JASMIN2 Data Manual

Thomas Pronk, pronkthomas@gmail.com
Version 1.3, 2018-08-02

Table of Contents

Introduction

This manual describes data maintained and produced by JASMIN2 tasks. The manual assumes the data has already been decoded into CSV files; for finding out how to do this, see SANDRA Tutorial 2: Decoding Trial Data. The manual consists of separate sections for each of the types of data produces by JASMIN2, namely:

- **task_start**, which contains metadata, such as the variant of a task being administered (e.g. a valence or approach/avoidance IAT), and information about the web-browser.
- **task**, which contains the properties and responses of each trial presented to the participant.
- **slideshow**, which contains the properties and responses to each slide presented to the participant.
- **screen**, which contains information about the screen state, such as when the screen got and lost focus, was resized, etc.

Variable Types

Each variable logged is of a certain type. Below is a list of possible variable types.

Type	Example Values	Description
string	I am a string	A sequence of characters (that may not contain any "tab" or "newline" characters, since these are used to separate values and rows from each other).

int	-23, 0, 11789	A whole number that may be positive or negative.
float	-23.001, 0.914	A number with decimals, that may be positive of negative
bool	TRUE, FALSE	A dichotomous variable that can only be TRUE or FALSE.

Task data

JASMIN2 task data consists of a set of variables shared across tasks, which are described in the next sub-section, and a set of variables that are unique to each task, which are described in successive sub-sections.

General Variables

Variable	Type	Description
block_type	string	Identifies a particular block (such as "practice" or "training"), but generally has no function beyond that.
cat	string	Category that the stimulus presented this trial belonged to (such as "practice", "test", or "control")
stim	string	Stimulus presented.
response	int	Type of response given this trial. Possible values: • 1. Correct • 2. Incorrect • 3. Timeout (no response was given within the response window) • 4. Invalid (an invalid keyboard key was pressed, or an invalid location on the screen was clicked/touched) • NA. The trial was aborted before any response could be registered.
rt	float	Response time of trial response in milliseconds
id	string	• For keyboard responses, the key code of the key pressed. For an overview of key codes, see: https://www.cambiaresearch.com/articles/15/javascript-char-codes-key-codes • For mouse and touch responses, the ID of the HTML Element clicked or touched.
modality	string	• For keyboard responses: **keydown** (key was pressed) or **keyup** (key was released) • For mouse responses: **mousedown** (mouse button was pressed) or **mouseup** (mouse button was released) • For touch responses: **touchstart** (screen was touched) or **touchend** (screen was let go)
label	string	Response mapped to response identifiers that are unified across keyboard, mouse, and touch modalities. Values vary across tasks.
fb_response	string	Response given to a feedback slide (that, for example, informs participants that they pressed an invalid key or were too slow with responding). Values are similar to the label variable. NA if no feedback slide was shown this trial.
fb_rt	String	Response time of response to feedback slide in milliseconds. NA if no feedback slide was shown this trial.

fb_id	String	ID of response to feedback slide. Same values as id variable. NA if no feedback slide was shown this trial.
block	int	Counts blocks. Starts counting with 0.
trial	int	Counts trials per block. Starts counting with 0.
retry	bool	If TRUE, then this trial has been retried. If FALSE, then after this trial the participant went on to the next one.
attempt	int	Counts attempts per trial, which can be used to identify the first response to a trial, if a trial is repeated upon an incorrect response. Starts counting with 0.
time_xxx	float	Time in milliseconds since start of task that a certain trial event occurred. The type of events vary per task (for example "time_start", "time_fix", etc.).
logger_sn	int	Logger serial number, identifies unique trial logs. Starts counting with 0. Each successive trial that is logged should have a value that is one higher than the previous one.
logger_time	float	Time in milliseconds since start of task that this trial data was logged.
UserID	int	LOTUS variable
RunID	Int	LOTUS variable
rel_time	float	Time between two succeeding logs, if this value is negative for any trial, then there might have been something wrong with the computer clock of your participant.
sequence_report	string	Reports on suspicious things in the logs. If sequence report is not empty, the data is not to be trusted. See the SANDRA Manuals for an overview of possible values in the sequence_report.

AAT (Approach Avoidance Task)

Variable	Type	Description
trial_type	string	approach or avoid
approach_left	bool	If TRUE, then left rotated pictures should have been approached (and right rotated pictures avoided). If FALSE, vice versa.
sust	bool	If TRUE, then after pressing a button (for starting the zooming), this button was not released until the trial was completed. If FALSE, then the button was released while zooming.
final_response	int	The **response** variable contains the initial response given by the participant. In contrast, **final_response** contains the response given at the end of the trial. To clarify: if a participant first chose to zoom in, then released the zoom in button, chose to zoom out and completed the trial as such, then **response** and **final_response** have different values. Also, in all these cases, **sust** should be FALSE.

Go/Nogo

Variable	Type	Description

trial_type	string	go or nogo
response	int	In the case of the Go/Nogo, no response 3 (timeout) is registered, since not responding in the response window is itself a correct or incorrect response.

