Unbalanced designs — Specifications for unbalanced designs

Description   Syntax   Options   Remarks and examples   Also see

Description

This entry describes the specifications of unbalanced designs for two-sample studies, including power and sample-size analysis for two-sample hypothesis tests and precision and sample-size analysis of two-sample CIs. See [PSS-2] **power** for a general introduction to the `power` command for power analysis and [PSS-3] **ciwidth** for a general introduction to the `ciwidth` command for precision analysis.

Syntax

Two samples, compute sample size for unbalanced designs

Compute total sample size

cmdname ..., nratio(*numlist*) [nfractional] ...

Compute one group size given the other

cmdname ..., n#(*numlist*) compute(N1 | N2) [nfractional] ...

Two samples, specify sample size for unbalanced designs

Specify total sample size and allocation ratio

cmdname ..., n(*numlist*) nratio(*numlist*) [nfractional] ...

Specify one of the group sizes and allocation ratio

cmdname ..., n#(*numlist*) nratio(*numlist*) [nfractional] ...

Specify total sample size and one of the group sizes

cmdname ..., n(*numlist*) n#(*numlist*) ...

Specify group sizes

cmdname ..., n1(*numlist*) n2(*numlist*) ...

cmdname can be either `power` for power analysis or `ciwidth` for precision analysis.

twosampleopts	Description
*n(*numlist*)	total sample size
*n1(*numlist*)	sample size of the control group
*n2(*numlist*)	sample size of the experimental group
*nratio(*numlist*)	ratio of sample sizes, N2/N1; default is nratio(1), meaning equal group sizes
compute(N1 | N2)	solve for N1 given N2 or for N2 given N1
nfractional	allow fractional sample sizes

*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.

Options

Main

n(*numlist*) specifies the total number of subjects in the study.

When used with power, this sample size is used for power or effect-size determination. If n() is specified, the power is computed. If n() and power() or beta() are specified, the minimum effect size that is likely to be detected in a study is computed.

When used with ciwidth, this sample size is used to compute the CI width and probability of CI width.

n1(*numlist*) specifies the number of subjects in the control group.

When used with power, this sample size is used for power or effect-size determination.

When used with ciwidth, this sample size is used to compute the CI width and probability of CI width.

n2(*numlist*) specifies the number of subjects in the experimental group. It is used for the same computations as n1(*numlist*), as mentioned above.

nratio(*numlist*) specifies the sample-size ratio of the experimental group relative to the control group, N2/N1. The default is nratio(1), meaning equal allocation between the two groups.

When used with power, this ratio is used for power and effect-size determination for two-sample tests.

When used with ciwidth, this ratio is used for computing CI width and probability of CI width for two-sample CIs.

compute(N1 | N2) requests that one of the group sample sizes be computed given the other one, instead of the total sample size. To compute the control-group sample size, you must specify compute(N1) and the experimental-group sample size in n2(). Alternatively, to compute the experimental-group sample size, you must specify compute(N2) and the control-group sample size in n1().

nfractional specifies that fractional sample sizes be allowed. When this option is specified, fractional sample sizes are used in the intermediate computations and are also displayed in the output.

Remarks and examples

stata.com

Remarks are presented under the following headings:

By default, for two-sample tests and CIs, the `power` and `ciwidth` commands assume a balanced design, but you may request an unbalanced design. A common way of specifying an unbalanced design is by specifying the `nratio()` option. You can also specify group sample sizes directly in the `n1()` and `n2()` options.

All considered options that accept arguments allow you to specify either one value # or *numlist*, a list of values as described in **[U] 11.1.8 numlist**. For simplicity, we demonstrate these options using only one value.

Below we describe in detail the specifications of unbalanced designs for two-sample methods and the handling of fractional sample sizes. As can be seen in *Syntax*, the specification of sample sizes, either total or group sample sizes, is the same across the `power` and `ciwidth` commands. In *Two samples*, we primarily use `power` in our examples, but the `n()`, `n1()`, `n2()`, `nratio()`, and `compute()` options shown there are used in the same fashion with `ciwidth`. Similarly, in *Fractional sample sizes*, we primarily use `ciwidth` in our examples, but the `nfractional` option shown there would be used in the same fashion with `power`.

