# Independent- and paired-samples permutation tests

Version 1.1.0 of 29 July 2019

The original R scripts for independent- and paired-samples permutation tests (v. 1.0.0) were intended as supplemental information to our research article on the persistence of yellow fever vaccinal immunity. The main reason for this update was initially to tidy up the R code and hopefully make it easier to parse and safer to use by others. Along the way it became something more complex altogether, but methods are provided that replicate the computation of the p-value estimator as in the earlier version. Please note that the default algorithm for random number generation was recently changed in R 3.6.0, so that the p-value output by the `permtest()` function may change between versions of R; see the Examples section for a workaround.

**Main changes in this version**

- Improved the comments and made good the garbled indentations in the code.
- A single function takes care of the independent- and the paired-samples test.
- New method "`ratio`" (default) for the independent-samples test, based on an alternative statistic.
- The function now checks the arguments for type and data integrity.
- The plot of the permutation distribution is now arguably prettier.
- Streamlined the arguments of the `replicate()` function to shave off CPU time.

*jose-enrique.mejia /at/ inserm.fr*

| | |
|---|---|
| `permtest` | a function for two-tailed permutation tests |

**Description**

Function `permtest()` performs Monte Carlo permutation tests between two independent or paired sets of antibody concentrations or titres, and outputs an estimate of the p-value. The script depends on base R only, and was written and tested under R version 3.6.1.

**Usage**

```
permtest(x, y, perms = 99999, seed = 123, method = c("ratio", "diff", "paired"))
```

**Arguments**

| | |
|---|---|
| `x, y` | Numeric or integer vectors. The elements of `x` and `y` must be strictly > 0, and comprise neither missing (NA) nor infinite values. With `method = "paired"`, `x` and `y` must be of the same length. |
| `perms` | A single integer or numeric value, interpreted as an integer, denoting the number of Monte Carlo permutations to perform; the default is 99999. For a more precise, but slower, computation of the p-value, use 999999 permutations; for an instantaneous but comparatively less precise computation, use 9999 or 49999 permutations. |
| `seed` | A single value, interpreted as an integer, denoting the seed of the random number generator; the default is 123. |
| `method` | The type of test to be performed, one of "`ratio`", "`diff`" or "`paired`"; the default is "`ratio`". See the Details section. |

**Details**

- With `method = "diff"`, a permutation test on the difference of geometric means between two independent sets of antibody concentrations or titres is performed, equivalent to the independent-samples test in v. 1.0.0 despite differences of detail in the code. The test statistic is $S = \bar{y}_{geom.} - \bar{x}_{geom.}$ and the null hypothesis is that the true difference in geometric means is equal to 0.

- `method = "paired"` performs the same permutation-based sign test for paired samples as in v. 1.0.0. The test statistic is the sum of randomly-signed log(y) - log(x) pairwise differences, and the null hypothesis is that the median of the paired differences is equal to 0.

- `method = "ratio"` performs a permutation test on the log-ratio of geometric means between two independent samples. The test statistic is $S = \log(\bar{y}_{geom.} / \bar{x}_{geom.}) = \text{mean}(\log(y)) - \text{mean}(\log(x))$, and the null hypothesis is that the true ratio of geometric means is equal to 1. This method is probably more universally applicable than "`diff`" because, in some cases, it noticeably reduces the skew of the permutation distribution of the test statistic, which can lead to overly conservative p-values in the two-tailed test. In addition, CPU time is cut by half.

The Monte Carlo resampling approach assumes that the sizes n and m of sets 'x' and 'y' are so large as to make it impractical to compute the test statistic for the complete enumeration of possible permutations, i.e. $P = (n + m)! / (n! \times m!)$ for two independent samples, or $P= 2^n$ in the paired test. P exceeds 99999, the default number of permutations, for all n = m > 10 (independent samples) or n > 16 (paired samples). Because it relies on logarithmic transformation of the data, the test is applicable to strictly positive 'x' and 'y' values. For more generally applicable permutation procedures, including complete enumeration, see for example Michael Fay's 'perm' package on CRAN.

### Examples

```
# ~ Load the example dataset
load("pseudodata.RData")
class(pseudodata)
# [1] "list"
summary(pseudodata)
#    Length Class  Mode
# X1 43     -none- numeric
# Y1 47     -none- numeric
# P1 43     -none- numeric
# P2 43     -none- numeric

# ~ Older versions of R: preferably set the random number generator to the new default values in R 3.6.0,
#   https://bugs.r-project.org/bugzilla/show_bug.cgi?id=17494
RNGkind(
    kind = "Mersenne-Twister",
    normal.kind = "Inversion",
    sample.kind = "Rejection"
)
```

Independent-samples tests

```
x <- pseudodata$X1
y <- pseudodata$Y1

# With method = “ratio”
permtest(x, y, method = "ratio")
# Permutation test for independent samples based on 99999 Monte Carlo simulations
# Alternative hypothesis: true ratio of geometric means is not equal to 1
# Geom. mean of x Geom. mean of y
#        1.243507        1.959416
#
# Two-tailed p-value = 0.04554

# Instantaneous but comparatively less precise estimation of the p-value using fewer permutations.
permtest(x, y, 9999, method = "ratio")

# Slower but more precise estimation by increasing the number of permutations.
permtest(x, y, 999999, method = "ratio")
# Permutation test for independent samples based on 999999 Monte Carlo simulations
# Alternative hypothesis: true ratio of geometric means is not equal to 1
# Geom. mean of x Geom. mean of y
#        1.243507        1.959416
#
# Two-tailed p-value = 0.045368

# Now with a different seed for the random number generator.
permtest(x, y, 999999, seed = 7654, method = "ratio")
# Permutation test for independent samples based on 999999 Monte Carlo simulations
# Alternative hypothesis: true ratio of geometric means is not equal to 1
# Geom. mean of x Geom. mean of y
#        1.243507        1.959416
#
# Two-tailed p-value = 0.044923

# As above but with method = “diff” for comparison.
permtest(x, y, 999999, seed = 7654, method = "diff")
# Permutation test for independent samples based on 999999 Monte Carlo simulations
# Alternative hypothesis: true difference of geometric means is not equal to 0
# Geom. mean of x Geom. mean of y
#        1.243507        1.959416

Two-tailed p-value = 0.04748
```

Paired-samples test

```
x <- pseudodata$P1
y <- pseudodata$P2

permtest(x, y, method = "paired")
```

```
# Paired-samples sign test based on 99999 Monte Carlo simulations
# Alternative hypothesis: median of the paired differences is not equal to 0
# Geom. mean of x Geom. mean of y
#        1.978613        1.243507
#
# Two-tailed p-value = 0.03059

# Instantaneous but comparatively less precise estimation of the p-value using fewer permutations.
permtest(x, y, 9999, method = "paired")

# Slower but more precise estimation by increasing the number of permutations.
permtest(x, y, 999999, method = "paired")
# Paired-samples sign test based on 999999 Monte Carlo simulations
# Alternative hypothesis: median of the paired differences is not equal to 0
# Geom. mean of x Geom. mean of y
#        1.978613        1.243507
#
# Two-tailed p-value = 0.030194

# Now with a different seed for the random number generator.
permtest(x, y, 999999, seed = 7654, method = "paired")
# Alternative hypothesis: median of the paired differences is not equal to 0
# Geom. mean of x Geom. mean of y
#        1.978613        1.243507
#
# Two-tailed p-value = 0.030825
```