Package: cbn (via r-universe)

September 8, 2024

Version 0.3.2

Title Tools and replication materials for Caliskan, Bryson, and Narayanan (2017)

Description This package allows users to replicate the analysis in the paper and also provides general purpose tools for working with a large word vector file and comparing groups of words with permutation statistics from the original paper. Alternative bootstrapped versions with confidence intervals are also available.

URL https://conjugateprior.github.io/cbn

BugReports https://github.com/conjugateprior/cbn/issues

License GPL-3

Encoding UTF-8

LazyData true

ByteCompile true

Depends R (>= 2.10)

Imports Rcpp, gender

LinkingTo Rcpp

SystemRequirements C++11

RoxygenNote 6.1.1

Suggests knitr, rmarkdown, ggplot2, testthat

VignetteBuilder knitr

Repository https://conjugateprior.r-universe.dev

RemoteUrl https://github.com/conjugateprior/cbn

RemoteRef HEAD

RemoteSha 3a578d5c0a74e64e53130a12bd0235ef11c95c67

Contents

---

cbn *CBN*

Description

This package contains tools and experimental items necessary to replicate Caliskan, Bryson, and Narayanan (2017). 'Semantics derived automatically from language corpora contain human-like biases'

Source

A. Caliskan, J. J. Bryson, and A. Narayanan (2017) 'Semantics derived automatically from language corpora contain human-like biases' *Science*. 356:6334 http://doi.org/10.1126/science.aal4230.

cbn_cosine *Calculates Cosine Similarity Between Matrix Rows*

Description

This function calculates the cosine similarity matrix between all rows of a matrix `x`. When `x` and `y` are vectors it calculates the cosine similarity between them. When `x` is a vector and `y` is a matrix it calculates the cosine between `x` and each row of `y`.

Usage

```
cbn_cosine(x, y = NULL)
```

Arguments

`x`	A vector or a matrix (e.g., a document-term matrix).
`y`	A vector with compatible dimensions to x. If NULL, use all columns of `x`.

Details

This code is taken directly from the `lsa` package but adjusted to operate rowwise.

Value

An `ncol(x)` by `ncol(x)` matrix of cosine similarities, a scalar cosine similarity, or a vector of cosine simialrities of length `nrow(y)`.

Source

The original code is from the `cosine` function by Fridolin Wild (f.wild@open.ac.uk) in the `lsa` package.

---

cbn_extract_word_vectors
Extract Word Vectors From Current Vector File

Description

This function provides a more convenient wrapper for `extract_words`. It uses the current vector file, whose location can be found using `cbn_get_vectorfile_location` and assigned with `cbn_set_vectorfile_location`.

Usage

```
cbn_extract_word_vectors(words, verbose = FALSE, report_every = 1e+05)
```

Arguments

words	words to get vectors for
verbose	whether to report on progress
report_every	how often to check in to see if we should stop

Value

a matrix with word vectors as rows

---

cbn_gender_name_stats *Gender Proportions for Names in the US Population*

Description

This data set contains for each name used in *any* of the studies, (not just those in WEFAT 2) and its the gender proportions in the US population. It was generated by the gender package, which uses US Social Security Administration data.

Usage

```
cbn_gender_name_stats
```

Format

An object of class data.frame with 210 rows and 6 columns.

Details

The columns of the data set are name, the name, proportion_male and proportion_female, gender (a best guess from the proportions), and the years within which the SSA search was performed. This data set can merged with several of the study item sets, but is most useful for replicating the second WEFAT study, as shown in the replication vignette.

This data should typically be joined e.g. using merge, to other item information using the columns 'name' and 'Word' (assuming that information comes from cbn_get_items). The replication vignette has an example.

cbn_gender_name_stats_census1990 *Gender Proportions for Names in the US Population from 1990 Census*

Description

This data set contains gender information *only* for names used in WEFAT 2. It is a slightly normalized version of the original Dataverse materials.

Usage

```
cbn_gender_name_stats_census1990
```

Format

An object of class `data.frame` with 50 rows and 5 columns.

Details

The columns are `name`, `gender.score` a numerical score derived (somehow) using -1 to be female, 0 to mean unisex, and 1 to mean male, and `percentage.in.population`, `percentage.in.male.population`, and `percentage.in.female.population`. Apparently these three are some measure of the prevalence of the name in the US population and two gender subpopulations.

