power twomeans, cluster — Power analysis for a two-sample means test, CRD

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

Description

`power twomeans, cluster` computes group-specific numbers of clusters, group-specific cluster sizes, power, or the experimental-group mean for a two-sample means test in a cluster randomized design (CRD). It computes group-specific numbers of clusters given cluster sizes, power, and the values of the control-group and experimental-group means. It also computes group-specific cluster sizes given numbers of clusters, power, and the values of the control-group and experimental-group means. Alternatively, it computes power given numbers of clusters, cluster sizes, and the values of the control-group and experimental-group means, or it computes the experimental-group mean given numbers of clusters, cluster sizes, power, and the control-group mean. See [PSS-2] **power twomeans** for a general discussion of power and sample-size analysis for a two-sample means test. Also see [PSS-2] **power** for a general introduction to the `power` command using hypothesis tests.

Quick start

Numbers of clusters for a test of $H_0\colon \mu_1 = \mu_2$ versus $H_a\colon \mu_1 \neq \mu_2$ given alternative control- and experimental-group means $m_1 = 8$ and $m_2 = 12$ with common standard deviation of 9 and cluster sizes of 5 using default intraclass correlation of 0.5, power of 0.8, and significance level $\alpha = 0.05$

```
power twomeans 8 12, m1(5) m2(5) sd(9)
```

Same as above, but assume the intraclass correlation is 0.4

```
power twomeans 8 12, m1(5) m2(5) sd(9) rho(0.4)
```

Same as above, and assume that cluster sizes vary with an average of 5 and coefficient of variation of 0.8

```
power twomeans 8 12, m1(5) m2(5) sd(9) rho(0.4) cvcluster(0.8)
```

Group-specific numbers of clusters using a ratio of experimental to control clusters of 0.5

```
power twomeans 8 12, m1(5) m2(5) kratio(0.5) sd(9)
```

Cluster sizes for a test of $H_0\colon \mu_1 = \mu_2$ versus $H_a\colon \mu_1 \neq \mu_2$ given alternative control-group and experimental-group means $m_1 = 1$ and $m_2 = 1.5$ for 60 equal-sized clusters in the control group and 30 clusters in the experimental group using default intraclass correlation of 0.5, standard deviation of 1, power of 0.8, and significance level $\alpha = 0.05$

```
power twomeans 1 1.5, k1(60) k2(30)
```

Same as above, but compute experimental-group cluster size given the control-group cluster size of 5

```
power twomeans 1 1.5, k1(60) k2(30) m1(5) compute(M2)
```

Power for 20 clusters with cluster sizes of 5 in the control and experimental groups

```
power twomeans 1 1.5, k1(20) k2(20) m1(5) m2(5)
```

Power for multiple numbers of clusters in the experimental group

```
power twomeans 1 1.5, k1(20) k2(20(5)50) m1(5) m2(5)
```

Same as above, and display results in a graph of power versus the number of clusters in the experimental group

```
power twomeans 1 1.5, k1(20) k2(20(5)50) m1(5) m2(5) graph
```

Effect size and target experimental-group mean with power of 0.8

```
power twomeans 1, k1(20) k2(20) m1(5) m2(5) power(0.8)
```

Menu

Statistics > Power, precision, and sample size

Syntax

Compute numbers of clusters

```
power twomeans m1 m2 , { mspec | nspec cluster } [ options ]
```

Compute cluster sizes

```
power twomeans m1 m2 , kspec [ options ]
```

Compute power

```
power twomeans m1 m2 , kspec { mspec | nspec } [ options ]
```

Compute effect size and experimental-group mean

```
power twomeans m1 , kspec { mspec | nspec } power(numlist) [ options ]
```

where m_1 is the mean in the control (reference) group and m_2 is the mean in the experimental (comparison) group. m_1 and m_2 may each be specified either as one number or as a list of values in parentheses (see **[U] 11.1.8 numlist**).

