Intro to APACE – Accelerated Permutation Inference for ACE models

APACE is currently developed to analyze heritability using data from twins and siblings. The relatedness of twins and full siblings is used to estimate heritability while the data from half-siblings only contributes to measuring sample variance. Although the genetic relationship between half-siblings is not taken into account, the derived heritability estimator is valid and unbiased.

See the script "WK_ParellelOutline.m" for the outline of the software. There are 5 main parts: (1) input preparation, (2) permutations using LR-SD, (3) bootstrapping using LR-SD, (4) permutations and bootstrapping using Aggregate Heritability (pair-wise correlation) method, and (5) APACE results summary. (LR-SD stands for Linear Regression on Squared Differences, the essential trick to the "acceleration").

Note that APACE requires SPM8 or SPM12 and (for CIFTI support) CIFTIReaderWriter which in turn requires workbench.

(1) PREPARATION

This part is separated into 4 steps. Steps A-D must be run once first to prepare the input file "ACEfit_Par.mat" for the following computation.

A) Specify the input variables as part of a structure:

`ACEfit_Par.P_nm`	A filelist of image paths, one subject per line; see "FileFormats.txt" for file types supported and other ways of specifying the input data. (REQUIRED)
`ACEfit_Par.InfMx`	A CSV file of kinship information of 4 columns with headers: SubjectID (number), MotherID (number), FatherID (number), Zygosity ('MZ', 'NotMZ', 'NotTwin'). The order of subjects in the rows should match the order of image paths or subjects in 'ACEfit_Par.P_nm'. (REQUIRED)
`ACEfit_Par.ResDir`	Path for the results directory. (REQUIRED)
`ACEfit_Par.Pmask`	Brain mask image (if omitted, the whole volume/surface is analyzed).
`ACEfit_Par.Dsnmtx`	Design matrix - must have number of rows equal to the length of 'ACEfit_Par.P_nm', but any number of columns (if omitted, an all-ones vector is used).
`ACEfit_Par.Nlz`	Data normalization by inverse Gaussian transform; 0 – don't normalize, 1 – normalize before forming residuals, 2 – normalize after forming residuals (1 is recommended and is the default if omitted).
`ACEfit_Par.AggNlz`	Data normalization for computing aggregate heritability; 0 – mean centered data, 1 – same as 0, but with variance normalization, 2 – undo mean centering by adding the mean back, 3 – same as 2, but with variance normalization (0 is the default if omitted).
`ACEfit_Par.ContSel`	Optionally select a single contrast to consider, ignoring all others (only valid for a 4D NIFTI file—one per subject—or CIFTI image, where contrasts are indexed over the last

	dimension; not compatible with a single file containing all subject's data). (No such selection done by default.)
`ACEfit_Par.NoImg`	Set to 1 to optionally only compute summary measures and suppress image-wise inference (faster when only summaries are of interest). (By default, voxel/element- and (if 3D data is used) cluster-wise computations are done.)

B) With the **PrepData** function, check the data formats and prepare input data for reading, producing an updated 'ACEfit_Par'.

C) With the **ACEfit** function, fit the model to the original data and output the result figures and images (see "README_APACE_outputs.pdf" for a complete list of outputs), producing an updated 'ACEfit_Par' structure. This requires the setting of the 'alpha_CFT' variable:

`ACEfit_Par.alpha_CFT`	Cluster-forming threshold for cluster inference (made if 3D data is used), specified as an uncorrected p-values (e.g. 0.05, 0.01). (If omitted, 0.05 is used by default.)

D) With the **PrepParallel** function, prepare for computation (optionally parallelized), saving 'ACEfit_Par' to a mat file named "ACEfit_Par.mat" in the results directory. This requires the setting of 'nPerm', 'nBoot' and 'nParallel':

`ACEfit_Par.nPerm`	Number of permutations. (REQUIRED)
`ACEfit_Par.nBoot`	Number of bootstrap replicates. (REQUIRED)
`ACEfit_Par.nParallel`	Number of parallel runs. (If omitted, set to be 1 by default without parallelization.)

