

# sampsizepwr

Sample size and power of test

## Syntax

```
nout = sampsizepwr(testtype,p0,p1)
nout = sampsizepwr(testtype,p0,p1,pwr)

pwrout = sampsizepwr(testtype,p0,p1,[],n)

plout = sampsizepwr(testtype,p0,[],pwr,n)

__ = sampsizepwr(testtype,p0,p1,pwr,n,Name,Value)
```

## Description

`sampsizepwr` computes the sample size, power, or alternative parameter value for a hypothesis test, given the other two values. For example, you can compute the sample size required to obtain a particular power for a hypothesis test, given the parameter value of the alternative hypothesis.

`nout = sampsizepwr(testtype,p0,p1)` returns the sample size, `nout`, required for a two-sided test of the type specified by `testtype` to have a power (probability of rejecting the null hypothesis when the alternative hypothesis is true) of 0.90 when the significance level (probability of rejecting the null hypothesis when the null hypothesis is true) is 0.05. `p0` specifies parameter values under the null hypothesis. `p1` specifies the value, or an array of values, of the single parameter being tested under the alternative hypothesis.

`nout = sampsizepwr(testtype,p0,p1,pwr)` returns the sample size, `nout`, that corresponds to the specified power, `pwr`, and the parameter value under the alternative hypothesis, `p1`.

[example](#)

`pwrout = sampsizepwr(testtype,p0,p1,[],n)` returns the power achieved for a sample size of `n` when the true parameter value is `p1`.

[example](#)

`plout = sampsizepwr(testtype,p0,[],pwr,n)` returns the parameter value detectable with the specified sample size, `n`, and the specified power, `pwr`.

[example](#)

`__ = sampsizepwr(testtype,p0,p1,pwr,n,Name,Value)` returns any of the previous arguments using one or more name-value pair arguments. For example, you can change the significance level of the test, or specify a right- or left-tailed test. The name-value pairs can appear in any order but must begin in the sixth argument position.

[example](#)

## Examples

[collapse all](#)

### ▼ Compute Sample Size for Selected Power Value

A company runs a manufacturing process that fills empty bottles with 100 mL of liquid. To monitor quality, the company randomly selects several bottles and measures the volume of liquid inside.

Determine the sample size the company must use for a *t*-test to detect a difference between 100 mL and 102 mL with a power of 0.80. Assume that a standard deviation is 5 mL.