kspec is one of

```
k1()  k2()
k1()  [kratio()]
k2()  [kratio()]
```

mspec is one of

```
m1()  m2()
m1()  [mratio()]
m2()  [mratio()]
```

nspec is one of

```
n1()  n2()
n1()  [nratio()]
n2()  [nratio()]
```

options	Description
Main	
`cluster`	perform computations for a CRD; implied by `k1()`, `k2()`, `m1()`, or `m2()`
* `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)`
* `k1(`*numlist*`)`	number of clusters in the control group
* `k2(`*numlist*`)`	number of clusters in the experimental group
* `kratio(`*numlist*`)`	cluster ratio, K2/K1; default is `kratio(1)`
* `m1(`*numlist*`)`	cluster size of the control group
* `m2(`*numlist*`)`	cluster size of the experimental group
* `mratio(`*numlist*`)`	cluster-size ratio, M2/M1; default is `mratio(1)`
* `n1(`*numlist*`)`	sample size of the control group
* `n2(`*numlist*`)`	sample size of the experimental group
* `nratio(`*numlist*`)`	sample-size ratio, N2/N1; default is `nratio(1)`
`compute(K1 | K2 | M1 | M2)`	solve for the number of clusters or cluster size in one group given the other group
`nfractional`	allow fractional numbers of clusters, cluster sizes, and sample sizes
* `diff(`*numlist*`)`	difference between the experimental-group mean and the control-group mean, $m_2 - m_1$; specify instead of the experimental-group mean m_2
* `sd(`*numlist*`)`	common standard deviation of the control and the experimental groups assuming equal standard deviations in both groups; default is `sd(1)`
* `sd1(`*numlist*`)`	standard deviation of the control group; requires `sd2()`
* `sd2(`*numlist*`)`	standard deviation of the experimental group; requires `sd1()`
* `rho(`*numlist*`)`	intraclass correlation; default is `rho(0.5)`
* `cvcluster(`*numlist*`)`	coefficient of variation for cluster sizes
`direction(upper|lower)`	direction of the effect for effect-size determination; default is `direction(upper)`, which means that the postulated value of the parameter is larger than the hypothesized value
`onesided`	one-sided test; default is two sided
`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 value for numbers of clusters, cluster sizes, or experimental-group mean
`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

*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).

column	Description	Symbol
`alpha`	significance level	α
`power`	power	$1-\beta$
`beta`	type-II-error probability	β
`K1`	number of clusters in the control group	K_1
`K2`	number of clusters in the experimental group	K_2
`kratio`	ratio of numbers of clusters, experimental to control	K_2/K_1
`M1`	cluster size of the control group	M_1
`M2`	cluster size of the experimental group	M_2
`mratio`	ratio of cluster sizes, experimental to control	M_2/M_1
`N`	total number of observations	N
`N1`	number of observations in the control group	N_1
`N2`	number of observations in the experimental group	N_2
`nratio`	ratio of sample sizes, experimental to control	N_2/N_1
`delta`	effect size	δ
`m1`	control-group mean	μ_1
`m2`	experimental-group mean	μ_2
`diff`	difference between the experimental-group mean and the control-group mean	$\mu_2-\mu_1$
`sd`	common standard deviation	σ
`sd1`	control-group standard deviation	σ_1
`sd2`	experimental-group standard deviation	σ_2
`rho`	intraclass correlation	ρ
`CV_cluster`	coefficient of variation for cluster sizes	$\mathrm{CV}_{\mathrm{cl}}$
`target`	target parameter; synonym for `m2`	
`_all`	display all supported columns	

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

Column `N` is shown in the table if specified.

Columns `N1` and `N2` are shown in the default table if `n1()` or `n2()` is specified.

Columns `nratio`, `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 the `k1()`, `k2()`, `m1()`, or `m2()` option is specified. `cluster` is required to compute the numbers of clusters when *nspec* is used to specify sample sizes instead of *mspec* for cluster sizes.

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

`k1(`*numlist*`)` specifies the number of clusters in the control group.

`k2(`*numlist*`)` specifies the number of clusters in the experimental group.

`kratio(`*numlist*`)` specifies the ratio of the numbers of clusters of the experimental group relative to the control group, `K2`/`K1`. The default is `kratio(1)`, meaning equal numbers of clusters in the two groups.

`m1(`*numlist*`)` specifies the cluster size of the control group. `m1()` may contain noninteger values.

`m2(`*numlist*`)` specifies the cluster size of the experimental group. `m2()` may contain noninteger values.

`mratio(`*numlist*`)` specifies the ratio of cluster sizes of the experimental group relative to the control group, `M2/M1`. The default is `mratio(1)`, meaning equal cluster sizes in the two groups.

`n1()`, `n2()`, `nratio()`; see [PSS-2] **power**.

`compute(K1 | K2 | M1 | M2)` solve for the number of clusters or cluster size of one group given the other group.

`nfractional`; see [PSS-2] **power**. The `nfractional` option displays fractional (without rounding) values of the numbers of clusters, cluster sizes, and sample sizes.

`diff()`, `sd()`, `sd1()`, `sd2()`; see [PSS-2] **power twomeans**.

