Ira A. Fulton Schools of Engineering
Arizona State University

CSE 572: Data Mining

# Project 2: Unsupervised Learning

## Purpose

In this project you will apply clustering techniques to detect the number of clusters in the extracted brain slices of resting state functional magnetic resonance imaging (rs-fMRI) scans

## Objectives

Students will be able to:

- Perform cluster detection in the brain slices images.

## Technology Requirements

Python 3.6 to 3.9

## Project Description

for this project, you will write a python program, that takes a patient's dataset, performs brain slice extraction on it (assignment 1 part a) and then detect the number of clusters present in every extracted brain slice.

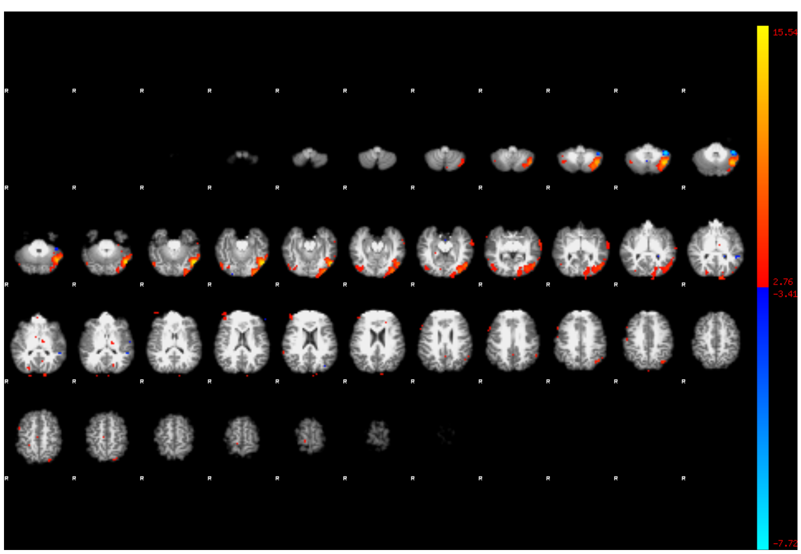

Figure 1

## Directions

There are two main parts to the process:

a) Extract the brain slices in every image (already done in assignment 1).
b) Once you have the brain slices images, you will apply clustering techniques to detect the number of clusters present in every slice. To extract the noticeable big enough cluster, you only need to report the number of clusters whose pixel value is greater than 135 pixels. For example, number of clusters detected in figure 2 is 1.

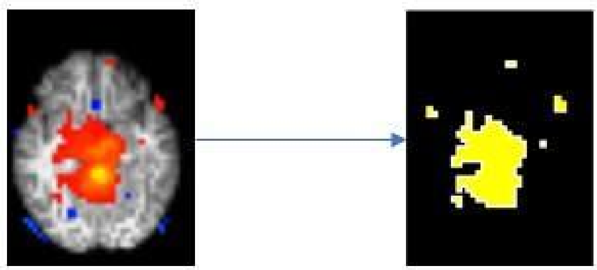

Figure 2

**Dataset:**

You will be given one patient's dataset which will contain approximately 100 spatial ICs of that patient's rs-fMRI scan. All the scans will look like figure 1 (except the blood activation (red/blue clusters) part).

## Submission Directions for Project Deliverables

Submit your Python code in a zip file on the Canvas. Please submit a zipped file containing your code files as

"**yourfirstname_lastname_Assignment1.zip**". Do not create an additional folder, just zip the files directly.

Submit two python files,

a) clustering.py
b) test.py
c) Template image (if your code uses it)

The clustering.py will read all the images (images those end with word "thresh") from the given data and perform slices extraction (you can use your assignment 1a code here). Once you have brain slices images, you will count number of clusters every slice contains using clustering techniques like DBSCAN (this part will be discussed in class). Import all your clustering.py functions to test.py and make test.py read the 'testPatient' folder. The test.py will output two folders named '**Slices' and 'Clusters".** 'Slices' folder will further have 'N' number of folders where N is number of images that ends with 'thresh". Every image folder should contain the brain slices images of that IC_thresh image. **Similarly, another folder 'Clusters' will also have N number of folders and every folder will have clusters detected images (similar to RHS of figure 2) along with one 'csv' file which will report the number of clusters for every slice in that image folder (similar to figure 3).**

**For example, if IC_thresh image has 33 brain slices images then that IC image folder will contain cluster detected images of those 33 slices along with a 'csv' file reporting their count.**

| SliceNumber | ClusterCount |
|---|---|
| 1 | 2 |
| 2 | 1 |
| 3 | 5 |
| 4 | 4 |
| 5 | 2 |

Figure 3

## Evaluation

We will run your test.py on a new patient's data ('testPatient'). Your code should output two folders' "Slices" and "Clusters" which should further have N number of folders each. Each folder will further have brain slices images in case of **Slices** folder and cluster detected images along with csv file reporting the cluster's count in case of **Clusters** folder.

Successful execution of your code will result in 70 points

The remaining 30 points will be based on number of clusters accurately reported by your code.