluster size. This option or the **n()** option is required to compute the number of clusters, power, or effect size. **m()** may contain noninteger values. In this case or if the **cvcluster()** option is specified, **m()** represents the average cluster size.

n(numlist) specifies the number of observations. This option or the **m()** option is required to compute the number of clusters, power, or effect size.

nfractional; see [PSS-2] **power**. The **nfractional** option is allowed when computing the number of clusters and cluster size to display fractional (without rounding) values of the number of clusters, cluster size, and sample size.

diff(), **sd()**; see [PSS-2] **power onemean**.

rho(numlist) specifies the intraclass correlation. The default is **rho(0.5)**.

`cvcluster(numlist)` specifies the coefficient of variation for cluster sizes. This option is used with varying cluster sizes.

`direction()`, `onesided`, `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 for the number of clusters or cluster size for sample-size determination or the initial value for the mean for the effect-size determination. The default is to use a closed-form normal approximation to compute an initial value for the estimated parameter.

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

The following option is available with `power onemean`, `cluster` but is not shown in the dialog box: `notitle`; see [PSS-2] [power](#).

Remarks and examples

stata.com

Remarks are presented under the following headings:

- Using power onemean, cluster*
- Computing number of clusters*
- Computing cluster size*
- Computing power*
- Computing effect size and target mean*
- Performing hypothesis tests on mean in a CRD*

`power onemean`, `cluster` requests that computations for the `power onemean` command be done for a CRD. In a CRD, groups of subjects or clusters are randomized instead of individual subjects, so the sample size is determined by the number of clusters and the cluster size. The sample-size determination thus consists of the determination of the number of clusters given cluster size or the determination of cluster size given the number of clusters. For a general discussion of using `power onemean`, see [PSS-2] [power onemean](#). The discussion below is specific to the CRD.

Using power onemean, cluster

If you specify the `cluster` option, include `k()` to specify the number of clusters or include `m()` to specify the cluster size, the `power onemean` command will perform computations for a one-sample mean test in a CRD.

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 can specify the `onesided` option to request a one-sided test.

To compute the number of clusters, you must specify the means under the null and alternative hypotheses as command arguments m_0 and m_a , respectively, and specify the cluster size in the `m()` option. Instead of specifying the `m()` option, you may specify the sample size in the `n()` option and specify the `cluster` option, so that `power onemean` will perform its computation for a cluster randomized design instead of the default individual-level design. You may also specify the power of the test in the `power()` option.

To compute cluster size, you must specify the null mean m_0 , the alternative mean m_a , and the number of clusters in the `k()` option. You may also specify the power of the test in the `power()` option.

To compute power, you must specify the number of clusters in the `k()` option, the cluster size in the `m()` option or the sample size in the `n()` option, the null mean m_0 , and the alternative mean m_a .

Instead of the alternative mean m_a , you may specify the difference $m_a - m_0$ between the alternative mean and the null mean in the `diff()` option when computing the number of clusters, cluster size, or power.

The effect size δ is defined as the standardized difference between the alternative and null means. In a CRD, the effect size δ is also adjusted for the cluster design; see [Methods and formulas](#).

To compute effect size and the corresponding target mean, you must specify the number of clusters in the `k()` option, the cluster size in the `m()` option or the sample size in the `n()` option, the power in the `power()` option, and the null mean m_0 . You may also specify the direction of the effect in the `direction()` option. The direction is upper by default, `direction(upper)`; see [Using power onemean](#) in [PSS-2] **power onemean** for other details.

The computations for a CRD are based on a z test that relies on (asymptotic) normality of the data and assumes known standard deviation, which you may specify in the `sd()` option. Otherwise, the default value of one is used.

All computations assume an intraclass correlation of 0.5. You can change this by specifying the `rho()` option. Also, all clusters are assumed to be of the same size unless the coefficient of variation for cluster sizes is specified in the `cvcluster()` option.

By default, the computed number of clusters, cluster size, and sample size is rounded up. However, you can specify the `nfractional` option to see the corresponding fractional values; see [Fractional sample sizes](#) in [PSS-4] **Unbalanced designs** for an example. If the `cvcluster()` option is specified when computing cluster size, then cluster size represents the average cluster size and is thus not rounded. When sample size is specified in the `n()` option, fractional cluster size may be reported to accommodate the specified number of clusters and sample size.

