gPPI TOOLBOX
Version 7.12

A Generalized Form of Context-Dependent Psychophysiological Interactions

DONALD G MCLAREN, PhD

Contact Info:
University of Wisconsin – Madison,
Massachusetts General Hospital,
ENRM Veteran's Hospital, and
Harvard Medical School
mclaren.donald@gmail.com

Functional MRI allows one to study task-related regional responses and task dependent connectivity analysis using psychophysiological interaction (PPI) methods. The latter affords the additional opportunity to understand how brain regions interact in a task-related manner. In other words, a PPI analysis assesses how the activity within brain networks is modulated by varying psychological states within an fMRI task. The current implementation of PPI in Statistical Parametric Mapping (SPM8) is primarily set up to assess connectivity differences between two task conditions, when in practice fMRI tasks frequently employ more than two conditions. Thus, we evaluated how a generalized form of context-dependent PPI (gPPI), which is configured to automatically accommodate more than two task conditions in the same PPI model by spanning the entire experimental space, compares to the standard implementation in SPM8, using both simulations and an empirical dataset. In the simulated dataset, we compare the interaction beta estimates to their expected values and model fit using the Akaike Information Criterion (AIC). We found that interaction beta estimates in gPPI were robust to different simulated data models, were not different from the expected beta value, and had better model fits than when using standard PPI (sPPI) methods. In the empirical dataset, we compare the model fit of the gPPI approach to sPPI. We found that the gPPI approach improved model fit compared to sPPI. There were several regions that became non-significant with gPPI. These regions all showed significantly better model fits with gPPI. Also, there were several regions where task-dependent connectivity was only detected using gPPI methods, also with improved model fit. Regions that were detected with all methods had more similar model fits. These results suggest that gPPI may have greater sensitivity and specificity than standard implementation in SPM. This notion is tempered slightly as there is no gold standard; however, data simulations with a known outcome support our conclusions about gPPI. In sum, the generalized form of context-dependent PPI approach has increased flexibility of statistical modeling, and potentially improves model fit, specificity to true negative findings, and sensitivity to true positive findings.

Importantly, the scripts are covered by their own license (utility_license.m).

Table Of Contents

1. **DOWNLOAD**
   The peak_nii toolbox can be downloaded from: www.nitrc.org/progjects/gppi
   The file will be in the form: PPPI*.tar

2. **INSTALLATION**
   Untar the downloaded file and move the files to the toolbox directory in SPM8. This will create a folder called pathtoSPM8/toolbox/PPPI.
   In MATLAB, make sure pathtoSPM8/toolbox/PPPI is in the search path.
   This can be set by using the addpath command or from File→ Set Path.
   The toolbox also requires SPM8 to be in the MATLAB path.

3. **TESTING TOOLBOX INSTALLATION**
   MAC/LINUX
   (1) Download running_gPPI file from NITRC
   (2) Uncompress the file.
   (3) Open a shell terminal, cd to *running_gPPI_generic*, then prepare the test set by typing the following at the prompt: *./prepareToRunOnSampleData.sh*
   (4) Launch MATLAB
      a. Add SPM8 to the MATLAB search path
      b. Add PPPI to the MATLAB search path and its parent directory
      c. cd to *running_gPPI_generic*
      d. To run gPPI on the test data, enter the following at the command prompt: *RunOnSampleData*
   (5) Check the output against README.txt file in running_gPPI against your output. They should be the same, except for the PPPI version line and lines with directory names.
   (6) Check your output files in (*PPI_rmedPrec_testsample*) against the files in (*sampleData/stats/PPI_rmedPrec*). These should also be the same.

   <mark>WINDOWS</mark>
   (1) Get running_gPPI from NITRC
   (2) Uncompress the file.
   (3) Change the filename *RunOnSampleData.m.Windows.Template* to *RunOnSampleData.m*
   (4) Launch MATLAB
      a. Add SPM8 to the MATLAB search path
      b. Add PPPI to the MATLAB search path and its parent directory
      c. cd to *running_gPPI_generic*
      d. To run gPPI on the test data, enter the following at the command prompt: *RunOnSampleData*
   (5) Check the output against README.txt file in running_gPPI_generic against your output. They should be the same, except for the PPPI version line and lines with directory names.
   (6) Check your output files in (*PPI_rmedPrec_testsample*) against the files in (*sampleData/stats/PPI_rmedPrec*). These should also be the same.

   If you get any error messages OR the files do not match, then the installation is not correct. Please follow the instructions again or contact mclaren.donald@gmail.com.

