



psychopy_ext: A framework for streamlining research workflow in neuroscience and psychology

Jonas Kubilius

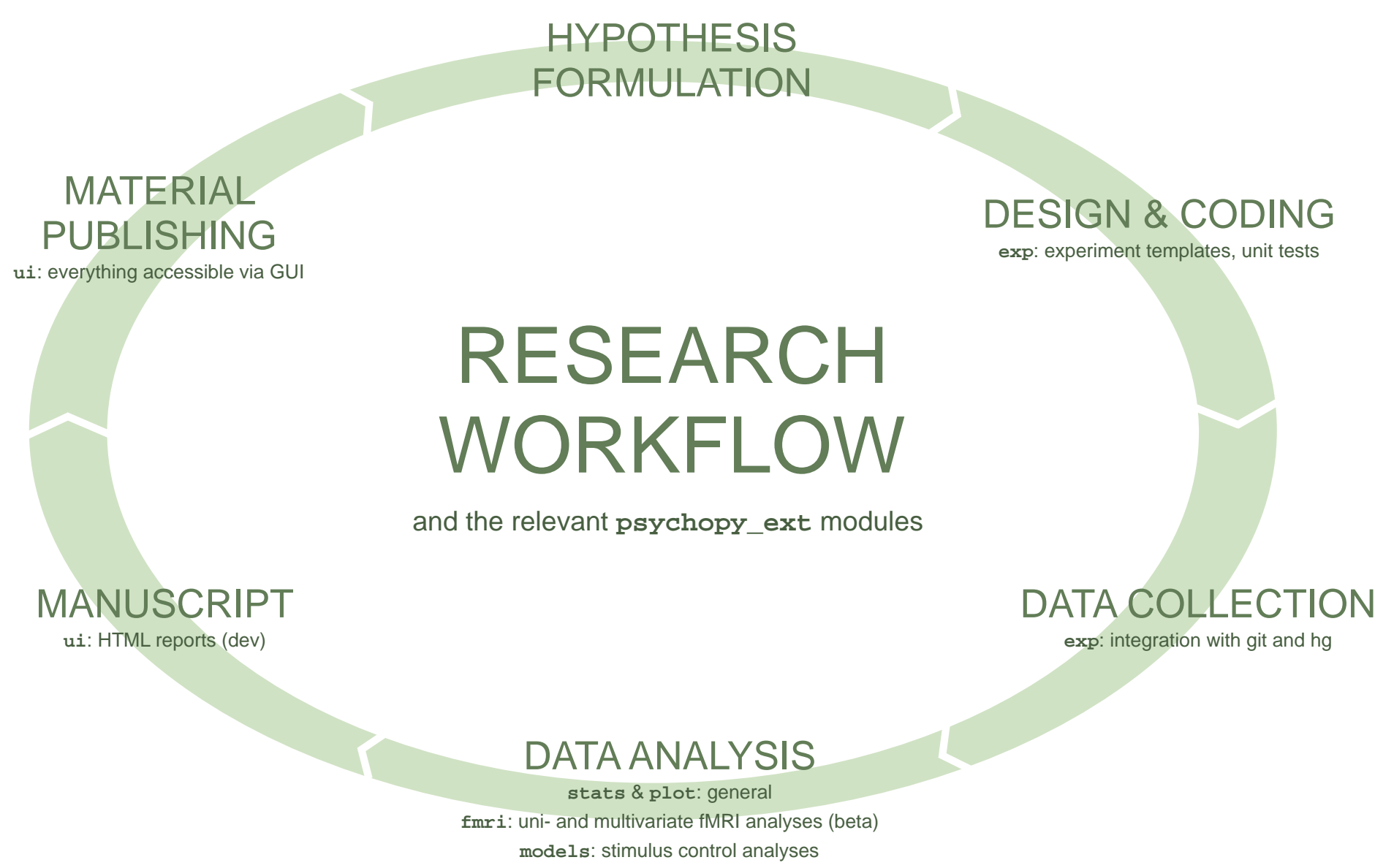
Laboratories of Biological and Experimental Psychology, KU Leuven (Belgium)

get psychopy_ext



psychopy_ext.klab.it

Streamlining research



Goals:

- Automate as much of workflow as possible
- Improve project organization and reproducibility

Implemented as a Python package that wraps:

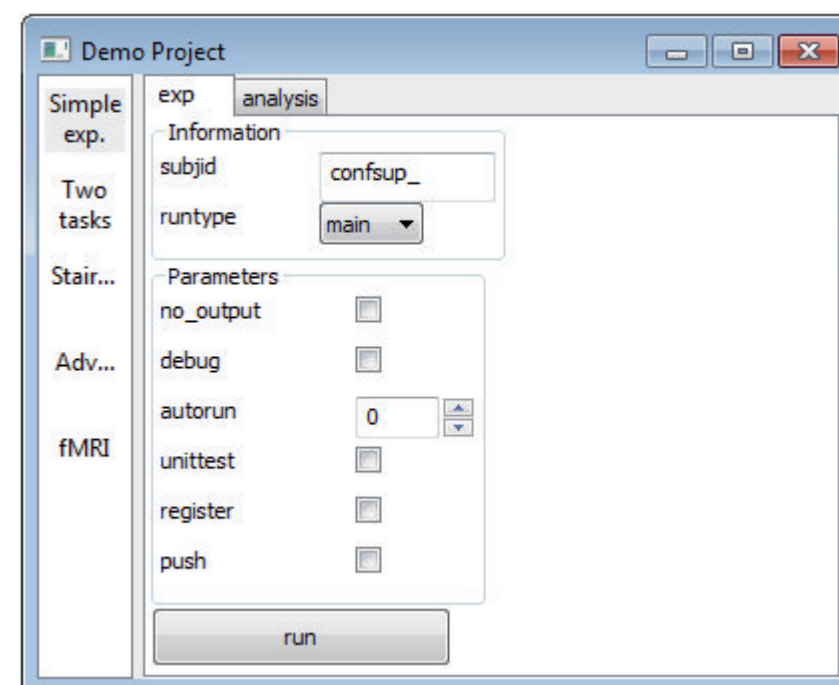
- PsychoPy
- pandas
- matplotlib / seaborn
- pymvpa2



Features

Object-oriented

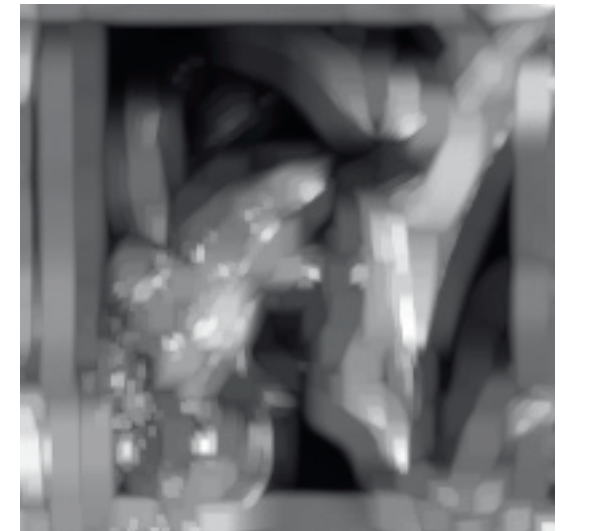
Many experiments share a common structure. Why not start from a template (a class) that you can reuse? Many useful routines are built-in: experiment loop, automatic testing, data and log handling...



Simple interface

A GUI is generated on the fly based on your project. Run and reproduce everything with ease! Command-line ninjas get their interface too.

- project
 - exp1
 - analysis
 - data
 - logs
 - exp2
 - scripts
 - __init__.py
 - computer.py
 - exp1.py
 - exp2.py
 - __init__.py
 - run.bat
 - run.py



Simulations

Are there simple confounds in your stimuli that could explain your findings? psychopy_ext has pixelwise, GaborJet, and HMAX'99 models built-in for a quick check.

Neat project organization

All project materials have a clear organization and naming conventions. No more mess.

