A Quick Guide for the MixfMRI Package

Wei-Chen Chen[1] **and Ranjan Maitra**[2]

[1]pbdR Core Team
Silver Spring, MD, USA

[2]Department of Statistics
Iowa State University
Ames, IA, USA

Contents

This publication was typeset using LaTeX.

Warning: The findings and conclusions in this article have not been formally disseminated by the U.S. Food and Drug Administration and should not be construed to represent any determination or policy of any University, Institution, Agency, Adminstration and National Laboratory.

This document is written to explain the main function of **MixfMRI** (Chen and Maitra 2018b), version 0.1-0. Every effort will be made to ensure future versions are consistent with these instructions, but features in later versions may not be explained in this document.

1. Introduction

The main purpose of this vignettes is to demonstrate basic usages of **MixfMRI** which is prepared for a developed methodology, simulation sutdies, data analyses in "Improved Activation Detection in Single-Subject fMRI Studies" (Chen and Maitra 2018a). The methodology mainly utilizes model-based clustering (unsupervised) of fMRI data that identifies regions of brain activation associated with stimula. The implemented methods includes 2D and 3D clustering analyses and segmentation analyses for fMRI signals. For simplification, only 2D clustering is demonstrated in this vignettes. In this package, the fMRI signals for brain activation, e.g. response to stimulate of interesting, are represented by p-values for every voxel. The clustering and segmentation analyses mainly identify active voxels/signals (in terms of small p-values) from normal brain behaviors within a single subject.

Note that the p-values may be derived from statistical models where typically a proper experimial design is required. The methods and analyses are for exprotorary purpose only, so that no claims nor statements about significance level of p-values associated with active voxels/signals will be needed/made. However, the analyses allow to prespecify expected upper bounds for the proportion of active voxels/signals that can ease burdens of selecting unreasonable amounts of active vosels.

For large datasets, the methods and analyses are also implemented in a distributed manner especially using SPMD programming framework. The package also includes workflows which utilize SPMD techniques. The workflows serve as examples of data analyses and large scale simulation studies. Several workflows are also built in for automatically process clusterings, hypotheses, merging clusters, and visualizations. See Section 1.5 and files in `MixfMRI/inst/workflow/` for information.

1.1. Dependent Packages

The **MixfMRI** depends on other R packages to be best functioning even though they are not required. Some examples, functions and workflows of the **MixfMRI** may need utilities of those dependent packages. For instance,

Imports: MASS, **Matrix**, **RColorBrewer**, **fftw**, **MixSim**, **EMCluster**.
Enhances: pbdMPI, **AnalyzeFMRI**, **oro.nifti**.

1.2. The Main Function

The main function, `fclust()`, implements model-based clustering using EM algorithm (McLachlan and Krishnan 1996) for fMRI signal data and provides unsupervised clustering results that identify active regions in brain. The `fclust()` contains an initialization method and EM algorithms for clustering fMRI signal data which have two parts:

- `PV.gbd` for p-value of signals associated with voxels, and
- `X.gbd` for voxel information/locations in either 2D or 3D,

where `PV.gbd` is of length N (number of voxels) and `X.gbd` is of dimension $N \times 2$ or $N \times 3$ (for 2D or 3D). Each signal (per voxel) is assumed to follow a mixture distribution of `K` components with mixing proportion `ETA`. Each component has two independent coordinates (one for each part) with density functions: Beta and multivariate Normal distributions, for each part of fMRI signal data.

Beta Density:
The first component (`k = 1`) is restricted by `min.1st.prop` and $Beta(1, 1)$ distribution. The rest `k = 2, 3, ..., K - 1` components have different $Beta(alpha, beta)$ distributions with `alpha < 1 < beta` for all `k > 1` components. This coordinate mainly represents the results of testing statistics for determining activation of voxels (small p-values). Note that the testing statistics may be developed/smoothed/computed from a time course model associated with voxel behaviors. See the main paper Chen and Maitra (2018a) for information.

