SSPM Manual

SSPM represents **S**patial **S**tatistical **P**arametric **M**apping. It was developed by UNC-CH Biostatistics and Imaging Analysis Lab (BIAS). It includes MAGEE, FADTTS and FMPM.

How to start SSPM

Set SSPM package folder to be the Matlab current directory. In Matlab command window, enter *SSPM*, SSPM Graphical User interface (GUI) occurs, as shown in Figure 1.

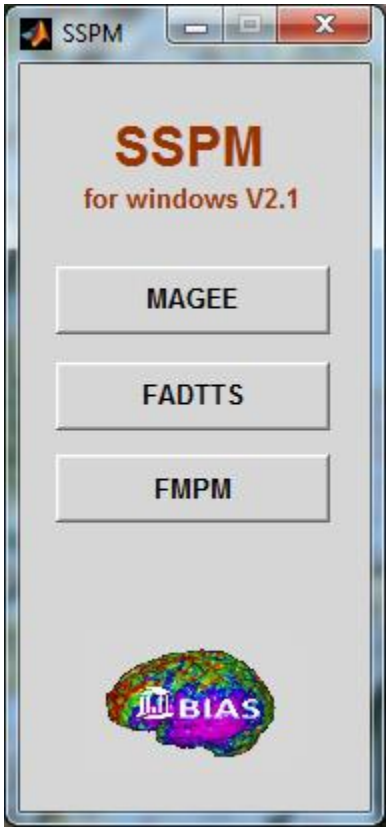

Figure 1

Chapter 1 MAGEE

MAGEE represents the **M**ultiscale **A**daptive **G**eneralized **E**stimating **E**quation. It was developed specifically for analyzing multivariate neuroimaging data in 3-dimensional volume (or on 2-dimensional surface, such as spherical harmonics representation) from longitudinal neuroimaging studies. It integrates Adaptive Smoothing methods with Generalized Estimating Equations at each voxel for spatial and adaptive analysis of multivariate continuous neuroimaging measures.

Motivation: Neuroimaging studies aim to analyze imaging data with complex spatial activation patterns in a large number of locations (called voxels) on a two-dimensional (2D) surface or in a 3D volume. Following spatial normalization, imaging observations for each subject are observed in a large number of locations (called voxels), that number in the thousands to millions, on a common two-dimensional (2D) surface or in a common 3 dimensional (3D) volume. Conventional analyses of those high-dimensional imaging data are often executed in two sequential steps: spatially smoothing imaging data and then independently fitting a statistical model at each voxel, called voxel-wise method.

Most smoothing methods are independent of imaging data and apply the same amount of smoothing throughout the whole image. See, for example, Yue, Loh and Lindquist (2010) for overviews of smoothing methods in the neuroimaging literature. As shown in Polzehl and Spokoiny (2000, 2003, 2006), Qiu (2005, 2007), and Tabelow et al. (2006, 2008a, b, c), those smoothing methods can be very problematic near the edges of the activated regions. Polzehl and Spokoiny (2000, 2003, 2006) proposed a powerful propagation$-$separation (PS) approach to adaptively and spatially smooth images from a single subject. Tabelow et al. (2006, 2008a, b, c) used the original PS idea to develop a multiscale adaptive linear model to adaptively and spatially denoise fMRI and diffusion tensor images from a single subject.

The existing voxel-wise methods for analyzing high-dimensional data involves fitting a statistical model, such as general linear model (LM), to neuroimaging data from all subjects at each voxel, and then generating a statistical parametric map of test statistics and p-values (Beckmann, Jenkinson, and Smith, 2003; Nichols and Hayasaka, 2003; Worsley et al., 2004). The existing voxel-wise methods have some obvious limitations for the analysis of neuroimaging data, which underscore the great need for further methodological development. As pointed out by Tabelow et al. (2006), **voxel-wise methods treat all voxels as independent units and do not model the spatial properties of the imaging observations explicitly**. Neuroimaging data, however, are spatially dependent in nature, where we often observe spatially

contiguous regions of activation with rather sharp edges in many neuroimaging studies. Moreover, most smoothing methods are independent of statistical models in voxel-wise methods. As shown in Hecke et al. (2009) and Jones et al. (2005), **voxel-wise methods can suffer from the arbitrary choice of smoothing extent and dramatically increase the numbers of false positives and false negatives.**

Spatially modeling neuroimaging data in the 3D volume (or 2D surface) represents both computational and theoretical challenges. It is common to use conditional autoregressive (CAR), Markov random field (MRF), and other spatial correlation priors to characterize spatial dependence among spatially connected voxels (Besag, 1986; Banerjee, Carlin, and Gelfand, 2004), but calculating the normalizing factor of MRF and estimating spatial correlation for a large number of voxels in the 3D volume (or 2D surface) **are computationally prohibitive** (Zhu, Gu, and Peterson, 2007; Bowman, 2007). Moreover, it is common that MRF can **oversmooth the boundary of activation regions**, while it can be **restrictive to assume a specific type of correlation structure**, such as CAR and MRF, for the whole 3D volume (or 2D surface). Bowman (2007) proposed to model the spatial correlation among the summary statistics of a small number of regions of interest (ROIs). As discussed in Snook et al. (2007), **the major drawbacks of such ROI analysis include the difficulty in identifying the meaningful ROIs, the instability of statistical results obtained from ROI analysis, and the partial volume effect in relative large ROIs.**

