Package ‘bldR’

May 30, 2017

Type Package

Title Bilingual-Led Diversity Library

Version 0.1.0

Author T. Mark Ellison <bldR@markellison.net>

Maintainer T. Mark Ellison <bldR@markellison.net>

Description This library contains functions for simulating and testing models of Bilingual-Led Divergence, such as described in the paper: Ellison, TM & L Miceli (2017 - 93(2):255-287) Language Monitoring in Bilinguals as a Mechanism for Rapid Lexical Divergence. Language. The (R6) classes include TensorModel implementing the cognitive models in the paper, PopulationSimulation proving agent-based simulations where each agent implements the cognitive model, and L2017 which constructs the graphs and statistics presented in the paper.

License GNU Public License Version 3 - GPLv3

LazyData FALSE

RoxygenNote 6.0.1

R topics documented:

`bldR` *bldR*

Description

`bldR` is a package for modelling bilingual-led divergence of lexica.

Details

bldR provides three R6 classes: `TensorAgent`, `PopulationSimulation`, and `L2017`.

Each are described on their own pages.

See Also

Most uptodate version at https://github.com/tyrannomark/bldR

`EM2017_ParameterFit` *EM2017_ParameterFit*

Description

This data set contains the fit of the cognitive model described in Ellison & Miceli (2017 - Language 93(2):255-287) - hereafter EM - to the data returned by the experiment also described in the paper.

Format

A data frame with 35968726 rows and 9 columns. Of these, 3 columns are for internal use and can be ignored. The others fields are the following:

Form The form returned used (one doppel per stimulus, and abstract values indicating non-doppels used in English and Dutch).

Meaning Names for th 41 distinct meanings tested in the experiment. These take the form of "S"+<stimulus-number+"_"+(allcaps)<meaning>.

Language The language the simulated participant is trying to express their form in: either English or Dutch. In the experiment reported in EM, articipants were only aiming to produce forms in English, so the values for `Language="Dutch"` are not used in parameter-fitting.

lm Language mode varying from 0.00 to 1.00 in increments of 0.01.

ml Monitoring level varying from 0.00 to 1.00 in increments of 0.01.

p The conditional probability of the form given the parameter values - basically, how well did the model do at predicting the frequency with which this form would be used by bilinguals.

Author(s)

T. Mark Ellison <`m.ellison@anu.edu.au`>

References

`https://github.com/tyrannomark/bldR`

`EM2017_Responses` *EM2017_Responses*

Description

This data set contains the responses returned in the experiment described in Ellison & Miceli (2017 - Language 93(2):255-287).

Format

A data frame with 2049 rows and 5 columns. The columns are as follows:

Condition Either "Monolingual" or "Bilingual" indicating the linguistic background of the participant.

ParticipantId The identifying code for the participant.

StimulusId The identifying code for the stimulus.

Response The response given by this participant to this stimulus.

IsDoppel A numeric value with 1 if the response is a doppel, and 0 if not.

Author(s)

T. Mark Ellison `<m.ellison@anu.edu.au>`

References

`https://github.com/tyrannomark/bldR`

`EM2017_Stimuli` *EM2017_Stimuli*

Description

This data set contains the stimuli used in the experiment described in Ellison & Miceli (2017 - Language 93(2):255-287).

Format

A data frame with 41 rows and 7 columns. The columns are as follows:

StimulusId An identifying number associated with this particular stimulus.

DutchDoppel The Dutch half of the doppel which is a potential response to the stimulus.

EnglishDoppel The English half of the doppel which is a potential response to the stimulus. This is a potential response to the stimulus.

EnglishNonDoppelExample An example word which could also be a response to the stimulus, but which has no doppel in Dutch.

DutchContext This is the paragraph immediately preceding the sentence frame, tightening the semantic context and (when used with EnglishFrame) pushing the participants into bilingual mode.

EnglishContext This is the paragraph immediately preceding the sentence frame used with monolingual English speakers.

EnglishFrame This is the sentence frame (with a non-initial gap) used to elicit a response from the participants.

Author(s)

T. Mark Ellison <m.ellison@anu.edu.au>

References

https://github.com/tyrannomark/bldR

`L2017` *L2017*

Description

This class contains methods for creating the graphs in Ellison & Miceli (2017 - Language 93(2):255-287). These graphs are based on simulations from the classes `TensorAgent` and `PopulationSimulation`. There is a method corresponding to each graph, which creates the graph in PNG format and writes it to the current working directory.

Usage

```
L2017
```

Format

`R6Class` object.

Details

A further method uses the Student t test to evaluate the significance of the difference in distribution between the monolingual and bilingual results from the experiment.

Value

An object of class L2017.

Object of `L2017` with methods for creating graphs for Ellison & Miceli (2017) published in Language 93(2):255-287.

Methods

`$new()` Creates a new, empty lexicon object.

`$draw_1ab(graph_number,language,side)` Draws graphs 1a and 1b, depending on arguments (see example).

`$t_test()` Performs the Student t test comparing doppel rates between monolinguals and bilinguals in the experimental results.

`$draw_2()` Draws graph 2.

`$draw_3a()` Draws graph 3a.

`$draw_3b()` Draws graph 3b.

`$draw_4a()` Draws graph 4a.

`$draw_4b(graphnum="4b")` Draws graph 4b.

`$draw_4c()` Draws graph 4c.

