

# FOLD

This is the GitHub repository for Bayesian Clustering via Fusing of Localized Densities. In this article, we present Fusing of Localized Densities (FOLD), a decision theoretic clustering method that focuses on grouping the *localized densities* from a mixture model. Here, you will find the code required to reproduce all plots, the simulation studies, and the application to the cell line dataset. Here is a quick summary of what files are included in this repository.

## Dependencies

After downloading the repository, run the following code in your console to install all dependencies.  
<source("dependencies.R")>

## Code for Reproducing Results

- Application to the GSE81861 cell line dataset (Figures 4-6, Supplement Figure H.7): `cells.R`.
- Gaussian mixture, skew Gaussian mixture, and skew-symmetric mixture simulation studies (Figures A.2-A.4): `SimStudy_mvn.Rmd`, `SimStudy_mvsn.Rmd`, and `SimStudy_nongauss.Rmd`, respectively.
- Illustration of oracle clusterings (Figures B.5 and B.6): `convergence_plots.R`.
- Credible ball simulation study: `simulatespirals.R`.

## Code for Reproducing Additional Figures

- Example of over-clustering (Figure 1): `introexample.R`.
- Clustering the moons data (Figures 2 and 3): `locationmoons.R`.
- Contour plots of simulation examples (Supplement Figure A.1): `contourmaker.R`.
- Plot of the spirals data (Supplement Figure I.8): `spiralsplot.R`.

## Main Functions

- The pairwise Hellinger distance matrix is computed in the file `rcppfuncts/mnorm_D_arma.cpp`. `mnorm_D_arma()` takes a given sample of localized densities and computes their pairwise Hellinger distance matrix. `makeHellingerAvg()` computes the average distance matrix across MCMC samples.
- The oracle pairwise Hellinger distance matrix is computed in `rcppfuncts/oracle.cpp` (for location-scale GMMs) and in `rcppfuncts/oracle_convergence.cpp` (for location GMMs).
- The Gibbs sampler for the location-scale GMM used in the cell line data application and the simulations is `rfuncts/mvnorm_gibbs.R`. A similar implementation for location GMMs can be found in `rfuncts/locationnorm_gibbs.R`.
- Functions for computing the credible ball can be found in `rfuncts/foldball.R` (for location-scale GMMs) and `rfuncts/loc_foldball.R` (for location GMMs).
- The implementation of the SALSO algorithm is given in `rfuncts/r_simple_SALSO.R`.