MAGEE has three computational features: being spatial, being hierarchical and being adaptive. MAGEE builds a sphere with a given radius at all voxels, and then uses these consecutively overlapping spheres to capture local and global spatial dependence among different voxels. Thus, MAGEE explicitly utilizes the spatial information to carry out statistical inference. MAGEE also builds hierarchically nested spheres by increasing the radius of a spherical neighborhood around each voxel and utilizes information in each of the nested spheres across all voxels. Finally, MAGEE combines all observations with adaptive weights in the voxels within the sphere of the current voxel to adaptively calculate parameter estimates and test statistics. Due to its hierarchical and adaptive nature, although MAGEE increases the amount of computational time in computing parameter estimates and testing statistics, MAGEE can efficiently utilize available information in the neighboring voxels of each voxel to increase the precision of parameter estimates and the power of test statistics in detecting subtle changes of brain structure and function.

MAGEE has strong theoretical underpinning. We establish consistency and asymptotic normality of the adaptive estimator and the asymptotic distribution of the adaptive test statistic for MAGEE as the number of subjects (or images) increases to infinity. In contrast, the existing theory for the PS approach only covers the consistency

of the adaptive estimator under the propagation-separation condition when imaging data follow a class of exponential families and come from a single subject (Polzehl and Spokoiny, 2006). The adaptive weights in MAGEE differ from those in the PS approach and the covariance estimate of the adaptive estimator in MAGEE has a simpler form. Our new theoretical results show that in MAGEE, the adaptive weighting idea of the novel PS approach is valid without imposing the propagation-separation condition. Our results show that it is critical to choose appropriate kernel functions in constructing adaptive weights, which depend on both the numbers of subjects and voxels, in order to have appropriate asymptotic results to carry out statistical inference including hypothesis testing on parameters of interest.

Input files:

Several input files are needed to run MAGEE program. First, you need to know six parameters:
Dimension of response;
Number of subjects;
Number of covariates for each dimension of response,
Maximum number of time points among all subjects.
Number of rows of the constraint matrix.
Number of iterations.

For example, consider a longitudinal study with 30 subjects. Among all subjects, 28 subjects were measured at 5 time points, while 2 subjects were measured at 8 time points. At each voxel, there are two measurements, such as FA and MD in a diffusion tensor study. For each measurement, we fit a linear model with 3 covariates including intercept, gender and age. In this software, the number of iterations must be no less than 5. For efficiency, we suggest that the number of adaptation iterations (Number of iterations) be set to be between 5 and 20. In this case,

Dimension of response =1;

Number of subjects =30;

Number of covariates for each dimension of response =3;

Maximum number of time points among all subjects =8.

“Number of rows of the constraint matrix” will be explained later. .

Second, you need to prepare four text files.

TimePointFilename.txt : A text file containing the numbers of time points for all subjects and the exact times. For example, in longitudinal data design, you collect 3

time point data for each subject and you collect at 0, 1 and 2 years. So the file should be: 3 0 1 2 for one subject. If you have three subjects, it should be:

3 3 3
0 1 2 0 1 2 0 1 2

In the first line, the n-th number is the number of time points for the n-th subject. Here, each number is equal to 3. That is to say, each subject has 3 time points. In the second line, the first three numbers 0 1 2 are the exact times for the first subject, the second three numbers 0 1 2 are the exact times for the second subject, and the third three numbers 0 1 2 are the exact times for the third subject.

Here is another example. In a longitudinal data design, you have four subjects and each subject has different number of time points. For instance, the first subject only has 1 observation at 2.4 year, the second subject only has 2 observations at 1.5 and 6 years, and other two subjects have four time points at 5, 6, 7, and 10 years. The file should be

1 2 4 5
2.4 1.5 6 5 6 7 10 5 6 7 10

In the first line, the first number "1" denotes that the first subject has only time point, which is 2.4 in the second line. The third number "4" in the first line denotes that the third subject has 4 time points, which are 5, 6, 7 in the second line.

DesignMatrixName.txt: A text file containing the design matrix for a single measure of the response vector. The number of rows of the matrix is equal to the total images. Each column corresponds to a specific covariate. For instance, if you want to include an intercept, then you need to put all 1's in that column. If you want to include the covariate gender and age, the design matrix should look like as follows:

1 1 5.2
1 1 8.9
1 0 12
1 1 7.8
1 0 11
1 0 15.1
1 0 9

In the above matrix, the first column represents the intercept and the second column represents the gender, for example, 1 may denote male and 0 female. The third column represents the age.

Since age is a time-dependent covariate, we allow age changing at different time points. Gender is a time-independent covariate, we fix it for different subjects.

Maskimage.hdr, Maskimage.img or Maskimage.nii: You can use the mask image file to eliminate the unused voxels. In the mask image, you need to put any number bigger than 0.1 in the regions of interest (ROI) and put any number smaller than 0.1 outside ROI.

linearConstraintMatrixName.txt : A text file containing a linear constraint matrix used to test statistics. For example, we have beta_0, beta_1 and beta_2 in the model. We want to test whether beta_1=a. Then, in this file linear constraint matrix should look like:

0 1 0
a
2

In the first line, the second number is a nonzero number 1, this means: we want to test whether beta_1=a. The number 2 in the third line represents beta_1. If we want to test beta_0=1. Then this file should look like:

1 0 0
1
1

Corresponding to above examples,

Number of rows of constraint matrix=1.

