Package ‘TInPosition’

January 31, 2019

Type Package
Title Inference Tests for TExPosition
Version 0.13.6.1
Date 2013-12-09
Author Derek Beaton, Jenny Rieck, Herve Abdi
Maintainer Derek Beaton <exposition.software@gmail.com>
Description Non-parametric resampling-based inference tests for TExPosition.
License GPL-2
Depends prettyGraphs (>= 2.1.4), TExPosition (>= 2.6.10), ExPosition
   (>= 2.8.19), InPosition (>= 0.12.7)
BugReports http://code.google.com/p/exposition-family/issues/list
NeedsCompilation no
Repository CRAN
Date/Publication 2019-01-31 17:27:50 UTC

R topics documented:

  TInPosition-package .................................................. 2
  boot.compute.fi.fj ................................................... 3
  print.tepBADA.inference.battery .................................... 4
  print.tepDICA.inference.battery .................................... 4
  print.tinpoAllBoots ................................................ 5
  print.tinpoBoot .................................................... 5
  print.tinpoBootTests ............................................... 6
  print.tinpoComponents ............................................. 6
  print.tinpoLOO .................................................... 7
  print.tinpoOmni ................................................... 7
  print.tinpoOutput ................................................. 8
  print.tinpoR2 ..................................................... 8
  tepBADA.inference.battery ........................................ 9
  tepDICA.inference.battery ....................................... 10
  tinGraphs .......................................................... 12
**Description**

TInPosition provides multiple forms of inference tests for the TExPosition package.

**Details**

- **Package:** TInPosition
- **Type:** Package
- **Version:** 0.13.6
- **Date:** 2013-12-09
- **Depends:** R (>=2.15.0), prettyGraphs (>= 2.1.4), TExPosition (>= 2.6.10), ExPosition (>= 2.8.19), InPosition (>= 0.12.7)
- **License:** GPL-2
- **URL:** http://www.utdallas.edu/~derekbeaton/software/ExPosition

**Author(s)**

Questions, comments, compliments, and complaints go to Derek Beaton <exposition.software@gmail.com>. Also see the bug-tracking and live update website for ExPosition: http://code.google.com/p/exposition-family/

Primary authors and contributors are: Derek Beaton, Jenny Rieck, and Hervé Abdi

**References**

Permutation:

Bootstrap:

Two-table specific cases:

See Also

* tepBADA.inference.battery, tepDICA.inference.battery

Examples

# For more examples, see each individual function (as noted above).

```r
boot.compute.fi.fj

Description

Provides bootstrap projections for $f_i$ and $f_j$ from TExPosition methods.

Usage

`boot.compute.fi.fj(DATA, DESIGN, res)`

Arguments

- **DATA**: The original data matrix to be bootstrapped. Rows will be bootstrapped and are assumed to be observations. Resampling will be constrained to within groups based on DESIGN.
- **DESIGN**: A design matrix (in disjunctive coding). Required for TExPosition and TInPosition analyses.
- **res**: of class tepxOutput. Results from one of the TExPosition methods (e.g., tepDICA, tepBADA).

Value

- **FBX**: a set of factor scores of the measures (columns, $f_j$) for the bootstrapped data.
- **FBY**: a set of factor scores of the groups ($f_i$) for the bootstrapped data.

Author(s)

Derek Beaton
print.tepBADA.inference.battery

Print tepBADA inference results

Description
Print tepBADA inference results.

Usage
```r
## S3 method for class 'tepBADA.inference.battery'
print(x,...)
```

Arguments
- `x`: an list that contains items to make into the tepBADA.inference.battery class.
- `...`: inherited/passed arguments for S3 print method(s).

Author(s)
Derek Beaton, Cherise Chin-Fatt

print.tepDICA.inference.battery

Print tepDICA.inference.battery results

Description
Print tepDICA Inference results.

Usage
```r
## S3 method for class 'tepDICA.inference.battery'
print(x,...)
```

Arguments
- `x`: an list that contains items to make into the tepDICA.inference.battery class.
- `...`: inherited/passed arguments for S3 print method(s).

Author(s)
Derek Beaton, Cherise Chin-Fatt
Description

Print bootstrap results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoAllBoots'
print(x, ...)
```

