

CARP : The Clustering Algorithms' Referee Package

Version 1.0 – Manual

Volodymyr Melnykov*
Department of Statistics
North Dakota State University
Fargo, ND 58103
USA.

E-mail: `volodymyr.melnykov@ndsu.edu`

Ranjan Maitra*
Department of Statistics
Iowa State University
Ames, IA 50011-1210
USA.

E-mail: `maitra@iastate.edu`

April 9, 2010

*Supported in part by the National Science Foundation (NSF) through its CAREER Award No. DMS-0437555.

Contents

1	Description	3
1.1	Project homepage	3
1.2	Dependencies	3
2	Installation	3
3	Usage	4
3.1	Parameters	4
3.2	Details	4
3.3	Other capabilities	5
4	Illustrations	6
4.1	Example: Evaluating clustering algorithms with spherical dispersion structures and equal representation	6
4.2	Example: Evaluating clustering algorithms with general dispersion structures and unequal representation	10
4.3	Example: Evaluating a clustering algorithm from data simulated earlier and stored in files	11
4.4	Example: Calculating overlap between identified classes in a dataset	13
4.5	Example: Use of C-MixSim as a standalone package	13
5	Conclusions	15

1 Description

The C-package `CARP` is a convenient and easy tool for evaluating performance of clustering algorithms. The underlying methodology is based on first simulating Gaussian mixture models according to prespecified levels of average and maximum pairwise overlaps. The concept of overlap is defined as the sum of two misclassification probabilities (Maitra and Melnykov, 2010). Datasets are then simulated from the realized Gaussian mixtures. The software implementing this phase is called `C-MixSim` and can be invoked standalone. This concludes the first phase of the procedure. In the second phase, the clustering algorithm being evaluated is run on the generated datasets. We provide an example here using an agglomerative hierarchical clustering algorithm `hierclust` which is included. The third phase compares obtained and true groupings. By default, the comparison measure is the Adjusted Rand index of Hubert and Arabie (1985) but the user can also provide some other measure in executable form. Upon conclusion, `CARP` provides a distribution of the desired performance measure for the clustering method being evaluated at the preferred setting. This provides for a detailed understanding of the performance of the clustering algorithm being evaluated. `CARP` is released under the GNU GPL license.

1.1 Project homepage

The project homepage is located on the Worldwide Web at:

- <http://www.public.iastate.edu/~maitra/Software/CARP.html>

The webpage contains instructions, an updated version of this manual and the source code in a zipped format.

1.2 Dependencies

`CARP` has the following dependencies required to be installed before installing `CARP`:

- `gcc` compiler
- `glibc` library

2 Installation

`CARP` can be installed in two easy steps:

1. Extract files from `CARP_v1.0.tar.gz` using

```
tar -xzf CARP_v1.0.tar.gz
```
2. Compile the files running the command

```
make CARP
```

This creates the executable files: `CARP`, `C-MixSim`, `unittest`, `AdjRand` and `hierclust`. The last two are optional files and the command to not create either may be removed from the included `makefile` if needed.

To check the integrity of the package, run the command

- `make check`.

To remove the installed files, use the command

- `make clean`.

3 Usage

3.1 Parameters

To run CARP, a command of the following format has to be used:

```
./CARP set-of-parameters
```

The following parameters can be specified in the above command line:

- b: average overlap (no default value)
- m: maximum overlap (no default value)
- p: number of dimensions (2 by default)
- K: number of mixing components (2 by default)
- n: number of observations generated from every mixture (0 by default)
- #: number of simulated mixtures (1 by default)
- 0: name of clustering program (if this option is not specified, only C-MixSim is run)
- 1: name of partition analyzing program (Adjusted Rand index is used by default; program AdjRand)
- s: spherical covariance matrix structure (non-spherical by default if option is unspecified)
- e: maximum eccentricity (0.90 by default)
- z: smallest mixing proportion (equal mixing proportions $1/K$ by default)
- u: upper bound for Uniform(0, *upper-bound*) distribution from which mean vectors are generated
- r: maximum number of resimulations (100 by default)
- a: accuracy of estimation (1e-06 by default)
- l: maximum number of integration terms used by `qfc` function (1e06 by default)
- P: file containing mixing proportions (`Pi.dat` by default)
- M: file containing mean vectors (`Mu.dat` by default)
- S: file containing covariance matrices in triangular form (`LTSigma.dat` by default)
- D: working directory (`DATA` by default)
- I: file containing numbers of observations generated from every cluster (`Nk.dat` by default)
- i: file containing estimated classifications (`idEst.dat` by default)
- X: file containing simulated datasets (`x.dat` by default)
- W: file containing maps of pairwise overlaps (`overMap.dat` by default)
- C: file containing characteristics of simulated mixtures (`overBarMax.dat` by default)
- R: file containing index values (Adjusted Rand index and `AR.dat` by default)

3.2 Details

Upon launching CARP, three stages of the program are processed. The first stage is responsible for the simulation of Gaussian mixtures with prespecified level of complexity expressed in terms

of pairwise overlap and for the generation of datasets from these mixtures. At this stage, CARP invokes C-MixSim. The admissible options are the following: `-b`, `-m`, `-p`, `-K`, `-n`, `-#`, `-s`, `-e`, `-z`, `-u`, `-r`, `-a`, `-l`, `-P`, `-M`, `-S`, `-D`, `-I`, `-X`, `-W`, `-C`.

At the second stage, a user-specified clustering method has to be run for the simulated datasets. For illustration purposes, the hierarchical clustering algorithm `hierclust` is employed. User's clustering program has to be able to accept the following options: `-p`, `-K`, `-n`, `-#`, `-D`, `-i`, `-X`. The program has to obtain partitionings and write them into the file specified by the option `-i`.

At the third stage, true classification vectors are compared with obtained ones to assess the performance of the clustering algorithm. The Adjusted Rand index is incorporated as a default measure of similarity between the two classifications. A user, however, can specify some other program for comparing the obtained and true partitionings. Then, the program should comply with the following options: `-K`, `-n`, `-#`, `-D`, `-I`, `-i`, `-R`. Calculated values of the provided similarity measure have to be written to the file specified by the option `-R`.

If both options `-b` and `-m` are specified (see Example 4.1), CARP produces a mixture satisfying both characteristics, average and maximum overlap. If one option, `-b` or `-m` is specified (see Example 4.2), a mixture satisfying the prespecified value is generated. The working directory is specified by the option `-D`. The obtained parameters will be saved to the files specified by options `-P`, `-M`, and `-S` while samples drawn from the mixtures will be saved into the file specified by the option `-X`. The file provided with the option `-I` contains sample sizes for every mixture component. In addition, the map of misclassification probabilities will be saved into the file specified by the option `-W` while average and maximum overlaps as well as the row and column numbers of the components that produced maximum overlap are stored in the file given by the option `-C`. The element with the index (i, j) in the misclassification map represents the probability that X simulated from the i th component is classified to the j th component.

3.3 Other capabilities

If both options, `-b` and `-m` are not provided, CARP reads parameters from the files specified by the options `-P`, `-M`, `-S` and computes misclassification probabilities for the components of the given mixture model. Note that the options `-p` and `-K` have to be correctly specified. The options `-n` and `-#` specify the sample size of a sample drawn from every generated mixture and the number of such mixtures correspondingly. If it is desired to use datasets stored in a file specified by the option `-X`, the sample size should be given as a negative number, for example: `-n-100` instead of the traditional `-n100`. In that case, new datasets will not be simulated. Note that this capability is available only in the mode when mixture parameters are not simulated but rather read from files. This capability can be useful in the case when it is desired to run several clustering algorithms on the same set of simulated data. Example 4.3 provides an illustration on the use of the these features.

Another application of CARP's capabilities is for assessing the level of clustering complexity of derived and existing groups in datasets based on the set of estimated mixture model parameters. Example 4.4 provides an illustration on the use of this feature for learning the properties of the famous *Iris* dataset.