`$draw_6()` Draws graph 6.

`$draw_7a()` Draws graph 7a.

`$draw_7b()` Draws graph 7b.

`$draw_8()` Draws graph 8.

`$draw_9()` Draws graph 9.

`$draw_10a(rx=0.2,ry=0.2,xc=0.5,yc=0.5,A=100,B=100,AB=100,linetypes=c("solid","dashed"),linecolou`
Draws graph 10a - graphs 10b, 10c build on this function, changing the default parameter values.

`$draw_10b()` Draws graph 10b.

`$draw_10c()` Draws graph 10c.

`$draw_all()` Draws graphs all graphs from the paper.

Examples

```
library(bldR)
self$t_test();
self$draw_1ab("1a","SG","left");
self$draw_1ab("1b","E","right");
self$draw_2();
self$draw_3a();
self$draw_3b();
self$draw_4a();
self$draw_4b();
self$draw_4c();
self$draw_6();
self$draw_7a();
self$draw_7b();
self$draw_9();
self$draw_10a();
self$draw_10b();
self$draw_10c();
```

TensorAgent *TensorAgent: a model of bilingual lexical selection.*

Description

This class implements the model of individual lexical selection described in Ellison & Miceli (2017 - Language 93(2):255-287) - hereafter EM. It is implemented using the TensorA, because that class facilitates linear calculations parameterised over a variable number of parameter dimensions.

Usage

```
TensorAgent
```

Format

`R6Class` object.

Value

Object of class `TensorAgent`.

Fields

Documentation Here are `TensorModel`'s fields. EM abbreviates Ellison & Miceli (2017).

`$LanguageMode` This is the language mode parameter on a scale $[0,1]$.

`$MonitoringLevel` The is the effort put into monitoring on a scale $[0,1]$.

`$Meanings` The meanings currently in the lexicon (a list of strings).

`$Languages` The languages modelled in the lexicon (a list of strings).

`$NumberOfLanguages` The number of languages modelled in the lexicon (integer).

`$LexicalTensor` The tensor with the frequency of occurrence for each language-meaning-form combination.

`$version` An integer tracking whether updates to values to avoid repeated calculation of the same values.

`$delta_lt` Kronecker delta (1 if $l=t$ zero otherwise), see Equation 2 in EM.

`$p_l_t__b` Probability of using language l given a target language t and language mode b.

`$p_f_sl` Probability of using form f to represent meaning s in language l, see Equation 3 in EM.

`$p_f_st__b` Probability of using form f to representing meaning s when trying to use language t and the bilingual mode is b, see Equation 4 in EM.

`$p_l_fst__bm` Probability of identifying language l as the source of form f when trying to express meaning s in language t, see Equation 8 in EM.

`$p_f_st__bm` Probability of using form f to express meaning s when aiming to speak language t, given bilingual mode b and monitoring level m, see Equation 11 in EM.

`$p_l_t__b_version` The version number associated with the current value of `p_l_t__b`.

`$p_f_sl_version` The version number associated with the current value of `p_f_sl`.

`$p_f_st__b_version` The version number associated with the current value of `p_st__b`.

`$p_l_fst__bm_version` The version number associated with the current value of `p_l_fst__bm`.

`$p_f_st__bm_version` The version number associated with the current value of `p_f_st__bm`.

Methods

`$new()` Creates a new, empty `TensorModel` object.

`$clearLexicon()` Empties the lexical memory, removing all meaning-language-form triples.

`$setLanguageMode(languageMode)` Set the level of interaction between target and other languages.

`$setMonitoringLevel(monitoringLevel)` Set the level of monitoring exerted by the agent.

`$addExample(meaning,language,form,ct=1)` Adds meaning-language-form triple to the model if it does not exist already. It then adds `ct` to the frequency recorded for this triple.

`$normalise(t,overIndices)` Normalise a tensor `t` over some vector of indices `overIndices`.

`$constructDataTensor()` Uses the tuples entered using `$addExample(..)` to build a tensor representing the distribution over experienced meaning-language-form combinations.

`$makeMeLanguagePairs()` Constructs , a rank-2 tensor, both of whose indices range over languages, and whose values for each dimension combination is 1.0.

`$make_p_l_t__b()` Calculate the probability of a language given a target language.

`$make_p_f_sl()` Calculate the probability of using form f to represent meaning s in language l.

`$make_p_f_st__b()` Calculate the probability of using form f to representing meaning s when trying to use language t and the bilingual mode is b.

`$make_p_l_fst__bm()` Calculate the probability of identifying language l as the source of form f when trying to express meaning s in language t.

`$make_p_f_st__bm()` Calculate the probability of using form f to express meaning s when aiming to speak language t, given bilingual mode b and monitoring level m.

`$as.data.frame()` Create a data frame with columns `Meaning`, `Language`, `Form` and probability of form `p`, giving the distribution `p_f_st__bm`.

Examples

```
library(bldR)
ta <- TensorAgent$new();
ta$clearLexicon();
ta$addExample("PHOTO","English","foto",    ct=0.5);
ta$addExample("PHOTO","English","picture", ct=0.5);
ta$addExample("PHOTO","Dutch",  "foto",    ct=1.0);
ta$constructDataTensor();
ta$setLanguageMode(0.4);
ta$setMonitoringLevel(0.8);
ta$make_p_f_st__bm();
df <- ta$as.data.frame();
print(df);
```

Index