The original materials are a tab separated file located at `system.file("extdata", "censusNames1990.tsv", package = "cbn")`.

Presumably some Bayes theorem with the addition of the population gender balance recreates the quantity of substantive interest: P(gender | name). This has not been done.

cbn_get_items *Get the Items in a Study*

Description

Returns a data frame containing the items from one of the studies (WEAT1 through WEAT10 or WEFAT1 or WEFAT2) or a vector containing all items from all studies if `type == "all"`.

Usage

```
cbn_get_items(type = c("all", "WEAT", "WEFAT"), number = 1)
```

Arguments

`type`	"all" (the default), "WEAT", or "WEFAT"
`number`	study number (default: 1) Ignored if `type = "all"`

Value

a data frame of items in columns or a vector of all items

---

cbn_get_item_vectors *Get Vectors for Items in a Study*

Description

Returns a matrix containing word vectors for the items used in one of the studies (WEAT1 through WEAT10 or WEFAT1 or WEFAT2). If `type == "all"` then vectors for all items used in any of the studies is returned. Words are row names.

Usage

```
cbn_get_item_vectors(type = c("all", "WEAT", "WEFAT"), number = 1)
```

Arguments

`type`	"all" (the default), "WEAT", or "WEFAT"
`number`	study number (default: 1) Ignored if `type = "all"`

Value

a matrix with word vectors as rows

---

cbn_get_vectorfile_location *Get the Location of the Vectors File*

Description

Returns the full path to the file of word vectors. If there is no environment variable `CBN_VECTORS_LOCATION` in the current environment it prompts to set a location with `cbn_set_vectorfile_location`

Usage

```
cbn_get_vectorfile_location()
```

Details

If you want prefer the location of your downloaded vectors to persist across sessions, add `CBN_VECTORS_LOCATION=/Users/m` or similar to your `~/.Renviron` file (creating the file if necessary).

Value

a full path to the vectors file

See Also

cbn_set_vectorfile_location

---

cbn_items *All Items Used in All Studies*

Description

This data frame contains all the items used in all the studies. It is the data source for cbn_get_items. Most of the time you should probably use that.

Usage

```
cbn_items
```

Format

An object of class data.frame with 730 rows and 4 columns.

Source

A. Caliskan, J. J. Bryson, and A. Narayanan (2017) 'Semantics derived automatically from language corpora contain human-like biases' *Science*. 356:6334 http://doi.org/10.1126/science.aal4230.

---

cbn_item_cosines *Cosine Similarity for Every Pair of Study Items*

Description

A matrix of cosine similarities between each item and every other one. Uses cbn_items.

Usage

```
cbn_item_cosines
```

Format

An object of class matrix with 457 rows and 457 columns.

cbn_item_vectors *Vectors for All Items Used in All Studies*

Description

A 457 x 300 matrix of (row) vectors for all study items, extracted from the 840B word Common Crawl data on Jun 30th, 2018.

Usage

```
cbn_item_vectors
```

Format

An object of class `matrix` with 457 rows and 300 columns.

Source

J. Pennington, R. Socher, and C. D. Manning (2014) 'GloVe: Global vectors for word representation' https://nlp.stanford.edu/projects/glove/.

cbn_make_items *Make items*

Description

Make items

Usage

```
cbn_make_items(studyname, words, conditions, roles = NULL)
```

Arguments

`studyname`	Name of your study
`words`	a vector of words
`conditions`	a vector of condition labels (must be the same length as `words`)
`roles`	An optional vector of role description labels (must be the same length as `words`). Values are either `target` or `attribute`

Value

a set of items

cbn_set_vectorfile_location *Set the Location of the Vectors File*

Description

This function adds the location of the file of vectors to the current environment (as the value of CBN_VECTORS_LOCATION). If persist is TRUE it also adds this key to ~/.Renviron so that it is retained across R sessions.

Usage

```
cbn_set_vectorfile_location(f, persist = FALSE)
```

Arguments

f	path where you unzipped your vectors file
persist	Whether to add this to your R startup file

Details

To recover the current location, use cbn_get_vectorfile_location.

Value

Nothing

See Also

cbn_get_vectorfile_location

summary.cbn_study *Summary Method for Study Items*

Description

A summary method for study items extracted via cbn_get_items.

