

UNIVERSITY OF BERGEN



PsychoPy II

From questionnaires to experiments

Sebastian Jentschke

UNIVERSITY OF BERGEN



Welcome to our second session about creating computerized experiments using PsychoPy.



Overview

- online experiments
- importing data from CSV
- considerations when preparing experiments:
 - randomization
 - stimuli (and how to prepare them)
 - tasks for the participants



Today, we will first discuss how to “transfer” experiments that you created locally to PsychoPy’s web platform Pavlovia.

Afterwards, I will introduce how to read the data that you produced in your questionnaire.

Finally, we will discuss several aspects of «real» experiments: randomization, stimuli (and how to prepare them) and tasks for the participants.



Online experiments

I would like to begin the lecture with making a brief introduction to online experiments with PsychoPy. This is easily done with exporting / uploading experiments from within PsychoPy to a web platform called pavlovia.org

Until recently, pavlovia was even free of charge. For now, I bought some credits that you can use.

Please use mapsyk302 as login and the password:

Or\$S0x? U}n0~wd_

As a couple of employees is interested in using PsychoPy and Pavlovia, the ISP will purchase a site licence that would permit unlimited use in the future.



PsychoPy: Online experiments

- register at pavlovia.org

Full name

Username

Email

Email confirmation

Password

Minimum length is 8 characters

Register

Pavlovia
/pævlɒviə/
noun. where behaviour is studied

PAGE 4

In any case it is smart to register for an account there. Lot's of people publicise their experiments on Pavlovia so it is also a great resource for learning by just looking at those experiments and adopting them for your own purposes.

For creating an account, go to pavlovia.org, choose "Sign in / Register", and enter the necessary details.



PsychoPy: Online experiments

- login from within PsychoPy:

trial

0 1 2 3

<https://gitlab.pavlov.org/sjentsch>

Logout

Sebastian Jentschke

Full name: Sebastian Jentschke
Organization: University of Bergen
Bio (250 chrs):

OK

Sign In/Register

Sign In Register

Username or email

Password

Remember me

[Forgot your password?](#)

Sign in

Explore Help About GitLab

PAGE 5

After created and confirmed your account via an e-mailed link, you can log in using the icon with the globe and a stylized person. In principle, you would be ready now to upload and run experiments online.

I show, how exactly to do this on the next slides.



PsychoPy: Online experiments

- submit the project to pavlovia.org:

The image illustrates the workflow for uploading a PsychoPy project to Pavlovia. It shows three main windows:

- PsychoPy Interface:** A window with a menu bar (File, Edit, View, Tools, Experiment, Demos, Pavlovia.org, Window, Help) and a toolbar. A globe icon in the toolbar is highlighted with a red box, and a red arrow points from it to the 'Create a project' dialog.
- 'Create a project' Dialog:** A small window with the message 'This file doesn't belong to any existing project.' and two buttons: 'Create a project' (highlighted with a red box) and 'Abbrechen'.
- Project Details Form:** A larger window for entering project information. Fields include:
 - Name: BriefAgitationRatingScale
 - Group/owner: sjentsch
 - Local folder: /home/sjentsch/Documents/Teaching/MAPSYK3
 - Description: Some description
 - Tags (comma separated): PsychoPy, Builder, Coder
 - Public:
 A red box highlights the 'Description' field. A red arrow points from the 'Create project on Pavlovia' button to the 'First commit' dialog.
- 'First commit' Dialog:** A window with a 'First commit' button highlighted in blue and a 'Details of changes (optional)' text area. A red arrow points from the 'First commit' button to the 'OK' button in the bottom right corner.

The University of Bergen logo is visible in the bottom right corner of the composite image.