Two samples

All two-sample methods, such as `power twomeans` and `ciwidth twomeans`, support the following options for specifying sample sizes: the total sample size `n()`, individual sample sizes `n1()` and `n2()`, and allocation ratio `nratio()`. The `compute()` option is useful if you want to compute one of the group sizes given the other one, instead of the total sample size.

We first describe the specifications and then demonstrate their use in real examples. The example below uses the `power` command, but the same principle applies to the `ciwidth` command.

We start with the sample-size determination—the default computation performed by the `power` command. The "switch" option for sample-size determination is the `power()` option. If you do not specify this option, it is implied with the default value of 0.8 corresponding to 80% power.

By default, group sizes are assumed to be equal; that is, the `nratio(1)` option is implied.

```
. power ..., [nratio(1)] ...
```

You can supply a different allocation ratio, n_2/n_1, to `nratio()` to request an unbalanced design.

```
. power ..., nratio(#) ...
```

To compute power or effect size, you must supply information about group sample sizes to `power`. Similarly, to compute CI width or probability of CI width, you must supply information about group sample sizes to `ciwidth`. There are several ways for you to do this. The simplest one, perhaps, is to specify the total sample size in the `n()` option.

```
. power ..., n(#) ...
```

The specification above assumes a balanced design in which the two group sizes are the same.

To request an unbalanced design, you can specify the desired allocation ratio between the two groups in the `nratio()` option:

```
. power ..., n(#) nratio(#) ...
```

The `nratio()` options assumes that the supplied values are the ratios of the second (experimental or comparison) group to the first (control or reference) group.

Alternatively, you can specify the two group sizes directly,

```
. power ..., n1(#) n2(#) ...
```

or you can specify one of the group sizes and the allocation ratio:

```
. power ..., n1(#) nratio(#) ...
. power ..., n2(#) nratio(#) ...
```

Also supported, but perhaps more rarely used, is a combination of the total sample size and one of the group sizes:

```
. power ..., n(#) n1(#) ...
. power ..., n(#) n2(#) ...
```

Below we demonstrate the described specifications using the `power twomeans` command, which provides power and sample-size analysis for tests of two independent means; see [PSS-2] **power twomeans** for details. In all examples, we use a value of 0 for the control-group mean, a value of 1 for the experimental-group mean, and the default values of the other study parameters.

▷ Example 1: Sample-size determination for a balanced design

By default, `power twomeans` computes sample size for a balanced design.

```
. power twomeans 0 1
Performing iteration ...
Estimated sample sizes for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated sample sizes:
            N =        34
  N per group =        17
```

The required total sample size is 34, with 17 subjects in each group.

The above is equivalent to specifying the `nratio(1)` option:

```
. power twomeans 0 1, nratio(1)
Performing iteration ...
Estimated sample sizes for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated sample sizes:
            N =        34
  N per group =        17
```

◁

▷ Example 2: Sample-size determination for an unbalanced design

To compute sample size for an unbalanced design, we specify the ratio of the experimental-group size to the control-group size in the `nratio()` option. For example, if we anticipate twice as many subjects in the experimental group as in the control group, we compute the corresponding sample size by specifying `nratio(2)`:

```
. power twomeans 0 1, nratio(2)
Performing iteration ...
Estimated sample sizes for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
        N2/N1 =    2.0000
Estimated sample sizes:
            N =        39
           N1 =        13
           N2 =        26
```

The required total sample size is 39, with 13 subjects in the control group and 26 subjects in the experimental group. Generally, unbalanced designs require more subjects than the corresponding balanced designs. This is the case for precision and sample size as well.

◁

▷ Example 3: Power determination for a balanced design

To compute power for a balanced design, we specify the total sample size in the `n()` option:

```
. power twomeans 0 1, n(30)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        30
  N per group =        15
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.7529
```

Equivalently, we specify one of the group sizes in the `n1()` or `n2()` option:

```
. power twomeans 0 1, n1(15)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        30
           N1 =        15
           N2 =        15
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.7529
```

Both specifications imply the `nratio(1)` option.

◁

▷ Example 4: Power determination for an unbalanced design

As we described in *Two samples*, there are a number of ways for you to request an unbalanced design for power and precision determination. Below we provide an example for each specification.