`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 numbers of clusters or cluster sizes for sample-size determination or the initial value for the experimental-group 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 twomeans, cluster` but is not shown in the dialog box:

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

Remarks and examples

Remarks are presented under the following headings:

Using power twomeans, cluster
Computing numbers of clusters
Computing number of clusters in one group
Computing cluster sizes
Computing power
Computing effect size and experimental-group mean
Testing hypotheses about two means in a CRD

`power twomeans, cluster` requests that computations for the `power twomeans` 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 numbers of clusters and the cluster sizes. The sample-size determination thus consists of the determination of the numbers of clusters given cluster sizes or the determination of cluster sizes given the numbers of clusters. For a general discussion of using `power twomeans`, see [PSS-2] **power twomeans**. The discussion below is specific to the CRD.

Using power twomeans, cluster

If you specify the `cluster` option, include `k1()` or `k2()` to specify the number of clusters or include `m1()` or `m2()` to specify the cluster size, the `power twomeans` command will perform computations for a two-sample means 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. By default, all computations assume a balanced or equal-allocation design, meaning equal numbers of clusters and cluster sizes in both groups; see [PSS-4] **Unbalanced designs** for a description of how to specify an unbalanced design.

To compute the number of clusters in both groups, you must provide cluster sizes for both groups. There are multiple ways to supply cluster sizes, but the most common is to specify the cluster size of the control group in the `m1()` option and the cluster size of the experimental group in the `m2()` option. See *mspec* and *nspec* under *Syntax* for other specifications. When *nspec* is specified, the `cluster` option is also required to request that `power twomeans` perform computations for a CRD. The number of clusters is assumed to be equal in the two groups, but you can change this by specifying the ratio of the numbers of clusters in the experimental to the control group in the `kratio()` option. Other parameters are specified as described in *Using power twomeans* in [PSS-2] **power twomeans**.

To compute the cluster sizes in both groups, you must provide the numbers of clusters in both groups. There are several ways to supply the numbers of clusters; see *kspec* under *Syntax*. The most common is to specify the numbers of clusters in the control group and the experimental group in the `k1()` and `k2()` options, respectively. Equal cluster sizes are assumed in the two groups, but you can change this by specifying the ratio of the cluster sizes in the experimental to that of the control group in the `mratio()` option. Other parameters are specified as described in *Using power twomeans* in [PSS-2] **power twomeans**.

You can also compute the number of clusters or the cluster size in one of the groups given the number of clusters or the cluster size in the other group by specifying the `compute()` option. For example, to compute the number of clusters in the control group, you specify `compute(K1)` and provide the number of clusters in the experimental group in `k2()`. Likewise, to compute the cluster size in the control group, you specify `compute(M1)` and provide the cluster size of the experimental group in `m2()`. You can compute the number of clusters or cluster size for the experimental group in a similar manner.

The power and effect-size determination is the same as described in *Using power twomeans* in [PSS-2] **power twomeans**, but the sample-size information is supplied as the numbers of clusters *kspec* and either cluster sizes using *mspec* or, less commonly, sample sizes using *nspec*.

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 numbers of clusters, cluster sizes, and sample sizes are 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 sizes, then cluster sizes represent average cluster sizes and are thus not rounded. When sample sizes are specified using *nspec*, fractional cluster sizes may be reported to accommodate the specified numbers of clusters and sample sizes.

Some of `power twomeans, cluster`'s computations require iteration, such as to compute the numbers 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 numbers of clusters

To compute the numbers of clusters in each group, you must either provide the cluster size for each group using *mspec* or specify the `cluster` option and provide the sample sizes of both groups using *nspec*. The most common method is to use *mspec* of `m1()` and `m2()`. In addition, the control- and experimental-group means must be specified.

▷ Example 1: Numbers of clusters for a two-sample means test in a CRD, specify cluster sizes

Consider an example from Ahn, Heo, and Zhang (2015, 37) of a hypothetical cluster randomized trial in which the goal is to assess the effect of a health promotion program on increasing the level of physical activity measured in kcal/kg/day of individuals in church congregations. In this study, the church congregation is the unit of randomization and the individual participant is the unit of analysis. Churches will be randomly assigned either to the experimental group that participates in the promotion program or to the control group. Investigators plan to recruit 20 church members from each participating church and would like to detect a mean difference of 1.1 kcal/kg/day between the experimental and control groups. From previous studies, the common standard deviation is 3.67 kcal/kg/day. The investigator assumes an intraclass correlation of 0.025.