PAGE 6

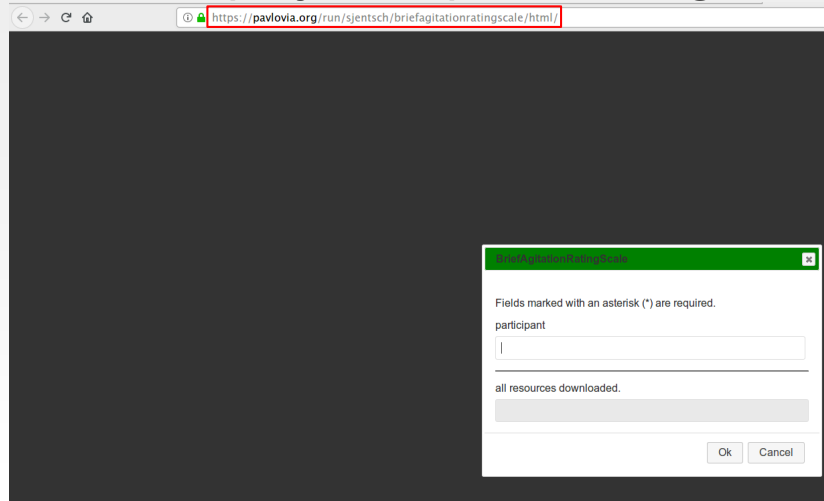
To publish and upload an experiment, you choose the icon with the globe and the running person. If it is the first time you upload that experiment, you will be first be asked to create a new project. The next window allows you to put in details (like the name of the project, a description, etc.). It also allows you to make the code of the experiment public so that other people can use it (and make yourself famous if you produced something substantial... ;-)

Then there is a last window that allows you to make a "commit". When writing software, you typically solve a certain "problem" (e.g., creating instructions) and then commit it. By doing so it is also possible to go back to earlier stages of your work if something goes wrong. For now, we don't need to care about that. You may create a comment describing what you did (that helps you remember that if you go try to understand it after a half year).



PsychoPy: Online experiments

- run the project at pavlovia.org



Tusen takk for
svarene og din
deltakelse!

Fortsett med
mellomromtasten

Once you did all this, you can run your experiment online. Typically, you would do this via a web address like:

`pavlovia.org/run/[username]/[experiment]/html`

Please note that I currently don't have credits, so the web address shown on the slide may not work right now.

Depending on what you aim to do for your M.Sc. thesis it might be worthwhile to consider this option. We are in the process to get the institute to finance a site license for UiB.



Reading data from PsychoPy

PsychoPy (and pavlovia.org) create text files that contain what your participants responded.

It is perhaps worthwhile to run your experiment a couple of times and remove the output from elements that you are not interested in (i.e., which don't have anything to do with the aim of your experiment). For example, PsychoPy – which by default stores everything – would also register how long your participants spent on reading the instructions or the thanks for participating note. You can switch off storing those information in the “Data”-tab of the respective component (Store: “Nothing”).



Reading data from PsychoPy

- «manually»:
 - PsychoPy produces Excel or CSV-files that can be read easily
 - the file contains output from the components that you asked PsychoPy to store and they should be arranged in columns
 - typically, you need to transpose them (Ctrl + Shift + V) → Transpose

PAGE 9

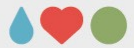


When you treat the data manually, especially the work for the first data set is a little more comprehensive: Here, you have to create the variable names, etc. whereas for later data sets you can just add the values you are interested in.

When you open the created file, you will find that the first column either contains the item code (if you used one) or the question that was asked. You typically would be interested in what participants replied and perhaps their reaction time. I would recommend that you create the variable names from the item code with adding a “_R” (for the response) and a “_T” (for the reaction time) at the end. You can use the function

=CONCATENATE(..., "_R") for that purpose

Now, you should have several columns (at least two, with the response and the variable name for it). You mark those columns and copy them (Ctrl + C).