(2) PERMUTATIONS

The permutation inference helps calculate more accurate p-values with few assumptions. The empirical distribution of the maximum statistics is constructed by permutations in order to control the false positives over the whole image and allow the family-wise error correction of the p-values.

There are 3 steps in this part for permutation inference. Step A includes the code that can be parallelized, step B sews all parallel results together, and step C generates the output files. If nPerm=0, then this part is omitted.

A) With the **ACEfit_Perm_Parallel** function, specify & run the set of parallel jobs for permutations. There are code snippets that you can use to create parallel jobs. Here below is an example of a shell script showing how you'd do that:

```
#!/bin/bash
# Shell script for a quad-core laptop
# Starts 4 simultaneous matlab processes
nParallel=4
for ((r=1;r<=nParallel;r++)) ; do
   matlab -nodisplay << EOF > matlab_${r}.log &
   load ACEfit_Par
```

```
    RunID = $r
    ACEfit_Perm_Parallel(ACEfit_Par,RunID);
    EOF
    sleep 60 # Wait 1 minute to let Matlab start
done
```

OR, if you have a SGE cluster, you can use the "task array" method to start multiple jobs, with this script:

```
#!/bin/bash
#$ -o $HOME/ACE/out.$TASK_ID.stdout
#$ -e $HOME/ACE/error.$TASK_ID.stderr
#$ -l h_vmem=4G,h_rt=24:00:00
#$ -cwd
#
# Shell script for a SGE cluster
# Starts simultaneous matlab jobs, with task-array setting the
# run number needed by our code
r=$SGE_TASK_ID
matlab -nodisplay << EOF > matlab_${r}.log &
load ACEfit_Par
RunID = $r
ACEfit_Perm_Parallel(ACEfit_Par,RunID);
EOF
```

Here 'RunID' is the index for the parallel job; the set of jobs run (in the end) must have RunID's that are equal to 1:nParallel.   Each time "ACEfit_Perm_Parallel.m" is run it will create a .mat file named "ACEfit_Parallel_XXXX.mat", where XXXX is 'RunID'.

B) With the **ACEfit_Perm_Parallel_Results** function, merge all the results from various parallel runs.   It creates an "ACEfit_Perm.mat" file in the results directory containing 6 vectors, each being a heritability summary of in-mask regions for all permutations and the original data:

mean_ACE	Mean of heritability estimates.
wh2_ACE	Variance-weighted average of heritability estimates.
med_ACE	Median of heritability estimates.
q3_ACE	Third quartile of heritability estimates.
mGmed_ACE	Mean of heritability estimates greater than the median.
mGq3_ACE	Mean of heritability estimates greater than the third quartile.

If image-wise inference is made (ACEfit_Par.NoImg=0 or not specified), 3 vectors of maximum statistics (2 for cluster inference & 1 for voxel/element-wise inference), each comprised of values for all permutations and the original data, and a matrix of uncorrected p-values, each element of this matrix for each voxel/element, are also saved:

max_K_ACE	Maximum suprathreshold cluster size distribution (saved if 3D data is

	used).
max_M_ACE	Maximum suprathreshold cluster mass distribution (saved if 3D data is used).
max_T_ACE	Maximum statistic distribution.
unPval_ACE	Uncorrected permutation-based p-values.

C) With the **ACEfit_Results** function, save and print to screen permutation distributions and p-values of summary statistics including mean(h^2), wh^2, median(h^2), Q3(h^2), mean(h^2>median) and mean(h^2>Q3); p-values of summary statistics are saved (in that order) as a column vector in "Pvals_h2.mat". Various PDF plots are also saved, recording the permutation distributions of these summary statistics. If image-wise inference is made (ACEfit_Par.NoImg=0 or not specified), voxel/element-wise uncorrected, FWE- and FDR-corrected p-value images as well as (if 3D data is used) cluster size and mass images and FWE-corrected p-value images for cluster size and mass are written. Figures of permutation distributions and p-values of maximum statistics for voxel/element, cluster size and mass are saved as PDF files and printed on screen; p-values are saved in "Pvals_Max_h2.mat" ('p_T' for voxel/element, (if 3D data is used) 'p_K' for cluster size and 'p_M' for cluster mass). See "README_APACE_outputs.pdf" for a complete list of output images.

