Package 'VICatMix'

November 27, 2024

Type Package

Title Variational Mixture Models for Clustering Categorical Data

Version 1.0

Date 2024-11-25

Maintainer Jackie Rao <jackie.rao@mrc-bsu.cam.ac.uk>

Description A variational Bayesian finite mixture model for the clustering of categorical data, and can implement variable selection and semi-supervised outcome guiding if desired. Incorporates an option to perform model averaging over multiple initialisations to reduce the effects of local optima and improve the automatic estimation of the true number of clusters. For further details, see the paper by Rao and Kirk (2024) <doi:10.48550/arXiv.2406.16227>.

License GPL (>= 3)

URL https://github.com/j-ackierao/VICatMix

BugReports https://github.com/j-ackierao/VICatMix/issues

Imports klaR, matrixStats, mcclust, Rcpp, stats, gtools

Suggests doParallel, doRNG, foreach, parallel

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

RoxygenNote 7.3.1

Depends R (>= 2.10)

NeedsCompilation yes

Author Jackie Rao [aut, cre],
Paul D.W Kirk [ths],
Sara Wade [ctb],
Colin Starr [ctb],
John Maddock [cph] (Author of original version of digamma header (digamma.h).)

Repository CRAN

Date/Publication 2024-11-27 12:40:06 UTC

Contents

`generateSampleDataBin` *generateSampleDataBin*

Description

Generate sample clustered binary data with cluster labels. The probability of a '1' in each cluster for each variable is randomly generated via a Beta(1, 5) distribution, encouraging sparse probabilities which vary across clusters. For noisy variables, the probability of a '1' is also generated by a Beta(1, 5) distribution but this probability is the same regardless of the cluster membership of the observation.

Usage

```
generateSampleDataBin(n, K, w, p, Irrp, yout = FALSE)
```

Arguments

`n`	Number of observations in dataset.
`K`	Number of clusters desired.
`w`	A vector of mixture weights (proportion of population in each cluster).
`p`	Number of clustering variables/covariates in dataset.
`Irrp`	Number of irrelevant/noisy variables/covariates in dataset. Note that these variables will be the final Irrp columns in the simulated dataset. Total data dimension is p + Irrp.
`yout`	Default FALSE. Indicate whether a binary outcome associated with clustering is required.

Value

A list with the following components:

`data`	A matrix consisting of the simulated data.
`trueClusters`	A vector with the simulated cluster assignments.
`outcome`	If yout = TRUE, this will be a vector with the outcome variable.

Examples

```
# example code
generatedData <- generateSampleDataBin(1000, 4, c(0.1, 0.2, 0.3, 0.4), 100, 0)
```

`generateSampleDataCat` *generateSampleDataCat*

Description

Generate sample clustered categorical data with cluster labels. The probability of a '1' in each cluster for each variable is randomly generated via a Dirichlet (1, ..., cat) distribution, where cat is the number of categories for each variable. For noisy variables, the probability of a '1' is also generated by a Dirichlet (1, ..., cat) distribution but this probability is the same regardless of the cluster membership of the observation. An outcome variable associated with the clustering structure can be generated with a different number of categories, also generated with a Dirichlet distribution. Package 'gtools' must be installed for this function.

Usage

```
generateSampleDataCat(n, K, w, p, Irrp, yout = FALSE, cat = 2, ycat = 2)
```

Arguments

`n`	Number of observations in dataset.
`K`	Number of clusters desired.
`w`	A vector of mixture weights (proportion of population in each cluster).
`p`	Number of clustering variables/covariates in dataset.
`Irrp`	Number of irrelevant/noisy variables/covariates in dataset. Note that these variables will be the final Irrp columns in the simulated dataset. Total data dimension is p + Irrp.
`yout`	Default FALSE. Indicate whether an outcome associated with clustering is required.
`cat`	Number of categories in each covariate. Default is 2.
`ycat`	Number of categories for the outcome varaible. Default is 2.

Value

A list with the following components:

`data`	A matrix consisting of the simulated data.
`trueClusters`	A vector with the simulated cluster assignments.
`outcome`	If yout = TRUE, this will be a vector with the outcome variable.

Examples

```
# example code
generatedData <- generateSampleDataCat(1000, 4, c(0.1, 0.2, 0.3, 0.4), 100, 0, cat = 3)
```

`minVI` *Minimize the posterior expected Variation of Information*

Description

Finds a representative partition of the posterior by minimizing the lower bound to the posterior expected Variation of Information from Jensen's Inequality.