Reading data from PsychoPy

- «manually»:
 - PsychoPy produces Excel or CSV-files that can be read easily
 - the file contains output from the components that you asked PsychoPy to store and they should be arranged in columns
 - typically, you need to transpose them (Ctrl + Shift + V) → Transpose



The contents that you just copied are going to be pasted into a new Excel-file that shall contain the of all participants. You need to choose the option that is called “Paste Special” or something similar, and there you need to choose transpose (to make the rows in which you data are currently arranged into columns). Remember that you also need to add a first column with the participant code, and that you have to copy the variable names that you created into the first line.

For the next participants, you “only” need to copy the values (i.e., responses and perhaps reaction times) and to add the participant code (it is part of the filename and written under the entry “participant” in the block underneath the heading “extraInfo” at the bottom of the Excel-file.



Reading data from PsychoPy

- using R:
 - `list.files(PATH, '*.csv')`
 - `read.csv()`
 - select columns: you need – your data (e.g., responses, reaction times), the participant code and an index (trial number)
 - concatenate them: `rbind`
 - `reshape` them and `write.csv`

PAGE 11



The alternative is to use R. This perhaps involves weighing how many files you have in relation to the effort to create such a R-script. It certainly pays off if you have to handle a large number of files.

You can find some examples how to conduct such task in the module for the lecture “From jamovi to R” (the file is called ReadData.R) and I put out another example called readPsychoPy.R in the module for the specialization. It requires that you store your data as “Save csv file (trial-by-trial)” (otherwise “participant” is not saved within each row).

The slide above describes the general steps involved if you were to create such a script yourself:

Reading each file (`read.csv`) selecting those columns you are interested in, concatenating them (adding rows; `rbind`) and finally converting the data from long (each trial as a row) into wide (trials as columns) format (using `reshape`). Finally, you write the result using `write.csv` (or `saveRDS`).



Preparing experiments

In the final part, I will give some practical hints regarding topics you should be attentive to when preparing experiments. A lot of those tips is related to maximize experimental control (i.e., to increase the internal validity of the experiment).



Randomization

- how many different conditions do you have that have to be balanced?
- stimulus type / characteristics
- if several stimuli are presented together, position
- ➔ all these conditions have to be permuted so that they appear with equal probability



The first aspect is randomization. Here, we have to weigh between using unconstrained randomization (where the order of trials and conditions is completely random) and pseudo-randomization (where we constrain certain conditions). If we, e.g., run an emotion induction experiment based upon music or film clips, we perhaps may not want to have to pieces / clips that induce the same emotion one after another. Or if we have an experiment where we have a relatively rare event that we are interested in, we would like to have those “rare” occurrences to be spaced a couple of trials apart. Doing such balancing in a good way requires perhaps skills that exceed yours and you should discuss this / obtain help from your supervisor.



Preparing stimuli

- generally: try to use validated stimuli (i.e., stimuli from published studies) → check OSF, Pavlovia
- if you create them, ensure that the stimuli are as comparable to each other as possible
 - words: word frequency, length (characters / words), semantic category, possible interactions (priming)
 - sounds: length, intensity / loudness, pleasantness
 - images: size (whole picture, main object), pixel intensity



When you use stimuli such as texts (in questionnaires), images, sounds, or video clips, you should try to go for validated stimuli. This is because: (a) those stimuli have been tested and are supposedly working well, (b) it may be easier when trying to publish your own work when you can refer to an “established” set of stimuli instead of justifying your own choice of stimuli, and (c) it typically decreases your work load. Often the publications mention where the stimuli for a particular study can be found, otherwise the study authors can be contacted and asked. Finally, there should be a lot of suitable stimuli available on web pages such as the Open Science Framework (osf.io) or Pavlovia (pavlovia.org).

If you decide to create stimuli yourself, they should be comparable to each other as possible or at least the mean for certain stimuli properties should be similar between conditions.



