Package ‘IAT’

October 12, 2022

**Title**  Cleaning and Visualizing Implicit Association Test (IAT) Data

**Description**  Implements the standard D-Scoring algorithm  
(Greenwald, Banaji, & Nosek, 2003) for Implicit Association Test (IAT)  
data and includes plotting capabilities for exploring raw IAT data.

**Version**  0.3

**Author**  Dan Martin <dpmartin42@gmail.com>

**Maintainer**  Dan Martin <dpmartin42@gmail.com>

**Depends**  R (>= 3.0)

**Imports**  dplyr (>= 0.4.3), ggplot2 (>= 2.1.0), lazyeval (>= 0.1.10),  
stat (< =3.2.0)

**License**  Apache License 2.0

**LazyData**  true

**RoxygenNote**  5.0.1

**NeedsCompilation**  no

**Repository**  CRAN

**Date/Publication**  2016-04-30 00:51:43

**R topics documented:**

- cleanIAT  .......................................................... 2
- IATData ........................................................... 3
- plotIIV ............................................................. 4
- plotIndVar .......................................................... 5
- plotItemErr .......................................................... 5
- plotItemVar .......................................................... 6

**Index**  7
cleanIAT

Clean IAT data using the updated D-Scoring algorithm

Description

Transform a dataframe with trial latencies (stored as one line per trial) for a standard format IAT (7 blocks) into a one line summary per subject of the IAT effect using the standard scoring algorithm recommended in Greenwald, Nosek, & Banaji (2003). The goal is to prepare IAT data for subsequent analysis. However, this does not relieve the researcher from making conceptual decisions about how best to analyze IAT data. There are decisions to make about how the function is applied, and the function does not remove participants. All subject exclusions must be made deliberately by the researcher. Note that the output of this function is identical to that of the standard SAS macro (link in reference) for all meaningful columns.

Usage

```
cleanIAT(my_data, block_name, trial_blocks, session_id, trial_latency, trial_error, v_error, v_extreme, v_std)
```

Arguments

- **my_data**: The raw dataframe to be used
- **block_name**: A string of the variable name for the blocks
- **trial_blocks**: A vector of the four essential blocks in the seven-block IAT (i.e., B3, B4, B6, and B7).
- **session_id**: A string of the variable name identifying each unique participant.
- **trial_latency**: A string of the variable name for the latency of each trial.
- **trial_error**: A string of the variable name identifying whether a trial was an error or not, where 1 indicates an error.
- **v_error**: If 1 (current standard), then means are calculated for the entire set of latencies. If 2, error latencies will be replaced by the block mean + 600ms
- **v_extreme**: If 1, then no extreme value treatment. If 2 (current standard), delete trial latencies < 400ms
- **v_std**: If 1 (current standard), block SD is performed including error trials (corrected or not). If 2, block SD is performed on correct responses only

Value

Outputs a dataframe that must be saved to an object. The variable IAT is the calculated D-Score for each individual. SUBEXCL notes any exclusion criteria, with 0 being inclusion data, 1 for exclusion due to fast response, and 2 for exclusion due to missing blocks. C indicates standard deviation for combined blocks (correct trial only), while A indicates standard deviations for combined blocks (all trials). M (mean), E (percent error), N (number of trials used), and F (percent fast responses), are reported for each block included in the original dataframe.
Examples

# Get Ps who receive Math-Male sorting task in first blocks
cong_first <- IATData[IATData$isCongruentFirst == 1, ]
dscore_first <- cleanIAT(my_data = cong_first,
    block_name = "BLOCK_NAME_S",
    trial_blocks = c("BLOCK2", "BLOCK3", "BLOCK5", "BLOCK6"),
    session_id = "SESSION_ID",
    trial_latency = "TRIAL_LATENCY",
    trial_error = "TRIAL_ERROR",
    v_error = 1, v_extreme = 2, v_std = 1)

# Get Ps who receive Math-Female sorting task in first blocks
cong_second <- IATData[IATData$isCongruentFirst == 0, ]
dscore_second <- cleanIAT(my_data = cong_second,
    block_name = "BLOCK_NAME_S",
    trial_blocks = c("BLOCK2", "BLOCK3", "BLOCK5", "BLOCK6"),
    session_id = "SESSION_ID",
    trial_latency = "TRIAL_LATENCY",
    trial_error = "TRIAL_ERROR",
    v_error = 1, v_extreme = 2, v_std = 1)

d_score <- rbind(dscore_first, dscore_second)

IATData

Sample Gender Stereotype Implicit Association Test data

Description

A dataframe containing data from a Gender Stereotype Implicit Association Test. Data was taken from college students in a differential equations classroom taught by a female professor.