4. **PPPI**
   PPPI is a command line utility aimed at automating gPPI and PPI analyses. The command will extract the seed region(s), form the regressors, build the design matrix, and estimate the first level PPI contrasts.

   4.1. USAGE
   PPPI('parameters.mat')
   PPPI(parameters,structfile)
   PPPI(parameters,structfile,tsdata)
   PPPI('parameters.mat',[],tsdata)

4.2. INPUTS

'parameters.mat' --   A mat-file containing the parameters variable (see section 3.2.1)
parameters –   A structure variable/file specifying how PPI should be conducted
structfile --   A string that contains the output for the structure file
tsdata --   A variable that contains the time-series data of the ROI.

4.2.1. PARAMETERS (FIELD LIST)

Required Options:

subject --   A string with the subject number
directory --   Either a string with the path to the first-level SPM.mat directory, or if you are only estimating a PPI model, then path to the first-level PPI directory.
VOI --   Either a string with a filename and path OR a structure variable (details will be in a future version of the manual) defining the seed region. The file should be a VOI.mat file or an image file of the ROI (see create_sphere_image package).
Region --   A string containing the basename of output file(s), if doing physiophysiological interaction, then two names separated by a space are needed.
analysis --   Specifies psychophysiological interaction (*'psy'*); physiophysiological interaction (*'phys'*); or psychophysiophysiological interactions (*'psyphy'*).
method --   Specifies traditional SPM PPI (*'trad'*) or generalized condition-specific PPI (*'cond'*). It is recommend that the *'cond'* approach is always selected (see McLaren et al. 2012 in NeuroImage for details).

Optional Fields:

extract --   Specifies the method of ROI extraction, eigenvariate (*'eig'*) or mean (*'mean'*). Default is *'eig'*.
Contrast --   Contrast to adjust for. Adjustments remove the effect of the null space of the contrast. Set to 0 for no adjustment. Set to a number, if you know the contrast number. Set to a contrast name, if you know the name.
The default is: 'Omnibus F-test for PPI Analyses'.
equalroi --   Specifies the ROIs must be the same size in all subjects. *Default=1 (true); set to 0 to lift the restriction.*
FLmask --   Specifies that the ROI should be restricted using the mask.img from the first-level statistics. *Default=0, 1 means to use the mask.img from the 1st level model*.
VOI2 --   Either a string with a filename and path OR a structure variable (details will be added in future manual version for details) defining the second seed region for physiophysiological interactions.
Weights --   For traditional PPI, you must specify weight vector for each task.
Tasks --   In the generalized condition-specific PPI, you should specify the tasks to include in the analyses, but put *'0'* or *'1'* in front of them to specify if they must exist in all sessions. For the '*trad'* approach the task must appear in all runs to make the proper, so the number should not be input first. For the *'cond'* approach task has to occur in at least 1 run, which is why you have the option. This field should be entered as a cell array.
Estimate --   Specifies whether or not to estimate the PPI design. 1 means to esimate the design, 2 means to estimate the design from already created regressors (must be of the OUT structure), 0 means not to estimate. *Default is set to 1, so it will estimate the model.*
CompContrasts --   0 not to estimate any contrasts; 1 to estimate contrasts; 2 to only use PPI txt file for 1st level (not recommended); 3 to only use PPI txt file for 1st level and estimate contrasts (not recommended); 2&3 are not recommended as they potentially do not include all tasks effects in the mode. Use at your own risk. 3 cannot weight the contrasts based on the number of trials. *Default is 0*.
Contrasts --   cell array of tasks to create contrasts to evaluate OR it is a structure variable

(see 3.2.2.). If left blank and *CompContrasts=1*, then it defines all possible T contrasts for task components and across runs. This is only feasible with less than four tasks.

Weighted -- Default is not to weight tasks by number of trials (0); to change this, specify which tasks should be weighted by trials. If you want to weight trials, then specify a duration longer than your events. If you have a mixed block event related design, then you can average your events based on number of trials and the blocks won't be averaged IF Weighted is set to be a number that is shorter than the block duration and longer than your events.

SPMver -- SPM version used to create SPM.mat files at the first level.

maskdir -- location to store ROI mask files.

outdir -- This is the name of the output directory if you want to store the PPI analysis in a different location than the first-level SPM.mat file.

GroupDir -- This is the location you want to copy the con_ files to for easier group analyses.

ConcatR -- Under development, but can be used to concat sessions to reduce collineaity between task and PPI regressors.

preservevarcorr -- preserves the variance correction estimated from the first level model. This will save time and also means all regions will have the same correction applied.