Multivariate Normal Density:
`model.X = "I"` is for identity covariance matrix of multivariate Normal distribution, and `"V"` for unstructured covariance matrix. `ignore.X = TRUE` is to ignore `X.gbd` and normal density, i.e. only Beta density is used. Note that this coordinate (for each axis) is recommended to be normalized in $(0, 1)$ scale which is the same scale of Beta density. No effects are expected from the rescaling `X.gbd` from modeling perspective.

In this package, the two parts, `PV.gbd` and `X.gbd`, are assumed independent because the dependence were typically ignored in the designed models (time course of voxel activation associated with the stimula) from which the `PV.gbd` are dervided. Even though, the goal of the main function is to provide spatial clusters (in addition to the `PV.gbd`) indicating spatial correlations.

Currently, APECMa (Chen and Maitra 2011) and EM algorithms are implemented with EGM algorithm (Chen *et al.* 2013) to speed up convergence when MPI and **pbdMPI** (Chen *et al.* 2012) are available. RndEM initialization (Maitra 2009) is also implemented for obtaining good initial values that may increase chance of convergence.

1.3. Datasets

The package has been built with several datasets including

- three 2D phantoms, `shepp0fMRI`, `shepp1fMRI`, and `shepp2fMRI`,
- one 3D voxels dataset, `pstats`, in p-values,
- two small 2D voxels datasets, `pval.2d.complex` and `pval.2d.mag`, in p-values
- two toy examples, `toy1` and `toy2`.

1.4. Examples

The scripts in `MixfMRI/demo/` have several examples that demonstrate the main function, the example datasets and other utilities in this package. For quick start,

- the scripts `MixfMRI/demo/fclust2d.r` and `MixfMRI/demo/fclust3d.r` show the basic usage of the main function `fclust()` using two toy datasets,
- the scripts `MixfMRI/demo/maitra_2d.r` and `MixfMRI/demo/shepp.r` show and visualize examples on how to generate simulated datasets with given overlap levels, and
- the scripts `MixfMRI/demo/alter_*.r` show alternative methods.

1.5. Workflows

The package also have several workflows established for simulation studies. The main examples are located in `MixfMRI/inst/workflow/simulation/`. See the file `create_simu.txt` that generates scripts for simulations.

The files under `MixfMRI/inst/workflow/spmd/` have main scripts of workflows. Note that MPI and **pbdMPI** are required for workflows because these simulations require potentially long computing time.

2. Demonstration

The examples presented below may not be real nor represent any meaningful brain activity. However, the purpose is to assess the performance of our proposed clustering methodology when any signal detected from voxels which can be active in vary ways.

2.1. 2D Phantoms

Three 2D phantoms built in the **MixfMRI** can be plotted in R from the demo as simple as

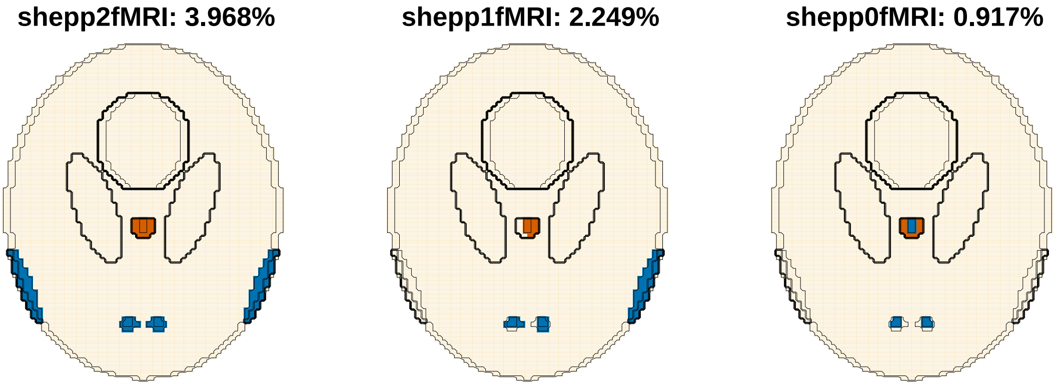
Maitra's Phantoms

```
R> demo(maitra_phantom, package = "MixfMRI", ask = F)
```

which performs the code in `MixfMRI/demo/maitra_phantom.r`. The R command should give a plot similar with Figure 1 containing three 2D slices of a brain. Each phantom may have different amounts of activated voxels (in terms of small p-values). Colors represent different activations such as distributions of p-values that can be computed by hypothesis testing of prespecified statistical models associated with experiments designed to discover brain functions responding to prespecified stimula. The total proportions of active voxels are listed in the title of each phantom.

