Code Review Checklist

Most important before paper

Important every session

Unclear

Usability	**Comments (at least Yes / No)**
Is the code easy to install? Are there setup instructions and a list of requirements?	- Yes, it is easy to run the code. There is a description of which file to run in the README. - Set up instructions don't appear to be needed, and everything can be run from a single file. - There isn't a requirement text file that specifies the packages and their versions that the code was built on. May have implications later if packages update and cause issues.
Is there an example script or a full pipeline that is easy to run and understand?	- There isn't a specific example script, but it wouldn't make sense to have one in this case. The analysis script does that job.
Data preparation	
Are data loading and analysis implemented as separate steps? Ideal: have a data loader class	- Every step (loading and analysis) is defined within a separate function within the data utilities file.
Is ALL data used available in the cluster	- All the data need to run the analysis script is on github (also all of the output of the code is on github).
Analysis & Plotting	
Are the different steps of the analysis clearly identified in the README?	- No. The README just specifies how to run the analysis. Steps within are not described.
Does the analysis code reflect what is described in the paper? If applicable	- NA

Is it clear what code is used to create each of the figures or panels in the paper?	- Yes. There is one file that plots all the data. However the file is long and it is slightly difficult to follow exactly what and when each plot is created.
Code quality	
Project is periodically updated in github, gitignore, README	- Difficult to tell. Seems like this repository was fairly recently created. - Every commit name is called “update” - may cause issues when trying to find a specific change in the future.
Project structure: folders: data, notebooks, scripts, figures	- Structure is intuitive. There are separate folders for data, figures, models, results.
Is the code well organized (functions, classes, modules, settings, ... as applicable)?	- There is an organisation of functions, which can be found in their own file. But there is a lack of organisation for other aspects of the code. For example, the settings used within the models are typically coded within either the analysis or the functions files. Maybe cleaner to have a separate file that contains these parameter values.
Are all functions and classes documented?	- It’s is fairly limited. Some functions have docstrings (but never input output, or explicit description of what the function is doing), but this isn’t for every function. Most functions are commented though, so there are some guides to how the code works.
Are some values hardcoded?	- Yes. Not just parameters, but some simpler things like the number of participants or sessions, that could be extracted

	from the data.
Can any of the code be replaced by existing packages?	- Have no idea!!!!
Are there any obvious optimisations that will improve performance?	
Is there any redundant code that should be removed?	- There seems to be some repeated loops, but unsure if this was needed for a specific reason.
Does the code agree with basic style guidelines (e.g. PEP8 for Python)	
Variables names are self explanatory (eg no a, b, c etc)	- Variable names seem to be well described
Are there any passwords in the repo or exposed in the code?	- Not that we can see
Is any identifying information unwillingly exposed?	- Other than it being her github, no.