Open in MATLAB  
Online

[View MATLAB Command](#)

```
nout = sampsizepwr('t',[100 5],102,0.80)
```

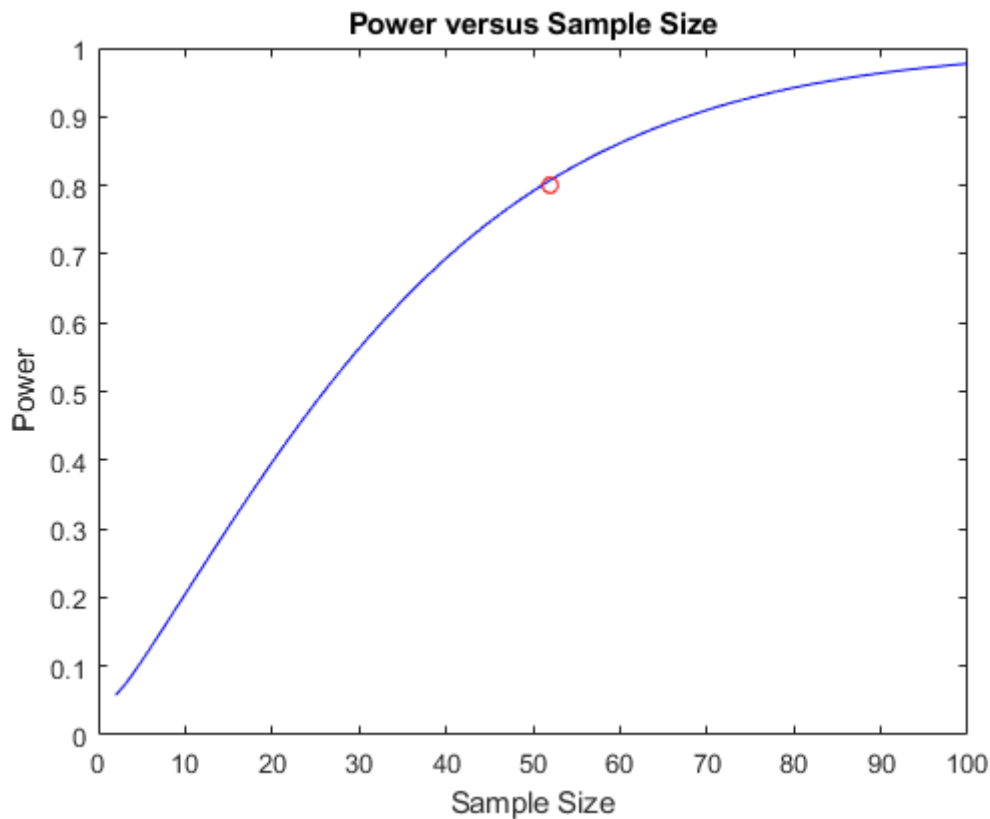
nout = 52

The company must test 52 bottles to detect the difference between a mean volume of 100 mL and 102 mL with a power of 0.80.

Generate a power curve to visualize how the sample size affects the power of the test.

```
nn = 1:100;
pwrout = sampsizewr('t',[100 5],102,[],nn);

figure;
plot(nn,pwrout,'b-',nout,0.8,'ro')
title('Power versus Sample Size')
xlabel('Sample Size')
ylabel('Power')
```



### ▼ Compute Power and Sample Size for One-Sided Test

An employee wants to buy a house near her office. She decides to eliminate from consideration any house that has a mean morning commute time greater than 20 minutes. The null hypothesis for this right-sided test is  $H_0: \mu = 20$ , and the alternative hypothesis is  $H_A: \mu > 20$ . The selected significance level is 0.05.

To determine the mean commute time, the employee takes a test drive from the house to her office during rush hour every morning for one week, so her total sample size is 5. She assumes that the standard deviation,  $\sigma$ , is equal to 5.

The employee decides that a true mean commute time of 25 minutes is too different from her targeted 20-minute limit, so she wants to detect a significant departure if the true mean is 25 minutes. Find the probability of incorrectly concluding that the mean commute time is no greater than 20 minutes.

Open in MATLAB  
Online

[View MATLAB Command](#)

Compute the power of the test, and then subtract the power from 1 to obtain  $\beta$ .

```
power = sampsizepwr('t',[20 5],25,[],5,'Tail','right');
beta = 1 - power
```

```
beta = 0.4203
```

The  $\beta$  value indicates a probability of 0.4203 that the employee concludes incorrectly that the morning commute is not greater than 20 minutes.

The employee decides that this risk is too high, and she wants no more than a 0.01 probability of reaching an incorrect conclusion. Calculate the number of test drives the employee must take to obtain a power of 0.99.

```
nout = sampsizepwr('t',[20 5],25,0.99,[],'Tail','right')
```

```
nout = 18
```

The results indicate that she must take 18 test drives from a candidate house to achieve this power level.

The employee decides that she only has time to take 10 test drives. She also accepts a 0.05 probability of making an incorrect conclusion. Calculate the smallest true parameter value that produces a detectable difference in mean commute time.

```
p1out = sampsizepwr('t',[20 5],[],0.95,10,'Tail','right')
```

```
p1out = 25.6532
```

Given the employee's target power level and sample size, her test detects a significant difference from a mean commute time of at least 25.6532 minutes.

## ▼ Compute Sample Size for a Binomial Test

Compute the sample size,  $n$ , required to distinguish  $p = 0.30$  from  $p = 0.36$ , using a binomial test with a power of 0.8.