4.2.2. CONTRASTS (FIELD LIST)

A contrast is defined based on a null hypothesis. For example, the null hypothesis is that task1=task2. The tasks on the left side, go into the left field and the ones on the right side go into the right field. This setup will show positive values for task1>task2 (e.g. left side greater than right side) and negative values for task1<task2 (e.g. right side greater the left side). SPM will only show positive values.

FIELDS:

left -- a cell array with tasks on left side of equation or 'none'

right -- a cell array with tasks on right side of equation or 'none'

Weighted -- either specified or from Weighted above. If not defined, *defaults to 0*.

STAT-- 'T' or 'F'

c -- contrast vector from createVec, automatically generated

name -- name of contrast, will be defined by task list if left blank. *NOTE: Windows users need to define this field as automatic names may be longer than allowed by Windows.*

Prefix -- prefix to the task name (optional), can be used to select each run (e.g. *'Sn(1)'*)

Contrail -- suffix after task name (e.g. parametric modulators, different basis function)

MinEvents -- The minimum number of events needed to compute the contrast. This is required. This is a number.

MinEventsPer -- The minimum number of events per task needed to compute the contrast. This is a number. Default is MinEvents/NumberOfTasks on each side of the contrast.

** *If you want to define the contrast by hand, then enter the contrast as a matrix in the left field and leave the right field empty.*

4.3.1. Common Issues (will be expanded as more issues occur)

(1) PPI Model Estimation fails because dataset is too large. If your data is larger than about 2GB, then PPI model estimation will fail. Please use the alternative version of spm_estimate_PPI.m

(2) VOI is larger than dataset. This is caused by the VOI going beyond the first level mask. The solution is to multiple your VOI by the group mask as this will constrain the VOI to only voxels in all subjects.

(3) Invalid Contrast Warning with Omnibus F-test. This is normal as it is checking all basis functions. Since bf(2) and bf(3) do not exist in most datasets, you will see this warning, but the program will continue.

5. Tutorials

    5.1. If doing these as part of a class/workshop, then steps 1-5 before the class/workshop

    5.2. Attention Dataset

        (1) Download the Attention Dataset from the SPM website (currently: http://www.fil.ion.ucl.ac.uk/spm/data/attention/)

            a. attention.zip

            b. ppi_spm_batch.m

        (2) Unzip attention.zip

        (3) Launch MATLAB

        (4) Add SPM8 and PPPI directories to the MATLAB search path (described above)

        (5) Run *ppi_spm_batch.m* (this will create the 1st level analysis and traditional PPI models)

        (6) cd to GLM directory

        (7) Steps 8-10 can also be found in *AttentionMATLABPrep.txt*

        (8) Traditional PPI

            a. Configure the PPPI Parameter structure

                i. P.subject='Attention_Tutorial';

                ii. P.directory=pwd;

                iii. P.VOI=[pwd filesep 'VOI_V2_1.mat'];

                iv. P.Region='VOI_V2_1';

                v. P.Estimate=1;

                vi. P.contrast=0;

                vii. P.extract='eig';

                viii. P.Tasks={'No-Attention' 'Attention'};

                ix. P.Weights=[-1 1];

                x. P.analysis='psy';

                xi. P.method='trad';

                xii. P.CompContrasts=0;

                xiii. P.Weighted=0;

            b. Run PPPI with the command: *PPPI(P,'attentiontutorial.mat')*

        (9) gPPI (VOI.mat file)

            a. Configure the PPPI Parameter structure

                i. P.subject='Attention_Tutorial';

                ii. P.directory=pwd;

                iii. P.VOI=[pwd filesep 'VOI_V2_1.mat'];

                iv. P.Region='VOI_V2_1_gPPI_Attention';

                v. P.Estimate=1;

                vi. P.contrast=0;

                vii. P.extract='eig';

                viii. P.Tasks={'1' 'Stationary' 'No-Attention' 'Attention'};

                ix. P.Weights=[];

                x. P.analysis='psy';

                xi. P.method='cond';

                xii. P.CompContrasts=1;

                xiii. P.Weighted=0;

                xiv. P.Contrasts(1).left={'Attention'};

                xv. P.Contrasts(1). right={'No-Attention'};

                xvi. P.Contrasts(1).STAT='T';

                xvii. P.Contrasts(1).Weighted=0;

                xviii. P.Contrasts(1).MinEvents=5;

                xix. P.Contrasts(1).name='Attention_minus_No-Attention';

                xx. P.Contrasts(2).left={'No-Attention'};