The examples used below mimic some active regions (in 2D) depending on different types of stimula that may trigger responses in the brain. Hypothetically, the voxels may be active by regions, but each region may not be active in the same way (or magnitude) even they may need to collectively respone to the stimula (e.g. due to time delay, responding orders, or sensitivity of study deisgns).

Figure 1: Maitra's Phantoms.



For example, only 3.968% of voxels in the `shepp2fMRI` phantom are active and indicate by two different colors (light blue and light brown) for different activation where p-values may be smaller than 0.05 and may follow two Beta distributions (with different configurations) for active voxles and one uniform distribution (i.e. $Beta(1, 1)$) for inactive voxels.

The following code provides some counts for each groups of active and inactive voxels in the `shepp2fMRI` phantom.

Summary of shepp2fMRI Phantoms

```
> table(shepp2fMRI, useNA="always")
shepp2fMRI
    0     1     2  <NA>
13408   472    82 51574
```

The summary says that this phantom has three kinds of activations with group ids: 0, 1, and 2. There are 13,408 voxels belonging to cluster 0 (inactive), followed by 472 voxels belonging to cluster 1 (active & high lighted in light blue in Figure 1), and 82 voxels belonging to cluster 2 (active & high lighted in light brown in Figure 1). There are 51,574 pixels (NA) of this imaging dataset which are not within the brain (contour by the black line in Figure 1). See Section 2.2 for information of generating p-values from a mixture of three Beta distributions.

2.2. 2D Simulations

The **MixfMRI** provides a function `gendataset(phantom, overlap)` to generate p-values of activations. The function needs two arguments: `phantom` and `overlap`. The `phantom` is a map containing voxel group id's where p-values will be simulated from a mixture Beta distribution with certain mixture level specified by the `overlap` argument. The example can be found in `MixfMRI/demo/maitra_2d.r` and can be done in R as simple as

Simulations of Active Voxels

```
R> demo(maitra_2d, package = "MixfMRI", ask = F)
```

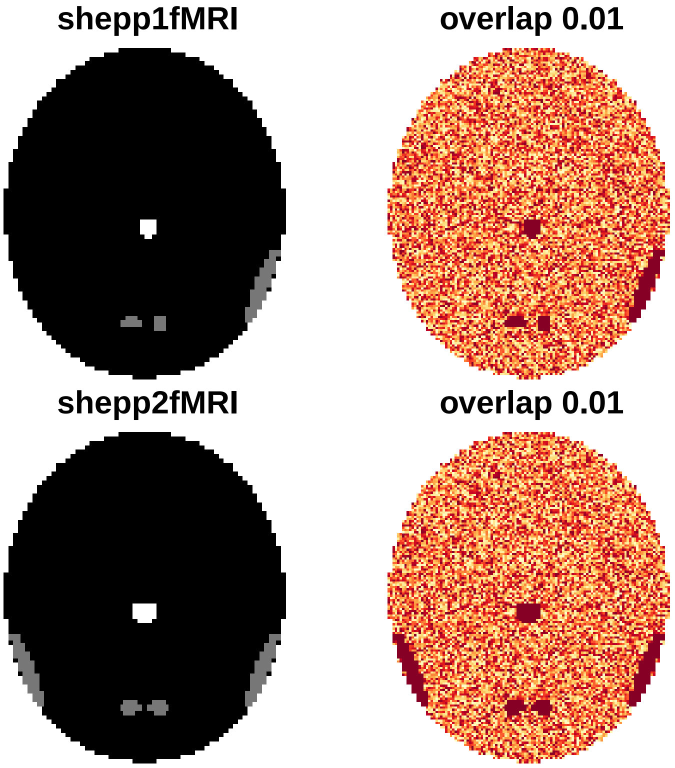
Note the overlap repesents similarity of activation signals. The higher the overlap, the more difficult to distinguish the clusters.

