Title

unbalanced designs — Specifications for unbalanced designs

Syntax   Description   Options   Remarks and examples   Also see

Syntax

Two samples, compute sample size for unbalanced designs

Compute total sample size

`power ..., nratio(numlist) [nfractional] ...`

Compute one group size given the other

`power ..., n#(numlist) compute(n1|n2) [nfractional] ...`

Two samples, specify sample size for unbalanced designs

Specify total sample size and allocation ratio

`power ..., n(numlist) nratio(numlist) [nfractional] ...`

Specify one of the group sizes and allocation ratio

`power ..., n#(numlist) nratio(numlist) [nfractional] ...`

Specify total sample size and one of the group sizes

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

Specify group sizes

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

twosampleopts	Description
*`n(numlist)`	total sample size; required to compute power or effect size
*`n1(numlist)`	sample size of the control group
*`n2(numlist)`	sample size of the experimental group
*`nratio(numlist)`	ratio of sample sizes, N_2/N_1; default is `nratio(1)`, meaning equal group sizes
`compute(n1|n2)`	solve for N_1 given N_2 or for N_2 given N_1
`nfractional`	allow fractional sample sizes

*Starred options may be specified either as one number or as a list of values; see **[U] 11.1.8 numlist**.

Description

This entry describes the specifications of unbalanced designs with the power command for two-sample hypothesis tests. See [PSS] **power** for a general introduction to the power command using hypothesis tests.

Options

Main

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

n1(*numlist*) specifies the number of subjects in the control group to be used for power or effect-size determination.

n2(*numlist*) specifies the number of subjects in the experimental group to be used for power or effect-size determination.

nratio(*numlist*) specifies the sample-size ratio of the experimental group relative to the control group, N_2/N_1 for power or effect-size determination for two-sample tests. The default is nratio(1), meaning equal allocation between the two groups.

compute(n1 | n2) requests that the power command compute one of the group sample sizes given the other one instead of the total sample size for two-sample tests. 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 a two-sample test, the power command assumes 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 a *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.

Two samples

All two-sample methods such as `power twomeans` and `power twoproportions` 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.

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`. 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 PSS analysis for tests of two independent means; see [PSS] **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
Ho: 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
Ho: 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
Ho: 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.

◁

▷ Example 3: Power determination for a balanced design

To computer 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
Ho: 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
Ho: 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 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
Ho: 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
Ho: 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
Ho: 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
Ho: 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` command handles fractional sample sizes.

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` command rounds sample sizes to integers and uses integer values in the computations. To ensure conservative results, the command rounds down the input sample sizes and rounds 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:

```
. power onemean 0 1.5
Performing iteration ...
Estimated sample size for a one-sample mean test
t test
Ho: m = m0  versus  Ha: m != m0
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.5000
           m0 =    0.0000
           ma =    1.5000
           sd =    1.0000
Estimated sample size:
            N =         6
```

We computed sample size for a one-sample mean test; see [PSS] **power onemean** for details.

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

```
. power onemean 0 1.5, nfractional
Performing iteration ...
Estimated sample size for a one-sample mean test
t test
Ho: m = m0  versus  Ha: m != m0
Study parameters:
        alpha =    0.0500
        power =    0.8000
        delta =    1.5000
           m0 =    0.0000
           ma =    1.5000
           sd =    1.0000
Estimated sample size:
            N =    5.6861
```

The sample size of 6 reported above is the ceiling for the fractional sample size 5.6861.

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

```
. power onemean 0 1.5, n(6)
Estimated power for a one-sample mean test
t test
Ho: m = m0  versus  Ha: m != m0
Study parameters:
        alpha =    0.0500
            N =         6
        delta =    1.5000
           m0 =    0.0000
           ma =    1.5000
           sd =    1.0000
Estimated power:
        power =    0.8325
```

The actual power corresponding to the sample size of 6 is larger than the specified power of 0.8 from the two previous examples because the sample size was rounded up.

◁

On the other hand, the `power` command rounds down the input sample sizes.

▷ Example 6: Input sample sizes

For example, let's use `power twomeans` to compute the power of a two-sample means test using a total sample size of 51 and the default settings for other parameters; see [PSS] **power twomeans** for details.

```
. power twomeans 0 1, n(51)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
Ho: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        51
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Actual sample sizes:
            N =        50
  N per group =        25
Estimated power:
        power =    0.9337
```

By default, `power 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 power was 50.

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

```
. power twomeans 0 1, n(51) nfractional
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
Ho: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =   51.0000
  N per group =   25.5000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.9382
```

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:

```
. power twomeans 0 1, n1(25) n2(26)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
Ho: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        51
           N1 =        25
           N2 =        26
        N2/N1 =    1.0400
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.9381
```

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

```
. power twomeans 0 1, n1(25) nratio(1.04)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
Ho: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =        51
           N1 =        25
           N2 =        26
        N2/N1 =    1.0400
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.9381
```

We obtain the same power of 0.9381.

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

```
. power twomeans 0 1, n1(25) nratio(1.3)
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
Ho: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
           N1 =        25
        N2/N1 =    1.3000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Actual sample sizes:
            N =        57
           N1 =        25
           N2 =        32
        N2/N1 =    1.2800
Estimated power:
        power =    0.9573
```

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:

```
. power twomeans 0 1, n1(25) nratio(1.3) nfractional
Estimated power for a two-sample means test
t test assuming sd1 = sd2 = sd
Ho: m2 = m1  versus  Ha: m2 != m1
Study parameters:
        alpha =    0.0500
            N =   57.5000
           N1 =   25.0000
           N2 =   32.5000
        N2/N1 =    1.3000
        delta =    1.0000
           m1 =    0.0000
           m2 =    1.0000
           sd =    1.0000
Estimated power:
        power =    0.9585
```

◁

Also see

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

[PSS] **Glossary**