  xxi. P.Contrasts(2).right={'Attention'};
  xxii. P.Contrasts(2).STAT='T';
  xxiii. P.Contrasts(2).Weighted=0;
  xxiv. P.Contrasts(2).MinEvents=5;
  xxv. P.Contrasts(2).name='No-Attention_minus_Attention';
  xxvi. P.Contrasts(3).left={'Attention'};
  xxvii. P.Contrasts(3).right={'Stationary'};
  xxviii. P.Contrasts(3).STAT='T';
  xxix. P.Contrasts(3).Weighted=0;
  xxx. P.Contrasts(3).MinEvents=5;
  xxxi. P.Contrasts(3).name='Attention_minus_Stationary';
 b. Run PPPI with the command: *PPPI(P)*

(10) gPPI (creating a sphere)
 a. Create the VOI using *create_sphere_image.m*
  i. create_sphere_image('SPM.mat',[15 -78 -9],{'VOI_V2_15_-78_-9'},6)
  ii. This creates the image -- *VOI_V2_15_-78_-9_mask.nii*
 b. Configure the PPPI Parameter structure
  i. P.subject='Attention_Tutorial';
  ii. P.directory=pwd;
  iii. P.VOI=[pwd filesep 'VOI_V2_15_-78_-9_mask.nii'];
  iv. P.Estimate=1;
  v. P.contrast=0;
  vi. P.extract='eig';
  vii. P.Tasks={'1' 'Stationary' 'No-Attention' 'Attention'};
  viii. P.Weights=[];
  ix. P.analysis='psy';
  x. P.method='cond';
  xi. P.CompContrasts=1;
  xii. P.Weighted=0;
  xiii. P.Contrasts(1).left={'Attention'};
  xiv. P.Contrasts(1). right={'No-Attention'};
  xv. P.Contrasts(1).STAT='T';
  xvi. P.Contrasts(1).Weighted=0;
  xvii. P.Contrasts(1).MinEvents=5;
  xviii. P.Contrasts(1).name='Attention_minus_No-Attention';
  xix. P.Contrasts(2).left={'No-Attention'};
  xx. P.Contrasts(2).right={'Attention'};
  xxi. P.Contrasts(2).STAT='T';
  xxii. P.Contrasts(2).Weighted=0;
  xxiii. P.Contrasts(2).MinEvents=5;
  xxiv. P.Contrasts(2).name='No-Attention_minus_Attention';
  xxv. P.Contrasts(3).left={'Attention'};
  xxvi. P.Contrasts(3).right={'Stationary'};
  xxvii. P.Contrasts(3).STAT='T';
  xxviii. P.Contrasts(3).Weighted=0;
  xxix. P.Contrasts(3).MinEvents=5;
  xxx. P.Contrasts(3).name='Attention_minus_Stationary';
 c. Run PPPI with the command: *PPPI(P)*

5.3. Repetition Dataset (Larger effects)
 (1) Download the Repetition Dataset from the SPM website (currently:
  http://www.fil.ion.ucl.ac.uk/spm/data/face_rep/)
  a. sots.mat

      b. all-conditons.mat
      c. rawepi.zip
      d. structural.zip
      e. face_rep_spm5_batch.m
      f. editfilenames.m

(2) Put all these files into a single directly, and then unzip the compressed files.
(3) Launch MATLAB
(4) Add SPM8 and PPPI directories to the MATLAB search path (described above)
(5) Change the directory from (2)
(6) *edit face_rep_spm5_batch.m*
      a. Change line 12 to contain the current directory
      b. Click Save
(7) Run *face_rep_spm5_batch.m*
(8) Change to the directory *categorical*
(9) Steps 10-12 can also be found in *RepetitionMATLABPrep.txt*
(10)   Traditional PPI #1
      a. Create the VOI using *create_sphere_image.m*
           i. create_sphere_image('SPM.mat',[0 -28 37],{'Tutorial_PCC'},6)
          ii. This creates the image -- *Tutorial_PCC_mask.nii*
      b. Configure the PPPI Parameter structure
           i. P.subject='Repetition_Tutorial';
          ii. P.directory=pwd;
         iii. P.VOI=[pwd filesep 'Tutorial_PCC_mask.nii'];
         iv. P.Region='PCC_traditional1'
          v. P.Estimate=1;
         vi. P.contrast=0;
        vii. P.extract='eig';
       viii. P.Tasks={'N1' 'N2' 'F1' 'F2'};
         ix. P.Weights=[1 -1 -1 1];
          x. P.analysis='psy';
         xi. P.method='trad';
        xii. P.CompContrasts=0;
       xiii. P.Weighted=0;
      c. Run PPPI with the command: *PPPI(P)*
(11)   Traditional PPI # 2
      a. Create the VOI using *create_sphere_image.m*
           i. create_sphere_image('SPM.mat',[0 -28 37],{'Tutorial_PCC'},6)
          ii. This creates the image -- *Tutorial_PCC_mask.nii*
      b. Configure the PPPI Parameter structure
         xiv. P.subject='Repetition_Tutorial';
          xv. P.directory=pwd;
         xvi. P.VOI=[pwd filesep 'Tutorial_PCC_mask.nii'];
        xvii. P.Region='PCC_traditional1'
       xviii. P.Estimate=1;
         xix. P.contrast=0;
         xx. P.extract='eig';
        xxi. P.Tasks={'N1' 'N2'};
        xxii. P.Weights=[1 -1];
       xxiii. P.analysis='psy';
       xxiv. P.method='trad';
        xxv. P.CompContrasts=0;
       xxvi. P.Weighted=0;
      c. Run PPPI with the command: *PPPI(P)*
(12)   gPPI Example