The command above should give a plot similar to Figure 2 containing group id's on the left and their associated p-values for stimulus responses on the right. The top row displays examples for phantom `shepp1fMRI`, and the bottom row displays examples for phantom `shepp2fMRI`.

- Inside the brain, the group id 2's are indicated by white (active is highly associated with stimula due to experiment design), 1's are indicated by light gray (slightly active), and 0's are indicated by dark gray (inactive). Note the white color was light blue and the light gray was light brown in Figure 1
- The simulated p-values are colored by red-orange-yellow from 0 to 1. Note that small p-values (red) may also occur at inactive voxels.

See Figure 3 for the distribution of simulated p-values for the phantom `shepp2fMRI`.

Figure 2: Voxels Activation Simulations



The methodology and analyses developed in this package are aiming for identifying those active voxels in spatial clusters. For example, regions of active voxels associated with playing certain sports. When an experiment was conducted/designed to detect brain behaviors, the statistical model and the p-values of the treatment effect should be able to reflact the voxel activations. Typically, the statistical tests are done independently voxel by voxel due to complexity of computation and modeling. This package provides post hoc clustering that adds spatial contents to p-vlaues and helps to isolate meaningful regions clouded with many small p-values. See Chen and Maitra (2018a) for information of clustering performance and comprehensive assessments for this post hoc approach.

2.3. 2D Clustering

The example can be found in `MixfMRI/demo/maitra_2d_fclust.r` as simple as

Clustering of Active Voxels

```
R> demo(maitra_2d_fclust, package = "MixfMRI", ask = F)
```

This demo (explained below) is to cluster the simulated p-values (see Section 2.2) using the developed method.

Code of `maitra_2d_fclust.r`

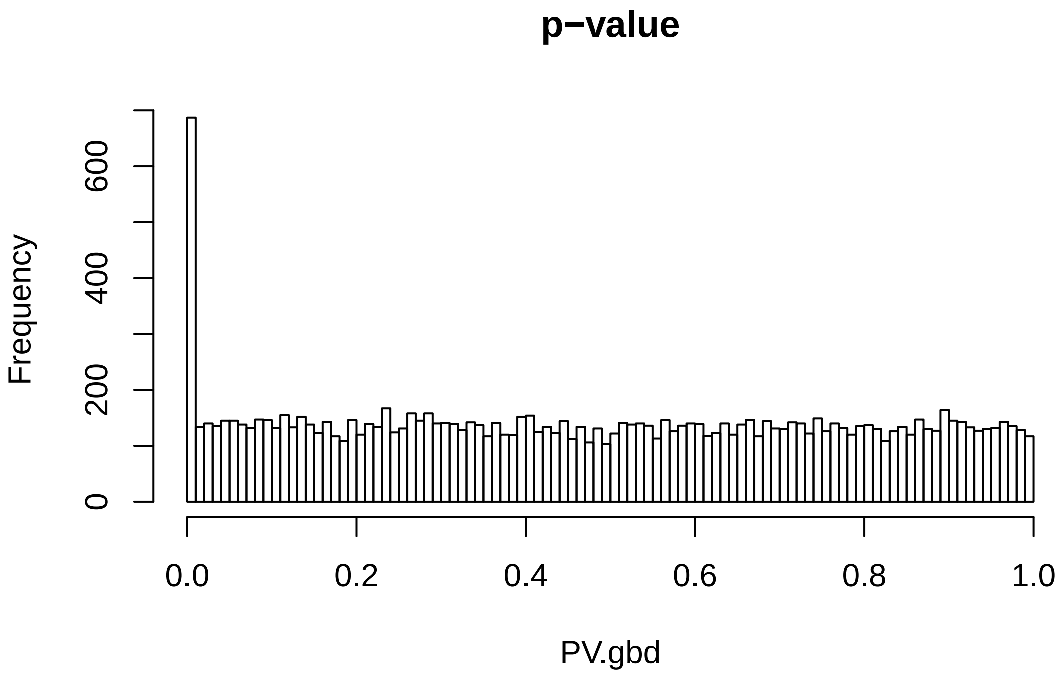
```
library(MixfMRI, quietly = TRUE)
set.seed(1234)
da <- gendataset(phantom = shepp2fMRI, overlap = 0.01)$pval

### Check 2d data.
id <- !is.na(da)
PV.gbd <- da[id]
# pdf(file = "maitra_2d_fclust.pdf", width = 6, height = 4)
hist(PV.gbd, nclass = 100, main = "p-value")
# dev.off()

### Test 2d data.
id.loc <- which(id, arr.ind = TRUE)
X.gbd <- t(t(id.loc) / dim(da))
ret <- fclust(X.gbd, PV.gbd, K = 3)
print(ret)

### Check performance
library(EMCluster, quietly = TRUE)
RRand(ret$class, shepp2fMRI[id] + 1)
```