Open in MATLAB  
Online

[View MATLAB Command](#)

```
napprox = sampsizepwr('p',0.30,0.36,0.8)
```

Warning: Values  $N > 200$  are approximate. Plotting the power as a function of  $N$  may reveal lower  $N$  values that have the required power.

```
napprox = 485
```

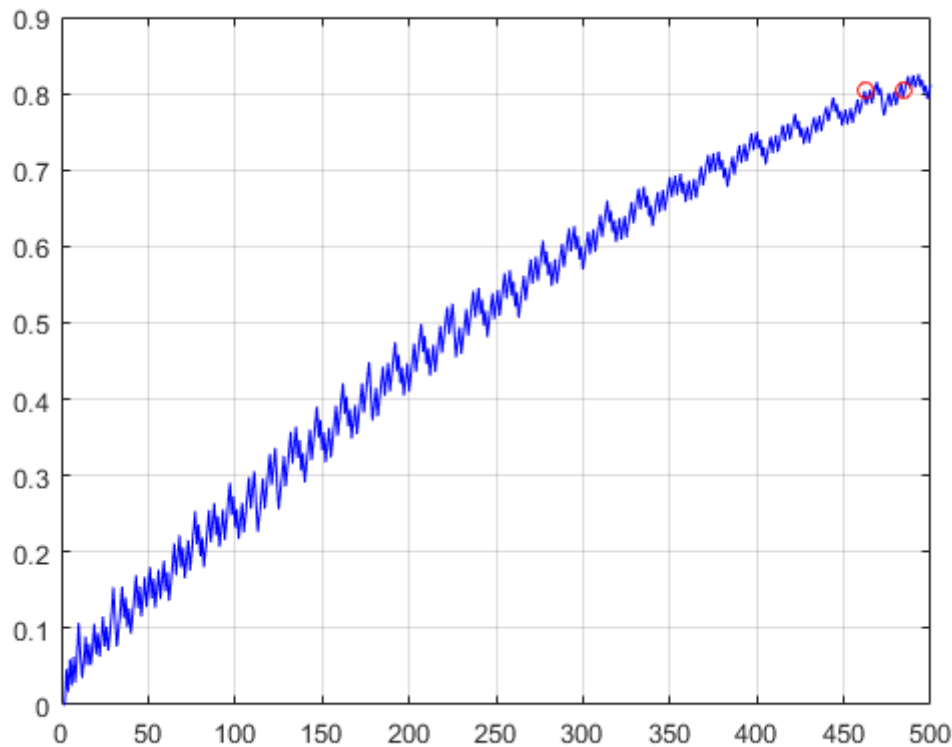
The result indicates that a power of 0.8 requires a sample size of 485. However, this result is approximate.

Make a plot to see if any smaller  $n$  values provide the required power of 0.8.

```
nn = 1:500;
pwrout = sampsizepwr('p',0.3,0.36,[],nn);
nexact = min(nn(pwrout>=0.8))
```

```
nexact = 462
```

```
figure
plot(nn,pwrout,'b-',[napprox nexact],pwrout([napprox nexact]),'ro')
grid on
```



The result indicates that a sample size of 462 also provides a power of 0.8 for this test.

### ▼ Compute Power for a Two-Sample t-Test

A farmer wants to test the impact of two different types of fertilizer on the yield of his bean crops. He currently uses Fertilizer A, but believes that Fertilizer B might improve crop yield. Because Fertilizer B is more expensive than Fertilizer A, the farmer wants to limit the number of plants he treats with Fertilizer B in this experiment.

Open in MATLAB  
Online

[View MATLAB Command](#)

The farmer uses a 2:1 ratio of plants in each treatment group. He tests 10 plants with Fertilizer A, and 5 plants with Fertilizer B. The mean yield using Fertilizer A is 1.4 kg per plant, with a standard deviation of 0.2. The mean yield using Fertilizer B is 1.7 kg per plant. The significance level of the test is 0.05.