To compute the numbers of churches required to detect a mean change in physical activity of 1.1 kcal/kg/day with 80% power using a 5%-level two-sided test, we type

```
. power twomeans 0 1.1, m1(20) m2(20) sd(3.67) rho(0.025)

Performing iteration ...

Estimated numbers of clusters for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1

Study parameters:

        alpha =    0.0500
        power =    0.8000
        delta =    1.1000
           m1 =    0.0000
           m2 =    1.1000
           sd =    3.6700

Cluster design:

           M1 =        20
           M2 =        20
          rho =    0.0250

Estimated numbers of clusters and sample sizes:

           K1 =        13
           K2 =        13
           N1 =       260
           N2 =       260
```

We find that with 20 members per church, 13 churches and thus a total of 260 members per group are required to detect a change of 1.1 kcal/kg/day in physical activity given the standard deviation of 3.67 kcal/kg/day with 80% power using a 5%-level two-sided test.

For power twomeans, the actual value of the control-group mean does not change the results as long as the difference between the means is fixed. In this example, we used the control-group mean of 0.

◁

▷ Example 2: Numbers of clusters for a two-sample means test in a CRD, varying cluster sizes

Instead of a constant number of members in each church as in example 1, we assume that the numbers of members selected from each church vary. Suppose that the numbers of members selected from each church have a mean of 20 and a standard deviation of 4 in both groups and thus have a coefficient of variation of 0.2. To compute the numbers of clusters when cluster sizes vary, we specify the coefficient of variation in the cvcluster() option.

```
. power twomeans 0 1.1, m1(20) m2(20) sd(3.67) rho(0.025) cvcluster(0.2)

Performing iteration ...

Estimated numbers of clusters for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1

Study parameters:

        alpha =    0.0500
        power =    0.8000
        delta =    1.1000
           m1 =    0.0000
           m2 =    1.1000
           sd =    3.6700

Cluster design:

   Average M1 =   20.0000
   Average M2 =   20.0000
          rho =    0.0250
        CV_cl =    0.2000

Estimated numbers of clusters and sample sizes:

           K1 =        14
           K2 =        14
           N1 =       280
           N2 =       280
```

The required number of churches in each group is 14, which is slightly larger than the required number of churches of 13 in example 1. When the number of members selected from each church varies, we need more churches to achieve the same power.

◁

▷ Example 3: Numbers of clusters for a two-sample means test in a CRD, specify sample sizes

Suppose that for our study, we can recruit only 200 members per group because of limited funding. We need to know the number of churches in each group and how many members to recruit in each church. In this case, we specify the n1(200) and n2(200) options. Because none of the k1(), k2(), m1(), or m2() options are specified, we also need to specify the cluster option so that computations are performed for a CRD instead of the conventional individual-level design.

```
. power twomeans 0 1.1, cluster n1(200) n2(200) sd(3.67) rho(0.025)
Performing iteration ...
Estimated numbers of clusters for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.1000
           m1 =    0.0000
           m2 =    1.1000
           sd =    3.6700
Cluster design:
           N1 =       200
           N2 =       200
          rho =    0.0250
Estimated numbers of clusters and cluster sizes:
           K1 =        30
           K2 =        30
   Average M1 =    6.6667
   Average M2 =    6.6667
```

To achieve the desired power, we need to recruit about 6.67 members on average per church from 30 churches to each of the control and experimental groups. `power twomeans, cluster` did not round the cluster sizes of 6.67 to meet our required total of 200 members per group. In practice, you can either decide to recruit 6 members from some churches and 7 from other churches to have roughly constant cluster sizes or decide to change the total number of members you want to recruit.

◁

Computing number of clusters in one group

To compute the number of clusters in one of the groups, you must specify the `compute()` option and the number of clusters in the other group. For example, to compute the number of clusters in the experimental group, you must specify the `compute(K2)` option and provide the number of clusters in the control group in the `k1()` option. Similarly, you can compute the number of clusters for the control group. In addition, you must provide cluster sizes of both groups using *mspec* or sample sizes of both groups using *nspec* and the control- and experimental-group means.