Usage

```
minVI(psm, cls.draw=NULL, method=c("avg","comp","draws","all"),
      max.k=NULL)
```

Arguments

`psm`	a posterior similarity matrix, which can be obtained from MCMC samples of clusterings through a call to `comp.psm`.
`cls.draw`	a matrix of the samples of clusterings of the `ncol(cls)` data points that have been used to compute `psm`. Note: `cls.draw` has to be provided if `method="draw"` or `"all"`.
`method`	the optimization method used. Should be one of `"avg"`, `"comp"`, `"draws"`, or `"all"`. Defaults to `"avg"`.
`max.k`	integer, if `method="avg"` or `"comp"` the maximum number of clusters up to which the hierarchical clustering is cut. Defaults to `ceiling(nrow(psm)/4)`.

Details

The Variation of Information between two clusterings is defined as the sum of the entropies minus two times the mutual information. Computation of the posterior expected Variation of Information can be expensive, as it requires a Monte Carlo estimate. We consider a modified posterior expected Variation of Information, obtained by swapping the log and expectation, which is much more computationally efficient as it only depends on the posterior through the posterior similarity matrix. From Jensen's inequality, the problem can be viewed as minimizing a lower bound to the posterior expected loss.

We provide several optimization methods. For `method="avg"` and `"comp"`, the search is restricted to the clusterings obtained from a hierarchical clustering with average/complete linkage and `1-psm` as a distance matrix (the clusterings with number of clusters `1:max.k` are considered).
Method `"draws"` restricts the search to the clusterings sampled. If `method="all"` all minimization methods are applied by default.

Value

cl	clustering with minimal value of expected loss. If method="all" a matrix containing the clustering with the smallest value of the expected loss over all methods in the first row and the clusterings of the individual methods in the next rows.
value	value of posterior expected loss. A vector corresponding to the rows of cl if method="all".
method	the optimization method used.

Author(s)

Sara Wade, <sara.wade@ed.ac.uk>

References

Meila, M. (2007) Bayesian model based clustering procedures, *Journal of Multivariate Analysis* **98**, 873–895.

Wade, S. and Ghahramani, Z. (2015) Bayesian cluster analysis: Point estimation and credible balls. Submitted. arXiv:1505.03339

Examples

```
set.seed(15)
generatedData <- generateSampleDataBin(100, 4, c(0.1, 0.2, 0.3, 0.4), 25, 0)
resultforpsm <- list()
for (i in 1:5){ #use 5 initialisations
  mix <- runVICatMix(generatedData$data, 10, 0.01, tol = 0.005)
  resultforpsm[[i]] <- mix$model$labels
}
p1 <- t(matrix(unlist(resultforpsm), 100, 5))
psm <- mcclust::comp.psm(p1)

labels_avg <- minVI(psm, method = 'avg',max.k = 10)$cl

print(labels_avg)
```

---

runVICatMix *runVICatMix*

Description

Perform a run of the VICatMix model on a data-frame with no variable selection imposed.

Usage

```
runVICatMix(data, K, alpha, maxiter = 2000, tol = 5e-08, verbose = FALSE)
```

Arguments

data	A data frame or data matrix with N rows of observations, and P columns of covariates.
K	Maximum number of clusters desired. Must be an integer greater than 1.
alpha	The Dirichlet prior parameter. Recommended to set this to a number < 1. Must be > 0.
maxiter	The maximum number of iterations for the algorithm. Default is 2000.
tol	A convergence parameter. Default is 5x10^-8.
verbose	Default FALSE. Set to TRUE to output ELBO values for each iteration.

Value

A list with the following components: (maxNCat refers to the maximum number of categories for any covariate in the data)

labels	A numeric vector listing the cluster assignments for the observations.
ELBO	A numeric vector tracking the value of the ELBO in every iteration.
Cl	A numeric vector tracking the number of clusters in every iteration.
model	A list containing all variational model parameters and the cluster labels: **alpha** A K-length vector of Dirichlet parameters for alpha. **eps** A K x maxNCat x P array of Dirichlet parameters for epsilon. **labels** A numeric vector listing the cluster assignments for the observations. **rnk** A N x K matrix of responsibilities for the latent variables Z.
factor_labels	A data frame showing how variable categories correspond to numeric factor labels in the model.

Examples

```
# example code
set.seed(20)
generatedData <- generateSampleDataBin(500, 4, c(0.1, 0.2, 0.3, 0.4), 100, 0)
result <- runVICatMix(generatedData$data, 10, 0.01)

print(result$labels)
```

runVICatMixAvg *runVICatMixAvg*

Description

An extension of 'runVICatMix' to incorporate model averaging/summarisation over multiple initialisations.

