|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MRI | | | |  |  |
| Name | | | Description | Folder | Type |
| Preprocessing | | |  |  |  |
|  | LEMO\_task\_preprocessing.R | | *….* |  | P |
|  | | | |  |  |
| Functions for preprocessing | | |  |  | F |
|  |  | | … |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  | | | |  |  |
| MRI analysis | | |  |  |  |
|  | First level | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
| *Functions for analysis* | | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
| Region of interest analysis | | |  |  |  |
|  | | LEMO\_create\_MarsbarROIs\_niftis.m | You input a set of ROI names and coordinates . It will use Marsbar functions ‘maroi\_sphere’ and ‘saveroi’ to create a sphere and save the coordinates in a .mat. Then ittransform this into the correct space of your study using ‘maroi’ function from Marsbar (for this it needs to read any of your preprocessed, corregistered and normalized epis). It saves it as nifti using ‘mars\_rois2img’ function from marsbars. |  |  |
|  | | LEMO\_ROI\_analysis.m | It calls function ‘LEMO\_func\_extract\_rois.m’ . Allows to specify the type of summary statistic from the betas that the function will generate (mean, median or eigenvariate). Reads the voxels of interest generated from Marsbar (or any other niftis with that information) . Those voxels will be input in to the function. Saves the coordinates read in a table with the subjects and the output values from the function (see function details below) |  |  |
|  | | LEMO\_gather\_betas2analyze.R | Gathers beta value for the different contrasts into a data set that can be later used for statistical analysis (e.g., t-tests). It saves large SPSS data sets (wide and long format). |  |  |
|  | | LEMO\_roi\_t-tests.R | Simply runs t-tests (against zero) for each column of the gathered rois , saves results in a large table with variables as rows and statistic values as columns. Additional output: ‘tidy’ table in cleaner format for copy pasting results in a report. |  |  |
|  | | LEMO\_roi\_lmm.R | Compares third 1 and third 3 for stimuli and feedback formula\_fixed <- as.formula(paste0(dependentvariable,'~ task\*third\*block') )in each ROI separately.  Uses dataset in long format. Calls function for linear mixed models excluding outliers. Additional output: ‘tidy’ table in cleaner format for copy pasting results in a report. |  |  |
|  | |  |  |  |  |
| *Functions for ROI analysis* | | |  |  |  |
|  | | LEMO\_func\_extract\_rois.m | Reads the voxels of interest from the nifti with the ROI (with spm\_read\_vols). Then uses spm\_get\_data to read to obtain the beta in the epi’s voxels of interest. Provides different summary statistics options: eigenvariate, median or mean. |  |  |
|  | | Function\_lmm\_exclOutliers\_lite.R | Runs a linear mixed model and iteratively excludes outliers based on residuals. Needs formulas for fixed and random effects, a threshold for the outliers and a dataset in long format. It will save information related to the exclusion (rows, percentage of data, iterations, etc) as well as the model fit object |  |  |
|  | |  |  |  |  |
|  | |  |  |  |  |
|  | |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Regions of interest analysis | | |  |  |
| Name | | Description | Folder | Type |
| Region of interest analysis | |  |  |  |
|  | [0]LEMO\_create\_MarsbarROIs\_niftis.m | You input a set of ROI names and coordinates . It will use Marsbar functions ‘maroi\_sphere’ and ‘saveroi’ to create a sphere and save the coordinates in a .mat. Then ittransform this into the correct space of your study using ‘maroi’ function from Marsbar (for this it needs to read any of your preprocessed, corregistered and normalized epis). It saves it as nifti using ‘mars\_rois2img’ function from marsbars. |  | P |
|  | [1]LEMO\_ROI\_analysis.m | It calls function ‘LEMO\_func\_extract\_rois.m’ . Allows to specify the type of summary statistic from the betas that the function will generate (mean, median or eigenvariate). Reads the voxels of interest generated from Marsbar (or any other niftis with that information) . Those voxels will be input in to the function. Saves the coordinates read in a table with the subjects and the output values from the function (see function details below) |  | P |
|  | [2]LEMO\_gather\_betas2analyze.R | Gathers beta value for the different contrasts into a data set that can be later used for statistical analysis (e.g., t-tests). It saves large SPSS data sets (wide and long format). |  | S |
|  | [3]LEMO\_roi\_t-tests.R | Simply runs t-tests (against zero) for each column of the gathered rois , saves results in a large table with variables as rows and statistic values as columns. Additional output: ‘tidy’ table in cleaner format for copy pasting results in a report. |  | P |
|  | [4]LEMO\_roi\_lmm.R | Compares third 1 and third 3 for stimuli and feedback formula\_fixed <- as.formula(paste0(dependentvariable,'~ task\*third\*block') )in each ROI separately.  Uses dataset in long format. Calls function for linear mixed models excluding outliers. Additional output: ‘tidy’ table in cleaner format for copy pasting results in a report. |  | P |
