Package 'MixedDeepMM'

November 25, 2026

Type Package

Title Deep Mixture Models for Continuous and Mixed-Type Data

Version 0.1.0

Description Fits Gaussian, Student-t and skew-t deep mixture models to continuous observations, and single-head and two-head latent mixture models to combinations of continuous, binary, ordinal, nominal and count measurements. Mixed-response estimation uses Monte Carlo expectation-maximisation with structured data-derived initialisation. Utilities for posterior clustering, Gower dissimilarity, silhouette evaluation, adjusted Rand index and misclassification rate are provided. Original deep Gaussian mixture models are described by Viroli and McLachlan (2019) <doi:10.1007/s11222-017-9793-z>.

License GPL (>= 3)

Encoding UTF-8

Depends R (>= 4.1.0)

Imports stats, utils, graphics

Suggests mvtnorm, corpcor, GIGrvg, cluster, mclust, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Config/roxygen2/version 8.1.1

Author Jinran Wu [aut, cre], Geoffrey J. McLachlan [aut]

Maintainer Jinran Wu <jinran.wu@uq.edu.au>

Repository CRAN

Date/Publication 2026-11-25 12:10:02 UTC

Contents

dstmm 2

mixeddeepmm 4
mixeddeepmm_ari 5
mixeddeepmm_diagnostics 6
mixeddeepmm_gower_distance 6
mixeddeepmm_gower_silhouette 7
mixeddeepmm_grid 7
mixeddeepmm_mr 8
predict.mixeddeepmm 9
simulate_mixeddeepmm 10
summary.mixeddeepmm 10

Index **12**

`dstmm` *Fit a Deep Skew-t Mixture Model*

Description

Fit a Deep Skew-t Mixture Model

Usage

```
dstmm(
  y,
  layers,
  k,
  r,
  it = 250,
  eps = 0.001,
  init = "kmeans",
  init_est = "factanal",
  seed = NULL,
  scale = FALSE,
  nu = 10,
  nu_structure = c("common", "first_layer", "pathway"),
  estimate_nu = TRUE,
  active_set = 1L,
  M = 1L,
  warm_start = NULL,
  psi_floor = 1e-06,
  nu_bounds = c(2.05, 200),
  min_iter = 50L,
  moving_window = 20L,
  alpha_tol = 1e-10,
  ridge = 1e-08,
  verbose = FALSE
)
```

Arguments

y	Finite numeric data matrix with observations in rows and variables in columns.
layers	Number of latent mixture layers (1, 2, or 3).
k	Integer vector containing the number of components in each layer.
r	Integer vector of latent dimensions, one per layer. Dimensions must decrease from the observed dimension.
it	Maximum number of Monte Carlo EM iterations (default 250).
eps	Positive convergence threshold for the relative change in moving-average log-likelihood.
init	Initial clustering method: "kmeans", "random", "hclass", or "mclust". The last method requires the suggested package mclust.
init_est	Initial latent parameter estimation method: "factanal" or "ppca".
seed	Optional random seed. NULL leaves the current random-number state unchanged.
scale	Logical; if TRUE, standardize the input variables before fitting.
nu	Initial degrees of freedom, strictly greater than 2. A scalar or a vector compatible with nu_structure.
nu_structure	How degrees of freedom are shared: "common" (one value), "first_layer" (per first-layer component), or "pathway" (per complete path).
estimate_nu	Logical; if TRUE, update the degrees of freedom during fitting; otherwise hold them fixed.
active_set	Indices of layers permitted to have nonzero skewness vectors (default 1L).
M	Number of Monte Carlo draws per observation-path pair. M = 1 corresponds to stochastic EM.
warm_start	Optional previously fitted robust deep mixture model object used for initialization; NULL initializes from data.
psi_floor	Positive numerical floor used to keep covariance matrices positive definite.
nu_bounds	Length-two vector of lower and upper bounds for degrees-of-freedom estimation, with lower bound greater than 2.
min_iter	Minimum iteration number before convergence can stop the algorithm.
moving_window	Number of iterations in each of the two log-likelihood windows used to assess convergence.
alpha_tol	Nonnegative numerical tolerance for treating the skewness contribution as zero.
ridge	Nonnegative regularization value used in latent parameter updates.
verbose	Logical; if TRUE, print progress messages during fitting.

Value

An object of class "dstmm" containing the fitted model and diagnostics.

`mixeddeepmm` *Fit deep mixture models for continuous or mixed responses*

Description

A unified model interface using existing continuous DGMM, RDMM and DStMM engines and a new single- or two-head mixed-response Monte Carlo EM estimator. Mixed Student-t and skew-t kernels extend the Gaussian formulation.