(3) BOOTSTRAPPING

The bootstrapping is applied to compute the confidence intervals for the above-mentioned summary statistics for h^2, c^2 and e^2. The bootstrap distributions of the summary statistics are generated and used to construct the confidence intervals.

There are 3 steps for bootstrapping; step A is what can be parallelized, step B merges together all the results from the multiple parallel jobs, and step C calculates the bootstrapping confidence intervals. If nBoot=0, this part is omitted.

A) With the **ACEfit_Boot_Parallel** function, specify & run the set of parallel jobs. Usage is demonstrated in "WK_ParellelOutline.m", and, as above in part (2-A), it can be parallelized.

'RunID' is the index for the parallel job; the set of jobs run (in the end) must have RunID's equal to 1:nParallel. Each time "ACEfit_Boot_Parallel.m" is run it will create a .mat file named "BootCI_Parallel_XXXX.mat", where XXXX is 'RunID'.

B) With the **ACEfit_Boot_Parallel_Results** function, merge all the results from parallel running together, create 18 column vectors saved to "ACEfit_Boot.mat", each being a summary of in-mask regions for all bootstrap replicates and the original data:

meanh2_ACE	Mean of h^2.
wh2_ACE	Variance-weighted average of h^2.
medh2_ACE	Median of h^2.
q3h2_ACE	Third quartile of h^2.
mGmedh2_ACE	Mean of h^2 greater than the median.
mGq3h2_ACE	Mean of h^2 greater than the third quartile.
meanc2_ACE	Mean of c^2.
wc2_ACE	Variance-weighted average of c^2.
medc2_ACE	Median of c^2.
q3c2_ACE	Third quartile of c^2.
mGmedc2_ACE	Mean of c^2 greater than the median.
mGq3c2_ACE	Mean of c^2 greater than the third quartile.
meane2_ACE	Mean of e^2.
we2_ACE	Variance-weighted average of e^2.
mede2_ACE	Median of e^2.
q3e2_ACE	Third quartile of e^2.
mGmede2_ACE	Mean of e^2 greater than the median.
mGq3e2_ACE	Mean of e^2 greater than the third quartile.

C) With the **Boot_CIs** function, compute the bootstrapping confidence intervals for all these 18 summary statistics, print the CI's on screen, and save the CI's to "Boot_CIs.mat"; saved information includes the alpha level (e.g. 0.05 for 95% CI's) and 3 matrices: 'CIs_h2', 'CIs_c2', 'CIs_e2'. The column order of these matrices is: mean, weighted mean, median, third quartile, mean of estimates greater than median, and mean of estimates greater than the third quartile. Each row of these matrices gives the CI for one summary measure.

(4) AgHe METHOD

Use the **AgHe_Method** function to get both permutation results and bootstrapping CI's for r_{MZ}-r_{DZ} and r_{DZ}-r_{Sib} using the Aggregate Heritability (AgHe, aka "Steve's Method") pair-wise correlation method. There are 3 output .mat files; "Ests_AgHe.mat" contains the measures of mean difference, "Pvals_AgHe.mat" includes the permutation-based p-values for 2-sample t-statistic and mean difference, and "CIs_AgHe.mat" gives the bootstrapping confidence intervals for mean difference.

(5) APACE SUMMARY

With the **APACEsummary** function, the APACE output results can be summarized. The created results summary table can be printed on screen and saved to a CSV file, and the estimates, p-values and confidence intervals for all considered statistics can be returned as matrices.