Some of `power onemean, cluster`'s computations require iteration, such as to compute the number of clusters for a two-sided test; see [Methods and formulas](#) for details and [PSS-2] **power** for the descriptions of options that control the iteration procedure.

Computing number of clusters

To compute the number of clusters, you must specify the means under the null and alternative hypotheses as command arguments m_0 and m_a , respectively, and specify the cluster size in the `m()` option. Instead of specifying the `m()` option, you may specify the sample size in the `n()` option and specify the `cluster` option, so that `power onemean` will perform its computation for a cluster randomized design instead of the default individual-level design. You may also specify the power of the test in the `power()` option.

► Example 1: Number of clusters for a one-sample mean test in a CRD, specifying cluster size

Consider an example from [Tamhane and Dunlop \(2000, 2009\)](#) that discusses the effectiveness of coaching programs in improving the verbal part of SAT scores. Previous studies found that students retaking the SAT exams without any coaching program improve their scores by 15 points on average with a standard deviation of about 40 points. The population standard deviation is assumed to be 40.

Unlike [Tamhane and Dunlop \(2000, 2009\)](#), we assume that students are sampled from a set of classes and that the scores of students from the same class are correlated. We plan on sampling 10 students from each class and assume that the intraclass correlation is 0.3.

A new coaching program claims to improve average SAT scores by 40 points. The changes in scores are assumed to be approximately normally distributed. The parameter of interest in this example is the mean change in the test scores. To test the claim, investigators wish to conduct another study and compute the number of classes that is required to detect a difference of 25 points when the mean change in scores increases from 15 points to 40 points with 80% power using a 5%-level two-sided test:

```
. power onemean 15 40, m(10) sd(40) rho(0.3)
Performing iteration ...
Estimated number of clusters for a one-sample mean test
Cluster randomized design, z test
HO: m = m0 versus Ha: m != m0
Study parameters:
    alpha =    0.0500
    power =    0.8000
    delta =    0.3249
    m0 =     15.0000
    ma =     40.0000
    sd =     40.0000
Cluster design:
    M =         10
    rho =       0.3000
Estimated number of clusters and sample size:
    K =          8
    N =         80
```

We find that given 10 students per class, 8 classes and thus a total of 80 students are required to detect a 25-point increase in the mean change in SAT scores given the standard deviation of 40 points with 80% power using a 5%-level two-sided test. The effect size (δ) is calculated using the given information about the null and alternative means, standard deviation, and cluster design; see [Methods and formulas](#).

◀

► Example 2: Number of clusters for a one-sample mean test in a CRD, with varying cluster sizes

Instead of a constant number of students in each class as in [example 1](#), we assume that the number of students selected from each class will vary. Suppose that the average of the class sizes is 10 and that the coefficient of variation for class sizes is 1.2. To account for varying cluster sizes, we specify the coefficient of variation in the `cvcluster()` option.

```

. power onemean 15 40, m(10) sd(40) rho(0.3) cvcluster(1.2)
Performing iteration ...
Estimated number of clusters for a one-sample mean test
Cluster randomized design, z test
H0: m = m0 versus Ha: m != m0
Study parameters:
    alpha = 0.0500
    power = 0.8000
    delta = 0.2868
    m0 = 15.0000
    ma = 40.0000
    sd = 40.0000
Cluster design:
    Average M = 10.0000
    rho = 0.3000
    CV_cl = 1.2000
Estimated number of clusters and sample size:
    K = 10
    N = 100

```

The required number of classes is 10 and total number of subjects is 100. When we compare this with the 8 classes required in [example 1](#), we see that when the number of students per class varies, we need more classes and thus more students to achieve the same power.

