# Hierarchical Mode Association Clustering Package - MATLAB

August 13, 2007

## 1 Introduction

HMAC is an agglomerative clustering method based on nonparametric kernel density estimate. It merges observations hierarchically by increasing the bandwidth in the kernel function. Ridge EM algorithm provides ridgelines between two clusters acquired from HMAC and it defines the pairwise separability. The clustering method using HMAC has properties combining clustering by mixture modeling and linkage and it does not require a cluster to be parametric, so it is robust for non-Gaussian clusters. In addition, clustering based on the separability helps small noisy clusters being merged to main clusters. [1] J. Li, S. Ray , B. G. Lindsay contains details on the algorithms.
This package is programmed on MATLAB 7.0.4.
Comments or Questions about the package: `ychung@psu.edu`

## 2 What is included?

This package includes `HMAC.m`, `REM.m`, `example.m`, built-in functions and several sample data(.txt)

1. `HMAC` performs the hierarchical mode association clustering, providing cluster levels for each bandwidth, and modes and cluster assignment of each observation.

2. `REM` provides ridgelines between all possible pair of clusters using REM, and calculates separabilities.

3. `example` contains Matlab code generating plots which help understanding clustering results.

## 3 Program Usages

`HMAC` and `REM` should be saved under your current MATLAB directory. M-files in `builtin functions` directory in this package are also supposed to be in the MATLAB directory which might be in Program Files. They might be already exist if you installed `ops`, `stats`, `matlab` and `bioinfo` toolboxes . If not, make it sure your folder containing them.

## 3.1 HMAC

### Syntax

```
[n_cluster,level, mode, member] = HMAC(data,sigmas)
```

### Description

1. Inputs

   - `data` is a matrix in which each row contains data for each observation and each column represent a variable.

   - `sigmas` is a sequence of bandwidth used for the kernel function.

2. Outputs

   - `n_cluster` is a vector with the same length as `sigmas` whose elements are the number of cluster obtained by corresponding bandwidth.

   - `level` is a vector with the same length as `sigmas` containing the level of clustering obtained by corresponding bandwidth.

   - `mode` is an structure that `mode.c1` includes a matrix in which modes of each cluster acquired by 1st bandwidth are stacked, and `mode.c2` includes one by 2nd bandwidth, and so on.

   - `member` is a matrix with size (# of observations) by (# of levels) whose $i$-th column contains assigned cluster index to corresponding observation in $i$-th level.

### Example

```
data=textread('glass2d.txt');
s=max(std(data));sigmas=s*0.1:s*1.9/19:s*2;
[n_cluster,level, modes, members]=HMAC(data, sigmas);

n_cluster =

    Columns 1 through 13

        21    11     4     3     3     3     2     1     1     1     1     1     1

    Columns 14 through 20

        1     1     1     1     1     1     1

 level =
    Columns 1 through 13

        1     2     3     4     4     4     5     6     6     6     6     6     6

    Columns 14 through 20

        6     6     6     6     6     6     6

modes =

    c1: [21x2 double]
    c2: [11x2 double]
    c3: [4x2 double]
    c4: [3x2 double]
    c5: [2x2 double]
    c6: [-1.2069 10.7834]

members =

     1     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     2     1     1     1     1     1
     3     1     1     1     1     1
     ....
```

## 3.2 REM

### Syntax

```
[ridges,S,sep,density]=REM(data, mode,sigma, member, alpha)
```

### Description

1. Inputs

   - `data` is a data matrix used for clustering which is the same as `data` in `HMAC`.

- `mode` is a matrix whose rows are mode of corresponding cluster. The matrix which is contained in a field of `mode` from `HMAC` output can be used. The number of columns in `data` should be the same as the number of columns in `mode`.

- `sigma` is a scalar used as a bandwidth in clustering. It might be an element of input `sigmas` for `HMAC` corresponding `mode`.

- `member` is a vector whose element shows the assigned cluster of each observation. It might be the column of output `member` in `HMAC` which obtained from clustering level matches with `sigma` and `mode`. The maximum value in `member` should be the same as the number of rows in `mode`.

- `alpha` is a sequence between 0 and 1 used for a grid of ridgelines.

2. Outputs

- `ridges` is a structure containing ridgelines. `ridges.c1c2` is a matrix of ridgeline between cluster 1 and 2, and `ridges.c1c3` is a ridgeline between cluster1 and 3, and so on.

- `S` is a matrix whose $(i, j)$ element is a pairwise separability between $i$-th and $j$-th cluster.

- `sep` is a matrix whose first column contains the cluster sizes and the second column contains separabilities.

- `density` is a structure containing the densities of each ridgelines. `density.c1c2` is a vector of density function along the ridgeline between cluster 1 and cluster2, evaluated at each grid point of `alpha`.

**Example**
This example continues from the example in `HMAC`.

```
i=3; %choose cluster level to get ridgelines and separability
k=min(find(level==i));
mode=eval(['modes.c',int2str(i)]); choose the modes at the i-th level
[ridge,S,sep,density]=REM(data,mode,sigmas(k), members(:,i), 0:0.05:1);


ridge =

    c1c2: [21x2 double]
    c1c3: [21x2 double]
    c1c4: [21x2 double]
    c2c3: [21x2 double]
    c2c4: [21x2 double]
    c3c4: [21x2 double]
```

```
S =

    1.0000    0.9904    1.0000    0.9347
    0.8808    1.0000    0.8053    0.9992
    0.9999    0.2834    1.0000    1.0000
    0.8386    0.9998    1.0000    1.0000

sep =

    73.0000    0.9347
    6.0000    0.8053
    1.0000    0.2834
    25.0000    0.8386

density =

    c1c2: [21x1 double]
    c1c3: [21x1 double]
    c1c4: [21x1 double]
    c2c3: [21x1 double]
    c2c4: [21x1 double]
    c3c4: [21x1 double]
```

## References

[1] J. Li, S. Ray, B. G. Lindsay, "A nonparametric statistical approach to clustering via mode identification," *Journal of Machine Learning Research*, under revision, 2007