CARP also has a capability to simulate finite mixture models and datasets from them without launching the second and third stages. Thus, the simulation stage would be the only one to run. In

order to do so, the option `-0` has to be omitted. The program `C-MixSim` will be run as illustrated in Example 4.5.

4 Illustrations

4.1 Example: Evaluating clustering algorithms with spherical dispersion structures and equal representation

This example is to illustrate the simulation of three 2-dimensional 4-component mixtures with spherical covariance matrices and equal mixing proportions according to pre-specified levels of maximum and average overlap, drawing samples of 100 observations from simulated mixtures, running the clustering program `hierclust`, and computing the Adjusted Rand index. Consider the following command:

```
./CARP -m0.1 -b0.05 -p2 -K4 -s -n100 -#3 -DDATA -Xdata.dat -IIDtrue.dat
-iIDest.dat -RAAdjRand.dat -0hierclust
```

CARP runs the following three steps:

1. *Running C-MixSim*

```
./C-MixSim -m0.1 -b0.05 -p2 -K4 -s -n100 -#3 -DDATA -Xdata.dat
-IIDtrue.dat
```

This command simulates three 2-dimensional 4-component mixtures with spherical covariance matrices, equal mixing proportions, maximum overlap 0.1, and average overlap 0.05. 100 observations will be generated from every mixture. The working directory is `DATA`. Simulated samples will be stored into the file `data.dat` while the classifications will be stored into the file `IDtrue.dat`. The corresponding output is provided below.

Output:

MIXTURE MODEL #1:

The desired overlap has been met...

Map of misclassification probabilities:

```
[0][0] : 1.000000 [0][1] : 0.015200 [0][2] : 0.003266 [0][3] : 0.033826
[1][0] : 0.012737 [1][1] : 1.000000 [1][2] : 0.000054 [1][3] : 0.035891
[2][0] : 0.003856 [2][1] : 0.000074 [2][2] : 1.000000 [2][3] : 0.036315
[3][0] : 0.049968 [3][1] : 0.064109 [3][2] : 0.044703 [3][3] : 1.000000
```

Average Overlap: 0.050000

Maximum Overlap: 0.100000 (components: 1 and 3)

Mixture parameters:

Pi:

```
0.250000 0.250000 0.250000 0.250000
```

```

Mu:
[0] : 0.471083 0.675651
[1] : 0.335962 0.337816
[2] : 0.980450 0.580291
[3] : 0.652717 0.362861
Sigma:
[0] :
0.007961 0.000000
0.000000 0.007961
[1] :
0.005868 0.000000
0.000000 0.005868
[2] :
0.010726 0.000000
0.000000 0.010726
[3] :
0.015040 0.000000
0.000000 0.015040
Dataset with cluster sizes Nk = 28 21 29 22 has been generated...

```

MIXTURE MODEL #2:

```

The desired overlap has been met...
Map of misclassification probabilities:
[0][0] : 1.000000 [0][1] : 0.030979 [0][2] : 0.029165 [0][3] : 0.001868
[1][0] : 0.013597 [1][1] : 1.000000 [1][2] : 0.010535 [1][3] : 0.006902
[2][0] : 0.048703 [2][1] : 0.036413 [2][2] : 1.000000 [2][3] : 0.056377
[3][0] : 0.002374 [3][1] : 0.019464 [3][2] : 0.043623 [3][3] : 1.000000
Average Overlap: 0.050000
Maximum Overlap: 0.100000 (components: 2 and 3)

```

```

Mixture parameters:
Pi:
0.250000 0.250000 0.250000 0.250000
Mu:
[0] : 0.203039 0.189259
[1] : 0.485454 0.285278
[2] : 0.338648 0.623985
[3] : 0.795138 0.499927
Sigma:
[0] :
0.010917 0.000000
0.000000 0.010917
[1] :

```

```

0.002621 0.000000
0.000000 0.002621
[2] :
0.025417 0.000000
0.000000 0.025417
[3] :
0.016861 0.000000
0.000000 0.016861
Dataset with cluster sizes  $N_k = 26 \ 22 \ 21 \ 31$  has been generated...