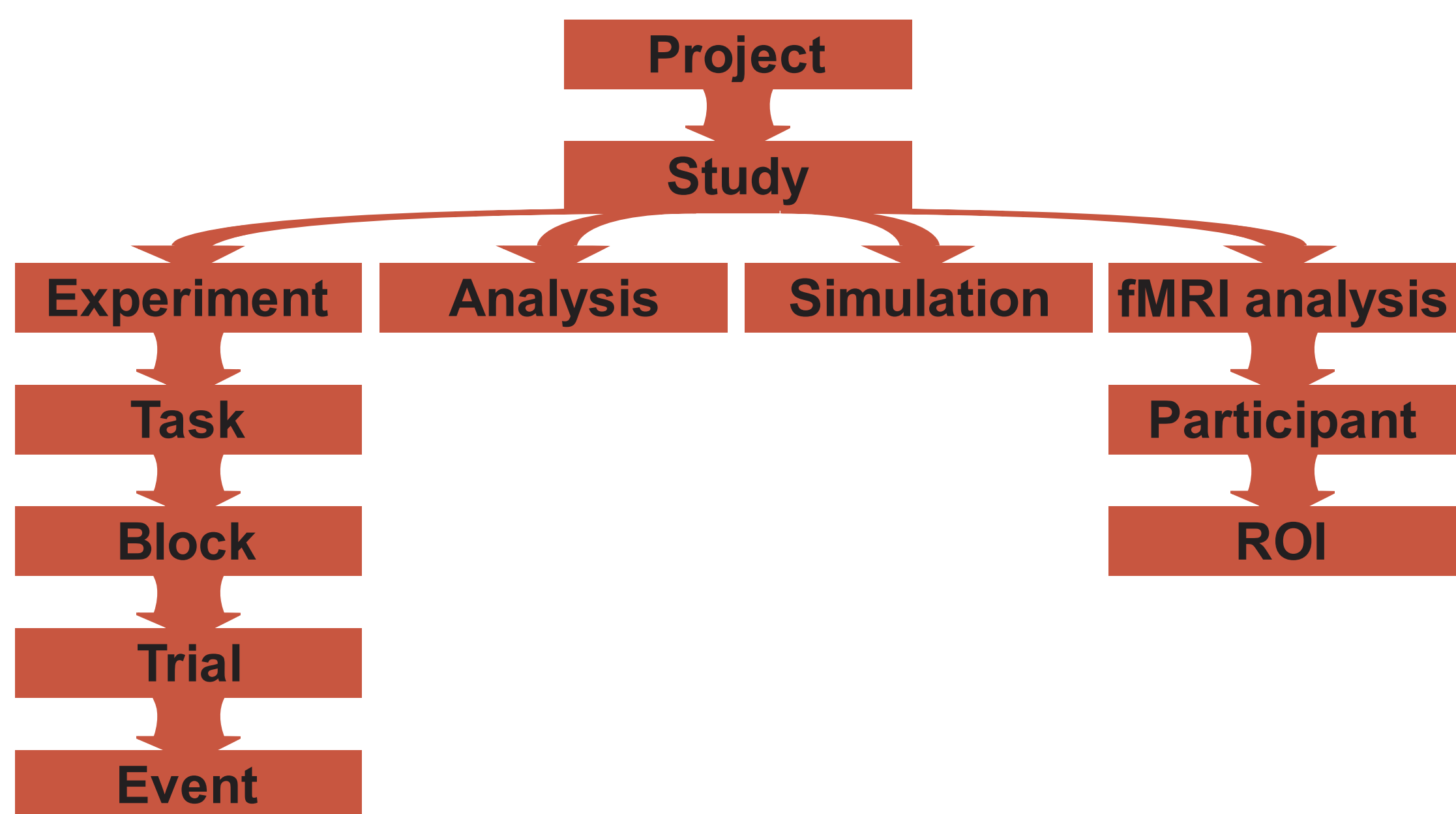
```
pattern = PATHS['data'] + '%s.csv'
df = exp.get_behav_df(self, pattern=pattern)
agg = stats.aggregate(df,
                      values='rt',
                      subplots='session',
                      cols='cond',
                      rows='kind',
                      yerr='subjid')

plt = plot.Plot()
plt.plot(agg, kind='bean')
plt.show()
```

Simple descriptive statistics

Compute accuracy across participants and plot in various formats with a couple of lines only. Nice formatting done for you by default.