Arguments

- `x` an list that contains items to make into the tinpoAllBoots class.
- `...` inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt

Description

Print bootstrap results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoBoot'
print(x, ...)
```

Arguments

- `x` an list that contains items to make into the tinpoBoot class.
- `...` inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt
Description

Print bootstrap ratio tests results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoBootTests'
print(x,...)
```

Arguments

- `x` an list that contains items to make into the tinpoBootTests class.
- `...` inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt

---

Description

Print Components permutation test results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoComponents'
print(x,...)
```

Arguments

- `x` an list that contains items to make into the tinpoComponents class.
- `...` inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt
print.tinpoLOO

Print results from TInPosition LOO

Description

Print LOO results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoLOO'
print(x, ...)
```

Arguments

- `x`: an list that contains items to make into the tinpoLOO class.
- `...`: inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt

print.tinpoOmni

Print results from TInPosition Omnibus Permutation Test

Description

Print Omnibus permutation test results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoOmni'
print(x, ...)
```

Arguments

- `x`: an list that contains items to make into the tinpoOmni class.
- `...`: inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt
print.tinpoOutput  

Print results from TInPosition

Description

Print results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoOutput'
print(x,...)
```

Arguments

- `x`: an list that contains items to make into the `tinpoOutput` class.
- `...`: inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt

See Also

tepBADA.inference.battery, tinGraphs

print.tinpoR2  

Print results from TInPosition R2 Permutation Test

Description

Print R2 permutation test results from the TInPosition.

Usage

```r
## S3 method for class 'tinpoR2'
print(x,...)
```

Arguments

- `x`: an list that contains items to make into the `tinpoR2` class.
- `...`: inherited/passed arguments for S3 print method(s).

Author(s)

Derek Beaton and Cherise Chin-Fatt
Barycentric Discriminant Analysis Inference Battery

Description

Barycentric Discriminant Analysis (BADA) Inference Battery via TlnPosition.

Usage

```r
tepBADA.inference.battery(DATA, scale = TRUE, center = TRUE, DESIGN = NULL, make_design_nominal = TRUE, group.masses = NULL, weights = NULL, graphs = TRUE, k = NULL, test.iters = 100, critical.value = 2)
```

Arguments

- **DATA**
  - original data to perform a BADA on.
- **scale**
  - a boolean, vector, or string. See `expo.scale` for details.
- **center**
  - a boolean, vector, or string. See `expo.scale` for details.
- **DESIGN**
  - a design matrix to indicate if rows belong to groups. Required for BADA.
- **make_design_nominal**
  - a boolean. If TRUE (default), DESIGN is a vector that indicates groups (and will be dummy-coded). If FALSE, DESIGN is a dummy-coded matrix.
- **group.masses**
  - a diagonal matrix or column-vector of masses for the groups.
- **weights**
  - a diagonal matrix or column-vector of weights for the column items.
- **graphs**
  - a boolean. If TRUE (default), graphs and plots are provided (via `epGraphs`)
- **k**
  - number of components to return.
- **test.iters**
  - number of iterations
- **critical.value**
  - numeric. A value, analogous to a z- or t-score to be used to determine significance (via bootstrap ratio).

Details

**tepBADA.inference.battery** performs barycentric discriminant analysis and inference tests on based on data and (row) design matrices.

If the expected time to compute the results (based on `test.iters`) exceeds 1 minute, you will be asked (via command line) if you want to continue.
Value

Returns two lists ($Fixed.Data and $Inference.Data). For $Fixed.Data, see tepBADA and corePCA for details on the descriptive (fixed-effects) results.