Usage

```
runVICatMixAvg(
  data,
  K,
  alpha,
  maxiter = 2000,
  tol = 5e-08,
  inits = 25,
  loss = "VoIcomp",
  parallel = FALSE,
  cores = getOption("mc.cores", 2L),
  verbose = FALSE
)
```

Arguments

data	A data frame or data matrix with N rows of observations, and P columns of covariates.
K	Maximum number of clusters desired. Must be an integer greater than 1.
alpha	The Dirichlet prior parameter. Recommended to set this to a number < 1. Must be > 0.
maxiter	The maximum number of iterations for the algorithm. Default is 2000.
tol	A convergence parameter. Default is 5x10^-8.
inits	The number of initialisations included in the co-clustering matrix. Default is 25.
loss	The loss function to be used with the co-clustering matrix. Default is VoIcomp. Options are "VoIavg", "VoIcomp" and "medv".
parallel	Logical value indicating whether to run initialisations in parallel. Default is FALSE.
cores	User can specify number of cores for parallelisation if parallel = TRUE. Package automatically uses the user's parallel backend if one has already been registered.
verbose	Default FALSE. Set to TRUE to output ELBO values for each iteration.

Value

A list with the following components: (maxNCat refers to the maximum number of categories for any covariate in the data)

`labels_avg`	A numeric N-vector listing the cluster assignments for the observations in the averaged model.
`init_results`	A list where each entry is the cluster assignments for one of the initialisations included in the model averaging.

See Also

`runVICatMix`

Examples

```
# example code
set.seed(20)
generatedData <- generateSampleDataBin(500, 4, c(0.1, 0.2, 0.3, 0.4), 100, 0)
result <- runVICatMixAvg(generatedData$data, 10, 0.01, inits = 10)

print(result$labels_avg)
```

`runVICatMixVarSel` *runVICatMixVarSel*

Description

Perform a run of the VICatMixVarSel model on a data-frame including variable selection. Includes an option to include an outcome variable for semi-supervised profile regression.

Usage

```
runVICatMixVarSel(
  data,
  K,
  alpha,
  a = 2,
  maxiter = 2000,
  tol = 5e-08,
  outcome = NA,
  verbose = FALSE
)
```

Arguments

data	A data frame or data matrix with N rows of observations, and P columns of covariates.
K	Maximum number of clusters desired. Must be an integer greater than 1.
alpha	The Dirichlet prior parameter. Recommended to set this to a number < 1. Must be > 0.
a	Hyperparameter for variable selection hyperprior. Default is 2.
maxiter	The maximum number of iterations for the algorithm. Default is 2000.
tol	A convergence parameter. Default is 5x10^-8.
outcome	Optional outcome variable. Default is NA; having an outcome triggers semi-supervised profile regression.
verbose	Default FALSE. Set to TRUE to output ELBO values for each iteration.

Value

A list with the following components: (maxNCat refers to the maximum number of categories for any covariate in the data)

labels	A numeric vector listing the cluster assignments for the observations.
ELBO	A numeric vector tracking the value of the ELBO in every iteration.
Cl	A numeric vector tracking the number of clusters in every iteration.
model	A list containing all variational model parameters and the cluster labels: **alpha** A K-length vector of Dirichlet parameters for alpha. **eps** A K x maxNCat x P array of Dirichlet parameters for epsilon. **c** A P-length vector of expected values for the variable selection parameter, gamma. **labels** A numeric vector listing the cluster assignments for the observations. **nullphi** A P x maxNCat matrix of maximum likelihood parameters for irrelevant variables. **rnk** A N x K matrix of responsibilities for the latent variables Z.
factor_labels	A data frame showing how variable categories correspond to numeric factor labels in the model.

Examples

```
# example code

set.seed(12)
generatedData <- generateSampleDataBin(500, 4, c(0.1, 0.2, 0.3, 0.4), 90, 10)
result <- runVICatMixVarSel(generatedData$data, 10, 0.01)

print(result$labels)
#clustering labels

print(result$c)
```

```
#expected values for variable selection parameter; 1 (or close to 1) indicates variable is relevant
```

---

runVICatMixVarSelAvg *runVICatMixVarSelAvg*

Description

An extension of 'runVICatMixVarSel' to incorporate model averaging/summarisation over multiple initialisations.