|  |  |  |  |  |
| *Functions for ROI analysis* | |  |  |  |
|  | LEMO\_func\_extract\_rois.m | Reads the voxels of interest from the nifti with the ROI (with spm\_read\_vols). Then uses spm\_get\_data to read to obtain the beta in the epi’s voxels of interest. Provides different summary statistics options: eigenvariate, median or mean. |  |  |
|  | Function\_lmm\_exclOutliers\_lite.R | Runs a linear mixed model and iteratively excludes outliers based on residuals. Needs formulas for fixed and random effects, a threshold for the outliers and a dataset in long format. It will save information related to the exclusion (rows, percentage of data, iterations, etc) as well as the model fit object |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Summary of fMRI results | | | | | |
| Name | | | Description | Folder | Type |
| Explore and summarize MRI results | | |  |  |  |
|  | [1] SavePictures\_Results\_SPM\_sliceOverlay.m | | This is used to explore results. Big part of this was found online. Find your 2nd level results, loop through contrasts and saves a table with results (uncorrected p) , a figure with the glass brain and several pictures with slices from the 3 planes. |  | S |
|  | .SavePictures\_Results\_SPM\_sliceOverlay\_regreModels.m | | The same as *SavePictures\_Results\_SPM\_sliceOverlay.m* but adjusted for the folders with 2nd level regressors. There were many subfolders to explore different behavioral scores so a loop was included. |  | S |
|  | [2] SaveResults\_summaryTable\_clusterThresh.R | | It reads the tables grenerated in *SavePictures\_Results\_SPM\_sliceOverlay.m.* Leaves clusters where the FPW correct p < 0.05 and saves the results in a simplified and trimmed xlsx file that can be used in a reported. This should be equivalent to manually doing the cluster extension threshold when viewing the results in SPM. |  | S |
|  | .SaveResults\_summaryTable\_clusterThresh\_regreModels.R | | Same as above but adjusted for the folders with 2nd level regressors. There were many subfolders to explore different behavioral scores so a loop was included. |  |  |
|  | [3] Search\_MRIcoordLabels.R | | Begins with a merger that will put together all tables from *SaveResults\_summaryTable\_clusterThresh.R.* (this is optional if they have not been combined).  The main script uses a function found online (github "yunshiuan/label4MRI"). Reads table with xyz coordinates*.* It saves the corresponding MNI anatomical labels of those coordinates . If no label is found for the coordinates it will look for the nearest one and save the distance. It generates a report-ready table with the info from the table read + the newly added anatomical labels |  | S |
|  | [4] pyScriptWriter\_feedlearn.R | psyScriptWriter\_\* | | These scripts are a bit cheap and dirty way of writing the python scripts for the visualizations in MRICROGL . They read the tables above and select some data based on your thresholds and then writes line-by-line the python scrpit that can be read by mricroGL. Specifically, it will place the coordinates x y z to create multiple plots of each effect in the table. The texts written in the variable ‘template’ are the ones written in a text .py file with ‘write\_lines’ |  | S |
| **Plot ROI beta values** | | | | | |
|  | | LEMO\_plot\_roi\_tests.R | Bar plot of ROI betas. Reads them from the 2nd level and loops thru contrasts. Saves a combined plot with face\_wrap grouping ROIs into sub groups. But also saves individual plots (one per ROI). Contrasts are the discrete variable in the x axis . |  | S |
|  | | LEMO\_plot\_roi\_tests\_FBLdiff.R | Similar to *LEMO\_plot\_roi\_tests.R* but it does read the data from both tasks (A and B) , calculates and plots the difference between them. Output is one plot per ROI with A-B and Betas for A and B in different panels. |  | S |
|  | |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PCs used in STAN models | | | | | |
| **computer-id** | **R version** | **Rstan version** | **Cmdstan version** | **Memory options on script** | **Notes** |
| KJPDPCNA92813.d.uzh.ch  (G.FragaGonzalez) | 4.0.2 | 2.21.2 | 0.2.2 | options(buildtools.check = NULL)  options(mc.cores = parallel::detectCores())  rstan\_options(auto\_write = TRUE)  memory.limit(9999999999)  memory.size(max = TRUE) | Cmdstan generates the chains (very large csv files). This PC was used to generate the chains. The chains are put in a list and read by rstan functions to make a stan fit object (rstan::sflist2stanfit). This often causes memory problems and crashes. |
| KJPDPCNA92828.d.uzh.ch | 4.1.2 | 2.21.3 | 0.4.0 | options(buildtools.check = NULL)  options(mc.cores = parallel::detectCores())  rstan\_options(auto\_write = TRUE)  memory.limit(9999999999)  memory.size(max = TRUE) | This PC was optimized for heavy computing (but not it has 32 Gb ram as the one above) |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Correlations | | | |  |  |
| Name | | | Description | Folder | Type |
| Cognitive tests – task performance correlations | | |  |  |  |
