COL 786 - Assignment 5: Develop General Linear Model Analysis Tool for fMRI Data

Due Date: 9:00pm, Thursday, 14$^{\text{th}}$ April, 2021
Max Marks : 100

Notes: The objective of this assignment is to develop a command-line based tool for General Linear Model (GLM) Analysis of fMRI data. The tool has two parts. The first part aims to develop a GLM tool for single subject analysis. The second part aims to develop a GLM tool for group analysis.

1 Tool for Single Subject GLM Analysis [60 Marks]

Create a Python command line tool with the name `glm.py` with the following specifications:

```
python glm.py <functional file> <ev file> <contrast file> <output prefix>
```

Your python program takes 4 arguments: The first argument is the `<functional file>` which is the name of the filtered and preprocessed 4 dimensional input fMRI data. The second argument is the `<ev file>` which represents the different experimental conditions. The experimental variables of EVs are in a four column format with the following interpretation: the first column represents the start of an experimental condition or a block (in seconds), the second column represents the duration of the experimental condition (in seconds), the third column represents the value of the experimental condition in the given time interval and the fourth column has an integer representing the experimental condition number. If there are three different types of experimental conditions, then the fourth column may take values of 1, 2 and 3 representing the three experimental conditions. The `<contrast file>` contains a list of numbers in each row with first number corresponding to the first EV, second number corresponding to the second EV and so on. If there are a total of three experimental conditions, then each row of the `<contrast file>` will contain three numbers representing the weight of each experimental condition in this contrast. Each row in the `<contrast file>` corresponds to a different contrast value.
Your program should create output files with the name beginning with the `<output prefix>`. The files that are to be created are:

1. Parameter estimate files (betas): `<output prefix>.pe<1>.nii.gz` to `<output prefix>.pe<n>.nii.gz` representing different experimental conditions. The number of pe files must be equal to the number of EVs in the `<ev file>`.

2. Contrast of parameter estimation files: `<output prefix>.cope<1>.nii.gz` to `<output prefix>.cope<m>.nii.gz` representing the contrast of parameter estimates ($c^t\beta$). The number of cope files should be equal to the number of rows in the `<contrast file>`.

3. t-statistics files: `<output prefix>.tstat<1>.nii.gz` to `<output prefix>.tstat<m>.nii.gz` representing the t-statistics of the contrast of parameter estimate. The number of tstat files should be equal to the number of rows in the `<contrast file>`.

4. z-statistics files: `<output prefix>.zstat<1>.nii.gz` to `<output prefix>.zstat<m>.nii.gz` representing the z-statistics of the contrast of parameter estimate. The t-statistics should be converted to a standard z-statistics value and stored in the zstat files for each of the contrasts. The number of zstat files should be equal to the number of rows in the `<contrast file>`.

Use your tool to generate every contrast in your assignment 2. Create a report comparing the parameter estimate files and the zstat files generated by your program with the corresponding files generated by FSL tool. For comparison, carry out a scatter plot of the PE (and zstat) values for every voxel in the brain with x-axis as the PE (or zstat) value as computed by FSL and y-axis as the values calculated by your program. Also, include brain maps of the zstat files (FSL computed and computed by your program) for a side-by-side comparison.

Submission Instructions: You need to submit your program `glm.py` and the input `<ev file>` and `<contrast file>` on moodle and your report on gradescope. In case your results don't match the FSL results, try to explain why they are different. Your evaluation will be based on correctness of your program and the quality of your report.

2 Tool for Group Analysis [40 Marks]

Create a python command line tool with the name `group_analysis.py` to carry out group analysis of fMRI data with the following specifications:
`python group_analysis.py <file list.txt> <output prefix>`.
Your program takes two arguments. The `<file list.txt>` is a file containing the list of registered file names on which the group analysis is to be done. These files may be parameter estimate (pe) files, contrast of parameter estimate (cope) files or zstat files generated during the first level analysis. The program should assume that all the files are registered to a standard atlas and carry out an independent t-test for every tests for every voxel. The null-hypothesis assumes that the voxel values are normally distributed with a zero mean and unknown variance and different files represent different independently and identically distributed samples of the variables. Your program should output the following files:

1. t-statistics file: `<output prefix>.tstat.nii.gz>` contains the t-statistic value for every voxel in the brain after the group-level hypothesis testing.

2. z-statistics file: `<output prefix>.zstat.nii.gz>` contains the z-statistics value converted from the t-statistic value.

Use your tool to generate every contrast in your assignment 2 for all the subjects that you analyzed in your assignment 3. Register the cope files to the standard atlas using the affine transformation (the mat files) of your assignment 3, and the FSL registration tool flirt (use the option -applyxfm to register using an existing affine transformation file). Now carry out the group level analysis using the `group_analysis.py` developed by you on the registered files.
Create a report comparing the group-level zstat files generated by your program with the corresponding files generated by FSL tool for each of the contrast that you studied. For comparison, carry out a scatter plot of zstat values for every voxel in the brain with x-axis as the group-zstat value as computed by FSL and y-axis as the values calculated by your program. Also, include brain maps of the zstat files for a side-by-side comparison.
Submission Instructions: You need to submit your program `group_analysis.py` on moodle and your report on gradescope. In your report, include a link to drive that has your intermediate PE, tstat and zstat files and final group-level tstat and zstat files for your contrasts. In case your results don't match the FSL results, try to explain why they are different. Your evaluation will be based on correctness of your program and the quality of your report.