a. Create the VOI using *create_sphere_image.m*
   i. create_sphere_image('SPM.mat',[0 -28 37],{'Tutorial_PCC'},6)
   ii. This creates the image -- *Tutorial_PCC_mask.nii*
b. Configure the PPPI Parameter structure
   i. P.subject='Repetition_Tutorial';
   ii. P.directory=pwd;
   iii. P.VOI=[pwd filesep 'Tutorial_PCC_mask.nii'];
   iv. P.Region='PCC_gPPI_Repetition'
   v. P.Estimate=1;
   vi. P.contrast=0;
   vii. P.extract='eig';
   viii. P.Tasks={'1' 'N1' 'N2' 'F1' 'F2'};
   ix. P.Weights=[];
   x. P.analysis='psy';
   xi. P.method='cond';
   xii. P.CompContrasts=1;
   xiii. P.Weighted=0;
   xiv. P.ConcatR=0;
   xv. P.preservevarcorr=0;
   xvi. P.Contrasts(1).left={'N1'};
   xvii. P.Contrasts(1).right={'N2'};
   xviii. P.Contrasts(1).STAT='T';
   xix. P.Contrasts(1).Weighted=0;
   xx. P.Contrasts(1).MinEvents=5;
   xxi. P.Contrasts(1).name='N1_minus_N2';
   xxii. P.Contrasts(2).left={'F1'};
   xxiii. P.Contrasts(2).right={'F2'};
   xxiv. P.Contrasts(2).STAT='T';
   xxv. P.Contrasts(2).Weighted=0;
   xxvi. P.Contrasts(2).MinEvents=5;
   xxvii. P.Contrasts(2).name='F1_minus_F2';
   xxviii. P.Contrasts(3).left={'N1' 'F2'};
   xxix. P.Contrasts(3).right={'N2' 'F1'};
   xxx. P.Contrasts(3).STAT='T';
   xxxi. P.Contrasts(3).Weighted=0;
   xxxii. P.Contrasts(3).MinEvents=5;
   xxxiii. P.Contrasts(3).name='N1_F2_minus_N2_F1';
c. Run PPPI with the command: *PPPI(P)*

6. **PPI Wrappers**
Researchers commonly use wrappers to run the same analysis on many subjects. This has the advantage that the process for running each subject is automatic and reduces the chance for human error. In MATLAB, wrappers are just functions and can be executed by typing their name at the matlab prompt. An example wrapper has been included in the release.

Key Elements:
   (1) Adding relevant paths to the MATLAB search path: PPPI, parent directory of PPPI, SPM8.
   (2) Subjects cell array. Each subject is in single quotes.
   (3) regionfile cell array. This provides the filename (recommended to have full path as well) of the region(s) you want to compute gPPI.
   (4) region cell array. This provides the region name for each file listed in regionfile. Each region name is used for naming the output directory.
   (5) Loading a master parameter file. Then we will change this based on the region and subject.

(6) Set the VOI field and Region fields, then save the new parameter file.
(7) for loop to loop through all subjects
- a. try statement to continue processing the next subject if a subject fails
- b. Loading the parameter file for this VOI and Region
- c. Set the directory of the first level subject. Can be something like: ['/Data/' Subjects{i} '/model/']
- d. Change to the subject directory
- e. Add the directory to the parameter file
- f. Add the subject name to the parameter file
- g. Save the parameter file
- h. Run PPPI on the saved parameter file