▷ Example 4: Number of clusters in the experimental group for a two-sample means test in a CRD

Continuing with example 1, suppose that we are designing a new study and we are planning to recruit 25 churches for the control group. We want to know the minimum number of churches we need to recruit to the experimental group. Given other study parameters from example 1, we compute the number of churches in the experimental group by specifying the `compute(K2)` option and the number of clusters in the control group of 25 in the `k1()` option.

```
. power twomeans 0 1.1, compute(K2) k1(25) m1(20) m2(20) sd(3.67) rho(0.025)
Performing iteration ...
Estimated experimental-group number of clusters for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.1000
           m1 =    0.0000
           m2 =    1.1000
           sd =    3.6700
Cluster design:
           K1 =        25
           M1 =        20
           M2 =        20
           N1 =       500
          rho =    0.0250
Estimated number of clusters and sample size:
           K2 =         9
           N2 =       180
```

With 25 churches in the control group, we need to recruit 9 churches for the experimental group.

◁

Computing cluster sizes

To compute cluster sizes in both groups, you must provide the numbers of clusters in both groups by using *kspec*. The most common method is to specify the numbers of clusters in the control and experimental groups in the `k1()` and `k2()` options, respectively. In addition, the control- and experimental-group means must be specified.

▷ Example 5: Cluster sizes for a two-sample means test in a CRD

Continuing with example 1, suppose that we are designing a new study and we are planning to recruit 30 churches with 15 churches in each group. Given other study parameters from example 1, we compute the numbers of members to recruit from each church by specifying 15 clusters in the `k1()` and `k2()` options.

```
. power twomeans 0 1.1, k1(15) k2(15) sd(3.67) rho(0.025)
Performing iteration ...
Estimated cluster sizes for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.1000
           m1 =    0.0000
           m2 =    1.1000
           sd =    3.6700
Cluster design:
           K1 =        15
           K2 =        15
          rho =    0.0250
Estimated cluster sizes and sample sizes:
           M1 =        17
           M2 =        17
           N1 =       255
           N2 =       255
```

With 15 churches per group, we need to recruit 17 members per church for a total of 255 members per group.

Computing power

To compute power in a CRD, you supply the sample-size information as the numbers of clusters by using *kspec* along with either the cluster sizes by using *mspec* or, less commonly, the sample sizes by using *nspec*. The most common method is to specify the `k1()`, `k2()`, `m1()`, and `m2()` options. In addition, the control- and experimental-group means must be specified.

Example 6: Power for a two-sample means test in a CRD

Continuing with example 1, suppose that we can recruit 20 members from each of 30 churches (15 churches per group) and we want to compute power for this design. Given other study parameters from example 1, we compute the power by specifying 15 in the `k1()` and `k2()` options and the cluster size of 20 in the `m1()` and `m2()` options:

```
. power twomeans 0 1.1, k1(15) k2(15) m1(20) m2(20) sd(3.67) rho(0.025)
Estimated power for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        delta =    1.1000
           m1 =    0.0000
           m2 =    1.1000
           sd =    3.6700
Cluster design:
           K1 =        15
           K2 =        15
           M1 =        20
           M2 =        20
           N1 =       300
           N2 =       300
          rho =    0.0250
Estimated power:
        power =    0.8560
```

The computed power is about 86%.

◁

▷ Example 7: Multiple values of study parameters

To investigate the effect of the number of clusters in the experimental group on power, we can specify a list of numbers of clusters in the `k2()` option:

```
. power twomeans 0 1.1, k1(15) k2(5(10)45) m1(20) m2(20) sd(3.67) rho(0.025)
> table(power K2)
Estimated power for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
```

power	K2
.5704	5
.856	15
.9221	25
.947	35
.9592	45

In this example, we also specified the `table(power K2)` option to list the only two columns that vary. As expected, as the number of clusters in the experimental group 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 experimental-group mean

Effect size δ for a two-sample means test is defined as the difference between the experimental-group mean and the control-group mean, $\delta = \mu_2 - \mu_1$. To compute effect size in a CRD, you supply the sample-size information as the numbers of clusters by using *kspec* along with either the cluster sizes by using *mspec* or, less commonly, the sample sizes by using *nspec*. The most common method is to specify the `k1()`, `k2()`, `m1()`, and `m2()` options. In addition, power and control-group mean must be specified. You must also decide on the direction of the effect, which is specified in the `direction()` option. For the default, upper, meaning $m_2 > m_1$, `power twomeans, cluster` uses `direction(upper)`. For lower, meaning $m_2 < m_1$, specify `direction(lower)`.