Compute the power of the test.

```
pwr = sampsizewr('t2',[1.4 0.2],1.7,[],5,'Ratio',2)
```

```
pwr = 0.7165
```

The farmer wants to increase the power of the test to 0.90. Calculate how many plants he must treat with each type of fertilizer.

```
n = sampsizewr('t2',[1.4 0.2],1.7,0.9,[])
```

```
n = 11
```

To increase the power of the test to 0.90, the farmer must test 11 plants with each type of fertilizer.

The farmer wants to reduce the number of plants he must treat with Fertilizer B, but keep the power of the test at 0.90 and maintain the initial 2:1 ratio of plants in each treatment group

Using a 2:1 ratio of plants in each treatment group, calculate how many plants the farmer must test to obtain a power of 0.90. Use the mean and standard deviation values obtained in the previous test.

```
[n1out,n2out] = sampsizepwr('t2',[1.4,0.2],1.7,0.9,[],'Ratio',2)
```

```
n1out = 8
```

```
n2out = 16
```

To obtain a power of 0.90, the farmer must treat 16 plants with Fertilizer A and 8 plants with Fertilizer B.

## Input Arguments

[collapse all](#)

### testtype — Test type

'z' | 't' | 't2' | 'var' | 'p'

Test type, specified as one of the following.

- 'z' — z-test for normally distributed data with known standard deviation.
- 't' — *t*-test for normally distributed data with unknown standard deviation.
- 't2' — Two-sample pooled *t*-test for normally distributed data with unknown standard deviation and equal variances.
- 'var' — Chi-square test of variance for normally distributed data.
- 'p' — Test of the *p* parameter (success probability) for a binomial distribution. The 'p' test is a discrete test for which increasing the sample size does not always increase the power. For *n* values larger than 200, there may exist values smaller than the returned *n* value that also produce the specified power.

### p0 — Parameter value under null hypothesis

scalar value | two-element array of scalar values

Parameter value under the null hypothesis, specified as a scalar value or a two-element array of scalar values.

- If **testtype** is 'z' or 't', then *p0* is a two-element array [*mu0*, *sigma0*] of the mean and standard deviation, respectively, under the null hypothesis.
- If **testtype** is 't2', then *p0* is a two-element array [*mu0*, *sigma0*] of the mean and standard deviation, respectively, of the first sample under the null and alternative hypotheses.
- If **testtype** is 'var', then *p0* is the variance under the null hypothesis.
- If **testtype** is 'p', then *p0* is the value of *p* under the null hypothesis.

**Data Types:** single | double

### p1 — Parameter value under alternative hypothesis

scalar value | array of scalar values | []

Parameter value under the alternative hypothesis, specified as a scalar value or as an array of scalar values.

- If **testtype** is 'z' or 't', then *p1* is the value of the mean under the alternative hypothesis.
- If **testtype** is 't2', then *p1* is the value of the mean of the second sample under the alternative hypothesis.
- If **testtype** is 'var', then *p1* is the variance under the alternative hypothesis.
- If **testtype** is 'p', then *p1* is the value of *p* under the alternative hypothesis.

If you specify `p1` as an array, then `sampsizepwr` returns an array for `nout` or `pwrout` that is the same length as `p1`.

To return the alternative parameter value, `p1out`, specify `p1` using empty brackets (`[]`), as shown in the [syntax description](#).

**Data Types:** single | double



#### **pwr — Power of the test**

0.90 (default) | scalar value in the range (0,1) | array of scalar values in the range (0,1) | `[]`

Power of the test, specified as a scalar value in the range (0,1) or as an array of scalar values in the range (0,1). The power of a test is the probability of rejecting the null hypothesis when the alternative hypothesis is true, given a particular significance level.

If you specify `pwr` as an array, then `sampsizepwr` returns an array for `nout` or `p1out` that is the same length as `pwr`.

To return a power value, `pwrout`, specify `pwr` using empty brackets (`[]`), as shown in the [syntax description](#).