Usage

```
mixeddeepmm(
  data,
  types = NULL,
  family = c("gaussian", "t", "skew-t"),
  heads = c("single", "two"),
  layers,
  control = list()
)
```

Arguments

`data`	Data frame or all-numeric matrix of observations. Missing values are not supported.
`types`	Named character vector assigning every column one of `continuous`, `binary`, `ordinal`, `nominal`, or `count`. Defaults to all continuous for an all-numeric dataset.
`family`	Latent kernel: Gaussian, Student-t or generalised-hyperbolic skew-t.
`heads`	One shared latent head, or separate continuous/discrete heads with shared deeper tail.
`layers`	List with `k` component counts and `r` dimensions. Single-head mixed example: `list(k=c(2,2),r=c(2,1))`; two-head example: `list(k=c(2,2,2),r=list(continuous=2,discrete`
`control`	Fitting controls: `max_iter` (35), `min_iter` (8), `tol` (0.002), `mc_samples` (48), `mc_eval` (192), `nstart` (1), `seed` (123), `nu` (8), `estimate_nu` (TRUE), `psi_floor` (0.03), `ridge` (0.001), `verbose` (FALSE), `init` ("kmeans"), `init_est` ("factanal"). Aliases `it`, `eps` and `M` are accepted. Mixed-response fits always use deterministic data-driven initialization; `seed` controls MCEM draws and the choice of deterministic initial partition, and `nstart` explores alternative deterministic variants. `init` and `init_est` apply only to legacy continuous-only engines.

Details

For mixed responses the latent Gaussian hierarchy and measurement model are fitted using posterior draws conditional on continuous responses, importance weighted by non-Gaussian response likelihoods. Each fit saves a separate Monte Carlo approximation to the observed log likelihood

for model comparison, with a delta-method integration-error estimate. The mixed Gaussian estimator uses the original paper's conditional Gaussian and GLLVM design with a component-specific two-head continuous measurement and one shallow Gaussian-mixture tail; it is not a full reimplementation of the published MDGMM estimation algorithm. Its structured PCA/indicator, clustering, and regression-based initialization is not the published NSEP algorithm; it uses no random initial parameter draws. MCEM integration still involves random sampling. Numerical integration and estimation sensitivity should be examined for the mixed Student-t and skew-t extensions. BIC counts the represented coefficients under reference-location constraints; residual factor rotational non-identifiability may affect interpretation.

Value

A fitted object of class `mixeddeepmm`, supporting `print`, `summary`, `predict`, `logLik`, `fitted`, and `plot`.

Examples

```
sim <- simulate_mixeddeepmm(n = 40, seed = 1)
fit <- mixeddeepmm(sim$data, sim$types, layers = list(k = c(2,2), r = c(2,1)),
  control = list(max_iter = 2, min_iter = 2, mc_samples = 8, mc_eval = 16))
print(fit)
```

`mixeddeepmm_ari` *Compute an adjusted Rand index*

Description

Evaluate an estimated partition against independent known labels.

Usage

```
mixeddeepmm_ari(truth, predicted)
```

Arguments

`truth`	Known cluster labels.
`predicted`	Estimated labels.

Value

Adjusted Rand index numeric scalar.

mixeddeepmm_diagnostics

Inspect numerical diagnostics from deep mixture fitting

Description

Summarise Monte Carlo error, log-likelihood trace, starts and iteration counts. The result omits an end-of-run convergence flag.

Usage

```
mixeddeepmm_diagnostics(object)
```

Arguments

object	A fitted mixeddeepmm object.

Value

List of fitting and Monte Carlo precision diagnostics.

mixeddeepmm_gower_distance

Mixed-response Gower dissimilarity

Description

Construct a single mixed-type dissimilarity matrix with equal variable weights using cluster::daisy(metric="gower"). Binary and nominal predictors are symmetric categorical variables; ordinal predictors are ordered; continuous/count predictors are range-normalised numeric.

Usage

```
mixeddeepmm_gower_distance(data, types)
```

Arguments

data	Complete mixed-response predictors, without the external truth.
types	Named character vector defining continuous, count, binary, nominal and ordinal response types.

Value

A dist/dissimilarity object.

See Also

mixeddeepmm_gower_silhouette

mixeddeepmm_gower_silhouette *Mean Gower silhouette width for a cluster partition*

Description

Compute the average silhouette width via `cluster::silhouette`. Returns `NA_real_` for partitions with fewer than two or at least n nonempty groups. This score evaluates a completed clustering, not an EM objective or a replacement for MCEM.

Usage

```
mixeddeepmm_gower_silhouette(clusters, distance)
```

Arguments

`clusters`	One cluster label per observation; external true labels are not used.
`distance`	Precomputed `dist` or `dissimilarity` object, typically from `mixeddeepmm_gower_distance`.

Value

Numeric mean silhouette width, or `NA_real_`.

See Also

`mixeddeepmm_gower_distance`

mixeddeepmm_grid *Fit and compare a grid of mixed deep mixture models*

Description

Fit a supplied list of candidate architectures and compare observed log likelihood approximations and BIC.

Usage

```
mixeddeepmm_grid(data, types = NULL, candidates, control = list())
```

Arguments

`data`	Data frame.
`types`	Response type specification; see `mixeddeepmm`.
`candidates`	List of named lists containing `family`, `heads` and `layers`.
`control`	Control settings passed to each model.