▷ Example 8: Effect size for a two-sample means test in a CRD

Continuing with example 6, we may also be interested in finding the minimum value of the difference in physical activity level between the two groups that can be detected with a sample of 15 churches per group, 20 members per church, and 80% power. To compute this, we specify the control-group mean of 0 as the command argument and the required options `k1(15)`, `k2(15)`, `m1(20)`, `m2(20)`, and `power(0.8)` and continue to use `sd(3.67)` and `rho(0.025)`.

```
. power twomeans 0, k1(15) k2(15) m1(20) m2(20) power(0.8) sd(3.67) rho(0.025)

Performing iteration ...

Estimated experimental-group mean for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1; m2 > m1

Study parameters:

        alpha =    0.0500
        power =    0.8000
           m1 =    0.0000
           sd =    3.6700

Cluster design:

           K1 =        15
           K2 =        15
           M1 =        20
           M2 =        20
           N1 =       300
           N2 =       300
          rho =    0.0250

Estimated effect size and experimental-group mean:

        delta =    1.0196
           m2 =    1.0196
```

Given 15 churches per group with 20 members per church and 80% power, the minimum detectable value of the difference in the physical activity level is about 1.02.

◁

Testing hypotheses about two means in a CRD

There are different ways to account for a CRD or for clustered data when performing hypothesis tests that compare means in two groups. With large samples or when you know the intraclass correlation and group-specific population standard deviations, the simplest way is to use a z test that accounts for clustered data; see [R] **ztest** for details. More commonly, two-level models such as those fit by `mixed` (see [ME] **mixed**) are used because they also allow adjusting for covariates.

In this section, we briefly demonstrate the `ztest` command for comparing means of two groups with clustered data.

▷ Example 9: Two-sample means test with clustered data

Consider example 6 in [R] **ztest** that compared the means of (log) BMI in two groups of patients with type-2 diabetes from a randomized controlled trial of patient-centered care of diabetes in general practice. The two groups included the comparison group that provided patients with routine care and an intervention group that provided patients with patient-centered care; see example 6 in [R] **ztest** for details. We replicate the analysis from that example below.

For clustered data, `ztest` requires that we specify the cluster identifier in the `cluster()` option and population intraclass correlation in the `rho()` option. We must also specify a common population standard deviation or group-specific population standard deviations in the respective options. We specify a common population standard deviation in the `sd()` option.

```
. use https://www.stata-press.com/data/r19/dcfd_trial
(BMI data from Diabetes Care from Diagnosis trial (Kinmonth et al., 1998))
. ztest lbmi, by(group) cluster(practice) rho(0.028) sd(0.35)

Two-sample z test
Cluster variable: practice

Group: Control                          Group: Interv.
  Number of clusters =        20          Number of clusters =        18
  Avg. cluster size  =      5.10          Avg. cluster size  =      7.67
  CV cluster size    =    0.5330          CV cluster size    =    0.5126
  Intraclass corr.   =    0.0280          Intraclass corr.   =    0.0280

------------------------------------------------------------------------------
       Group |      Obs        Mean    Std. err.   Std. dev.   [95% conf. interval]
-------------+----------------------------------------------------------------
     Control |      102     2.62954    .0372502         .35    2.556531    2.702549
      Interv |      138    2.749023    .0332182         .35    2.683916     2.81413
-------------+----------------------------------------------------------------
        diff |           -.1194831    .0499102               -.2173054   -.0216608
------------------------------------------------------------------------------
    diff = mean(Control) - mean(Interv.)                          z =  -2.3940
H0: diff = 0

    Ha: diff < 0                 Ha: diff != 0                 Ha: diff > 0
 Pr(Z < z) = 0.0083         Pr(|Z| > |z|) = 0.0167          Pr(Z > z) = 0.9917
```

There is statistical evidence to reject the null hypothesis that the two group means are the same at the 5% significance level.

Suppose that we want to use the results of this study to design another study that compares the two types of care of diabetes in the same population. Specifically, we want to compute the required number of clusters given the average cluster sizes of 5.1 and 7.67 in two groups, the intraclass correlation of 0.028, and the coefficient of variation of cluster sizes of 0.53, as shown in the output above. The coefficients of variation of cluster sizes are slightly different between the two groups—we use the larger value with `power twomeans`. We also use the observed mean estimates of 2.6 and 2.75 in the computation.

```
. power twomeans 2.6 2.75, m1(5.1) m2(7.67) cvcluster(0.53) rho(0.028) sd(0.35)
Performing iteration ...
Estimated numbers of clusters for a two-sample means test
Cluster randomized design, z test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    0.1500
           m1 =    2.6000
           m2 =    2.7500
           sd =    0.3500
Cluster design:
   Average M1 =    5.1000
   Average M2 =    7.6700
          rho =    0.0280
        CV_cl =    0.5300
Estimated numbers of clusters and sample sizes:
           K1 =        17
           K2 =        17
           N1 =        87
           N2 =       131
```

The required number of clusters for each group is 17. Given varying cluster sizes, we need to have a total of 87 patients in the control group and a total of 131 patients in the intervention group.