Specifying total sample size and allocation ratio

Similarly to example 2 but for power determination, we request an unbalanced design with twice as many subjects in the experimental group as in the control group by specifying the `nratio(2)` option:

```
. power twomeans 0 1, n(30) nratio(2)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        30
           N1 =        10
           N2 =        20
        N2/N1 =    2.0000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.7029
```

The computed power of 0.7029 is lower than the power of 0.7529 of the corresponding balanced design from example 3.

Specifying group sample sizes

Instead of the total sample size and the allocation ratio, we can specify the group sample sizes directly in the `n1()` and `n2()` options:

```
. power twomeans 0 1, n1(10) n2(20)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        30
           N1 =        10
           N2 =        20
        N2/N1 =    2.0000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.7029
```

Specifying one of the group sample sizes and allocation ratio

Alternatively, we can specify one of the group sizes and the allocation ratio. Here we specify the control-group size.

```
. power twomeans 0 1, n1(10) nratio(2)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        30
           N1 =        10
           N2 =        20
        N2/N1 =    2.0000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.7029
```

We could have specified the experimental-group size instead:

```
. power twomeans 0 1, n2(20) nratio(2)
  (output omitted)
```

Specifying total sample size and one of the group sample sizes

Finally, we can specify a combination of the total sample size and one of the group sizes—the control group:

```
. power twomeans 0 1, n1(10) n(30)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
H0: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        30
           N1 =        10
           N2 =        20
        N2/N1 =    2.0000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.7029
```

or the experimental group:

```
. power twomeans 0 1, n2(20) n(30)
  (output omitted)
```

◁

Options `n()`, `n1()`, and `n2()` require integer numbers. When you specify the `n1()` and `n2()` options, your sample sizes are guaranteed to be integers. This is not necessarily true for other specifications for which the resulting sample sizes may be fractional. See *Fractional sample sizes* for details about how the `power` and `ciwidth` commands handle fractional sample sizes.

We show examples using the `ciwidth` command, but the same principles apply to the `power` command.

Fractional sample sizes

Certain sample-size specifications may lead to fractional sample sizes. For example, if you specify an odd value for the total sample size of a two-sample study, the two group sample sizes would have to be fractional to accommodate the specified total sample size. Also, if you specify the `nratio()` option with a two-sample method, the resulting sample sizes may be fractional.

By default, the `power` and `ciwidth` commands round sample sizes to integers and use integer values in the computations. To ensure conservative results, the commands round down the input sample sizes and round up the output sample sizes.

▷ Example 5: Output sample sizes

For example, when we compute sample size, the sample size is rounded up to the nearest integer by default:

```
. ciwidth onemean, width(1) probwidth(0.9)
Performing iteration ...
Estimated sample size for a one-mean CI
Student's t two-sided CI
Study parameters:
        level =     95.00
     Pr_width =    0.9000
        width =    1.0000
           sd =    1.0000
Estimated sample size:
            N =        24
```

We computed sample size for a one-sample mean CI; see [PSS-3] **ciwidth onemean** for details.

We can specify the `nfractional` option to see the corresponding fractional sample size:

```
. ciwidth onemean, width(1) probwidth(0.9) nfractional
Performing iteration ...
Estimated sample size for a one-mean CI
Student's t two-sided CI
Study parameters:
        level =     95.00
     Pr_width =    0.9000
        width =    1.0000
           sd =    1.0000
Estimated sample size:
            N =   23.8582
```

The sample size of 24 reported above is the ceiling for the fractional sample size 23.86.

We can also compute the actual CI width corresponding to the rounded sample size:

```
. ciwidth onemean, n(24) probwidth(0.9)
Estimated width for a one-mean CI
Student's t two-sided CI
Study parameters:
        level =     95.00
            N =        24
     Pr_width =    0.9000
           sd =    1.0000
Estimated width:
        width =    0.9963
```

The actual CI width corresponding to the sample size of 24 is smaller than the specified CI width of 1 from the two previous examples because the sample size was rounded up.

◁

On the other hand, the power and ciwidth commands round down the input sample sizes.

