Package 'SemNetCleaner'

May 16, 2018

Title An Automated Cleaning Tool for Semantic and Linguistic Data

Version 0.1.1

Date 2018-05-11

Author Alexander Christensen

Maintainer Alexander Christensen <alexpaulchristensen@gmail.com>

Description Implements several functions that automatize the cleaning, removal of plurals and continuous strings, making the data binary, converging, and finalizing of linguistic data for semantic network analysis. Also provides a partial bootstrapped network function and plot.

Depends R (>= 3.3.0)

License GPL (>= 3.0)

Encoding UTF-8

LazyData true

Imports qdap, tm, foreach, parallel, doParallel, NetworkToolbox, ggplot2, dplyr, plyr, RColorBrewer, purrr, magrittr, grid

RoxygenNote 6.0.1.9000

NeedsCompilation no

Repository CRAN

Date/Publication 2018-05-16 15:15:15 UTC

R topics documented:

---

autoConverge *Autmated Converge Responses*

Description

Automated converge function merging of columns of binarized response data with another

Usage

```
autoConverge(rmat)
```

Arguments

rmat   A semnetcleaner filtered response matrix

Value

The response matrix with the **word** column merged and the **replace** column removed for all variables

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
convmat <- autoConverge(rmat)
```

---

autoDeStr   *Automated De-string Responses*

Description

Automated de-string responses after performing semnetcleaner

Usage

```
autoDeStr(rmat, char = 10)
```

Arguments

rmat	A semnetcleaner filtered response matrix
char	Minimum number of characters in a string to be checked for destr. Defaults to 10

Value

A question asking whether the response should be de-string-ed. If yes, destr will be applied. If no, the next response will be checked

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#create example stringed responses
stringed <- cbind(rowSums(cbind(rmat[,c(1,2)])),convmat)
#change name to stringed name
colnames(stringed)[1] <- "alligator.ant"


#automated de-string
convmat <- destr(stringed, 10)
```

---

converge	*Converge Responses*

Description

Merge a column of binarized response data with another

Usage

```
converge(rmat, word, replace)
```

Arguments

rmat	A semnetcleaner filtered response matrix
word	The column name that will incoporate the **replace** column's binarized responses (must be characters)
replace	The column name that should be merged with the **word** column (must be characters)

Value

The response matrix with the **word** column merged and the **replace** column removed

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#converge "kitten" into response of "cat"
rmat <- converge(rmat,"cat","kitten")
```

---

convert2cytoscape	*Convert Adjacency Matrix to Cytoscape Format*

Description

Converts an adjacency matrix to Cytoscape's sparse matrix format

Usage

```
convert2cytoscape(A)
```

Arguments

A	A cleaned, finalized response matrix ready to be visualized

Value

A sparse matrix formatted for Cytoscape

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#finalize rmatA
finalCmat <- finalize(convmat)
#finalize rmatB
finalRmat <- finalize(rmat)

#equate rmatA and rmatB
eq1 <- equate(finalCmat,finalRmat)

#obtain respective equated response matrices
eqCmat <- eq1$rmatA
eqRmat <- eq1$rmatB

#compute similarity matrix
cosC <- cosine(eqCmat)
cosR <- cosine(eqRmat)

#compute networks using NetworkToolbox
Cnet <- NetworkToolbox::TMFG(cosC)$A
Rnet <- NetworkToolbox::TMFG(cosR)$A

#covert to Cytoscape format
ctyoC <- convert2cytoscape(Cnet)
ctyoR <- convert2cytoscape(Rnet)
```

---

convmat	*Converged trial data of verbal fluency responses*

Description

Converged trial data of animal verbal fluency responses. The columns are responses and the rows are the participants.