◀

► Example 3: Number of clusters for a one-sample mean test in a CRD, specifying sample size

Suppose that for our study, we plan to recruit a total of 100 students, but we need to know the required number of classes and how many students to recruit in each class. In this case, we specify the `n(100)` option. Because neither the `k()` nor the `m()` option is specified, we also need to specify the `cluster` option for power computations to be performed for a CRD.

```

. power onemean 15 40, cluster n(100) sd(40) rho(0.3)
Performing iteration ...
Estimated number of clusters for a one-sample mean test
Cluster randomized design, z test
H0: m = m0 versus Ha: m != m0
Study parameters:
    alpha = 0.0500
    power = 0.8000
    delta = 0.2963
    m0 = 15.0000
    ma = 40.0000
    sd = 40.0000
Cluster design:
    N = 100
    rho = 0.3000
Estimated number of clusters and cluster size:
    K = 8
    Average M = 12.5000

```


To achieve the desired power, we need to recruit on average 12.5 students per class from 8 classes. `power onemean` did not round the cluster size of 12.5 to meet our requirement that there is a total of 100 students. In practice, you can decide to recruit 12 students from some classes and 13 from other classes to have roughly constant class sizes.

◀

Computing cluster size

To compute cluster size, you must specify the null mean m_0 , the alternative mean m_a , and the number of clusters in the `k()` option. You may also specify the power of the test in the `power()` option.

▶ Example 4: Cluster size for a one-sample mean test in a CRD

Continuing with [example 1](#), suppose that we are designing a new study but we would like to select 12 classes, and we need to know how many students to recruit from each class. Given other study parameters from [example 1](#), we compute the required number of students by specifying 12 clusters in the `k()` option.

```
. power onemean 15 40, k(12) sd(40) rho(0.3)
Performing iteration ...
Estimated cluster size for a one-sample mean test
Cluster randomized design, z test
H0: m = m0 versus Ha: m != m0
Study parameters:
      alpha = 0.0500
      power = 0.8000
      delta = 0.4941
      m0 = 15.0000
      ma = 40.0000
      sd = 40.0000

Cluster design:
      K = 12
      rho = 0.3000

Estimated cluster size and sample size:
      M = 3
      N = 36
```

With 12 classes, we need to recruit 3 students per class for a total of 36 students.

◀

Computing power

To compute power, you must specify the number of clusters in the `k()` option, the cluster size in the `m()` option or the sample size in the `n()` option, the null mean m_0 , and the alternative mean m_a .

▶ Example 5: Power for a one-sample mean test in a CRD

Continuing with [example 4](#), suppose that we can recruit 10 students from each of the 12 classes and we want to compute power for this design. Given other study parameters from [example 4](#), we compute the power by specifying 12 clusters in the `k()` option and the cluster size of 10 in the `m()` option:

```

. power onemean 15 40, k(12) m(10) sd(40) rho(0.3)
Estimated power for a one-sample mean test
Cluster randomized design, z test
H0: m = m0 versus Ha: m != m0
Study parameters:
      alpha =    0.0500
      delta =    0.3249
      m0 =    15.0000
      ma =    40.0000
      sd =    40.0000
Cluster design:
      K =      12
      M =      10
      N =     120
      rho =    0.3000
Estimated power:
      power =    0.9451

```

The computed power is about 95%.

◀

► Example 6: Multiple values of study parameters

To investigate the effect of the number of clusters on power, we can specify a list of numbers in the `k()` option:

```

. power onemean 15 40, k(4(2)12) m(10) sd(40) rho(0.3) table(power K)
Estimated power for a one-sample mean test
Cluster randomized design, z test
H0: m = m0 versus Ha: m != m0

```

power	K
.5379	4
.7112	6
.828	8
.9013	10
.9451	12

In this example, we also specified the `table(power K)` option to list the only two columns that vary. As expected, as the number of clusters increases, the power tends to get closer to 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 and target mean

The effect size δ is defined as the standardized difference between the alternative and null means. In a CRD, the effect size δ is also adjusted for the cluster design; see *Methods and formulas*.

To compute effect size and the corresponding target mean, you must specify the number of clusters in the `k()` option, the cluster size in the `m()` option or the sample size in the `n()` option, the power in the `power()` option, and the null mean m_0 . You may also specify the direction of the effect in the `direction()` option. The direction is upper by default, `direction(upper)`; see [Using power onemean](#) in [PSS-2] [power onemean](#) for other details.