Usage

```
## S3 method for class 'cbn_study'
summary(object, ...)
```

Arguments

object	A set of study items
...	Ignored

Value

Condition names, roles (target or attribute) and N for study items

Examples

```
its <- cbn_get_items("WEAT", 6)
summary(its)
```

`weat_boot` *WEAT via simple item bootstrap*

Description

A simple bootstrap for the WEAT calculations. The statistic of interest is an average difference of average differences.

Usage

```
weat_boot(items, vectors, x_name, y_name, a_name, b_name, b = 300,
  se.calc = c("sd", "quantile"))
```

Arguments

`items`	information about the items, typically from `cbn_get_items`
`vectors`	a matrix of word vectors for all the study items
`x_name`	the *name* of the target item condition, e.g. "Flowers" in WEAT 1
`y_name`	the *name* of the target item condition, e.g. "Insects" in WEAT 1
`a_name`	the name of the first condition, e.g. "Pleasant" in WEAT 1
`b_name`	the name of the second condition, e.g. "Unpleasant" in WEAT 1
`b`	number of bootstrap samples. Defaults to 300.
`se.calc`	how to compute lower and upper bounds on an approximate 95 interval for the difference of differences of cosines statistic. "se" (default) or "quantile".

Details

Schematically, the statistic is the average value of

(cosine(x names, a words) - cosine(x names, b words)) - (cosine(y names, a words) - cosine(y names, b words))

If a denotes a set of 'Pleasant' and b denotes a set of 'Unpleasant' words, x are names of 'Insects', and y are names of 'Flowers' (as in WEAT 1) then the statistic will take positive values when flowers are more pleasant than insects. That is, when the degree to which flower names are more similar to pleasant versus unpleasant words is stronger than the degree to which insect names are more similar to pleasant versus unpleasant words.

Uncertainty is quantified by bootstrapping each set of item vectors. That is, in each of the b bootstrap samples, vectors in each condition (a_name, b_name, x_name and y_name) are separately resampled with replacement, and the statistic is computed. The bootstrap sampling distribution of this statistic is summarized in the output by an approximate 95 statistic across bootstrap samples if se.calc is "sd", or as the 0.025 and 0.975 quantiles of the bootstrap sampling distribution if se.calc is "quantile".

If se.calc is "quantile" the data frame returned has an extra column containing the median of the statistic in the bootstrap samples. This should not be too far from the original statistic.

The sign of the statistic is arbitrary. If you wish to reverse the ordering just swap the values of a_name for b_name or x_name and y_name when calling it.

Note that this is not the statistic reported in the original paper. This bootstraps within each target categories (x and y) and within each attribute category (a and b).

Value

a data frame with first column the difference of differences of cosines statistic, the second and third columns the lower and upper bounds of an approximate 95 interval from the bootstrapped statistic. If se.calc is "quantile", the fourth column is the median value of the statistic across bootstrap samples.

Examples

```
its <- cbn_get_items("WEAT", 1)
its_vecs <- cbn_get_item_vectors("WEAT", 1)
res <- weat_boot(its, its_vecs,
                 x_name = "Flowers", y_name = "Insects",
                 a_name = "Pleasant", b_name= "Unpleasant",
                 se.calc = "quantile")
res
```

---

weat_perm *WEAT Permutation Test*

Description

The statistic computed by this function is the mean cosine similarity of each x item to the a attributes minus the mean cosine to the b attributes, summed over x items subtracted for the same quantity computed for the y items. See the paper for details of the statistic, and the effect size.

Usage

```
weat_perm(items, vectors, x_name, y_name, a_name, b_name, b = 1000)
```

Arguments

items	information about the items, typically from cbn_get_items
vectors	a matrix of word vectors for all the study items, typically from cbn_get_item_vectors
x_name	the *name* of the target item condition, e.g. "Flowers" in WEAT 1
y_name	the *name* of the target item condition, e.g. "Insects" in WEAT 1
a_name	the name of the first condition, e.g. "Pleasant" in WEAT 1
b_name	the name of the second condition, e.g. "Unpleasant" in WEAT 1
b	number of bootstrap samples. Defaults to 1000.

Details

The p value is constructed by permuting the assignment of words to the x and y conditions. (The a and b attribute items are fixed.) The p value is the proportion of times the statistic computed on the permuted labels is greater than the value of the statistic that is observed.