Usage

```
runVICatMixVarSelAvg(
  data,
  K,
  alpha,
  a = 2,
  maxiter = 2000,
  tol = 5e-08,
  outcome = NA,
  inits = 25,
  loss = "VoIcomp",
  var_threshold = 0.95,
  parallel = FALSE,
  cores = getOption("mc.cores", 2L),
  verbose = FALSE
)
```

Arguments

data	A data frame or data matrix with N rows of observations, and P columns of covariates.
K	Maximum number of clusters desired. Must be an integer greater than 1.
alpha	The Dirichlet prior parameter. Recommended to set this to a number < 1. Must be > 0.
a	Hyperparameter for variable selection hyperprior. Default is 2.
maxiter	The maximum number of iterations for the algorithm. Default is 2000.
tol	A convergence parameter. Default is 5x10^-8.
outcome	Optional outcome variable. Default is NA; having an outcome triggers semi-supervised profile regression.
inits	The number of initialisations included in the co-clustering matrix. Default is 25.

loss	The loss function to be used with the co-clustering matrix. Default is VoIcomp. Options are "VoIavg", "VoIcomp" and "medv".
var_threshold	Threshold for selection proportion for determining selected variables under the averaged model. Options are 0 < n <= 1 for a threshold. Default is 0.95.
parallel	Logical value indicating whether to run initialisations in parallel. Default is FALSE.
cores	User can specify number of cores for parallelisation if parallel = TRUE. Package automatically uses the user's parallel backend if one has already been registered.
verbose	Default FALSE. Set to TRUE to output ELBO values for each iteration.

Value

A list with the following components: (maxNCat refers to the maximum number of categories for any covariate in the data)

labels_avg	A numeric N-vector listing the cluster assignments for the observations in the averaged model.
varsel_avg	A numeric P-vector with a variable selection indicator for the covariates in the averaged model.
init_results	A list where each entry is the cluster assignments for one of the initialisations included in the model averaging.
init_varsel_results	A list where each entry is the expected value for the variable selection parameters ('c') for one of the initialisations included in the model averaging.

See Also

runVICatMixVarSel

Examples

```
# example code

set.seed(12)
generatedData <- generateSampleDataBin(500, 4, c(0.1, 0.2, 0.3, 0.4), 40, 10)
result <- runVICatMixVarSelAvg(generatedData$data, 10, 0.01, inits = 10)

print(result$labels_avg)
print(result$varsel_avg)
```

`VI.lb` *Compute the modified Variation of Information from swapping log and expectation.*

Description

Based on samples of partitions (eg. from MCMC or different clustering initialisations), computes the modified Variation of Information which switches the log and expectation in the usual Variation of Information.

Usage

```
VI.lb(cls, psm)
```

Arguments

`cls`	a matrix of partitions where the posterior expected (modified) Variation of Information is to be evaluated. Each row corresponds to a clustering of `ncol(cls)` data points.
`psm`	a posterior similarity matrix, which can be obtained from clusterings through a call to `comp.psm`.

Details

The Variation of Information (VoI) between two clusterings is defined as the sum of the entropies minus two times the mutual information. Computation of the posterior expected VoI can be expensive, as it requires a Monte Carlo estimate. The modified posterior expected VoI, obtained by swapping the log and expectation, is much more computationally efficient as it only depends on the posterior through the posterior similarity matrix. From Jensen's inequality, the problem of finding the optimal partition which minimizing the posterior expected modified VoI can be viewed as minimizing a lower bound to the posterior expected VoI.

Value

vector of length `nrow(cls)` of the posterior expected (modified) VoI.

Author(s)

Sara Wade, <sara.wade@ed.ac.uk>

References

Meila, M. (2007) Bayesian model based clustering procedures, *Journal of Multivariate Analysis* **98**, 873–895.

Wade, S. and Ghahramani, Z. (2015) Bayesian cluster analysis: Point estimation and credible balls. Submitted. arXiv:1505.03339.

See Also

minVI which locates the partition that minimizes the posterior expected modified VoI.

Examples

```
set.seed(15)
generatedData <- generateSampleDataBin(100, 4, c(0.1, 0.2, 0.3, 0.4), 25, 0)
resultforpsm <- list()
for (i in 1:5){ #use 5 initialisations
  mix <- runVICatMix(generatedData$data, 10, 0.01, tol = 0.005)
  resultforpsm[[i]] <- mix$model$labels
}
p1 <- t(matrix(unlist(resultforpsm), 100, 5))
psm <- mcclust::comp.psm(p1)

# Compute modified Variation of Information for each partition from VICatMix runs
VI.lb(p1, psm)
```

Index