InputImageFilename.txt: A text file that containing all image names that will be analyzed. In this file, all image names are listed by row without suffix. The number of rows is the same as the number of rows of design matrix. An example for InputImageFilename.txt is as bellow;

0021002_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM
0021003_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM
0021005_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM
0021006_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM
0021007_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM

0021008_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM
0021009_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM
0021010_mprage_noface_strip_cere_seg_reg_conv_RAVENSmap_WM

Output files: There are five sets of output image files. MAGEE outputs the images of parameter estimates, standard deviations, corrected –log10(p) p-values, uncorrected –log10(p) values and Wald test statistics at scales: h1, hs when the number of iterations s is no more than 5 or h1, h5, hs when the number of iterations s is more than 5.

The image files for parameter estimates are named as **BetaMA_hs_Pm.nii**, where s denotes the scale and varies from 0 to 20 and m denotes a specific parameter of beta and varies from 1 to dim(beta).

The image files for standard deviations are named as **BstdMA_hs_Pm.nii**, where s denotes the scale and varies from 0 to 20 and m denotes a specific parameter of beta and varies from 1 to dim(beta).

The image files for Wald test statistics are named as **BwaldMA_hs_Pm.nii**, where s denotes the scale and varies from 0 to 20 and m denotes a specific parameter of beta and varies from 1 to dim(beta).

The image files for uncorrected –log10(p) p-values are named as **BrawPMA_hs_Pm.nii** and, where s denotes the scale and varies from 0 to 20 and m denotes a specific parameter of beta and varies from 1 to dim(beta).

The image files for corrected –log10(p) p-values are named as **BcorPMA_hs_Pm.nii**, where s denotes the scale and varies from 0 to 20 and m denotes a specific parameter of beta and varies from 1 to dim(beta).

How to use MAGEE GUI

MAGEE GUI is a MATLAB graphical user interface software developed to do data processing by using MAGEE.

Before using **MAGEE GUI** to do data processing, you'd better create two folders: one contains the input images whose names are listed in the text file **InputImageFilename.txt.** Another is to place output images.

Click MAGEE button in SSPM interface, or set MAGEE in the SSPM package and type *mageegui* directly, the MAGEE GUI occurs, as shown in Figure 2.

In the "Enter the Setup Parameters" panel, type the corresponding numbers in to the white text box. In the "Enter Filenames" panel, click each toggle button and then select

a proper text file or image file. For example, if you click “Filename Containing Time Points” button, you will get a select box. Then you can select the folder contains the text file by name of **TimePointFilename.txt** and choose this file, then click “Open”. See Figure 3.

In the “Enter the input-image folder and the output-image folder” panel, click each toggle button to select the folder contains the input images and the folder you want to place the output images. For example, when you click the button “Output Image Folder”, a file select box will occur, see Figure4. Choose folder “Output” and click “OK”, the output folder will be input.

When all parameters and filenames were input, the GUI will look like Figure 5. If you select a wrong file or folder, you may click the corresponding toggle again and select a right file or folder.

When you are sure all parameters, files and folders are input correctly, click “RUN” to start data processing.

While the MAGEE GUI is running, some messages will appear in the MATLAB command window.

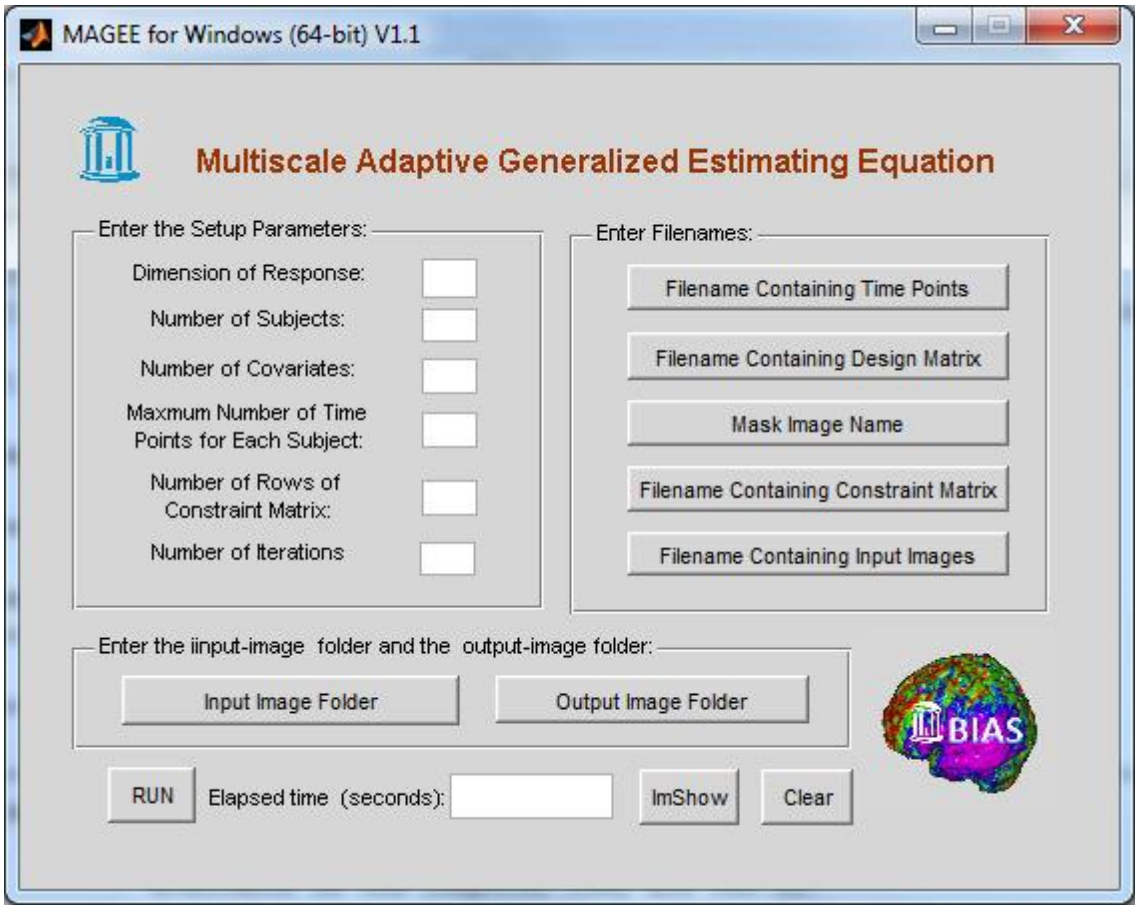
MAGEE for Windows (64-bit) V1.1
Multiscale Adaptive Generalized Estimating Equation
Enter the Setup Parameters:
Dimension of Response:
Number of Subjects:
Number of Covariates:
Maxmum Number of Time Points for Each Subject:
Number of Rows of Constraint Matrix:
Number of Iterations
Enter Filenames:
Filename Containing Time Points
Filename Containing Design Matrix
Mask Image Name
Filename Containing Constraint Matrix
Filename Containing Input Images
Enter the iinput-image folder and the output-image folder:
Input Image Folder
Output Image Folder
BIAS
RUN
Elapsed time (seconds):
ImShow
Clear