$Inference.Data returns:

- **omni** Permutation tests of components. p-values ($p.val) and distributions of total inertia ($inertia.perm)
- **r2** Permutation tests of R-squared value. p-values ($p.val) and distributions of R2s ($r2.perm)
- **components** Permutation tests of components. p-values ($p.vals) and distributions of eigenvalues ($eigs.perm) for each component
- **boot.data** Bootstrap tests for $fi and $fj. Contains distributions. See also `boot.ratio.test` output details.
- **loo.data** Leave one out cross-validation tests. Includes assignments ($loo.assign), factor scores ($loo.fii), LOO and fixed confusion matrices ($loo.confuse, $fixed.confuse), and accuracy ($loo.acc, $fixed.acc)

Author(s)

Derek Beaton, Jenny Rieck, Hervé Abdi

Examples

data(bada.wine)
data <- bada.wine$data
design <- bada.wine$design
bada.res <- tepBADA.inference.battery(data, scale=FALSE, DESIGN=design,
make_design_nominal=FALSE, test.iters=50)

Description

Discriminant Correspondence Analysis Inference Battery via TInPosition

Usage

tepDICA.inference.battery(DATA, make_data_nominal = FALSE, DESIGN = NULL,
make_design_nominal = TRUE,
group.masses = NULL, weights = NULL,
symmetric = TRUE, graphs = TRUE, k = 0,
test.iters = 100, critical.value = 2)
Arguments

DATA  original data to perform a DICA on. Data can be contingency (like CA) or
categorical (like MCA).
make.data_nominal  a boolean. If TRUE (default), DATA is recoded as a dummy-coded matrix. If
FALSE, DATA is a dummy-coded matrix.
DESIGN  a design matrix to indicate if rows belong to groups. Required for DICA.
make.design_nominal  a boolean. If TRUE (default), DESIGN is a vector that indicates groups (and
will be dummy-coded). If FALSE, DESIGN is a dummy-coded matrix.
group.masses  a diagonal matrix or column-vector of masses for the groups.
weights  a diagonal matrix or column-vector of weights for the column it
symmetric  a boolean. If TRUE (default) symmetric factor scores for rows.
graphs  a boolean. If TRUE (default), graphs and plots are provided (via epGraphs)
k  number of components to return.
test.iters  number of iterations
critical.value  numeric. A value, analogous to a z- or t-score to be used to determine signifi-
cance (via bootstrap ratio).

Details

tepDICA.inference.battery performs discriminant correspondence analysis and inference tests
on based on data and (row) design matrices.

If the expected time to compute the results (based on test.iters) exceeds 1 minute, you will
be asked (via command line) if you want to continue.

Value

Returns two lists ($Fixed.Data and $Inference.Data). For $Fixed.Data, see tepDICA and coreCA for
details on the descriptive (fixed-effects) results.
$Inference.Data returns:

omni  Permutation tests of components. p-values ($p.val) and distributions of total
inertia ($Inertia.perm)
r2  Permutation tests of R-squared value. p-values ($p.val) and distributions of R2s
($r2.perm)
components  Permutation tests of components. p-values ($p.vals) and distributions of eigen-
values ($eigs.perm) for each component
boot.data  Bootstrap tests for $fi and $fj. Contains distributions. See also boot.ratio.test
output details.
loo.data  Leave one out cross-validation tests. Includes assignments ($loo.assign), factor
scores ($loo.fii), LOO and fixed confusion matrices ($loo.confuse, $fixed.confuse),
and accuracy ($loo.acc, $fixed.acc)
Author(s)
Derek Beaton, Jenny Rieck, Hervé Abdi

Examples

data(dica.wine)
data<-dica.wine$data
design<-dica.wine$design
dica.res <-

tepDICA.inference.battery(data,DESIGN=design,
make_design_nominal=FALSE,test.ites=50)

Description
TInPosition plotting function which is an interface to prettyGraphs.

Usage
tinGraphs(res, DESIGN = NULL, x_axis = NULL, y_axis = NULL, inference.info = NULL,
color.by.boots = TRUE, boot.cols = c("plum4", "darkseagreen", "firebrick3"),
fi.col = NULL, fi.pch = NULL, fii.col = NULL, fii.pch = NULL,
fj.col = NULL, fj.pch = NULL, col.offset = NULL,
constraints = NULL, xlab = NULL, ylab = NULL, main = NULL,
bootstrapbars = TRUE, correlationPlotter = TRUE,
showHulls = 0.95, biplots = FALSE)

