Package ‘WordPools’

October 12, 2022

Type Package
Title Classical Word Pools Used in Studies of Learning and Memory
Version 1.1-1
Date 2018-12-14
Maintainer Michael Friendly <friendly@yorku.ca>
Description Collects several classical word pools used most often to provide lists of words in psychological studies of learning and memory. It provides a simple function, 'pickList' for selecting random samples of words within given ranges.

Suggests dplyr
Depends R (>= 2.10)
License GPL-2
LazyLoad yes
LazyData yes
RoxygenNote 6.1.1
Encoding UTF-8

Author Michael Friendly [aut, cre] (<https://orcid.org/0000-0002-3237-0941>)
Repository CRAN

Date/Publication 2018-12-15 14:10:07 UTC

R topics documented:

- WordPools-package ................................. 2
- Battig ........................................... 3
- CatProp .......................................... 4
- Paivio ........................................... 6
- pickList ........................................... 7
- plotDensity ...................................... 9
- TWP ............................................. 10
- within ......................................... 12
Description

This package collects several classical word pools used most often to provide lists of words in psychological studies of learning and memory.

Each word pool consists of a population of words, together with various descriptive measures (number of letters, number of syllables, word frequency, etc.) and normative measures (imagery, concreteness, etc.) that can be used in experimental designs to vary and control such factors.

Details

Package: WordPools
Type: Package
Version: 1.1-1
Date: 2018-12-14
License: GPL-2

At present, the package contains three main word pools:

Paivio - the Paivio etal. (1968) word list of 925 nouns
TWP - the Friendly etal. (1982) Toronto Word Pool of 1080 words in various grammatical classes
Battig - the Battig & Montague (1969) Categorized Word Norms, containing 5231 words listed in 56 taxonomic categories. Various measures on these categories are given in CatProp.

In addition, the function pickList provides the ability to select items from such lists with restrictions on the ranges of the measured variables.

Author(s)

Michael Friendly

Maintainer: Michael Friendly <friendly@yorku.ca>

References


Battig


See also http://memory.psych.upenn.edu/Word_Pools for other related word pools

---

**Description**

This dataset comprises a ranked list of 5231 words listed in 56 taxonomic categories by people who were asked to list as many exemplars of a given category ("a precious stone", "a unit of time", "a fruit", "a color", etc.). Participants had 30s to generate as many responses to each category as possible, after which time the next category name was presented.

Included in this dataset are all words from the Battig and Montague (1969) norms listed with freq > 1.

**Usage**

data(Battig)

**Format**

A data frame with 5231 observations on the following 9 variables.

- **word**: a character vector
- **catnum**: category number, a factor
- **catname**: category name, a factor
- **syl**: number of syllables
- **letters**: number of letters
- **freq**: Frequency of response
- **frequency**: Kucera-Francis word frequency
- **rank**: rank of freq within the category
- **rfreq**: rated frequency

**Details**

In our original dataset, words were truncated at 18 characters, so some are incomplete.
Source


References


Examples
data(Battig)
## maybe str(Battig) ; plot(Battig) ...

# select items from several categories
cats <- c("fish", "bird", "flower", "tree")
for (c in cats) {
  cat("\nCategory:", c, "\n")
  print(pickList(subset(Battig, catname==c), nitems=5))
}

# or, using sapply():
sapply(cats, function(c) pickList(subset(Battig, catname==c), nitems=5), simplify=FALSE)

---

CatProp | *Joelson-Hermann Category Properties*

Description


Usage

data(CatProp)

Format

A data frame with 56 observations on the following 24 variables.

catnum Category number, a numeric variable
catname Category name, a character variable
rnatr1 Rated naturalness 1..7, a numeric variable
rfamil  Rated familiarity 1..7, a numeric variable
rmeang  Rated meaningfulness 1..7 (Hunt & Hodge, 1971), a numeric variable
rfreq  Rated frequency 1..7 B&M, a numeric variable
genfreq  Generated category label frequency, a numeric variable
rageoq  Rated age of acquisition 1..10, a numeric variable
rsize  Estimated category size, a numeric variable
ts_30  Mean # types produced in 30 seconds, a numeric variable
rclasm  Recall asymptote, a numeric variable
rclrate  Recall rate parameter, a numeric variable
tas  Types across subjects, a numeric variable
cortas  Corrected types across subjects, a numeric variable
ntf  # of types produced first, a numeric variable
nmngox  # of dictionary meanings (Oxford), a numeric variable
nmngam  # of dictionary meanings (Am. Heritage), a numeric variable
catfreqp  category label K-F frequency, a numeric variable
rabcon  Rated abstract-concreteness 1..7, a numeric variable
rvagprc  Rated vague-precise 1..7, a numeric variable
exfreqp  Avg exemplar log K-F frequency, a numeric variable
intsam  Intersample correlation, a numeric variable
maxfreq  Maximum response frequency, a numeric variable
pagmt  Percent agreement on category membership, a numeric variable

