Package ‘peramo’

May 31, 2023

Type Package
Title Permutation Tests for Randomization Model
Version 0.1.3
Depends R (>= 4.3.0)
Imports magrittr (>= 2.0.3), dplyr (>= 1.1.2), lme4 (>= 1.1-33), parameters (>= 0.21.0), emmeans (>= 1.8.6), stats, utils
Suggests multcompView (>= 0.1-9)
BugReports https://github.com/phamdn/peramo/issues
License GPL-3
Encoding UTF-8
RoxygenNote 7.2.3
LazyData true
NeedsCompilation no
Author Duy Nghia Pham [aut, cre] (<https://orcid.org/0000-0003-1349-1710>), Inna M. Sokolova [ths] (<https://orcid.org/0000-0002-2068-4302>)
Maintainer Duy Nghia Pham <nghiapham@yandex.com>
Repository CRAN
Date/Publication 2023-05-31 16:20:02 UTC

R topics documented:

peramo-package .................................................. 2
AB ................................................................. 3
ctm ............................................................... 4
diffcalc ........................................................... 4
ensulizole .................................................... 5
peramo-package

nolesser .................................................... 5
owl .......................................................... 6
owlStat ...................................................... 7
twl ........................................................... 8
twlStat ....................................................... 9
tw_complex ................................................ 10
XY ............................................................ 12

Index 13

peramo-package  peramo: Permutation Tests for Randomization Model

Description


Copyright

peramo: Permutation Tests for Randomization Model.
Copyright (C) 2022-2023 Duy Nghia Pham & Inna M. Sokolova

peramo is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

peramo is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with peramo. If not, see https://www.gnu.org/licenses/.

Author(s)

Duy Nghia Pham & Inna M. Sokolova
**Two-Group Permutation Test**

**Description**

AB performs A/B testing for two-group experiments.

**Usage**

```
AB(a, b, rand = 9999, seed = 1)
```

**Arguments**

- **a**: the measurement of responses of the first group.
- **b**: the measurement of responses of the second group.
- **rand**: an integer, the number of randomization samples. The default value is 9999.
- **seed**: an integer, the seed for random number generation. Setting a seed ensures the reproducibility of the result. See `set.seed` for more details.

**Value**

AB returns an one-row data frame with 6 columns:

- **nA**: the sample size of the first group.
- **mean.A**: the mean responses of the first group.
- **nB**: the sample size of the second group.
- **mean.B**: the mean responses of the second group.
- **mean.dif**: the difference between two mean responses.
- **pval**: the p-value.

**References**


**Examples**

```
AB(c(19, 22, 25, 26), c(23, 33, 40))
```
ctm

Biomarker Responses of the Ragworms to Copper and Warming

Description

Biomarker Responses of the Ragworms to Copper and Warming

Usage

ctm_Cu

ctm_worm

Format

An object of class data.frame with 60 rows and 7 columns.
An object of class data.frame with 210 rows and 26 columns.

diffcalc

Calculate the Differences between Means

Description

Calculate the Differences between Means

Usage

diffcalc(vec, control)

Arguments

vec a numeric vector, the mean responses.
control a logical, whether the control group exists.
ensulizole

---

**Biomarker Responses of the Blue Mussels to Organic UV Filters**

### Description

Biomarker Responses of the Blue Mussels to Organic UV Filters

### Usage

- `mussel_SoS`
- `mussel_gill`
- `mussel_digest`

### Format

- An object of class `data.frame` with 15 rows and 7 columns.
- An object of class `data.frame` with 120 rows and 24 columns.
- An object of class `data.frame` with 120 rows and 24 columns.

### Source


---

**nolesser**

---

**Compare the Differences with Critical Values**

### Description

Compare the Differences with Critical Values

### Usage

```r
nolesser(obs, cric)
```

### Arguments

- `obs` a numeric, the observed difference.
- `cric` a numeric, the critical values of maximum absolute differences.
**One-Way Layout Permutation Test**

**Description**

owl performs the global test and multiple comparisons for single factor experiments.