Value

a data frame with first column the statistic, the second column the effect size, and the third column permutation test p value.

Examples

```
its <- cbn_get_items("WEAT", 1)
its_vecs <- cbn_get_item_vectors("WEAT", 1)
res <- weat_perm(its, its_vecs,
                 x_name = "Flowers", y_name = "Insects",
                 a_name = "Pleasant", b_name= "Unpleasant")
res
```

---

wefat *Compute the Paper's WEFAT statistic*

Description

Computes the WEFAT statistic from the paper. No standard error is currently computed.

Usage

```
wefat(items, vectors, x_name, a_name, b_name)
```

Arguments

items	information about the items, typically from cbn_get_items
vectors	a matrix of word vectors for all the study items, typically from cbn_get_item_vectors
x_name	twe *name* of the target word condition, e.g. "AndrogeynousNames" in WEFAT 2
a_name	the name of the first attribute, e.g. "MaleAttributes" in WEFAT 2
b_name	the name of the second attribute, e.g. "FemaleAttributes" in WEFAT 2

Value

a data frame with columns Word and S_wab, the value of the statistic.

Examples

```
its <- cbn_get_items("WEFAT", 2)
vecs <- cbn_get_item_vectors("WEFAT", 2)
wefs <- wefat(its, vecs, x_name = "AndrogynousNames",
              a_name = "MaleAttributes", b_name = "FemaleAttributes")
props <- cbn_gender_name_stats[, c('name', 'proportion_male')]
wefs_props <- merge(wefs, props, by.x = "Word", by.y = "name")
cor.test(wefs_props$S_wab, wefs_props$proportion_male)
```

---

wefat_boot *WEFAT via simple item bootstrap*

Description

A simple bootstrap for the WEFAT calculations. The statistic of interest is the difference between the cosine of each word in condition x_name e.g. "Careers", to the mean vector of condition a_name, e.g. "MaleAttributes" and the mean vector from condition b_name, e.g. "FemaleAttributes".

Usage

```
wefat_boot(items, vectors, x_name, a_name, b_name, b = 300,
  se.calc = c("sd", "quantile"))
```

Arguments

items	information about the items, typically from cbn_get_items
vectors	a matrix of word vectors for the study
x_name	the *name* of the target item condition, e.g. "Careers" in WEFAT 1
a_name	the name of the first condition, e.g. "MaleAttributes" in WEFAT 1 and 2
b_name	the name of the second condition, e.g. "FemaleAttributes" in WEFAT 1 and 2
b	number of bootstrap samples. Defaults to 300.
se.calc	how to compute lower and upper bounds on an approximate 95 interval for the difference of cosines statistic. "se" (default) or "quantile".

Details

Uncertainty is quantified by bootstrapping each set of item vectors. That is, in each of the b bootstrap samples, vectors in the a_name condition and vectors in the b_name condition are resampled (independently) with replacement, and the difference between the cosine of a target word and the mean of the a_name vectors and cosine of a target word and the mean of the b_name is recorded. The bootstrap sampling distribution of this difference of cosines statistic is summarized in the outpu by an approximate 95 statistic across bootstrap samples if se.calc is "sd", or as the 0.025 and 0.975 quantiles of the bootstrap sampling distribution if se.calc is "quantile".

If se.calc is "quantile" the data frame returned has an extra column containing the median of the statistic in the bootstrap samples. This should not be too far from the original statistic.

The output of this function is sorted by the value of the difference of cosines statistic. This direction is arbitrary, but if you wish to reverse the ordering just swap the values of a_name for b_name when calling it.

Note that this is not the statistic reported in the original paper.

Value

a data frame with first column x_name, second column the difference of cosines statistic, third and fourth columns the lower and upper bounds of an approximate 95 from the bootstrapped statistic. If se.calc is "quantile", the fifth column is the median value of the statistic across bootstrap samples. The data frame is sorted by the second column.

Examples

```
its <- cbn_get_items("WEFAT", 1)
its_vecs <- cbn_get_item_vectors("WEFAT", 1)
res <- wefat_boot(its, its_vecs, x_name = "Careers",
                  a_name = "MaleAttributes", b_name = "FemaleAttributes",
                  se.calc = "quantile")
```

Index