Processing JASMIN1 Data from a LOTUS Results File

*Thomas Pronk*

[*tpronk@uva.nl*](mailto:tpronk@uva.nl)

*Version 1.1, 2016-04-14*

This is a step-by-step guide on:

1. **Decoding JASMIN1 data encoded in a LOTUS results file into a trial data CSV file.**
2. **Calculating scores from trial data**
3. **Merging scores together into a single file**

Note: This tutorial assumes you have already installed the SANDRA library, which you’ve already done if you’ve followed the steps of Tutorial *1. Installing SANDRA.docx*.

## ****Step 1. Setup the analysis framework****

The right processing scripts can be obtained from the SANDRA Github repository.

1. Download the whole repo in a ZIP via the link below. See “Download ZIP” button in the top right corner of the screen. <https://github.com/tpronk/SANDRA>
2. Unzip the file. The folder *framework\_demos*  contains a SANDRA Analysis Framework, together with the processing scripts, and some example data to test it on.
3. Move the contents of *framework\_demos* to a comfortable location on your computer, then run “Install SANDRA (from GitHub).R” to set up the framework for that location.

If everything went well, you’ll see some messages saying that various directories already exist, and finally the message:

sandra::FrameworkFileIO. Succesfully constructed FileIO

## ****Step 2. Test the processing script with the example data****

1. The *scripts* folder contains a collection of scripts for processing your data. See the page below for a brief description of what each script does: <https://github.com/tpronk/sandra/tree/master/framework_demos>
2. In general, you’ll execute a t.1, t.2, and t.3 script in order. By default, the scripts are set up to process a demo dataset named “jasmin1\_data.csv” located in the *original* folder. Run one t.1, t.2, and t.3 script, and the files listed below should be produced in the *interim* folder. Check “date modified” to find out if they were actually just created.
   * t.1 decodes JASMIN data into trial data and metadata, the latter of which contains participant parameters and some reporting variables. See glossary below.
     1. It produces a *trialdata* file (one file per task; one row per trial) and *metadata* file (one file per dataset; one row per participation). The *trialdata* and *metadata* files can be joined together on the *set\_id­* variable.
     2. You can configure *fileSource* (which file to decode) and *participationID* (which columns identify a participation).
   * t.2 calculates scores (and/or split-halve reliabilities) from trial data.
     1. In the case of scores, the script produces a *scores* file (one per task; one row per participation), but in the case of split-halve reliabilities, it only prints a single value to the R Console.
     2. You can configure *fileSource* and *scorings* (how to score a task). Type **?calculateScores**into the R console to get information about how to setup the task scoring.
   * t.3 merges scores across tasks and sessions together into a single ‘wide’, with one row per participant and columns postfixed by task and session.
     1. It produces a *joined* file (one file per dataset, one row per participant)
     2. You can configure *fileSource*, participantID (which column identifies one participant), *sessionID* (which column identifies a session), and *tasks* (which task data to combine). Finally, you can drop artefacts before joining the files together via the *dropArtefacts* function.

## Step 3. Download a LOTUS Results File

Time to download your task data into a results file. To get it in the right format, take heed of the settings below:

1. Login to LOTUS.
2. Go to *Your project 🡪 View participants 🡪 Participants 🡪 Results*.
3. Click *Export*. At the export screen:
   1. Make sure to select any participant parameters you are interested in (such as *username* or *age)*.
   2. At Separator, remove the ‘;’ and enter ‘\t’ instead.
   3. At Escape, remove the ‘\’ and just leave that field empty.
   4. Click OK.
4. Unzip the downloaded file and put it in the *original* folder.
5. Be sure to give your unzipped results file an informative name, such as *pretest\_alcohol.csv*. Note that by default, Windows Explorer does not display the *extension* of a filename (the part after the dot, which is *csv* in the example. In that case, you don’t need to enter the extension when renaming the file; just name it *pretest\_alcohol*. However, when adjusting *fileSource* in the processing script, this extension needs to be specified. To summarize, in Explorer your file could be displayed as *pretest\_alcohol*, but in the scripts you need to refer to this file as *pretest\_alcohol.csv*.

## ****Step 4. Process your own data!****

Now configure the right t.1, t.2, and t.3 scripts and process your data.

* + At the very least, in every script, you’ll need to change *fileSource* to point to the results file you just downloaded. See “Step 2” and the comments inside the processing scripts for more information about how to configure each script.
  + Note that with big datasets, t.1 may take a while.
  + Don’t forget to set up scorings for each of the tasks encoded in your data
  + The manual *TP - CBM Tasks JASMIN Configuration.docx* provides more information about the meaning of all the variables in the trial data. You can find this manual in ADAPT Shared/Documentation/CBM Tasks

## ****Step 5. Inspect the output****

SANDRA will add a couple of variables to the metadata that tell you a bit about what happened during data processing. These variables are also added to the *scores* and *joined* files. The table below explains the meaning of these variables.

**Table 1.** Explanation about metadata variables

|  |  |
| --- | --- |
| **Folder** | **Explanation** |
| lotus\_says | Tells you how the task ended. There is only trialdata produced if the task completed successfully. Possible values:   * **task\_done**. Task completed successfully * **task\_start**. Task was restarted before it completed * **task\_error**. Task reported that an error occurred |
| sequence\_report | Reports suspicious data. If this variable is not empty, then the trial data of this set are not to be trusted. Possible values:   * **inconsistent.** Events with the same sequence number but different data * **missing.** Sequence numbers missing (1,2,3,5,6,7) * **negtime.** Client time decreased with successive sequence numbers * **invalid.** Task produced invalid (or no) data, which can happen when the participant did not complete any trials. |
| taskName | The type of task belonging to this set. |