Usage

```
data(convmat)
```

Format

A 15x95 response matrix

Examples

```
data(convmat)
```

---

cosine	*Cosine Similarity*

Description

Computes cosine similarity

Usage

```
cosine(data, addConstant = 0)
```

Arguments

data	A binarized dataset of verbal fluency or linguistic data
addConstant	A constant to be added to the cosine similarity matrix

Value

A cosine similarity matrix

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

---

destr	*De-string Responses*

Description

De-string responses after performing semnetcleaner

Usage

```
destr(rmat, rm.str)
```

Arguments

rmat	A semnetcleaner filtered response matrix
rm.str	The column number of the stringed response

Value

The response matrix with the string column merged into appropriate response columns and the string response removed

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#create example stringed responses
stringed <- cbind(rowSums(cbind(rmat[,c(1,2)])),convmat)
#change name to stringed name
colnames(stringed)[1] <- "alligator.ant"

#de-string
convmat <- destr(stringed, 1)
```

---

equate *Equate Group Responses*

Description

An automated cleaning function for matching groups' responses

Usage

```
equate(rmatA, rmatB)
```

Arguments

rmatA	A semnetcleaner filtered response matrix for group 1
rmatB	A semnetcleaner filtered response matrix for group 2

Value

A list of responses matched for group 1 (rmatA) and group 2 (rmatB)

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#finalize rmatA
finalCmat <- finalize(convmat)
#finalize rmatB
finalRmat <- finalize(rmat)

#equate rmatA and rmatB
eq1 <- equate(finalCmat,finalRmat)

#obtain respective equated response matrices
eqCmat <- eq1$rmatA
eqRmat <- eq1$rmatB
```

---

finalize   *Finalize Response Matrix*

Description

Finalizes the response matrix by keeping responses that are given by two or more people

Usage

```
finalize(rmat)
```

Arguments

rmat   A semnetcleaner filtered response matrix

Value

A matrix with responses given by two or more people

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
convmat <- autoConverge(rmat)
```

```
finalRmat <- finalize(convmat)
```

---

org.plot   *Organization function for partboot.plot*

Description

A function used in partboot. Not to be used individually

Usage

```
org.plot(bootData, bootPaired, sampsData, sampsPaired, len, measname, netmeas,
  pall, paired, CI, labels)
```

Arguments

bootData	bootData from partboot
bootPaired	bootPaired from partboot
sampsData	sampsData from partboot
sampsPaired	sampsPaired from partboot
len	Number of samples in data list
measname	Full network measure name
netmeas	Abbreviated network measure name
pall	Color palette to be used from RColorBrewer
paired	Is samples paired?
CI	Confidence interval to be used
labels	Labels to be used in plot.

Value

Returns plots for the specified measures

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
###NOT FOR INDIVIDUAL USE###
```

---

partboot   *Partial Bootstrapped Semantic Network Analysis*

Description

Bootstraps (without replacement) the nodes in the network and computes global network characteristics

Usage

```
partboot(data, paired = NULL, n, iter = 1000, corr = c("cor", "cosine"),
  cores, seeds = NULL)
```

Arguments

data	Cleaned response matrix
paired	Should bootstrapped nodes be paired? Defaults to NULL. Input a matrix, data frame or list containing another sample
n	Number of nodes for bootstrap. Defaults to round(ncol(data)/2,0) (i.g., 50% of nodes)
iter	Number of iterations in bootstrap. Defaults to 1000
corr	Association method to use. Defaults to "cosine"
cores	Number of computer processing cores to use for bootstrapping samples. Defaults to *n* - 1 total number of cores. Set to any number between 1 and maxmimum amount of cores on your computer
seeds	Seeds used in previous run. Defaults to NULL. Input a vector from previous run to replicate analyses

Value

Returns a list that includes the original semantic network measures (origmeas; ASPL, CC, Q, S), the bootstrapped semantic network measures (bootmeas), and Seeds that can be used to replicate analysis

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#finalize rmatA
finalCmat <- finalize(convmat)
#finalize rmatB
finalRmat <- finalize(rmat)

#equate rmatA and rmatB
```

```
eq1 <- equate(finalCmat,finalRmat)