Preparing sound stimuli

- speech (e.g., Google TTS), sine-wave tones, instruments (<https://philharmonia.co.uk/resources/sound-samples>)
- ensure that sound files have / are:
 - equal length
 - ramps (fade-in, fade-out)
 - (RMS) normalized
- Audacity (GUI), SoX (Sound exchange; CLI)



If you create sound stimuli, ensuring that they are similar is a key requirement. Ideally, they should have the same or very similar length and for minimizing “cracks” with the transition from one stimulus to the next, they should have ramps (i.e., a period – a couple of milliseconds long – where the sound is faded in and out. In addition, the stimuli should have the same average loudness: Typically, there is an option to “normalize” sounds to achieve that and ideally choose the option to normalize it according to the sound energy (i.e., the RMS – root mean square – of the total sound energy, i.e., how much the sound makes the air vibrate).

Software to help you with those tasks is, e.g., Audacity (audacityteam.org) or SoX (sox.sourceforge.net).



Preparing visual stimuli

- try to use validated stimuli, e.g., KDEF / AKDEF
 - if you select stimuli, ensure same / similar:
 - perspective (e.g., faces from front)
 - background (e.g., neutral grey)
 - size
 - intensity / normalizing
- GIMP (GUI), ImageMagick (CLI)



The same key requirement, the stimuli being as similar as possible also applies to visual stimuli. Here it is often recommended to search for possible existing and validated stimuli (e.g., the KDEF from Karolinska Institutet showing faces expressing different emotions).

If you have to create stimuli yourself, you have to ensure that they are similar with respect to central physical properties, e.g., the background (often neutral grey or blue is a good choice), their size (the whole picture as well as the target object in it) as well as their intensity (more or less colourful), contrast and brightness. Tools such as GIMP (gimp.org) or ImageMagick (imagemagick.org) can help you with such tasks (e.g., adjusting the brightness to a particular target value). It is important that you find a good compromise between the stimuli still looking “natural” and them being comparable in their physical characteristics.



Task for your participants?

- can they do sth. else? (e.g., watch a silent movie during sound experiments)
- tasks should be (relatively) simple and not give away the aim of the experiment
- tasks should ensure that the participant follow the stimulus presentation
- sound: different instrument / voice
images: «target» image



Finally, experiments typically take a certain amount of time to conduct which may mean that you may want to ensure that (a) the participants are holding up their attention and / or (b) they are not getting bored.

The second aspect could be ensured by presenting something in a different sensory modality together with you stimuli of interest (e.g., presenting images or a silent movie while presenting sequences of sounds as main stimuli).

For the first aspect, you have to ensure that it is not diverting too much attention from the main task and that it doesn't hint what the aim of the experiment is. If the participants react to such "target" stimuli, you can take this as indication that they followed the stimuli of interest. Such targets could be, e.g., sounds that deviate a bit from the "main" sounds presented or particular target images in a series of images.



Literature

Further resources for help

Finally, I would like to point you again to the literature and where to find further resources for help. They are the same as last week, but I thought twice is better...



PsychoPy: Literature and help

- Peirce, J., & MacAskill, M. (2018). *Building Experiments in PsychoPy*. Sage Publications.
- <https://www.psychopy.org/documentation.html>
- <https://discourse.psychopy.org/>
- <https://www.youtube.com/playlist?list=PLFB5A1BE51964D587>
https://www.youtube.com/playlist?list=PL-KTa_GY7VEMehFKqnBgIlg48KqbKwSj-a
<https://www.youtube.com/playlist?list=PLuqBA9VDSXk7Z06RtJ6Gh6Y5YznVrFrK6>

PAGE 19



You can borrow the Peirce and MacAskill-book from the library or from me. Unfortunately, the library informed me that buying an electronic copy is too expensive. But, including mine, there are currently three copies (one is pensum and must stay in the library).

Therefore, you preferably may use the electronic resources and video resources listed above.

Typically, the discourse is most helpful if you run into an error or get stuck otherwise (just copy the error message in the search field of your search engine [Google]).

There is a couple of YouTube-videos showing how to create experiments (first two entries), and how to use programming for more complex experiments (last entry).



**Thank you for your
interest and your
attention!**

I hope the lecture was helpful. Thanks for your attention!