

Introduction to the MLC-CUB R Functions

This set of R functions implements the Expectation–Maximization (EM) algorithm for the **Multivariate Latent Class CUB (MLC-CUB)** model, a model-based clustering method for multivariate ordinal (rating) data.

Core Function

```
EM(R, m, K, tol = 1e-10, EM.iter = 20, max_iter = 1500)
```

Runs the EM algorithm for a specified number of clusters K .

Arguments:

- `R`: matrix of ordinal responses ($n \times J$).
- `m`: vector containing the number of categories for each variable.
- `K`: number of clusters.
- `tol`: tolerance threshold for convergence of the log-likelihood.
- `EM.iter`: number of random initializations.
- `max_iter`: maximum number of EM iterations allowed.

Output:

A list including:

- `xi.est`, `pi.est`, `omega.est`: estimated parameters;
- `class`: cluster assignment for each observation;
- `LogLik`, `AIC`, `BIC`: model fit statistics;
- `taus`: posterior probabilities of cluster membership.

Wrapper Functions

```
EM.MLCCUB(R, m, K.vec, tol, EM.iter, max_iter)
```

Sequentially fits multiple MLC-CUB models for different values of K (given in `K.vec`). Useful for model selection using information criteria such as BIC.

```
EM.MLCCUB_parallel(R, m, K.vec, tol, EM.iter, max_iter)
```

Parallel version of the previous function, using the R `parallel` package. Each value of K is processed on a separate core to reduce computation time.

Bootstrap Functions

```
bootstrap_iteration(data, m, K, tol, EM.iter, max_iter)
```

Performs one bootstrap resampling and fits the MLC-CUB model on the resampled data. Returns both the fitted model and the corresponding cluster assignments.

```
bootstrap(data, n.parallel, n.boot = 100, m, K, tol, EM.iter,  
max_iter)
```

Executes multiple bootstrap iterations (optionally in parallel) to evaluate model stability and identifiability.
Returns:

- `model.boot.list`: list of models fitted on bootstrap samples;
- `clusters.sort.list`: list of cluster assignments for each bootstrap sample.