**Data Types:** single | double



#### **n — Sample size**

positive integer value | array of positive integer values

Sample size, specified as a positive integer value or as an array of positive integer values.

If `testtype` is `'t2'`, then `sampsizepwr` assumes that the two sample sizes are equal. For unequal sample sizes, specify `n` as the smaller of the two sample sizes, and use the `'Ratio'` name-value pair argument to indicate the sample size ratio. For example, if the smaller sample size is 5 and the larger sample size is 10, specify `n` as 5, and the `'Ratio'` name-value pair as 2.

If you specify `n` as an array, then `sampsizepwr` returns an array for `pwrout` or `p1out` that is the same length as `n`.

**Data Types:** single | double

## **Name-Value Pair Arguments**

Specify optional comma-separated pairs of `Name`, `Value` arguments. `Name` is the argument name and `Value` is the corresponding value. `Name` must appear inside quotes. You can specify several name and value pair arguments in any order as `Name1`, `Value1`, ..., `NameN`, `ValueN`.

**Example:** `'Alpha', 0.01, 'Tail', 'right'` specifies a right-tailed test with a 0.01 significance level.



#### **'Alpha' — Significance level**

0.05 (default) | scalar value in the range (0,1)

Significance value of the test, specified as the comma-separated pair consisting of `'Alpha'` and a scalar value in the range (0,1).

**Example:** `'Alpha', 0.01`

**Data Types:** single | double

▼ **'Ratio' — Sample size ratio**  
1 (default) | scalar value greater than or equal to 1

Sample size ratio for a two-sample *t*-test, specified as the comma-separated pair consisting of 'Ratio' and a scalar value greater than or equal to 1. The value of Ratio is equal to  $n_2/n_1$ , where  $n_2$  is the larger sample size, and  $n_1$  is the smaller sample size.

To return the power, `pwrout`, or alternative parameter value, `p1out`, specify the smaller of the two sample sizes for `n`, and use 'Ratio' to indicate the sample size ratio.

**Example:** 'Ratio',2

▼ **'Tail' — Test type**  
'both' (default) | 'right' | 'left'

Test type, specified as the comma-separated pair consisting of 'Tail' and one of the following:

- 'both' — Two-sided test for an alternative not equal to  $p_0$
- 'right' — One-sided test for an alternative larger than  $p_0$
- 'left' — One-sided test for an alternative smaller than  $p_0$

**Example:** 'Tail','right'

## Output Arguments

[collapse all](#)

▼ **nout — Sample size**  
positive integer value | array of positive integer values

Sample size, returned as a positive integer value or as an array of positive integer values.

If `testtype` is `t2`, and you use the 'Ratio' name-value pair argument to specify the ratio of the two unequal sample sizes, then `nout` returns the smaller of the two sample sizes.

Alternatively, to return both sample sizes, specify this argument as `[n1out,n2out]`. In this case, `sampsizepwr` returns the smaller sample size as `n1out`, and the larger sample size as `n2out`.

If you specify `pwr` or `p1` as an array, then `sampsizepwr` returns an array for `nout` that is the same length as `pwr` or `p1`.

▼ **pwrout — Power**  
scalar value in the range (0,1) | array of scalar values in the range (0,1)

Power achieved by the test, returned as a scalar value in the range (0,1) or as an array of scalar values in the range (0,1).

If you specify `n` or `p1` as an array, then `sampsizepwr` returns an array for `pwrout` that is the same length as `n` or `p1`.

**p1out — Parameter value for the alternative hypothesis**

scalar value | array of scalar values

Parameter value for the alternative hypothesis, returned as a scalar value or as an array of scalar values.

When computing p1out for the 'p' test, if no alternative can be rejected for a given null hypothesis and significance level, the function displays a warning message and returns NaN.

---

**See Also**

[binocdf](#) | [ttest](#) | [ttest2](#) | [vartest](#) | [ztest](#)

---

**Introduced in R2006b**

---