|  | LEMO\_corre\_excludeOutliers\_run.R | | *….* |  |  |
|  | | | |  |  |
| Functions for correlations | | |  |  |  |
|  | LEMO\_func\_gather\_corrs.R | | … |  |  |
|  | |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task performance (basic RT and accuracy analysis) | | | |  |  |
| Name | | | Description | Folder | Type |
| Gather performance | | |  |  |  |
|  | LEMO\_performance\_gather.R | | Goes through the files in the MRI preprocessing where it should find a copy of the log files from the task. It will gather accuracy (proportion of hits), and mean RTs per quartile and third of trials. It runs per task (you need to manually input the task). It will save .csv tables with (1) cumulative probabilities , (2) gathered performance in wide format (includes some additional calculation of mean across blocks and thirds ) and (3) gathered performance in long format. |  | S |
|  | LEMO\_performance\_gatheredMerge.R | | Used to merge the tables from the script *LEMO\_performance\_gather.R.* It also reads the master file’s subjet variable to create temporary wide formatted tables.  Main output: merged longitudinal performance table into a SPSS .sav file for analysis  Temporary output: Wide formatted data sets are merged into an .xlsx file ‘tmp\_merged.xlsx’ A version with colored background in the columns ‘tmp\_merged\_color.xlsx’ is saved. These are just temporal table to copy+paste in the master file. I tend to delete them after copying the content in my master (.xlsx) file. |  | S |
|  | LEMO\_merge\_masterSheets.R | | Simple merger to combine different sheets of the master .xls file (wide formatted ) into an SPSS wide formatted data set that can be used for analysis (e.g., correlations between task and cognitive tests). It merges the sheet with Demographics (which also contains filter variables), cognitive (cognitive tests) and task\_FBL (with the learning task). The output is called LEMO\_cogni\_fbl.sav to indicate it has cognitive tests and fbl | File Management | S |
| Task performance statistics | | |  |  |  |
|  | LEMO\_performance\_descriptives.R | | Not essential. Just used to make some simple tables of descriptive statistics |  |  |
|  |  | |  |  |  |
|  | | | |  |  |
|  | | |  |  |  |
|  |  | | … |  |  |
|  | |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BEHAVIORAL MODELS | | |  |  |
| Name | | Description | Folder | Type |
| Data preparation | |  |  |  |
|  | LEMO\_task\_preprocessing.R | Reads .txt/log files from learning task (selection based on master or based on logs in MRI 1st level ) . Concatenates all text files and preprocess: trim trials with no response, RT thresholded, column selection and coding new ones required for Stan models. Output is gathered data as .txt file . |  |  |
|  |  | |  |  |
| Run models (cmdStan package; saving chains and loglik) | | |  |  |
|  | LEMO\_model\_run\_rlddm.R | Runs RLDDM Stan models using ‘cmdstanr’ package. Extremely time consuming. Allows choosing which model to call, set initial values list, select sampling parameters: n chains, iterations etc. Once the fit is computed it saves log likelihood for later use in model comparison. Outputs: model fit (very large files), log likehood , copy of the script run with date atnd time as filename. |  |  |
|  | LEMO\_model\_run\_RWdelta.R | Runs Rescorla-Wagner (delta) model from HbayesDM 2. This is the model for the arm bandit 2 choice task. Allows setting up initial values and sampling parameters, saves fit and loglikelihood. + It modifies the task preprocessing and saves the resulting table in the output dir. The input required is Tsubj , choice (1,2) and outcome (-1,1) .They are obtained from iter, response and fb variables in the preprocessed table.  https://www.rdocumentation.org/packages/hBayesDM/versions/0.2.1/topics/bandit2arm |  |  |
|  | | |  |  |
| Functions: STAN models (Stan folder) | |  |  |  |
|  | LEMO\_rlddm\_v11.stan | Reinforcement learning drift diffusion model (rlddm)  V11 = 1 exponential decision boundary modifier (a\_mod) and 1 learning rate |  |  |
|  | LEMO\_rlddm\_v12.stan | V12 = 1 exponential decision boundary modifier and 2 learning rates |  |  |
|  | LEMO\_rlddm\_v21.stan | V21 = 1 linear decision boundary modifier and 1 learning rate |  |  |
|  | LEMO\_rlddm\_v22.stan | V22 = 1 linear decision boundary modifier and 2 learning rates |  |  |
|  | LEMO\_rlddm\_v31.stan | V31 = 1 no decision boundary modifier and 1 learning rate |  |  |
|  | LEMO\_rlddm\_v32.stan | V32 = 1 no decision boundary modifier and 2 learning rates |  |  |
|  |  |  |  |  |
|  | LEMO\_bandit2arm\_delta.stan | Simple reinforcement learning model from hBayesDM |  |  |
|  | | |  |  |
| Gather outputs | |  |  |  |
|  | LEMO\_gather\_modelOutput | Calls *LEMO\_func\_gatherOutput.R*  Gathers the relevant information from the model parameters for statistical analysis. Saves plot for model fit assessment. Depending on the model different paremeters will be collected. Time consuming as it has to load the model fit (very large files) |  |  |
|  | *LEMO\_func\_gatherOutput.R* | Function called by gather output script (…) |  |  |
|  |  |  |  |  |