**Usage**

```r
owl(df, rand = 9999, alpha.post = 0.05, type.post = "control", seed = 1)
```

**Arguments**

- `df`: a data frame with the name of experimental groups as the first column and the measurement of responses as the remaining columns.
- `rand`: an integer, the number of randomization samples. The default value is 9999.
- `alpha.post`: a numeric, the Type I error rate for multiple comparisons. The default value is 0.05.
- `type.post`: the way of multiple comparisons, "all" for pairwise comparisons or "control" for only comparisons with the control group.
- `seed`: an integer, the seed for random number generation. Setting a seed ensures the reproducibility of the result. See `set.seed` for more details.

**Details**

The first name appearing in the first column will determine the control group. The other names will be treatment groups.

**Value**

owl returns a list with 9 components:

- `n.obs`: the sample sizes.
- `avg.obs`: the mean responses.
- `T.obs`: the T statistic for global test.
- `pval`: the p-value for global test.
- `pval.round`: the reported form of p-value.
- `main.test`: the strength of evidence against the null hypothesis.
- `d.multi.obs`: the differences in means for multiple comparisons.
- `mad.cric`: the critical value of maximum absolute differences in means.
- `post.test`: TRUE if the differences are significant.
References


Examples

ernst2004 <- data.frame(group = factor(rep(c("style1", "style2", "style3"), each = 5 ), levels = c("style1", "style2", "style3")), speed = c(135, 91, 111, 87, 122, 175, 130, 514, 283, NA, 105, 147, 159, 107, 194))
owl(ernst2004, type.post = "all")

owlStat

Test Statistics for One-Way Layout Permutation Test

Description

owlStat computes statistics for owl. This is not meant to be called directly.

Usage

owlStat(lov, env = parent.frame())

Arguments

lov a list of vectors, responses by experimental groups.
env an environment, to access outer scope variables.

Value

owlStat returns a list with 5 components:

n the sample sizes.
avg the mean responses.
T the T statistic for global test.
d.multi the differences in means for multiple comparisons.
mad the maximum absolute differences in means.

References

Two-Way Layout Permutation Test

dtwl performs the global test and multiple comparisons for two-factor experiments.

Usage

dtwl(
  df,
  rand = 4999,
  seed = 1,
  mult = FALSE,
  simple = TRUE,
  control = TRUE,
  alpha = 0.05
)

Arguments

df a data frame with the first and second columns containing the levels of the two main factors and the third column containing the measurement of responses.
rand an integer, the number of randomization samples. The default value is 4999.
seed an integer, the seed for random number generation. Setting a seed ensures the reproducibility of the result. See set.seed for more details.
mult a logical, whether to perform multiple comparisons.
simple a logical, whether to perform comparisons for simple effects.
control a logical, whether to perform only comparisons with the control group.
alphaa numeric, the Type I error rate for multiple comparisons. The default value is 0.05.

Details

The first levels appearing in the first and second columns will determine the control groups (if any). The other levels will be treatment groups.

Value

dtwl returns a list with possible components:
n, n.main1, and n.main2
  the sample sizes.
avg, avg.main1, and avg.main2
  the mean responses.
Fs the F statistics, p-values, reported form of p-value, and strength of evidence against the null hypotheses.

d.main1sub and d.main2sub or d.main1 and d.main2 the differences in means for multiple comparisons.

mad.main1sub.cric and mad.main2sub.cric or mad.main1.cric and mad.main2.cric the critical value of maximum absolute differences in means.

mult.test.main1sub and mult.test.main2sub or mult.test.main1 and mult.test.main2 TRUE if the differences are significant.

References


Examples

manly2007 <- data.frame(
  month = factor(rep(c("jun", "jul", "aug", "sep"), each = 6 ),
  levels = c("jun", "jul", "aug", "sep")),
  size = factor(rep(c("small", "large"), each = 3, times = 4),
  levels = c("small", "large")),
  consume = c( 13,242,185,182,21,7,8,59,20,24,312,68,515,488,88,460,1223,990,18,44,21,140,40,27))
twl(manly2007)
twl(manly2007, mult = TRUE, simple = TRUE, control = FALSE)
#might take more than 5s in some machines

twlStat

Test Statistics for Two-Way Layout Permutation Test

Description

twlStat computes statistics for twl. This is not meant to be called directly.