► Example 7: Effect size for a one-sample mean test in a CRD

Continuing with [example 5](#), we may be interested in finding the minimum value of the mean that can be detected with a sample of 12 classes and 10 students per class and 80% power. To compute this, we specify the null value of 15 as the command argument and the required options `k(12)`, `m(10)`, and `power(0.8)` and continue to use `sd(40)` and `rho(0.3)`.

```
. power onemean 15, k(12) m(10) power(0.8) sd(40) rho(0.3)
Performing iteration ...
Estimated target mean for a one-sample mean test
Cluster randomized design, z test
H0: m = m0 versus Ha: m != m0; ma > m0
Study parameters:
    alpha =    0.0500
    power =    0.8000
    m0 =    15.0000
    sd =    40.0000

Cluster design:
    K =         12
    M =         10
    N =        120
    rho =    0.3000

Estimated effect size and target mean:
    delta =    0.2557
    ma =    34.6777
```

Given the null value of 15, the minimum detectable value of the mean is about 34.68, which corresponds to an effect size of about 0.26. The computed target mean is smaller than the alternative mean of 40 that was specified in previous examples. Because we use more classes here than the number of classes computed in [example 1](#), use larger cluster size than that computed in [example 4](#), and use lower power than that computed in [example 5](#), we can detect a smaller effect size and thus smaller target mean.

◀

Performing hypothesis tests on mean in a CRD

There are different ways to account for a CRD or for clustered data when performing hypothesis tests that compare the mean with a reference value. With large samples or when you know population standard deviation and intraclass correlation, the simplest way is to use a z test that accounts for clustered data.

In this section, we briefly demonstrate the `ztest` command for testing hypotheses about means with clustered data; see [R] [ztest](#) for details. We will use the same `sat.dta` we used in [Performing hypothesis tests on mean](#) in [PSS-2] [power onemean](#). Here we consider that SAT scores of students from the same class may be correlated.

▷ Example 8: Testing for mean with clustered data

As in [example 8](#) of [\[PSS-2\] power onemean](#), suppose that we want to test whether the mean verbal SAT score is equal to 600 while accounting for the correlated scores within classes.

The `ztest` command provides the `cluster()` option to account for clustered data. In addition to the cluster identifier in `cluster()`, we must specify the population standard deviation in `sd()` and the population intraclass correlation in `rho()`. Suppose that the population standard deviation is 132 and the population intraclass correlation is 0.7.

```
. use https://www.stata-press.com/data/r18/sat
(Fictional SAT data)
. ztest score == 600, cluster(class) sd(132) rho(0.7)
```

One-sample z test	Number of clusters =	15
Cluster variable: class	Cluster size =	5
	Intraclass corr. =	0.7000

Variable	Obs	Mean	Std. err.	Std. dev.	[95% conf. interval]
score	75	504.8	29.71222	132	446.5651 563.0349

```

mean = mean(score)                                z = -3.2041
HO: mean = 600
Ha: mean < 600                                Ha: mean != 600                                Ha: mean > 600
Pr(Z < z) = 0.0007                            Pr(|Z| > |z|) = 0.0014                            Pr(Z > z) = 0.9993
```

There is statistical evidence that the mean verbal SAT score is different from 600 (two-sided p -value = 0.0014) and that it is actually smaller than 600 (lower one-sided p -value = 0.0007).

If we were to design another similar study based on the estimates from this study, we could compute the required number of classes as follows:

```
. power onemean 600 505, sd(132) rho(0.7) m(5)
```

Performing iteration ...

Estimated number of clusters for a one-sample mean test
Cluster randomized design, z test
HO: $m = m_0$ versus Ha: $m \neq m_0$

Study parameters:

```

alpha = 0.0500
power = 0.8000
delta = -0.3692
m0 = 600.0000
ma = 505.0000
sd = 132.0000
```

Cluster design:

```

M = 5
rho = 0.7000
```

Estimated number of clusters and sample size:

```

K = 12
N = 60
```

Given 5 students per class, we need 12 classes and a total of 60 students to detect the difference between the null value of 600 and the alternative value of 505, assuming a standard deviation of 132 and an intraclass correlation of 0.7, with 80% power using a 5%-level two-sided test.

Compared with the required sample size of 18 students in [example 8](#) of [\[PSS-2\] power onemean](#), a sample size of 60 students is needed here to detect the same mean difference when observations are correlated.