#obtain respective equated response matrices
eqCmat <- eq1$rmatA
eqRmat <- eq1$rmatB


results <- partboot(eqCmat, eqRmat, iter = 10, corr = "cosine", cores = 4)
```

---

partboot.plot   *Plot for partboot*

Description

Bootstraps (without replacement) the nodes in the network and computes global network characteristics

Usage

```
partboot.plot(object, paired = FALSE, CI = 0.975, labels = NULL,
  measures = c("ASPL", "CC", "Q", "S", "MeanStrength"))
```

Arguments

object	An object from [partboot](#)
paired	Is object from a paired [partboot](#)?
CI	Confidence intervals to use for plot. Defaults to .975
labels	Labels to be used in plot. Defaults to NULL. Typed responses will be requested if NULL
measures	Measures to be plotted

Value

Returns plots for the specified measures

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#finalize rmatA
finalCmat <- finalize(convmat)
#finalize rmatB
finalRmat <- finalize(rmat)

#equate rmatA and rmatB
```

```
eq1 <- equate(finalCmat,finalRmat)

#obtain respective equated response matrices
eqCmat <- eq1$rmatA
eqRmat <- eq1$rmatB


results <- partboot(eqCmat, eqRmat, corr = "cosine", cores = 4)

#labels
labs <- c("eqCmat","eqRmat")
partboot.plot(results, paired = TRUE, labels = labs)
```

---

partboot.test   *Test for partboot*

Description

Bootstraps (without replacement) the nodes in the network and computes global network characteristics

Usage

```
partboot.test(object)
```

Arguments

object   An object from [partboot](#)

Value

Returns test statistics for specified measures

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#finalize rmatA
finalCmat <- finalize(convmat)
#finalize rmatB
finalRmat <- finalize(rmat)

#equate rmatA and rmatB
eq1 <- equate(finalCmat,finalRmat)

#obtain respective equated response matrices
eqCmat <- eq1$rmatA
```

```
eqRmat <- eq1$rmatB

results <- partboot(eqCmat, eqRmat, corr = "cosine", cores = 4)

partboot.test(results, paired = TRUE, labels = labs)
```

---

rmat *Cleaned trial data of verbal fluency responses*

Description

Cleaned trial data of animal verbal fluency responses. The columns are responses and the rows are the participants.

Usage

```
data(rmat)
```

Format

A 15x95 response matrix

Examples

```
data(rmat)
```

---

semnetcleaner *Semantic Network Cleaner*

Description

An automated cleaning function for semantic network data

Usage

```
semnetcleaner(data, miss = 99)
```

Arguments

data	A dataset of verbal fluency or linguistic data
miss	Value for missing data. Defaults to 99

Value

A list of a binary matrix of responses (binary; rows = participants, columns = responses) and cleaned response matrix (responses)

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

References

Hornik, K., & Murdoch, D. (2010). Watch Your Spelling!. *The R Journal*, 3(2), 22-28.

Examples

```
rmat<-semnetcleaner(trial)
```

---

semnetmeas   *Semantic Network Measures*

Description

Computes the average shortest path length (ASPL), clustering coefficient(CC), modularity (Q), small-worldness (S; defaults to "rand"), and mean network strength (MNS)

Usage

```
semnetmeas(A, iter, weighted = FALSE, swm = "rand")
```

Arguments

A	An adjacency matrix of network A
iter	Number of iterations for the small-worldness measure
weighted	Should weighted measures be computed? Defaults to FALSE. Set to TRUE for weighted measures
swm	Method for computing small-worldness. Defaults to "rand". See smallworldness for other options

Value

Returns a values for ASPL, CC, Q, and S

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
lowO <- subset(NetworkToolbox::animals, Group==1)[-1]

cosL <- cosine(lowO)

A <- NetworkToolbox::TMFG(cosL)$A

globmeas <- semnetmeas(A)
```

---

trial	*Trial data of verbal fluency responses*

Description

Trial data of animal verbal fluency responses. The columns are participants and the rows are their responses.

Usage

```
data(trial)
```

Format

A 49x15 response matrix

Examples

```
data(trial)
```

Index