IAT, BIAT, STIAT, and SCIAT

Variable	Type	Description
t1_left	bool	If TRUE, target1 is first presented left (IAT/STIAT) or first focal (BIAT). If FALSE, target1 is first presented right (IAT/STIAT) or first non-focal (BIAT).
a1_left	bool	If TRUE, attribute1 is first presented left (IAT/STIAT) or first focal (BIAT). If FALSE, attribute1 is first presented right (IAT/STIAT) or first non-focal (BIAT).
f1_left	bool	<not documented yet>
block_type	string	In the case of IAT/BIAT/STIAT/SCIAT, block_type has standard names: • **tar_disc.** Target discrimination block • **att_disc.** Attribute discrimination block • **tar1att1_1.** Target1 Attribute1 combination block. Practice part. • **tar1att1_2.** Target1 Attribute1 combination block. Critical part. • **tar_rev.** Reverse target discrimination block. • **tar1att2_1.** Target1 Attribute2 combination block. Practice part. • **tar1att2_2.** Target1 Attribute2 combination block. Critical part.

MAPST

Variable	Type	Description
trial_type	string	assignment or mood
confirm_answer	bool	If TRUE, then answers on assignments need to be confirmed via a "Continue" button in order to complete a trial.
pick_level	bool	If TRUE, then each trial an assignment from one of two difficulty levels can be selected.
progress_bar	bool	If TRUE, a progress bar fills up from empty to full across the mood and assignment trials of this block.
timer	bool	If TRUE, a timer counts up in tenths of seconds during assignment trials.
category	string	Category of assignments in this trial
trial_feedback	string	Text shown to participant after completing this trial.
level_rt, initial_rt, rt	float	Response times of: • level_rt. Level response (click on difficulty level) • initial_rt. Initial response (first click on an assignment or mood answer option) • rt. Final response (answer upon completing the trial).
initial_answer, answer	string or float	For assignment trials, answer contains the text of answer option selected. For mood trials, answer contains the VAS score selected, ranging from 0 to 1.
initial_position, position	int	1, 2, 3, 4, denoting the position of the answer option selected from left to right.

level_position	string	"left" or "right", indicating whether the left or right difficulty level was selected.
correct	int	Position of correct answer option, being 1,2, 3, or 4
level	int	1, 2, or 3, denoting difficulty level
name	string	Name of current assignment
option_1, option_2, option_3, option_4	string	Texts presented in each of the four answer options
question	string	Text presented as question

NB – The variables correct, level, name, option_1-4, and question, are also present in versions prefixed with "left_" and "right_". When level selection is enabled, these variables present the corresponding assignment properties for the left and right level, one of which being selected by the participant. When level selection is disabled, the preselected level properties are prefixed "left_", with "right_" having NA values.

SPT (Semantic Priming Task)

Variable	Type	Description
randomize	bool	If TRUE, then trial order in this block is randomized.
retry	bool	If TRUE, then this trial is repeated upon an unsuccessful response
feedback	bool	Determines when to show feedback upon an unsuccessful response. • If feedback is TRUE, then success is TRUE only on a correct response • If feedback is FALSE, then success is TRUE on a correct or incorrect response
button	string	Category associated with the response (1 or 2).
prime	string	ID of prime stimulus
prime_type	string	Type of prime stimulus: "txt' or "img"
prime_content	string	Content of prime stimulus: a text for "txt"; a path to a file for "img"
target	string	ID of target stimulus
target_type	string	Type of target stimulus: "txt' or "img"
target _content	string	Content of target stimulus: a text for "txt"; a path to a file for "img"
success	bool	If success is FALSE, then a feedback slide is shown and the trial is repeated

VST (Visual Search Task)

Variable	Type	Description
block_attempt	int	No. of attempts made to complete this block. Starts counting with 0.
block_correct	int	No. of correct responses made in this block so far.
practice	bool	If TRUE, this trial is a practice trial (featuring textual feedback on correct and incorrect responses).

pos	int	Position of target stimulus. The right-most position is indexed as 0, with higher numbers being successive positions in a clockwise fashion.
stims	string	Stimuli used in each of the X positions in the trial. This string is formatted as JSON indexed array with the number of elements begin equal to the number of positions on the circle. An example value for 3 positions would be: ["some_stim", "another_stim", "yet_another_stim"].

The VST offers an experimental feature in which state variables are variables that are tracked and updated while a participant is taking part in a task. The variables can be used for setting up feedback texts using the "trial_feedback" setting in the VST. These are all of the available state variables: attempt, block, block_attempt, block_correct, block_randomization, block_trial_count, block_type, cat, correct_duration, done, fixation_duration_max, fixation_duration_min, incorrect_duration, intertrial_duration_max, intertrial_duration_min, keyboard, language, max_attempts, min_correct, mouse, participant_id, pos, practice, repeat_on_incorrect, response, results, show_touch_buttons, stim, stim0, stim1, stim2, stim3, stim4, stim5, stim6, stim7, stim_angles, stim_count, stimuli, stimuli_term, tar_1_label, tar_2_label, target_1_left, task_trial, task_trial_count, touch, trial, trial_feedback.

Change Log

- **v1.3.** Added Semantic Priming Task to main documentation.