In the code above, the histogram of simulated p-values is plotted in Figure 3. Then, the `fclust(X.gbd, PV.gbd, K = 3)` groups voxels in three clusters. At the end, `ret` saves the clustering results. The `print(ret)` from the above code will show the results below in detail:

- `N` is the total number of voxels to be clustered.
- `K` is the total number of clusters.
- `n.class` is the number of voxels for each cluster.
- `ETA` is the mixing proportion of each cluster.
- `BETA` is the parameters of Beta distributions (by column).
- `MU` is the centers of clusters (location within the brain).
- `SIGMA` is the dispersion of clusters.

The numbers of voxels for each cluster are 13,394, 184, and 384 associated with new cluster ids: 0, 1, and 2, respectivelly. Comparing with the true classifications (see the table in Section 2.1), the adjusted Rand index gives 0.9749 indicating good consistence between true active and clustered results.

Figure 3: Activation (p-values) Distribution. The x-axis is for the p-values.



Outputs of Clustering

```
R> print(ret)
Algorithm: apecma  Model.X: I  Ignore.X: FALSE
- Convergence: 1  iter: 16  abs.err: 0.02091979  rel.err: 7.375343e-07
- N: 13962  p.X: 2  K: 3  logL: 28364.52
- AIC: -56693.04  BIC: -56557.25 ICL-BIC: -55712.18
- n.class: 13394 184 384
- init.class.method:
- ETA: (min.1st.prop: 0.8  max.PV: 0.1)
[1] 0.95266704 0.01307847 0.03425449
- BETA: (2 by K)
     [,1]         [,2]       [,3]
[1,]    1 1.127244e-01 0.04237128
[2,]    1 4.429518e+04 1.00000130
- MU: (p.X by K)
          [,1]      [,2]      [,3]
[1,] 0.5013538 0.3105460 0.5917377
[2,] 0.5076080 0.3718145 0.3749842
- SIGMA: (d.normal by K)
           [,1]         [,2]        [,3]
[1,] 0.01271198 0.0001859628 0.009462210
[2,] 0.02186662 0.0011582052 0.004284016

R> RRand(ret$class, shepp2fMRI[id] + 1)
```

```
    Rand  adjRand   Eindex
  0.9964   0.9749   1.7012
```

References

Chen WC, Maitra R (2011). "Model-based clustering of regression time series data via APECM – an AECM algorithm sung to an even faster beat." *Statistical Analysis and Data Mining*, **4**, 567–578.

Chen WC, Maitra R (2018a). "Improved Activation Detection in Single-Subject fMRI Studies." *manuscript*.

Chen WC, Maitra R (2018b). "MixfMRI: fMRI Clustering Analysis." R Package, URL http://cran.r-project.org/package=MixfMRI.

Chen WC, Ostrouchov G, Pugmire D, Prabhat M, Wehner M (2013). "A Parallel EM Algorithm for Model-Based Clustering with Application to Explore Large Spatio-Temporal Data." *Technometrics*, **55**, 513–523.

Chen WC, Ostrouchov G, Schmidt D, Patel P, Yu H (2012). "pbdMPI: Programming with Big Data – Interface to MPI." R Package, URL https://cran.r-project.org/package=pbdMPI.

Maitra R (2009). "Initializing Partition-Optimization Algorithms." *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **6**, 144–157.

McLachlan G, Krishnan T (1996). *The EM Algorithm and Extensions*. John Wiley & Sons.