Arguments
res results from TExPosition
DESIGN A design matrix to apply colors (by pallete selection) to row items
x_axis which component should be on the x axis?
y_axis which component should be on the y axis?
inference.info Inference data as output by TInPosition (of class tinpoOutput).
color.by.boots a boolean. If TRUE, items are colored by bootstrap ratio test. Items larger than
critical.value are colored 'plum4' on the horizontal component, 'darkseagreen' on the vertical component, or 'firebrick3' if the item is significant on both components (to be visualized). If FALSE, the color of the items will be used.
boot.cols vector of colors: c(horizontal component color, vertical component color, color when item
fi.col A matrix of colors for the group items. If NULL, colors will be selected.
fii.pch A matrix of pch values for the group items. If NULL, pch values are all 21.
**fii.col** A matrix of colors for the row items (observations). If NULL, colors will be selected.

**fii.pch** A matrix of pch values for the row items (observations). If NULL, pch values are all 21.

**fj.col** A matrix of colors for the column items. If NULL, colors will be selected.

**fj.pch** A matrix of pch values for the column items. If NULL, pch values are all 21.

**col.offset** A numeric offset value. Is passed to `createColorVectorsByDesign`.

**constraints** Plot constraints as returned from `prettyPlot`. If NULL, constraints are selected.

**xlab** x axis label

**ylab** y axis label

**main** main label for the graph window

**bootstrapBars** a boolean. If TRUE (default), bootstrap ratio bar plots will be created.

**correlationPlotter** a boolean. If TRUE (default), a correlation circle plot will be created. Applies to PCA family of methods (CA is excluded for now).

**showHulls** a value between 0 and 1 to make a peeled hull at that percentage. All values outside of 0-1 will not plot any hulls.

**biplots** a boolean. If FALSE (default), separate plots are made for row items ($fii$ and $fi$) and column items ($fj$). If TRUE, row ($fii$ and $fi$) and column ($fj$) items will be on the same plot.

**Details**

tinGraphs is an interface between TInPosition and prettyGraphs.

**Author(s)**

Derek Beaton

**See Also**

`prettyGraphs`, `inGraphs`

**Examples**

```r
#this is for TExPosition's iris data
data(ep.iris)
data <- ep.iris$data
design <- ep.iris$design
bada.iris <-
tepBADA.inference.battery(
data, DESIGN = design,
make_design_nominal = FALSE, test.ites = 50)
tinGraphs(bada.iris, x_axis = 1, y_axis = 2, biplots = TRUE)
```
Index

*Topic bootstrap
  boot.compute.fi.fj, 3
*Topic graphs
  tinGraphs, 12
*Topic misc
  tinGraphs, 12
*Topic multivariate
  tepBADA.inference.battery, 9
tepDICA.inference.battery, 10
tinGraphs, 12
TInPosition-package, 2
*Topic package
  TInPosition-package, 2
*Topic print
  print.tepBADA.inference.battery, 4
  print.tepDICA.inference.battery, 4
  print.tinpoAllBoots, 5
  print.tinpoBoot, 5
  print.tinpoBootTests, 6
  print.tinpoComponents, 6
  print.tinpoLOO, 7
  print.tinpoOmnii, 7
  print.tinpoOutput, 8
  print.tinpoR2, 8

boot.compute.fi.fj, 3
boot.ratio.test, 10, 11

coreCA, 11
corePCA, 10
createColorVectorsByDesign, 13

epGraphs, 9, 11
expo.scale, 9

inGraphs, 13

prettyGraphs, 12, 13
prettyPlot, 13
print.tepBADA.inference.battery, 4
print.tepDICA.inference.battery, 4
print.tinpoAllBoots, 5
print.tinpoBoot, 5
print.tinpoBootTests, 6
print.tinpoComponents, 6
print.tinpoLOO, 7
print.tinpoOmnii, 7
print.tinpoOutput, 8
print.tinpoR2, 8
tepBADA, 3, 10
tepBADA.inference.battery, 3, 8, 9
tepDICA, 3, 11
tepDICA.inference.battery, 3, 10
TExPosition, 2, 3
tinGraphs, 8, 12
TInPosition, 13
TInPosition (TInPosition-package), 2
TInPosition-package, 2