Figure 2

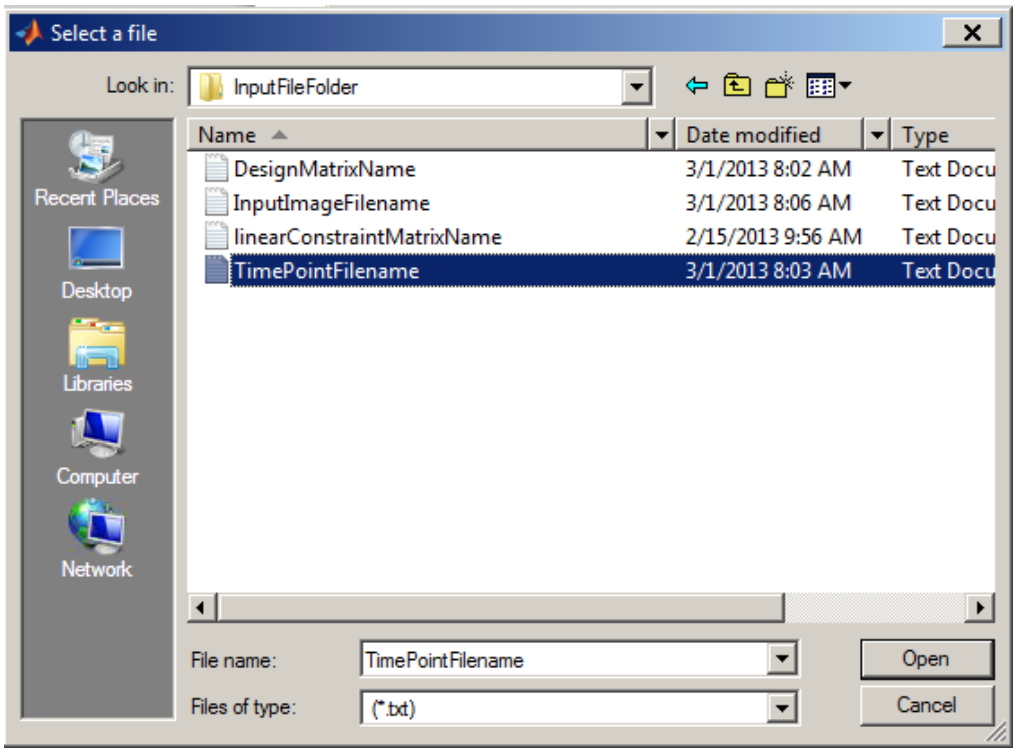

Figure 3

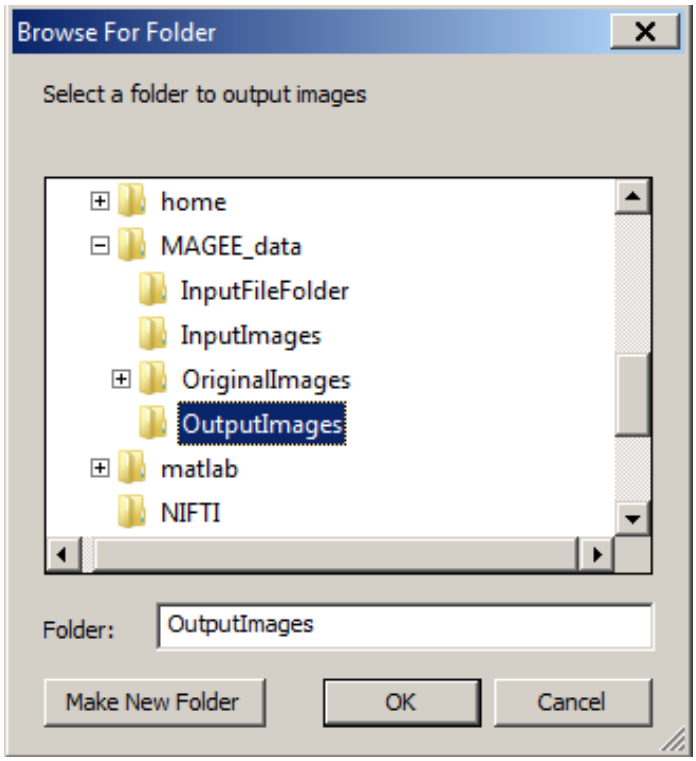

Figure 4

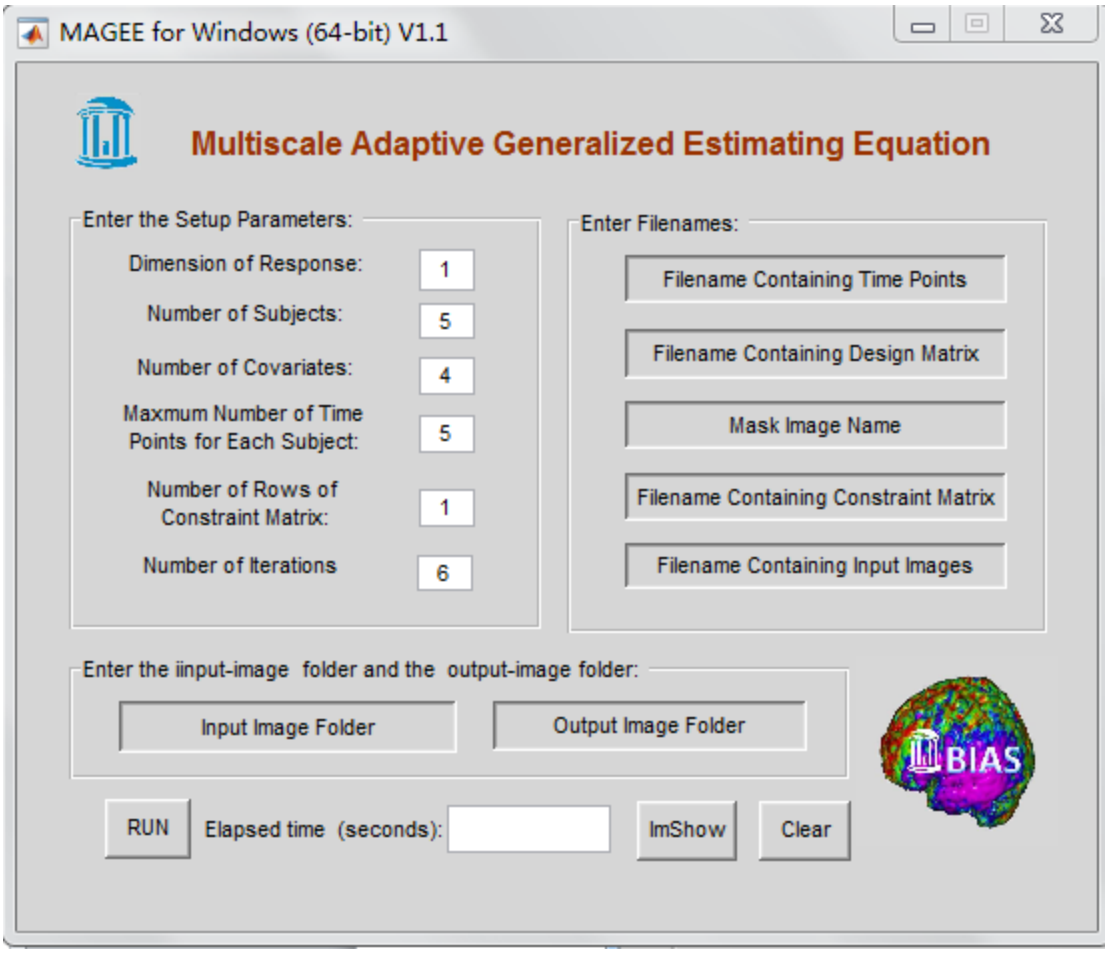

Figure 5

When the MAGEE GUI finishes its job, the all output images will be put into the “Output” folder. You may click “Imshow” button to choose an image to show, or choose an image directly in the “Output” folder and click it rightly to view it if there is an image show software in your computer, such as MRIcro.

If you want to do the same job again or another job by using MAGEE GUI, you’d better click “Clear” button to clear up all input.

You may run MAGEE program without Matlab GUI or submit it to a server, the main M-file is *MAGEE_main.m.* Type MAGEE_main in Matlab command window to run it.

Running information When the program go to end a text file that records some information containing several parameters and running process will be in the OUTPUT folder.

Example: In SSPM package, there is a folder 'MAGEE_example', which includes files and images you can use to display MAGEE GUI, as an example. You can enter parameters as shown in Figure 5.