Details

Includes data for all 56 of the Battig-Montague categories from a preprint of the Joelson-Hermann paper. Values for catfreqp were added for categories 3, 4, 8, 15, 24, 27, 32, 46, 47 & 56 from the Kucera-Francis norms, ignoring "part of", "unit of", and taking max of labels connected by "or".

Source


Examples

data(CatProp)
summary(CatProp)
plot(CatProp[,3:10])

# try a biplot
CP <- CatProp
rownames(CP) <- CP$catname
biplot(prcomp(na.omit(CP[,3:12]), scale=TRUE))
# select some categories where the rated age of acquisition is between 2-4
cats <- pickList(CatProp, list(rageoaq=c(2,4)))
cats[,2:9]

# pick some fruit
pickList(subset(Battig, catname=="fruit"))

## Description
The Paivio, Yuille & Madigan (1968) word pool contains 925 nouns, together with average ratings of these words on imagery, concreteness and meaningfulness, along with other variables.

## Usage
data(Paivio)

## Format
A data frame with 925 observations on the following 9 variables.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>itmno</td>
<td>item number</td>
</tr>
<tr>
<td>word</td>
<td>the word</td>
</tr>
<tr>
<td>imagery</td>
<td>imagery rating</td>
</tr>
<tr>
<td>concreteness</td>
<td>concreteness rating</td>
</tr>
<tr>
<td>meaningfulness</td>
<td>meaningfulness rating</td>
</tr>
<tr>
<td>frequency</td>
<td>word frequency, from the Kucera-Francis norms</td>
</tr>
<tr>
<td>syl</td>
<td>number of syllables</td>
</tr>
<tr>
<td>letters</td>
<td>number of letters</td>
</tr>
<tr>
<td>freerecall</td>
<td>Free recall proportion, added from Christian et al (1978)</td>
</tr>
</tbody>
</table>

## Details
The freerecall variable has 27 NAs.

## Source

pickList

Select Items from a Word Pool in Given Ranges

Description
This is a convenience function to provide the capability to select items from a given word pool, with restrictions on the range of any numeric variables.

Usage
pickList(data, ranges, nitems = 10, nlists = 1, replace = FALSE)
Arguments

- **data**: A data.frame, typically a word list like Paivio or TWP.
- **ranges**: A data.frame of two rows, and with column names corresponding to a subset of the column names in data. The two rows give the minimum and maximum values, respectively, of variables in data. Alternatively, you can supply a named list containing the minimum and maximum values for one or more variables in data.
- **nitems**: Number of items per list.
- **nlists**: Number of lists.
- **replace**: A logical value, indicating whether the sampling of items (rows) of data is to allow sampling with replacement.

Details

The `sample` will generate an error if fewer than `nitems * nlists` items are within the specified ranges and `replace=FALSE`.

Value

A data frame of the same shape as `data` containing the selected items prefixed by the list number.

Author(s)

Michael Friendly

References


See Also

- `sample`

Examples

data(Paivio)
# 2 lists, no selection on any variables
pickList(Paivio, nlists=2)

# Define ranges for low and high on imagery, concreteness, meaningfulness
# These go from low - median, and median-high on each variable
vars <- 3:5
(low <- as.data.frame(apply(Paivio[,vars], 2, fivenum))[c(1,3),])
(high <- as.data.frame(apply(Paivio[,vars], 2, fivenum))[c(3,5),])

# select two lists of 10 low/high imagery items
lowI <- pickList(Paivio, low[,"imagery", drop=FALSE], nitems=10, nl=2)
highI <- pickList(Paivio, high[,"imagery", drop=FALSE], nitems=10, nl=2)
# compare means
colMeans(lowI[,c(4:8)])
colMeans(highI[,c(4:8)])

