Package 'microclustr'

October 1, 2020

Type Package

Title Entity Resolution with Random Partition Priors for Microclustering

Version 0.1.0

Date 2020-09-15

Depends R (>= 3.2.4)

Imports Rcpp (>= 1.0.1), stats

Suggests knitr, rmarkdown

Description An implementation of the model in Betancourt, Zanella, Steorts (2020) <arXiv:2004.02008>, which performs microclustering models for categorical data. The package provides a vignette for two proposed methods in the paper as well as two standard Bayesian non-parametric clustering approaches for entity resolution. The experiments are reproducible and illustrated using a simple vignette. LICENSE: GPL-3 + file license.

VignetteBuilder knitr

License GPL-3

LinkingTo Rcpp

RoxygenNote 7.1.1.9000

NeedsCompilation yes

Author Rebecca C Steorts [aut, cre],
Brenda Betancourt [aut],
Giacomo Zanella [aut]

Maintainer Rebecca C Steorts <beka@stat.duke.edu>

Repository CRAN

Date/Publication 2020-10-01 08:30:02 UTC

R topics documented:

`microclustr-package` *A short title line describing what the package does*

Description

A more detailed description of what the package does. A length of about one to five lines is recommended.

Details

This section should provide a more detailed overview of how to use the package, including the most important functions.

Author(s)

Your Name, email optional.

Maintainer: Your Name <your@email.com>

References

This optional section can contain literature or other references for background information.

See Also

Optional links to other man pages

Examples

```
## Not run:
   ## Optional simple examples of the most important functions
   ## These can be in \dontrun{} and \donttest{} blocks.

## End(Not run)
```

DataRemap *Remap data to a list of consecutive integers per field*

Description

Remap data to a list of consecutive integers per field

Usage

```
DataRemap(data)
```

Arguments

data	Data frame containing only categorical variables

Value

Data frame of remapped values to consecutive list of integers

Examples

```
truePartition <- c(10,10,10,10)
numberFields <- 5
numberCategories <- rep(10,5)
trueBeta <- 0.01
data <- SimData(truePartition, numberFields, numberCategories, trueBeta)
DataRemap(data)
```

fdr_fun *Calculates FDR when ground truth is available*

Description

Calculates FDR when ground truth is available

Usage

```
fdr_fun(z, id)
```

Arguments

z	Vector of cluster assignments
id	Vector of true cluster assignments (ground truth)

Value

FDR

Examples

```
truePartition <- c(50,50,50,50)
maxPartitionSize<- length(truePartition)
uniqueNumberRecords <- sum(truePartition)
id <- rep(1:uniqueNumberRecords, times=rep(1:maxPartitionSize, times=truePartition))
fdr_fun(z = truePartition, id)
```

`fnr_fun` *Calculates FNR when ground truth is available*

Description

Calculates FNR when ground truth is available

Usage

```
fnr_fun(z, id)
```

Arguments

`z`	Vector of cluster assignments
`id`	Vector of true cluster assignments (ground truth)

Value

FNR

Examples

```
truePartition <- c(50,50,50,50)
maxPartitionSize<- length(truePartition)
uniqueNumberRecords <- sum(truePartition)
id <- rep(1:uniqueNumberRecords, times=rep(1:maxPartitionSize, times=truePartition))
fnr_fun(z = truePartition, id)
```

`mean_fdr` *Calculates average FDR when ground truth is available*

Description

Calculates average FDR when ground truth is available

Usage

```
mean_fdr(zm, id)
```

Arguments

zm	Matrix with posterior samples of cluster assignments
id	Vector of true cluster assignments (ground truth)

Value

Average FDR over posterior samples

Examples

```
truePartition <- c(50,50,50,50)
maxPartitionSize<- length(truePartition)
uniqueNumberRecords <- sum(truePartition)
id <- rep(1:uniqueNumberRecords, times=rep(1:maxPartitionSize, times=truePartition))
numberFields <- 5
numberCategories <- rep(10,5)
trueBeta <- 0.01
simulatedData <- SimData(truePartition, numberFields, numberCategories, trueBeta)
posteriorESCD <- SampleCluster(data=simulatedData, Prior="ESCD", burn=0, nsamples=10)
mean_fdr(zm = posteriorESCD$Z, id)
```

---

mean_fnr *Calculates average FNR when ground truth is available*

Description

Calculates average FNR when ground truth is available

Usage

```
mean_fnr(zm, id)
```

Arguments

zm	Matrix with posterior samples of cluster assignments
id	Vector of true cluster assignments (ground truth)

Value

Average FNR over posterior samples

Examples

```
truePartition <- c(50,50,50,50)
maxPartitionSize<- length(truePartition)
uniqueNumberRecords <- sum(truePartition)
id <- rep(1:uniqueNumberRecords, times=rep(1:maxPartitionSize, times=truePartition))
numberFields <- 5
numberCategories <- rep(10,5)
trueBeta <- 0.01
simulatedData <- SimData(truePartition, numberFields, numberCategories, trueBeta)
posteriorESCD <- SampleCluster(data=simulatedData, Prior="ESCD", burn=0, nsamples=10)
mean_fnr(zm = posteriorESCD$Z, id)
```

`SampleCluster` *Posterior samples of cluster assignments and Prior parameters*

Description

Posterior samples of cluster assignments and Prior parameters

Usage

```
SampleCluster(data, Prior, burn, nsamples, spacing = 1000, block_flag = TRUE)
```

Arguments

`data`	Data frame containing only categorical variables
`Prior`	Specify partition prior: "DP", "PY", "ESCNB"
`burn`	MCMC burn-in period
`nsamples`	MCMC iterations after burn-in
`spacing`	Thinning for chaperones algorithm (default 1000)
`block_flag`	TRUE for non-uniform chaperones (default)

Value

List with posterior samples for cluster assignments (Z), Prior parameters and distortion probabilities (Params)

SimData *Generates a simulated dataset based on a true partition*

Description

Generates a simulated dataset based on a true partition

Usage

```
SimData(true_L, nfields, ncat, true_beta)
```

Arguments

true_L	Vector of size max cluster size with number of clusters of each size
nfields	Number of fields
ncat	Vector with number of categories per field
true_beta	Distortion probability for the fields

Value

Simulated data set

Examples

```
truePartition <- c(2,2,2,2)
numberFields <- 2
numberCategories <- rep(5,2)
trueBeta <- 0.01
SimData(truePartition, numberFields, numberCategories, trueBeta)
```

Index