Chapter 2 FADTTS

FADTTS represents **F**unctional **A**nalysis of **D**iffusion **T**ensor **T**ract **S**tatistics. The aim of this tool is to implement a functional analysis pipeline, called FADTTS, for delineating the structure of the variability of multiple diffusion properties along major white matter fiber bundles and their association with a set of covariates of interest, such as age, diagnostic status and gender, in various diffusion tensor imaging studies. The FADTTS integrates five statistical tools: a multivariate varying coefficient model for allowing the varying coefficient functions to characterize the varying association between fiber bundles diffusion properties and a set of covariates, a weighted least squares estimation to estimate the varying coefficient functions, a functional principal component analysis to delineate the structure of the variability in fiber bundles diffusion properties, a global test statistic to test hypotheses of interest, and a simultaneous confidence band to quantify the uncertainty in the estimated coefficient function. FADTTS can be used to facilitate understanding normal brain development, the neural bases of neuropsychiatric disorders, and the joint effects of environmental and genetic factors on white matter fiber bundles.

Motivation: Diffusion Tensor Imaging (DTI), which can track the effective diffusion of water in the human brain in vivo, has been widely used to map the structure and orientation of the white matter fier tracts of the brain (Basser et al., 1994b,a). In the current literature, three major approaches to the group analysis of diffusion imaging data are region-of-interest (ROI) analysis, voxel based analysis, and fiber tract based analysis (Smith et al., 2006; O'Donnell et al., 2009; Snook et al., 2007). The ROI analysis used in some neuroimaging studies (Bonekam et al., 2008; Gilmore et al., 2008) primarily suffers from the difficulty in identifying meaningful ROIs. Voxel based analysis is used more commonly than ROI analysis in neuroimaging studies (Chen et al., 2009; Focke et al., 2008; Camara et al., 2007; Snook et al., 2005). The major drawbacks of voxel based analysis include the issues of alignment quality and the arbitrary choice of smoothing extent (Hecke et al., 2009; Ashburner and Friston, 2000; Smith et al., 2006; Jones et al., 2005). With the drawbacks mentioned of the ROI and voxel based analyses, there is a growing interest in the DTI literature in developing fiber tract based analysis of diffusion properties (Smith et al., 2006; O'Donnell et al., 2009; Yushkevich et

al., 2008; Goodlett et al., 2009; Zhu et al., 2010b). Statistically, diffusion properties along fiber bundles are functional data and its analysis requires advanced functional data analysis methods (Li and Hsing, 2010; Yao and Lee, 2006; Hall et al., 2006; Ramsay and Silverman, 2005, 2002).

There are several developments on the use of functional data analysis methods for the statistical analysis of diffusion properties along fiber tracts, which all are smoothing first, then estimation" procedures. However, their methods are not capable of delineating the structure of the variability in fiber bundles diffusion properties and for quantifying the uncertainty in the estimated coefficient functions. To specifically address the limitations in (Goodlett et al., 2009; Zhu et al., 2010b), FADTTS presents a functional analysis pipeline for delineating the structure of the variability of multiple diffusion properties along major white matter fiber bundles and their association with a set of covariates of interest, such as age, diagnostic status and gender, in various diffusion tensor imaging studies.

How to use FADTTS GUI

FADTTS GUI is a MATLAB graphical user interface software developed to do data processing by using FADTTS. There are four button groups, which are supposed to be executed in order. The four groups are **Load Raw Data**, **Basic Plots**, **Load Test Data**, and **P-value Plots.** There are three raw data sets, namely, **tract data**, **design data** and **diffusion data**. The test data sets include test design matrix and null hypothesis vector. All data sets must be in .mat. The package includes a sample Matlab code pre_address_data.m on how to set up data. After loading all raw data, GUI will transfer the raw data, estimate the coefficients, do spectral decomposition and estimate confidence bands. Then you can plot the raw tract data, the coefficient functions, spectral decomposition and confidence bands by pushing the corresponding buttons. If you want to do a test, you need to load the test design data. There are two types of test. One is to test individually and the other one is test all the diffusion properties together. Once you loaded the test design data, GUI will display what test type you requested. The test calculation may take a while. After Matlab finishes the computation, GUI will report the global test statistics and p-values. You also have the option to plot the local p-values.

Load Raw Data

tractData: the text file containing (x; y; z) coordinates of all locations on a given fiber tract. The data set should start from one end to the other end. tractData is an LX3 matrix, where L denotes the number of locations. 3 denotes the three coordinates.

designData: the text file containing covariates of interest. Please always include the intercept in the first column. designData is an nXp matrix, where n denotes the number of subjects and p denotes the number of covariates.

Diffusion: an mX1 cell containing the names of all fiber diffusion properties files, where m is the number of features. Each fiber bundle diffusion properties should contain an LXn matrix. Rows correspond to the columns in **tractData**, while columns correspond to the columns in **designData**.

Basic Plots

Diffusion: plot an mX1 vector of scales for each property, where m is the number of features.
Coefficients: plot a pX1 vector of coefficient functions, p-1 is the number of covariates.
Eigens: plot an L_0X (L_0+1) Xm matrix of eigenvalues of individual covariance matrix of etas.
CBands: plot a 2pXL_0Xm matrix of estimated confidence bands.

Load Test Data

CMatrix: an rXmp design matrix for characterizing the r linear constraints among mp parameters.

B0vector: an rXL_0 vector for hypothesis testing.

P-value Plots: plot p-values.

Click FADTTS button in SSPM interface, or run *FADTTS_GUI* directly, the FADTTS GUI occurs, as shown in Figure 6. Click buttons in order from left to right, following the directions to input files and data, and plot each kind of figure. The results will present in FADTTS Output panel.