Details

Monte Carlo likelihood error affects BIC uncertainty. BIC is an approximate fit criterion and does not itself measure clustering quality.

Value

List with comparison `table` and `fits`.

`mixeddeepmm_mr` *Compute a label-invariant misclassification rate*

Description

Compute the minimum misclassification rate (MR) after optimally matching predicted cluster labels to known class labels, using one-to-one assignment.

Usage

```
mixeddeepmm_mr(truth, predicted)
```

Arguments

`truth`	Nonempty vector of reference class labels, without missing values.
`predicted`	Nonempty vector of estimated cluster labels, without missing values.

Details

Cluster labels are arbitrary. The Hungarian algorithm finds the one-to-one correspondence maximising the diagonal contingency-table count, with zero padding for unequal numbers of reference classes and predicted clusters. Unmatched groups therefore contribute errors. Reference labels must be kept out of unsupervised fitting and model selection.

Value

A numeric scalar between 0 and 1: 0 is perfect agreement and smaller is better.

Examples

```
mixeddeepmm_mr(c("A", "A", "B", "B"), c(9, 9, 4, 4))
mixeddeepmm_mr(c("A", "A", "B", "B"), c(9, 9, 4, 9))
```

`predict.mixeddeepmm` *Predict pathway posteriors, clusters and responses*

Description

Obtain posterior probabilities of latent pathways, their marginal layer probabilities, classifications or model-implied response means/probabilities.

Usage

```
## S3 method for class 'mixeddeepmm'
predict(
  object,
  newdata = NULL,
  type = c("cluster", "posterior", "path", "layer", "response"),
  layer = NULL,
  mc_samples = NULL,
  cluster_level = c(
    "paper", "first", "shared_tail", "continuous_head", "discrete_head"
  ),
  ...
)
## S3 method for class 'mixeddeepmm'
fitted(object, ...)
```

Arguments

`object`	Fitted `mixeddeepmm` object.
`newdata`	Optional data frame with the same named columns and known categorical levels.
`type`	Return specified-layer MAP classification (`cluster`), complete-path posterior (`posterior`), MAP full pathway (`path`), layer marginal posterior (`layer`), or marginal link probabilities and expected counts (`response`).
`layer`	Optional integer layer index for `type="layer"` or `type="cluster"`; overrides cluster_level.
`mc_samples`	Override MC importance sample size when predicting mixed data.
`cluster_level`	Named layer when layer is NULL: paper uses the common first layer for single head and shared tail for two heads; first selects layer 1; shared_tail selects the last layer. Head options are available as indicated by the architecture.
`...`	Additional arguments.

Details

By default cluster_level=`"paper"` selects the first common head in single-head models and the shared tail in two-head models. When predicting link probabilities, observed responses inform the latent posterior. Thus the estimates are conditional fitted probabilities and are not leave-one-out predictive scores.

Value

Numeric vector, matrix or named list of fitted values, according to `type`.

`simulate_mixeddeepmm` *Simulate mixed observations and latent class labels*

Description

Simulate the manuscript's three-continuous and five-discrete response design, with three generating classes.

Usage

```
simulate_mixeddeepmm(n = 600L, seed = 2026L)
```

Arguments

`n`	Sample size (at least 30).
`seed`	Random seed.

Value

List with `data`, `types`, `group`, and `latent`.

Examples

```
sim <- simulate_mixeddeepmm(n = 50, seed = 42)
str(sim$data)
```

`summary.mixeddeepmm` *Summary, print, likelihood and plot methods for deep mixtures*

Description

Display goodness of fit, Monte Carlo precision, iteration count and likelihood history (without an end-of-run convergence flag).

Usage

```
## S3 method for class 'mixeddeepmm'
summary(object, ...)
## S3 method for class 'mixeddeepmm'
print(x, ...)
## S3 method for class 'summary.mixeddeepmm'
print(x, ...)
## S3 method for class 'mixeddeepmm'
logLik(object, ...)
## S3 method for class 'mixeddeepmm'
plot(x, ...)
```

Arguments

object	A mixeddeepmm fit.
x	A model or summary object.
...	Other options passed to plot, as appropriate.

Value

A summary list, an invisible object for print/plot, or logLik object.

Index

dstmm, 2

fitted.mixeddeepmm
(predict.mixeddeepmm), 9

logLik.mixeddeepmm
(summary.mixeddeepmm), 10

mixeddeepmm, 4
mixeddeepmm_ari, 5
mixeddeepmm_diagnostics, 6
mixeddeepmm_gower_distance, 6
mixeddeepmm_gower_silhouette, 7
mixeddeepmm_grid, 7
mixeddeepmm_mr, 8

plot.mixeddeepmm (summary.mixeddeepmm),
10
predict.mixeddeepmm, 9
print.mixeddeepmm
(summary.mixeddeepmm), 10
print.summary.mixeddeepmm
(summary.mixeddeepmm), 10

simulate_mixeddeepmm, 10
summary.mixeddeepmm, 10