```

MIXTURE MODEL #3:

```

The desired overlap has been met...
Map of misclassification probabilities:
[0][0] : 1.000000 [0][1] : 0.000000 [0][2] : 0.021907 [0][3] : 0.020430
[1][0] : 0.000000 [1][1] : 1.000000 [1][2] : 0.013279 [1][3] : 0.000000
[2][0] : 0.078093 [2][1] : 0.051412 [2][2] : 1.000000 [2][3] : 0.041697
[3][0] : 0.049043 [3][1] : 0.000000 [3][2] : 0.024139 [3][3] : 1.000000
Average Overlap:  0.050000
Maximum Overlap:  0.100000 (components:  0 and 2)

```

Mixture parameters:

```

Pi:
0.250000 0.250000 0.250000 0.250000

```

```

Mu:
[0] : 0.694165 0.531804
[1] : 0.166167 0.386339
[2] : 0.466549 0.391484
[3] : 0.927160 0.438973

```

```

Sigma:
[0] :
0.002244 0.000000
0.000000 0.002244
[1] :
0.001815 0.000000
0.000000 0.001815
[2] :
0.024865 0.000000
0.000000 0.024865
[3] :
0.009984 0.000000
0.000000 0.009984

```

Dataset with cluster sizes $N_k = 26 \ 24 \ 29 \ 21$ has been generated...

2. Running clustering algorithm

```
./hierclust -p2 -K4 -n100 -#3 -DDATA -Xdata.dat -iIDest.dat
```

At this stage, the program `hierclust` performing hierarchical clustering is run. The estimated classification will be written into the file `IDest.dat`. The output contains the estimated classification vectors.

Output:

OBTAINED CLASSIFICATIONS:

Dataset #1:

```
0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 2 1 0 1 2 1 2 2 0 3 1 3 1 2 1 3 2
3 1 1 0
```

Dataset #2:

```
0 0 0 0 3 0 0 0 0 3 0 0 0 0 0 0 3 0 0 0 0 3 0 0 0 0 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 0 2 3 2 3 2 2 2 0 2 2 2 2 2 2 2
2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 3 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1
```

Dataset #3:

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 0 0 1 3 3 3 3 1 0 0 3 3 3 3
1 3 3 0 1 3 0 3 0 1 3 3 0 1 3 2 2 2 2 2 2 2 2 0 2 0 2 2 2 2 2 2 2
2 2 2 2
```

3. Computing Adjusted Rand index

```
./AdjRand -K4 -n100 -#3 -DDATA -IIDtrue.dat -iIDest.dat -RAAdjRand.dat
```

This command computes three Adjusted Rand index values and saves them into the file `AdjRand.dat`. The following is the output from the program.

Output:

ADJUSTED RAND VALUES:

Dataset #1: 0.722711

Dataset #2: 0.753868

Dataset #3: 0.622881

4.2 Example: Evaluating clustering algorithms with general dispersion structures and unequal representation

This example illustrates simulating a 2-dimensional 4-component mixture with general covariance matrices and mixing proportion being no less than 0.1 according to the specified level of maximum overlap, drawing a sample of size 100 from the simulated mixture, running the clustering program `hierclust`, and computing the Adjusted Rand index. Consider the following command:

```
./CARP -m0.001 -p2 -K4 -n100 -z0.1 -0hierclust
```

The following is the description of the three stages of CARP.

1. *Running C-MixSim*