For more details about FADTTS, see *FADTTS_refference.pdf*: "*Matlab Tool: Functional Analysis of Diffusion Tensor Tract Statistics* " in FADTTS folder.

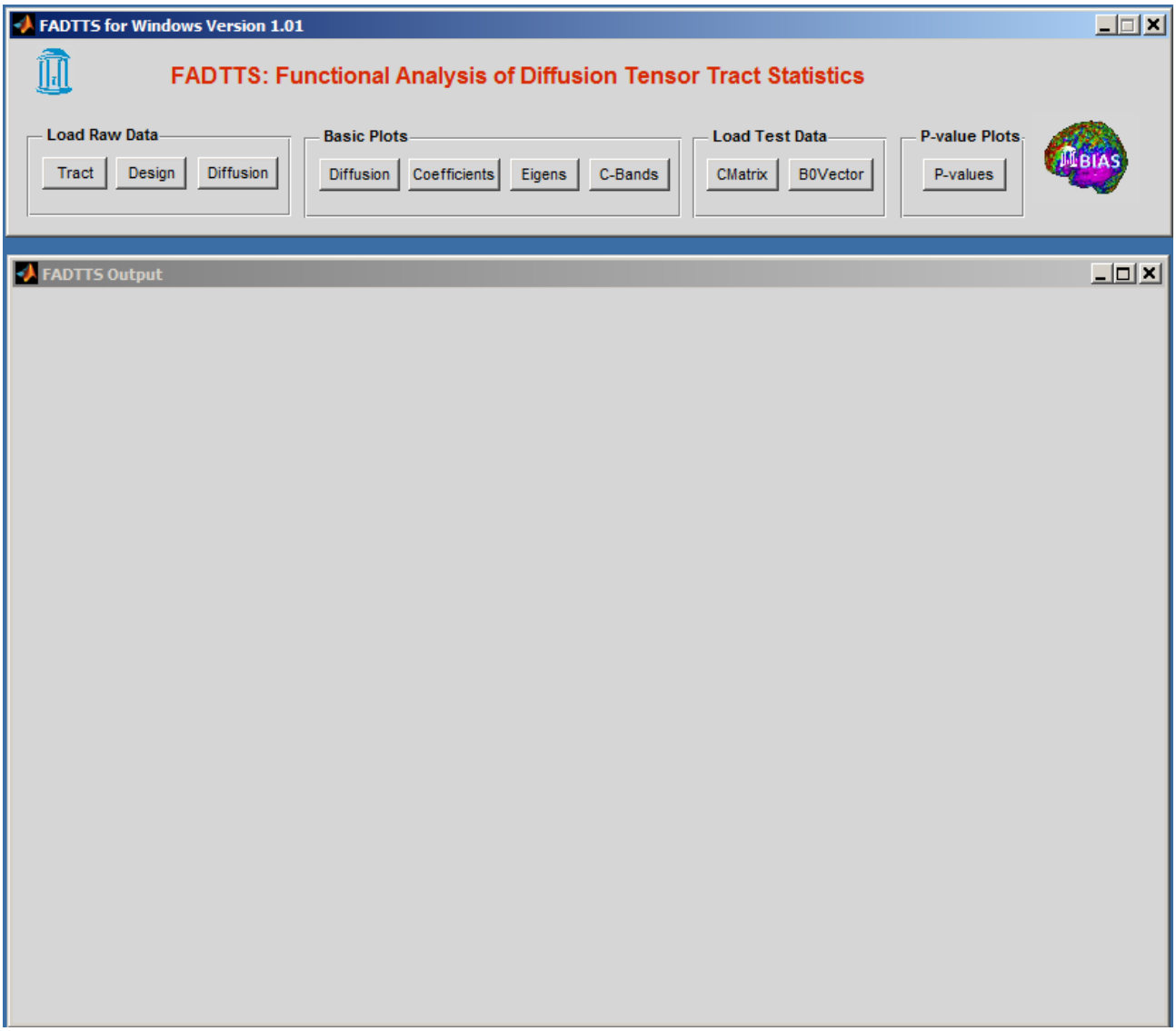

Figure 6

Chapter 3 FMPM

FMPM represents **F**unctional **M**ixed **P**rocesses **M**odels. The aim of this tool is to implement a functional analysis pipeline, for the joint analysis of longitudinally measured functional data and clinical data, for example age, gender and disease status. FMPM consists of a functional mixed effects model for characterizing the association of functional response with covariates of interest by incorporating complex spatial–temporal correlation structure, an efficient method for spatially smoothing varying coefficient functions, an estimation method for estimating the spatial– temporal

correlation structure, a test procedure with local and global test statistics for testing hypotheses of interest associated with functional response, and a simultaneous confidence band for quantifying the uncertainty in the estimated coefficient functions.