Architecture



The Experiment class

Experiment:

```

setup
  get runtime info
  set seed
  set logging
  create_win
before_exp
for task in self.tasks:
  run_task
after_exp
register or push output
  
```

Task:

```

setup_task
create_stimuli
create_trial
create_exp_plan
get_blocks
before_task
for block in self.blocks:
  set_TrialHandler
  run_block
after_task
register or push output
  
```

Block:

```

set clocks
before_block
for trial in self:
  run_trial
after_block
  
```

Trial:

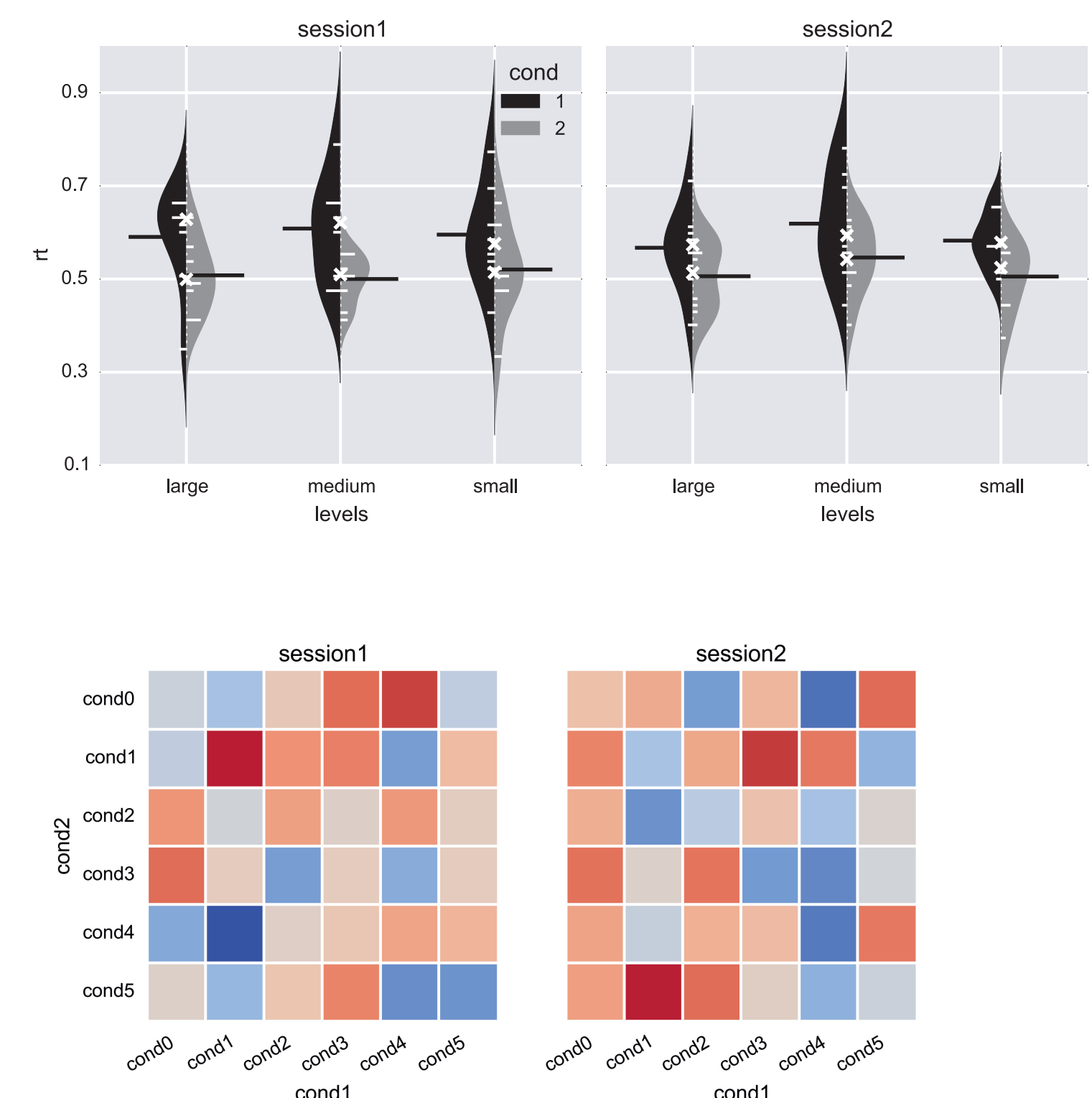
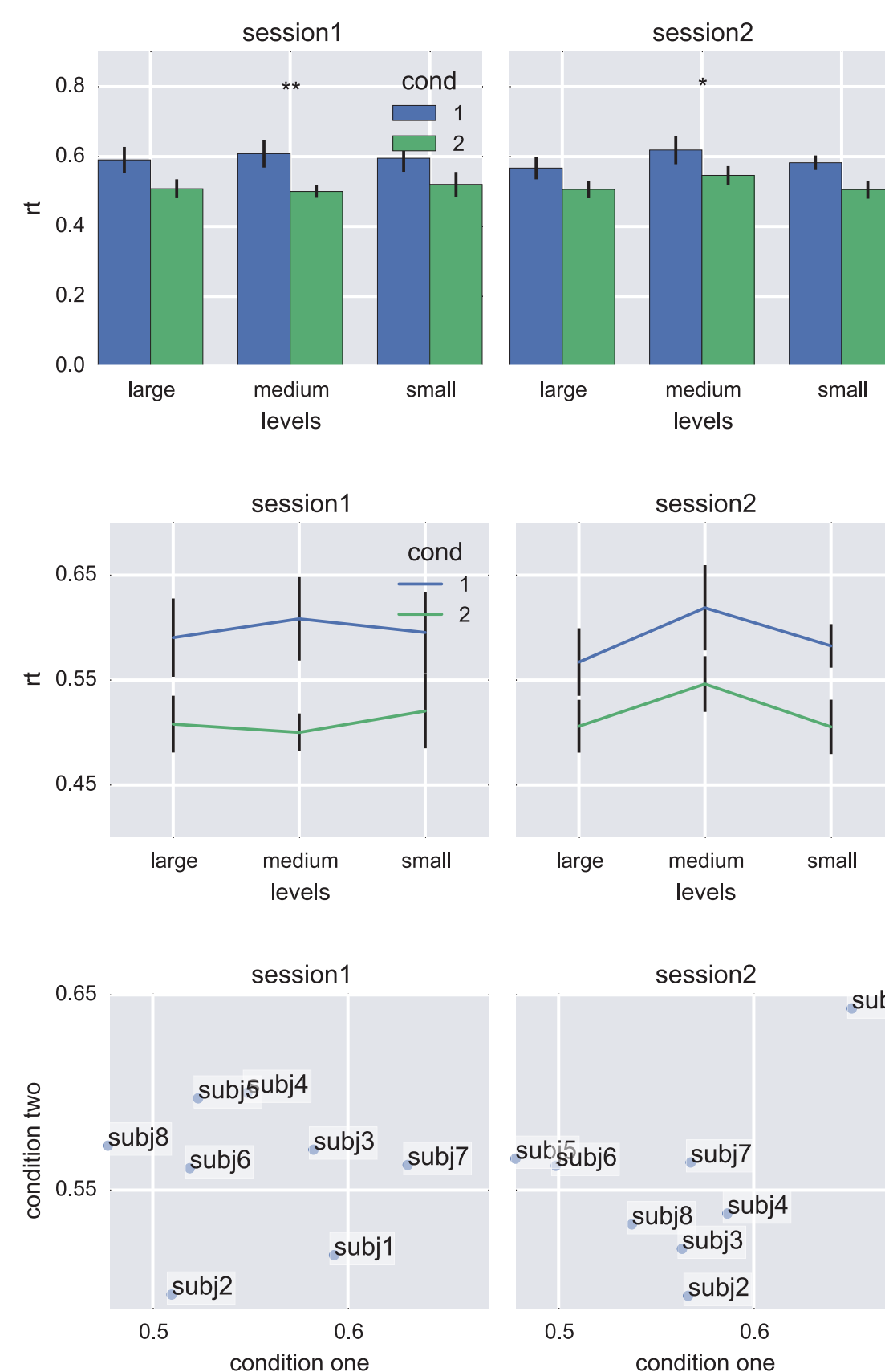
```

before_trial
for event in self.trial:
  run_event
post_trial
write to datafile
  
```

Event:

execute event function

Pretty plots

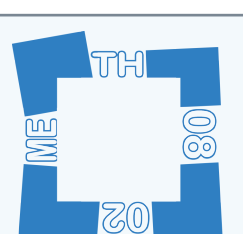


Limitations

- Presumably steeper learning curve because of object-oriented approach
- Not every experiment is possible in this framework
- fMRI analysis still in development

Future directions

- Automatic analysis report generation
- Project management tool
- GUI for experiment creation and data analysis (similar to Excel's PivotChart)
- Bayesian statistics



This work was supported by a Methusalem Grant (METH/08/02) awarded to Johan Wagemans from the Flemish Government. J. K. is a research assistant of the Research Foundation – Flanders (FWO).

jonas.kubilius@ppw.kuleuven.be | klab.It | @qbilius