▷ Example 6: Input sample sizes

For example, let's use `ciwidth twomeans` to compute the CI width for a two-means-difference CI, given a total sample size of 51. To be 95% certain that the width of a future CI for this sample size will be no larger than the value we estimate, we specify `probwidth(0.95)`. We use the default 95% confidence level; see [PSS-3] **ciwidth twomeans** for details.

```
. ciwidth twomeans, probwidth(0.95) n(51)
Estimated width for a two-means-difference CI
Student's t two-sided CI assuming sd1 = sd2 = sd
Study parameters:
        level =     95.00
            N =        51
     Pr_width =    0.9500
           sd =    1.0000
Actual sample sizes:
            N =        50
  N per group =        25
Estimated width:
        width =    1.3253
```

By default, `ciwidth twomeans` assumes a balanced design. To accommodate a balanced design, the command rounds down the group sample sizes from 25.5 to 25 for an actual total sample size of 50.

When the specified sample sizes differ from the resulting rounded sample sizes, the actual sample sizes used in the computations are reported. In our example, we requested a total sample size of 51, but the actual sample size used to compute the CI width was 50.

We can specify the `nfractional` option to request that fractional sample sizes be used in the computations.

```
. ciwidth twomeans, probwidth(0.95) n(51) nfractional

Estimated width for a two-means-difference CI
Student's t two-sided CI assuming sd1 = sd2 = sd

Study parameters:

        level =     95.00
            N =   51.0000
  N per group =   25.5000
     Pr_width =    0.9500
           sd =    1.0000

Estimated width:

        width =    1.3097
```

The fractional group sample sizes of 25.5 are now used in the computations.

If we want to preserve the total sample size of 51 and ensure that group sample sizes are integers, we can specify the group sizes directly:

```
. ciwidth twomeans, probwidth(0.95) n1(25) n2(26)

Estimated width for a two-means-difference CI
Student's t two-sided CI assuming sd1 = sd2 = sd

Study parameters:

        level =     95.00
            N =        51
           N1 =        25
           N2 =        26
        N2/N1 =    1.0400
     Pr_width =    0.9500
           sd =    1.0000

Estimated width:

        width =    1.3099
```

Alternatively, we can specify one of the group sizes (or the total sample size) and the corresponding allocation ratio $n_2/n_1 = 26/25 = 1.04$:

```
. ciwidth twomeans, probwidth(0.95) n1(25) nratio(1.04)

Estimated width for a two-means-difference CI
Student's t two-sided CI assuming sd1 = sd2 = sd

Study parameters:

        level =     95.00
            N =        51
           N1 =        25
           N2 =        26
        N2/N1 =    1.0400
     Pr_width =    0.9500
           sd =    1.0000

Estimated width:

        width =    1.3099
```

We obtain the same CI width of 1.31.

In the command above, the sample-size ratio we specified resulted in integer sample sizes. This may not always be the case. For example, if we specify the sample-size ratio of 1.3,

```
. ciwidth twomeans, probwidth(0.95) n1(25) nratio(1.3)
Estimated width for a two-means-difference CI
Student's t two-sided CI assuming sd1 = sd2 = sd
Study parameters:
        level =    95.00
           N1 =       25
        N2/N1 =   1.3000
     Pr_width =   0.9500
           sd =   1.0000
Actual sample sizes:
            N =       57
           N1 =       25
           N2 =       32
        N2/N1 =   1.2800
Estimated width:
        width =   1.2352
```

the experimental-group size of 32.5 is rounded down to 32. The total sample size used in the computation is 57, and the actual sample-size ratio is 1.28.

As before, we can specify the `nfractional` option to use the fractional experimental-group size of 32.5 in the computations:

```
. ciwidth twomeans, probwidth(0.95) n1(25) nratio(1.3) nfractional
Estimated width for a two-means-difference CI
Student's t two-sided CI assuming sd1 = sd2 = sd
Study parameters:
        level =    95.00
            N =  57.5000
           N1 =  25.0000
           N2 =  32.5000
        N2/N1 =   1.3000
     Pr_width =   0.9500
           sd =   1.0000
Estimated width:
        width =   1.2300
```

◁

Also see

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

[PSS-3] **ciwidth** — Precision and sample-size analysis for CIs

[PSS-5] **Glossary**