```
./C-MixSim -m0.001 -p2 -K4 -n100 -z0.1
```

The command simulates one 2-dimensional 4-component mixture with general covariance matrices, mixing proportions greater or equal to 0.1, and maximum overlap of 0.001. 100 observations will be generated from the mixture. The default names will be used for all output files. The default working directory (DATA) is used.

Output:

The desired overlap has been met...

Map of misclassification probabilities:

[0][0] : 1.000000 [0][1] : 0.000000 [0][2] : 0.000000 [0][3] : 0.000921

[1][0] : 0.000000 [1][1] : 1.000000 [1][2] : 0.000000 [1][3] : 0.000000

[2][0] : 0.000000 [2][1] : 0.000000 [2][2] : 1.000000 [2][3] : 0.000000

[3][0] : 0.000079 [3][1] : 0.000000 [3][2] : 0.000000 [3][3] : 1.000000

Average Overlap: 0.000167

Maximum Overlap: 0.001000 (components: 0 and 3)

Mixture parameters:

Pi:

0.344318 0.297326 0.148341 0.210016

Mu:

[0] : 0.537026 0.212149

[1] : 0.235874 0.026386

[2] : 0.866591 0.579785

[3] : 0.451681 0.219715

Sigma:

[0] :

0.000131 0.000118

0.000118 0.000274

[1] :

0.000277 - 0.000187

-0.000187 0.000272

```
[2] :
0.000042 - 0.000028
-0.000028 0.000109
[3] :
0.000387 - 0.000310
-0.000310 0.000555
Dataset with cluster sizes Nk = 36 27 14 23 has been generated...
```

2. *Running clustering algorithm*

```
./hierclust -p2 -K4 -n100
```

This runs the program `hierclust` which performs hierarchical clustering. The obtained estimated classification is written into the default file `idEst.dat`.

Output:

```
OBTAINED CLASSIFICATIONS:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3
```

3. *Computing Adjusted Rand index*

```
./AdjRand -K4 -n100
```

The above command computes the Adjusted Rand index and writes it into the default file `AR.dat`.

Output:

```
ADJUSTED RAND VALUES:
1.000000
```

4.3 **Example: Evaluating a clustering algorithm from data simulated earlier and stored in files**

This example is on reading a 2-dimensional 4-component mixture and a dataset of size 100 from a file (storing such data, upon a first invocation to `C-MixSim`), running the clustering program `hierclust`, and computing the Adjusted Rand index. Consider the following command:

```
./CARP -p2 -K4 -n-100 -0hierclust
```

This command reads parameters and one dataset (see Example 4.2) from the default files and runs `hierclust`. After that, the Adjusted Rand index is computed.

1. *Running C-MixSim*

```
./C-MixSim -p2 -K4 -n-100
```

The command reads one 2-dimensional 4-component mixture from the default files for mixing proportions, means, and dispersions. 100 observations are read from the default file `x.dat`.

Output:

```
The desired overlap has been met...
Map of misclassification probabilities:
[0][0] : 1.000000 [0][1] : 0.000000 [0][2] : 0.000000 [0][3] : 0.000921
[1][0] : 0.000000 [1][1] : 1.000000 [1][2] : 0.000000 [1][3] : 0.000000
[2][0] : 0.000000 [2][1] : 0.000000 [2][2] : 1.000000 [2][3] : 0.000000
[3][0] : 0.000079 [3][1] : 0.000000 [3][2] : 0.000000 [3][3] : 1.000000
Average Overlap:  0.000167
Maximum Overlap:  0.001000 (components:  0 and 3)
```

2. *Running clustering algorithm*

```
./hierclust -p2 -K4 -n-100
```

This command runs the program `hierclust` which performs hierarchical clustering.

Output:

```
OBTAINED CLASSIFICATIONS:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 3 3
3 3 3 3
```

3. *Computing Adjusted Rand index*

```
./AdjRand -K4 -n-100
```

The above command computes the Adjusted Rand index and writes it into the default file `AR.dat`.

Output:

```
ADJUSTED RAND VALUES:
1.000000
```

4.4 Example: Calculating overlap between identified classes in a dataset

Here we provide an example that illustrates the use of the program in calculating the overlap between identified groups in a dataset, such as the celebrated *Iris* dataset (included). Consider the following command:

```
./CARP -p4 -K3 -D"TEST" -P"Pi.iris" -M"Mu.iris" -S"LTSigma.iris"
```

Here we provide an illustration based on the famous *Iris* dataset. Files `Pi.iris`, `Mu.iris`, and `LTSigma.iris` at the directory `TEST` contain parameters of the Gaussian mixture model with 3 components computed based on the true classification. CARP runs only the first stage evaluating pairwise overlaps for provided parameters. The second and third stages will not be run as the option `-0` is not specified. No data are simulated as the option `-n` is not specified.

1. Running C-MixSim

```
./C-MixSim -p4 -K3 -DTEST -PPi.iris -MMu.iris -SLTSigma.iris
```

The command reads parameters for *Iris*. From the output, it is clearly seen that the second and third components of the mixture produce substantial overlap while the other two pairs do not yield any noticeable overlap.

Output:

```
Map of misclassification probabilities:
[0][0] : 1.000000 [0][1] : 0.000000 [0][2] : 0.000000
[1][0] : 0.000000 [1][1] : 1.000000 [1][2] : 0.023023
[2][0] : 0.000000 [2][1] : 0.026294 [2][2] : 1.000000
Average Overlap: 0.016439
Maximum Overlap: 0.049318 (components: 1 and 2)
```

4.5 Example: Use of C-MixSim as a standalone package

Here we illustrate the use of CARP in using C-MixSim standalone, specifically in the case of simulating two 2-dimensional 2-component mixtures with general covariance matrices and mixing proportions no less than 0.3 according to an average overlap of 0.2 and generating datasets of size 100. Consider the following command:

```
./CARP -b0.2 -p2 -K2 -#2 -n100 -z0.3
```

Two 2-dimensional 2-component mixtures will be generated according to the average overlap value of 0.2 and samples of size 100 will be drawn from them. CARP runs only the simulation stage as the option `-0` is not specified. The files will have their default names. The smallest mixing proportion allowed is 0.3. As there are only two components, specifying the value of average overlap is equivalent to providing that of maximum overlap.

1. Running C-MixSim

```
./C-MixSim -b0.2 -p2 -K2 -#2 -n100 -z0.3
```

Output from the program is given below.

Output:

MIXTURE MODEL #1:

The desired overlap has been met...

Map of misclassification probabilities:

[0][0] : 1.000000 [0][1] : 0.065789

[1][0] : 0.134211 [1][1] : 1.000000

Average Overlap: 0.200000

Maximum Overlap: 0.200000 (components: 0 and 1)

Mixture parameters:

Pi:

0.476254 0.523746

Mu:

[0] : 0.482694 0.868947

[1] : 0.228367 0.105629

Sigma:

[0] :

0.068956 - 0.061422

-0.061422 0.138259

[1] :

0.057771 0.007065

0.007065 0.216991

Dataset with cluster sizes $N_k = 53 \ 47$ has been generated...

MIXTURE MODEL #2:

The desired overlap has been met...

Map of misclassification probabilities:

[0][0] : 1.000000 [0][1] : 0.167245

[1][0] : 0.032755 [1][1] : 1.000000

Average Overlap: 0.200000

Maximum Overlap: 0.200000 (components: 0 and 1)

Mixture parameters:

Pi:

0.315557 0.684443

Mu:

[0] : 0.290308 0.104932

```

[1] : 0.360821 0.169447
Sigma :
[0] :
0.001618 0.000344
0.000344 0.001961
[1] :
0.001450 - 0.001001
-0.001001 0.002347
Dataset with cluster sizes Nk = 28 72 has been generated...

```

5 Conclusions

The package CARP is a software tool devoted to evaluating performances of finite mixture modeling and clustering algorithms. The underlying technique involves producing Gaussian finite mixture models with prespecified level of average and maximum pairwise overlap. The Adjusted Rand Index is used by default, for assessing the performance of the clustering method under investigation but any other user-specified measure may be used.

References

- L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.
- R. Maitra and V. Melnykov. Simulating data to study performance of finite mixture modeling and clustering algorithms. *Accepted for publication in the Journal of Computational and Graphical Statistics*, 2010. doi: 10.1198/jcgs.2009.08054.