Stored results

power onemean, cluster stores the following in `r()`:

Scalars

<code>r(alpha)</code>	significance level
<code>r(power)</code>	power
<code>r(beta)</code>	probability of a type II error
<code>r(delta)</code>	effect size
<code>r(K)</code>	number of clusters
<code>r(M)</code>	cluster size
<code>r(N)</code>	number of subjects
<code>r(nfractional)</code>	1 if <code>nfractional</code> is specified, 0 otherwise
<code>r(onesided)</code>	1 for a one-sided test, 0 otherwise
<code>r(m0)</code>	mean under the null hypothesis
<code>r(ma)</code>	mean under the alternative hypothesis
<code>r(diff)</code>	difference between the alternative and null means
<code>r(sd)</code>	standard deviation
<code>r(rho)</code>	intraclass correlation
<code>r(CV_cluster)</code>	coefficient of variation for cluster sizes
<code>r(separator)</code>	number of lines between separator lines in the table
<code>r(divider)</code>	1 if <code>divider</code> is requested in the table, 0 otherwise
<code>r(init)</code>	initial value for estimated parameter
<code>r(maxiter)</code>	maximum number of iterations
<code>r(iter)</code>	number of iterations performed
<code>r(tolerance)</code>	requested parameter tolerance
<code>r(deltax)</code>	final parameter tolerance achieved
<code>r(ftolerance)</code>	requested distance of the objective function from zero
<code>r(function)</code>	final distance of the objective function from zero
<code>r(converged)</code>	1 if iteration algorithm converged, 0 otherwise

Macros

<code>r(type)</code>	test
<code>r(method)</code>	onemean
<code>r(design)</code>	CRD
<code>r(direction)</code>	upper or lower
<code>r(columns)</code>	displayed table columns
<code>r(labels)</code>	table column labels
<code>r(widths)</code>	table column widths
<code>r(formats)</code>	table column formats

Matrices

<code>r(pss_table)</code>	table of results
---------------------------	------------------

Methods and formulas

For the computation in a CRD, we assume that the standard deviation is known; see *Known standard deviation* under *Methods and formulas* in [PSS-2] [power onemean](#) for the common notation for a one-sample mean test.

Methods and formulas are presented under the following headings:

Equal cluster sizes

Unequal cluster sizes

Equal cluster sizes

In a CRD, let K be the number of clusters, M be the number of observations in each cluster, and n be the total number of subjects, where $n = MK$. Let y_{ij} be the j th ($j = 1, 2, \dots, M$) observation from the i th cluster ($i = 1, 2, \dots, K$), which is assumed to follow a normal distribution with mean μ and variance σ^2 . Let ρ be the intraclass correlation.

Let

$$\bar{y} = \frac{1}{n} \sum_{i=1}^K \sum_{j=1}^M y_{ij}$$

be the observed mean. Let μ_0 be the mean under the null hypothesis, and let DE be the design effect,

$$\text{DE} = 1 + \rho(M - 1)$$

The sampling distribution of the test statistic $z = \sqrt{n}(\bar{y} - \mu_0)/(\sigma\sqrt{\text{DE}})$ under the null hypothesis follows a standard normal distribution; see, for example, [Ahn, Heo, and Zhang \(2015\)](#). In a CRD, the observed mean has a variance of $\sigma^2\text{DE}$.

Let α be the significance level, β be the probability of a type II error, and $z_{1-\alpha}$ and z_β be the $(1 - \alpha)$ th and the β th quantiles of the standard normal distribution.

The power $\pi = 1 - \beta$ is computed using

$$\pi = \begin{cases} \Phi(\sqrt{n}\delta - z_{1-\alpha}) & \text{for an upper one-sided test} \\ \Phi(-\sqrt{n}\delta - z_{1-\alpha}) & \text{for a lower one-sided test} \\ \Phi(\sqrt{n}\delta - z_{1-\alpha/2}) + \Phi(-\sqrt{n}\delta - z_{1-\alpha/2}) & \text{for a two-sided test} \end{cases} \quad (1)$$

where $\Phi(\cdot)$ is the c.d.f. of the standard normal distribution and $\delta = (\mu_a - \mu_0)/(\sigma\sqrt{\text{DE}})$ is the effect size.