Motivation: FMPM framework is motivated by the emerging demand to analyze massive functional (or imaging) data collected in large-scale longitudinal biomedical studies, such as the Alzeimer's disease neuroimaging initiative (Evans and Group, 2006). Although there is an extensive literature on statistical methods for the analysis of univariate (or multivariate) variables measured longitudinally, little has been done on the development of statistical methods to analyze longitudinal functional data. This could be due to at least three major challenges including (i) infinite dimensional functional responses measured at multiple time points, (ii) complex spatial–temporal correlation structure, and (iii) complex spatial smoothness. According to the best of our knowledge, a longitudinal functional principal component analysis in Greven et al. (2010) is the first statistical method for the analysis of repeated functional responses, in which an estimation procedure was proposed to estimate both fixed effect curves and spatial–temporal covariance operators. However, in Greven et al. (2010), there is a lack of several formal statistical inference tools, such as a test statistic. Moreover, in Geng et al. (2012), a functional mixed effects model proposed by Guo (2002) was used to analyze fiber-tract diffusion properties from longitudinal studies. However, since the functionalmixed effects model in Guo (2002)was developed to model the datawith functional responses (of time or distance) measured only once for each subject, directly applying suchmodel in Guo (2002) to functional responses measured multiple times essentially accounts for only the spatial correlations, but ignores the within-subject temporal correlations. The aim of this paper is to present a functional analysis pipeline (FMPM) with several formal statistical inference tools for delineating the dynamic changes of fiber-tract statistics and their associations with a set of covariates obtained from longitudinal studies.

How to use FMPM

Hypothesis Testing

The whole procedure for hypothesis testing can be found in HT _FMPM_Example.m.
When you use the procedure, what you need to do is to

1. Prepare the data in the required format

Necessary Data and Data Format are as follows:

1) Ydesign --- $\sum_{i=1}^{n} r_i \times L_0$ response matrix;

2) Xdesign --- $\sum_{i=1}^{n} r_i \times p_x$ design matrix including intercept;

3) Zdesign --- $\sum_{i=1}^{n} r_i \times p_z$ design matrix including intercept;

4) Indicator --- $n \times 1$ cell, where each cell is a $r_i \times 1$ vector, containing the indices for each subject *i*;

5) arclength --- $L_0 \times 1$ column vector of the arclength from one end to the other end

where

n --- number of subjects;

L_0 --- number of location of fiber tract;

p_x --- number of covariates with respect to fixed effects;

p_z --- number of covariates with respect to random effects;

r_i--- number of repeated measurements for each subject;

2. Specify the contrast matrix and the corresponding zero vector

For example, suppose there are three covariates including intercept, age and gender. If you want

to test the effect of age, you need to specify the contrast matrix and the corresponding zero

vector as follows:

Cdesign=[0,1,0];

B0vector=zeros(1,L0);

3) specify ExpVar. E.g., ExpVar=0.99;

Simultaneous Confidence Interval

The whole procedure for estimating the simultaneous confidence interval can be found in SB

_FMPM_Example.m. When you use the procedure, what you need to do is to

1) prepare the data in the required format, see Hypothesis Testing part.

2) specify ExpVar, see Hypothesis Testing part.

How to use FMPM GUI

FMPM GUI is a MATLAB graphical user interface software developed to do data processing.

Before using **FMPM GUI** to do data processing, you have to create a mat file containing Xdesign matrix, Ydesign matrix, Zdesign matrix, Indicator cell and arclength vector. You also need to create a mat file containing Cdesign matrix. Then put these two files into a folder which will accept outputs.

Click **FMPM** button in SSPM interface, or set **FMPM** in the SSPM package and type *FMPMgui* directly, the **FMPM** GUI occurs, as shown in Figure 7.

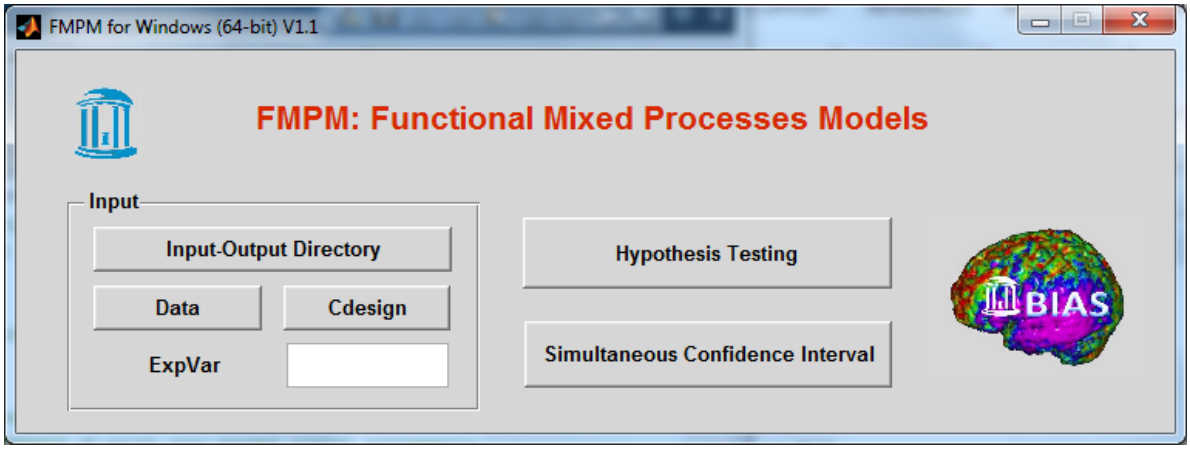

Figure 7

Click ‘Input-Output’ to choose the folder containing the mat files you created. Click ‘Data’ to choose the mat file containing Xdesign matrix, Ydesign matrix, Zdesign matrix, indicator cell and arclength vector. Click ‘Cdesign’ to choose a mat file containing Cdesign matrix. Type a number in the ‘ExpVar’ box. Click ‘Hypothesis Testing’ to do hypothesis testing’, click ‘Simultaneous Confidence Interval’ to calculate confidence intervals simultaneously. All results will be put into the ‘Input-Output’ folder.