◁

Stored results

`power twomeans, cluster` 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(K1)`	number of clusters in the control group
`r(K2)`	number of clusters in the experimental group
`r(kratio)`	ratio of numbers of clusters, `K2/K1`
`r(M1)`	cluster size of the control group
`r(M2)`	cluster size of the experimental group
`r(mratio)`	ratio of cluster sizes, `M2/M1`
`r(N)`	total sample size
`r(N1)`	sample size of the control group
`r(N2)`	sample size of the experimental group
`r(nratio)`	ratio of sample sizes, `N2/N1`
`r(nfractional)`	`1` if `nfractional` is specified, `0` otherwise
`r(onesided)`	`1` for a one-sided test, `0` otherwise
`r(m1)`	control-group mean
`r(m2)`	experimental-group mean
`r(diff)`	difference between the experimental- and control-group means
`r(sd)`	common standard deviation of the control and experimental groups
`r(sd1)`	standard deviation of the control group
`r(sd2)`	standard deviation of the experimental group
`r(rho)`	intraclass correlation
`r(CV_cluster)`	coefficient of variation for cluster sizes

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 estimated parameter
r(maxiter)	maximum number of iterations
r(iter)	number of iterations performed
r(tolerance)	requested parameter tolerance
r(deltax)	final parameter tolerance achieved
r(ftolerance)	requested distance of the objective function from zero
r(function)	final distance of the objective function from zero
r(converged)	1 if iteration algorithm converged, 0 otherwise

Macros

r(type)	test
r(method)	twomeans
r(design)	CRD
r(direction)	upper or lower
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

For the computation in a CRD, we assume the standard deviations of the two groups are known. See *Known standard deviations* under *Methods and formulas* in [PSS-2] **power twomeans** for the common notation for a two-sample means test.

Methods and formulas are presented under the following headings:

Introduction
Equal cluster sizes
Unequal cluster sizes

Introduction

In a CRD, let K_1 and K_2 be the numbers of clusters in the control and experimental groups, respectively, and M_1 and M_2 be the cluster sizes of the control and experimental groups, respectively. We have $n_1 = K_1M_1$ and $n_2 = K_2M_2$. Let R_k be the ratio of the numbers of clusters, K_2/K_1, and R_m be the ratio of the cluster sizes, M_2/M_1. Let ρ be the intraclass correlation coefficient and DE_1 and DE_2 be the design effect in the control and experimental groups, with

$$\text{DE}_1 = 1 + \rho(M_1 - 1) \qquad \text{and} \qquad \text{DE}_2 = 1 + \rho(M_2 - 1)$$

Similarly to the discussion for the two-sample means test in the individual-level design, the test statistic is

$$\text{TS} = \frac{(\overline{x}_2 - \overline{x}_1) - (\mu_2 - \mu_1)}{\sigma_D}$$

where $\overline{x}_1$ and $\overline{x}_2$ are the sample means of the two groups and σ_D is the standard deviation of the mean difference. The sampling distribution of the test statistic TS under the null hypothesis follows a standard normal distribution; see, for example, Ahn, Heo, and Zhang (2015).

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

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

where $\Phi(\cdot)$ is the c.d.f. of a standard normal distribution, $\delta = \mu_2 - \mu_1$, and σ_D is defined in the subsequent sections.

Equal cluster sizes

When the cluster sizes are equal, the standard deviation of the mean difference σ_D in (1) is computed as

$$\sigma_D = \sqrt{\sigma_1^2 \mathrm{DE}_1 / n_1 + \sigma_2^2 \mathrm{DE}_2 / n_2}$$

where σ_1^2 and σ_2^2 are group-specific variances.

Given the cluster sizes M_1 and M_2 and the ratio of the numbers of clusters R_k, the numbers of clusters K_1 and K_2 for a one-sided test are computed as follows. K_1 is computed by inverting a one-sided power equation from (1)

$$K_1 = \left(\frac{z_{1-\alpha} - z_\beta}{\mu_2 - \mu_1}\right)^2 \left(\frac{\sigma_1^2 \mathrm{DE}_1}{M_1} + \frac{\sigma_2^2 \mathrm{DE}_2}{M_2 R_k}\right) \quad (2)$$

Then, K_2 is computed using $K_2 = R_k K_1$.