Given the cluster size, the number of clusters K for a one-sided test is computed by inverting a one-sided power equation from (1):

$$K = \left(\frac{z_{1-\alpha} - z_\beta}{\delta\sqrt{M}} \right)^2 \quad (2)$$

Given the number of clusters K , the cluster size M for a one-sided test is computed from (2), after substituting for δ ,

$$M = \frac{1 - \rho}{\frac{K(\mu_a - \mu_0)^2}{\sigma^2(z_{1-\alpha} - z_\beta)^2} - \rho} \quad (3)$$

The absolute value of the effect size for a one-sided test is computed as follows:

$$|\delta| = \frac{(z_{1-\alpha} - z_\beta)}{\sqrt{n}} \quad (4)$$

The minimum detectable value of the mean for a one-sided test is computed as

$$\mu_a = \mu_0 + (z_{1-\alpha} - z_\beta)\sigma\sqrt{\text{DE}}/\sqrt{n} \text{ when } \mu_a > \mu_0 \text{ and as}$$

$$\mu_a = \mu_0 - (z_{1-\alpha} - z_\beta)\sigma\sqrt{\text{DE}}/\sqrt{n} \text{ when } \mu_a < \mu_0.$$

The number of clusters, cluster size, and minimum detectable value of the mean for a two-sided test are computed iteratively using the two-sided power equation from (1). The initial values are obtained from (2), (3), and (4), with $\alpha/2$.

Unequal cluster sizes

For unequal cluster sizes, we assume that the cluster sizes are independent and identically distributed and are small relative to the number of clusters; see [Ahn, Heo, and Zhang \(2015\)](#) for details. Let the coefficient of variation of the cluster sizes be CV_{cl} . According to [van Breukelen, Candel, and Berger \(2007\)](#) and [Campbell and Walters \(2014\)](#), to adjust for varying cluster sizes, define the relative efficiency (RE) of unequal versus equal cluster sizes as

$$RE = 1 - \lambda(1 - \lambda)CV_{cl}^2$$

where $\lambda = \rho M / (\rho M + 1 - \rho)$. With unequal cluster sizes, the effect size δ becomes

$$\delta = \frac{\mu_a - \mu_0}{\sigma \sqrt{DE/RE}} \quad (5)$$

The number of clusters for a one-sided test is computed using (2) after substituting δ from (5).

The effect size for a one-sided test is computed using (4) and the minimum detectable value of the mean is computed as

$$\mu_a = \mu_0 + (z_{1-\alpha} - z_\beta)\sigma \sqrt{DE/RE}/\sqrt{n} \text{ when } \mu_a > \mu_0 \text{ and as}$$

$$\mu_a = \mu_0 - (z_{1-\alpha} - z_\beta)\sigma \sqrt{DE/RE}/\sqrt{n} \text{ when } \mu_a < \mu_0.$$

In all other cases, parameters are computed iteratively using the power equations in (1) with δ as defined in (5).

References

- Ahn, C., M. Heo, and S. Zhang. 2015. *Sample Size Calculations for Clustered and Longitudinal Outcomes in Clinical Research*. Boca Raton, FL: CRC Press.
- Campbell, M. J., and S. J. Walters. 2014. *How to Design, Analyse and Report Cluster Randomised Trials in Medicine and Health Related Research*. Chichester, UK: Wiley.
- Gallis, J. A., F. Li, H. Yu, and E. L. Turner. 2018. `cvcrand` and `cpctest`: Commands for efficient design and analysis of cluster randomized trials using constrained randomization and permutation tests. *Stata Journal* 18: 357–378.
- Tamhane, A. C., and D. D. Dunlop. 2000. *Statistics and Data Analysis: From Elementary to Intermediate*. Upper Saddle River, NJ: Prentice Hall.
- van Breukelen, G. J. P., M. J. J. M. Candel, and M. P. F. Berger. 2007. Relative efficiency of unequal versus equal cluster sizes in cluster randomized and multicentre trials. *Statistics in Medicine* 26: 2589–2603. <https://doi.org/10.1002/sim.2740>.

Also see

[PSS-2] **power onemean** — Power analysis for a one-sample mean test

[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] **ztest** — z tests (mean-comparison tests, known variance)

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on [citing Stata documentation](#).