# using a list of ranges
L <- list(imagery=c(1,5), concreteness=c(1,4))
pickList(Paivio, L)

---

**plotDensity**  
Enhanced density plot for WordPools

**Description**
Plots the distribution of a variable with a density estimate and a rug plot

**Usage**

```r
plotDensity(data, var, adjust = 1, lwd = 2, fill = rgb(1, 0, 0, 0.2),
            xlab = NULL, main = NULL, anno = FALSE, ...)
```

**Arguments**
- `data`: A data.frame
- `var`: Name of the variable to be plotted
- `adjust`: Adjustment factor for the bandwidth of the density estimate
- `lwd`: Line width
- `fill`: Color to fill the area under the density estimate
- `xlab`: Label for the variable
- `main`: Title for plot
- `anno`: If TRUE
- `...`: Other arguments passed to `plot.density`

**Value**
Returns the result of `density`

**Examples**

```r
plotDensity(Paivio, "imagery", anno=TRUE)
plotDensity(Paivio, "imagery", anno=TRUE, adjust=1.5)
plotDensity(Paivio, "syl")

plotDensity(TWP, "imagery", anno=TRUE)
```
Description

The Toronto Word Pool consists of 1080 words in various grammatical classes together with a variety of normative variables.

The TWP contains high frequency nouns, adjectives, and verbs taken originally from the Thorndike-Lorge (1944) norms. This word pool has been used in hundreds of studies at Toronto and elsewhere.

Usage

data(TWP)

Format

A data frame with 1093 observations on the following 12 variables.

- item number
- the word
- imagery rating
- concreteness rating
- number of letters
- word frequency, from the Kucera-Francis norms
- a measure of first order approximation to English. In a first-order approximation, the probability of generating any string of letters is based on the frequencies of occurrence of individual letters in the language.
- a measure of second order approximation to English, based on bigram frequencies.
- Orthographic neighbor ratio, taken from Landauer and Streeter (1973). It is the ratio of the frequency of the word in Kucera and Francis (1967) count divided by the sum of the frequencies of all its orthographic neighbors.
- dictionary codes, a factor indicating the collection of grammatical classes, 1-5, for a given word form. In the code, "1" in any position means the item had a dictionary definition as a noun; similarly, a "2" means a verb, "3" means an adjective, "4" means an adverb, and "5" was used to cover all other grammatical categories (but in practice was chiefly a preposition). Thus an entry "2130" indicates an item defined as a verb, noun, and an adjective in that order of historical precedence.
- percent noun usage. Words considered unambiguous based on dictcode are listed as 0 or 100; other items were rated in a judgment task.
- a factor indicating an alternative Canadian spelling of a given word

Details

The last 13 words in the list are alternative Canadian spellings of words listed earlier, and have duplicate itemno values.
TWP

Source


References


Examples

data(TWP)
str(TWP)
summary(TWP)
# quick view of distributions
boxplot(scale(TWP[, 3:9]))

plotDensity(TWP, "imagery")
plotDensity(TWP, "concreteness")
plotDensity(TWP, "frequency")

# select low imagery, concreteness and frequency words
R <- list(imagery=c(1,5), concreteness=c(1,4), frequency=c(0,30))
pickList(TWP, R)

# dplyr now makes this much more flexible
if (require(dplyr)) {
  # select items within given ranges
  selected <- TWP %>%
    filter( canadian == 0 ) %>%
    filter( imagery <= 5, concreteness <= 4, frequency <= 30 ) %>%
    select(word, imagery:frequency )

  str(selected)

  # get random samples of selected items
  nitems <- 5
  nlists <- 2
  lists <- selected %>%
    sample_n(nitems*nlists, replace=FALSE) %>%
    mutate(list = rep(1:nlists, each=nitems))

  str(lists)
  lists
}
within

Select observations within a given range

Description

This function masks `base::within` and so is no longer exported. Eventually it will be removed.

Usage

```r
within(x, a, b)
```

Arguments

- `x`: A vector
- `a`: Lower limit
- `b`: Upper limit

Value

A logical vector of the same length as `x`

Examples

```r
WordPools::within(1:10, 2, 5)
```
Index

* **datagen**
  pickList, 7

* **datasets**
  Battig, 3
  CatProp, 4
  Paivio, 6
  TWP, 10
  Battig, 2, 3
  CatProp, 2, 4
  density, 9
  Paivio, 2, 6, 8
  pickList, 2, 7
  plot.density, 9
  plotDensity, 9
  sample, 8
  TWP, 2, 8, 10
  within, 12
  WordPools (WordPools-package), 2
  WordPools-package, 2