For a one-sided test, to compute the number of clusters in one group given that of the other one—for example, to compute K_1 given K_2—we use the following formula:

$$K_1 = \frac{\sigma_1^2 \mathrm{DE}_1 / M_1}{(\mu_2 - \mu_1)^2 / (z_{1-\alpha} - z_\beta)^2 - \sigma_2^2 \mathrm{DE}_2 / K_2 M_2} \quad (3)$$

Similarly, we can compute K_2 given K_1.

Given the numbers of clusters K_1 and K_2 and the cluster-size ratio R_m, the cluster sizes M_1 and M_2 for a one-sided test are computed as follows. M_1 is computed as

$$M_1 = \frac{(1-\rho)(\sigma_1^2 / K_1 + \sigma_2^2 / K_2 R_m)}{(\mu_2 - \mu_1)^2 / (z_{1-\alpha} - z_\beta)^2 - \rho(\sigma_1^2 / K_1 + \sigma_2^2 / K_2)} \quad (4)$$

and $M_2 = R_m M_1$.

For a one-sided test, to compute the cluster size in one group given that of the other one—for example, to compute M_1 given M_2—we use the following formula:

$$M_1 = \frac{(1-\rho)(\sigma_1^2 / K_1)}{(\mu_2 - \mu_1)^2 / (z_{1-\alpha} - z_\beta)^2 - \sigma_2^2 \mathrm{DE}_2 / K_2 M_2 - \rho \sigma_1^2 / K_1} \quad (5)$$

Similarly, we can compute M_2 given M_1.

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

$$|\delta| = \sigma_D(z_{1-\alpha} - z_\beta) \tag{6}$$

Note that the magnitude of the effect size is the same regardless of the direction of the test.

The experimental-group mean for a one-sided test is then computed as

$$\mu_2 = \begin{cases} \mu_1 + (z_{1-\alpha} - z_\beta)\sqrt{\sigma_1^2 \mathrm{DE}_1/n_1 + \sigma_2^2 \mathrm{DE}_2/n_2} & \text{when } \mu_2 > \mu_1 \\ \mu_1 - (z_{1-\alpha} - z_\beta)\sqrt{\sigma_1^2 \mathrm{DE}_1/n_1 + \sigma_2^2 \mathrm{DE}_2/n_2} & \text{when } \mu_2 < \mu_1 \end{cases} \tag{7}$$

The numbers of clusters, cluster sizes, and minimum detectable value of the experimental-group mean for a two-sided test are computed iteratively using the two-sided power equation from (1). The initial values are obtained from the corresponding one-sided equations [(2) through (6)] 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 $\mathrm{CV}_{\mathrm{cl}}$. 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

$$\mathrm{RE}_i = 1 - \lambda_i(1 - \lambda_i)\mathrm{CV}_{\mathrm{cl}}^2$$

where $\lambda_i = \rho M_i/(\rho M_i + 1 - \rho)$, where $i = 1$ corresponds to the control group and $i = 2$ corresponds to the experimental group. Under unequal cluster sizes, the standard deviation of the mean difference σ_D becomes

$$\sigma_D = \sqrt{\frac{\sigma_1^2 \mathrm{DE}_1}{n_1 \mathrm{RE}_1} + \frac{\sigma_2^2 \mathrm{DE}_1}{n_2 \mathrm{RE}_2}} \tag{8}$$

By substituting σ_D in (1) and replacing $\sigma_1^2\mathrm{DE}_1$ and $\sigma_2^2\mathrm{DE}_2$ with $\sigma_1^2\mathrm{DE}_1/\mathrm{RE}_1$ and $\sigma_2^2\mathrm{DE}_2/\mathrm{RE}_2$ in (2), (3), and (7), we can obtain the formulas for computing the numbers of clusters and experimental-group mean for a one-sided test. In all other cases, parameters are computed iteratively using the power equations (1) with σ_D as defined in (8).

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. https://doi.org/10.1201/b17822.

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. https://doi.org/10.1002/9781118763452.

Gallis, J. A., F. Li, H. Yu, and E. L. Turner. 2018. cvcrand and cptest: Commands for efficient design and analysis of cluster randomized trials using constrained randomization and permutation tests. *Stata Journal* 18: 357–378.

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 twomeans** — Power analysis for a two-sample means 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**

[ME] **mixed** — Multilevel mixed-effects linear regression+

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

Stata, Stata Press, Mata, NetCourse, and NetCourseNow are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow is a trademark of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on citing Stata documentation.