Usage

twlStat(df, env = parent.frame())
Arguments

df a data frame with the levels of the two main factors as the first and second columns and the measurement of responses as the third column.

env an environment, to access outer scope variables.

Value

twlStat returns a list with at least 4 components:

Fs the F statistics for global test.

F.main1 and F.main2 the F statistics for the first main factor and the second main factor.

F.int the F statistic for the interaction.

In case of multiple comparisons, additional components are:

avg or avg.main1 and avg.main2 the mean responses for multiple comparisons.

d.main1sub and d.main2sub or d.main1 and d.main2 the differences in means.

mad.main1sub and mad.main2sub or mad.main1 and mad.main2 the maximum absolute differences in means.

References


tw_complex  

*Permutation Test for Two-Way Layout with Extra Factors*

Description

tw_complex performs the permutation test for ANOVA of two-factor experiments with complex design.

Usage

tw_complex(df, res, mains, nested, nuis, seed = 1, rand = 1999, emm = FALSE)
Arguments

- **df**: a data frame with at least three columns.
- **res**: a character string, name of response variable.
- **mains**: two character strings, names of two main factors.
- **nested**: (optional) a character string, name of the nested factor.
- **nuis**: (optional) a character string, name of the nuisance factor.
- **seed**: an integer, the seed for random number generation. Setting a seed ensures the reproducibility of the result. See `set.seed` for more details.
- **rand**: an integer, the number of randomization samples. The default value is 1999.
- **emmm**: a logical, whether to compute estimated marginal means.

Details

- **res**, **mains**, **nested**, and **nuis** refer to column names in **df**. While **nuis** column must be a numeric vector, **mains** and **nested** columns must be factors. **res** can be a numeric or logical vector.

**tw_complex** currently support linear models with only **mains**, generalized linear mixed-effects models with **mains** and **nested**, and linear mixed-effects models with **mains**, **nested**, and **nuis**.

Value

**tw_complex** returns a list with 3 main components:

- **lm, glmer, or lmer**: model results.
- **anova**: anova table.
- **perm**: permutation test results with F-statistics, p-values, and strength of evidence.

References


See Also

- `lm`, `glmer`, and `lmer`. 
Examples

tw_complex(df = subset(ctm_Cu, run == "Jan", select = c("copper", "temp", "sediment")), res = "sediment", mains = c("copper", "temp"))

# might take more than 5s in some machines

---

Permutation Test for Correlation Between Paired Samples

Description

XY performs permutation test on correlation coefficients.

Usage

XY(a, b, rand = 9999, seed = 1, method = c("pearson", "kendall", "spearman"))

Arguments

<table>
<thead>
<tr>
<th>a</th>
<th>a numeric vector, the first variable.</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>a numeric vector, the second variable.</td>
</tr>
<tr>
<td>rand</td>
<td>an integer, the number of randomization samples. The default value is 9999.</td>
</tr>
<tr>
<td>seed</td>
<td>an integer, the seed for random number generation. Setting a seed ensures the reproducibility of the result. See set.seed for more details.</td>
</tr>
<tr>
<td>method</td>
<td>correlation coefficient, &quot;pearson&quot;, &quot;kendall&quot;, or &quot;spearman&quot;.</td>
</tr>
</tbody>
</table>

Value

XY returns an one-row data frame with 2 columns:

<table>
<thead>
<tr>
<th>cor</th>
<th>the correlation coefficient.</th>
</tr>
</thead>
<tbody>
<tr>
<td>pval</td>
<td>the p-value.</td>
</tr>
</tbody>
</table>

See Also

cor.test

Examples

with(subset(ctm_Cu, run == "Jan"), XY(sediment, porewater))
Index

* datasets
  ctm, 4
  ensulizole, 5

AB, 3

cor.test, 12
ctm, 4
ctm_Cu(ctm), 4
ctm_worm(ctm), 4
diffcalc, 4
ensulizole, 5
glmer, 11
lm, 11
lmer, 11

mussel_digest (ensulizole), 5
mussel_gill (ensulizole), 5
mussel_SoS (ensulizole), 5

nolesser, 5

owl, 6
owlStat, 7

peramo-package, 2

set.seed, 3, 6, 8, 11, 12
tw_complex, 10
twl, 8
twlStat, 9

XY, 12