Format

A dataframe with 11792 observations of 16 variables (88 students in total)

- BLOCK_NAME_S: string of blocknames
- BLOCK_PAIRING_DEFINITION_S: string of block pairings
- TRIAL_NAME_S: word/picture used in sorting trial
• SESSION_ID: ID of participant
• TRIAL_NUMBER: number of trial within block
• TRIAL_ERROR: indicates whether trial was an error (1 = YES)
• TRIAL_LATENCY: reaction time for trial
• isCongruentFirst: indicates if stereotype congruent blocks came first

Author(s)
Dan Martin dpmartin42@gmail.com

References
http://projectimplicit.net/fpi/researchers.html

plotIIV

Plot intraindividual variability of reaction time

Description
Plot intraindividual variability in reaction time, faceted by the four essential blocks.

Usage
plotIIV(my_data, data_type, block_name, trial_blocks, session_id, trial_number, trial_latency)

Arguments
my_data The raw dataframe to be used
data_type A string of "raw" for no cleaning, or "clean" for cleaned data (no error trials, RT < 10,000ms, and RT > 180ms)
block_name A string of the variable name for the blocks
trial_blocks A vector of the four essential blocks in the seven-block IAT (i.e., B3, B4, B6, and B7).
session_id A string of the variable name identifying each unique participant.
trial_number A string of the variable identifying the trial number.
trial_latency A string of the variable name for the latency of each trial.
**plotIndVar**

*Plot individual variability in the IAT*

**Description**

Plot mean participant reaction time with 95% confidence intervals to see how reaction time varies by participant. The data is automatically cleaned (i.e., no error trials, trials with RT > 10000 or < 180 are deleted) to avoid over/underinflation of mean estimates and only includes trials from essential blocks.

**Usage**

```r
plotIndVar(my_data, block_name, trial_blocks, session_id, trial_latency, trial_error)
```

**Arguments**

- `my_data`: The raw dataframe to be used
- `block_name`: A string of the variable name for the blocks
- `trial_blocks`: A vector of the four essential blocks in the seven-block IAT (i.e., B3, B4, B6, and B7).
- `session_id`: A string of the variable name identifying each unique participant.
- `trial_latency`: A string of the variable name for the latency of each trial.
- `trial_error`: A string of the variable name identifying whether a trial was an error or not (1 = error)

---

**plotItemErr**

*Plot proportion of errors per item in the IAT*

**Description**

Plot proportion of errors in the IAT to see if any items yield high error rates. The data is automatically cleaned (i.e., trials with RT > 10000 or < 180 are deleted) to avoid over/underinflation of mean error estimates.

**Usage**

```r
plotItemErr(my_data, item_name, trial_latency, trial_error)
```

**Arguments**

- `my_data`: The raw dataframe to be used
- `item_name`: A string of the variable identifying the items
- `trial_latency`: A string of the variable name for the latency of each trial.
- `trial_error`: A string of the variable name identifying whether a trial was an error or not (1 = error)
plotItemVar

Plot IAT item variability

Description
Plot mean item reaction time with 95% confidence intervals to see how reaction time varies by item. The data is automatically cleaned (i.e., no error trials, and trials with RT > 10000 or < 180 are deleted) to avoid over/underinflation of mean estimates and only includes trials from essential blocks.

Usage
plotItemVar(my_data, block_name, trial_blocks, item_name, trial_latency, trial_error)

Arguments
- my_data: The raw dataframe to be used
- block_name: A string of the variable name for the blocks
- trial_blocks: A vector of the four essential blocks in the seven-block IAT (i.e., B3, B4, B6, and B7).
- item_name: A string of the variable identifying the items
- trial_latency: A string of the variable name for the latency of each trial.
- trial_error: A string of the variable name identifying whether a trial was an error or not (1 = error)
Index

* Association
  IATData, 3
* IAT
  IATData, 3
* Implicit
  IATData, 3
* Test
  IATData, 3

cleanIAT, 2
IATData, 3
plotIIV, 4
plotIndVar, 5
plotItemErr, 5
plotItemVar, 6