Package: copre (via r-universe)

October 24, 2026

Type Package

Title Tools for Nonparametric Martingale Posterior Sampling

Version 0.2.2

Description Performs Bayesian nonparametric density estimation using Martingale posterior distributions including the Copula Resampling (CopRe) algorithm. Also included are a Gibbs sampler for the marginal Gibbs-type mixture model and an extension to include full uncertainty quantification via a predictive sequence resampling (SeqRe) algorithm. The CopRe and SeqRe samplers generate random nonparametric distributions as output, leading to complete nonparametric inference on posterior summaries. Routines for calculating arbitrary functionals from the sampled distributions are included as well as an important algorithm for finding the number and location of modes, which can then be used to estimate the clusters in the data using, for example, k-means. Implements work developed in Moya B., Walker S. G. (2022). <doi:10.48550/arxiv.2206.08418>, Fong, E., Holmes, C., Walker, S. G. (2021) <doi:10.48550/arxiv.2103.15671>, and Escobar M. D., West, M. (1995) <doi:10.1080/01621459.1995.10476550>.

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.3.1

LinkingTo Rcpp, RcppArmadillo, BH

Imports Rcpp, pracma, abind, dirichletprocess

Suggests ggplot2

NeedsCompilation yes

Author Blake Moya [cre, aut], The University of Texas at Austin [cph, fnd]

Maintainer Blake Moya <blakemoya@utexas.edu>

Repository https://cran.r-universe.dev

Date/Publication 2025-10-21 17:00:10 UTC

RemoteUrl https://github.com/cran/copre

RemoteRef HEAD

RemoteSha f6c6b9a3d33204d107f064cfbbf76cb2f10656c4

Contents

copre-package 2
[[.seqreresult 3
$.grideval_result 4
antimodes 5
autoplot.copre_result 5
autoplot.grideval_result 6
autoplot.seqre_result 7
base_measure 7
copre 8
functional 9
G_normls 10
gibbsmix 10
length.grideval_result 11
modes 12
moment 13
plot.copre_result 13
plot.grideval_result 14
plot.seqre_result 15
register_autoplot_s3_methods 15
register_s3_method 16
seq_measure 16
Sq_dirichlet 17
Sq_gnedin0 17
Sq_pitmanyor 18

Index **19**

`copre-package` *CopRe Tools for Nonparametric Martingale Posterior Sampling*

Description

Performs Bayesian nonparametric density estimation using Martingale posterior distributions including the Copula Resampling (CopRe) algorithm. Also included are a Gibbs sampler for the marginal Gibbs-type mixture model and an extension to include full uncertainty quantification via a predictive sequence resampling (SeqRe) algorithm. The CopRe and SeqRe samplers generate random nonparametric distributions as output, leading to complete nonparametric inference on posterior summaries. Routines for calculating arbitrary functionals from the sampled distributions are included as well as an important algorithm for finding the number and location of modes, which can then be used to estimate the clusters in the data using, for example, k-means. Implements work developed in Moya B., Walker S. G. (2022).

Author(s)

Blake Moya blakemoya@utexas.edu

References

- Fong, E., Holmes, C., Walker, S. G. (2021). Martingale Posterior Distributions. arXiv. DOI: doi:10.48550/arxiv.2103.15671
- Moya B., Walker S. G. (2022). Uncertainty Quantification and the Marginal MDP Model. arXiv. DOI: doi:10.48550/arxiv.2206.08418
- Escobar M. D., West, M. (1995) Bayesian Density Estimation and Inference Using Mixtures. Journal of the American Statistical Association. DOI: doi:10.1080/01621459.1995.10476550

`[[.seqreresult` *Sequence Resampling*

Description

A function that samples predictive distributions for univariate continuous data using exchangeable predictive extension.

Usage

```
## S3 method for class 'seqreresult'
obj[[i]]

seqre(obj, inc = 1000, eps = 0.001, max_it = 100)
```

Arguments

`obj`	A `seqre_result` object, usually output from `gibbsmix()`.
`i`	A numeric vector of sample indices.
`inc`	A positive integer increment value for the number of predictive samples to take each convergence check.
`eps`	An error value which determines the convergence approximation.
`max_it`	A positive integer maximum number of iterations before halting.

Value

A `seqre_result` object, or a list of two `seqre_result` objects if `keep_marg` is `TRUE`.

Functions

- `[[`: Subset method for `seqre_result` objects

See Also

`gibbsmix()`

`$.grideval_result` *Grid evaluation of* `copre_result` *and* `seqre_result` *objects*

Description

Grid evaluation of `copre_result` and `seqre_result` objects

Usage

```
## S3 method for class 'grideval_result'
obj$name

## S3 method for class 'grideval_result'
obj[[i]]

grideval(obj, grd = NULL, func = "density", nthreads = 1)

## S3 method for class 'copre_result'
grideval(obj, grd = NULL, func = "density", nthreads = 1)

## S3 method for class 'seqre_result'
grideval(obj, grd = NULL, func = "density", nthreads = 1)
```

Arguments

`obj`	A `copre_result` or `seqre_result` object.
`name`	The name of the attribute to access (i.e. `func`, `grid`, or `args`).
`i`	A numeric vector of sample indices.
`grd`	For `seqre_result` objects, a numeric vector of `m` grid points.
`func`	Either 'distribution', 'density', or 'gradient'.
`nthreads`	The number of parallel threads to launch with OpenMP.

Value

A `grideval_result` object, which is a matrix with dimension `[k, m]` of evaluated sample functions, with the following attributes:

- `func`: The evaluated function.
- `grid`: The grid points on which each of the `k` rows was evaluated.
- `args`: A copy of the `args` entry from `obj`.

Methods (by class)

- `grideval(copre_result)`: Grid evaluation method for `copre_result` objects.
- `grideval(seqre_result)`: Grid evaluation method for `seqre_result` objects.

Functions

- `$`: Attribute access method for `grideval_result` objects
- `[[`: Subset method for `grideval_result` objects

---

antimodes *Antimode Extractor*

Description

Extracts the antimodes from a `copre_result` or `seqre_result` object.

Usage

```
antimodes(obj, mean = FALSE, grd = NULL, idx = FALSE)
```

Arguments

`obj`	A `copre_result` or `mdp_result` object.
`mean`	A logical value indicating whether to extract the modes of the mean density of each of the individual sampled density.
`grd`	For `mdpolya_result`, a grid on which to evaluate the object.
`idx`	A logical value indicating whether to also return the index within `grd` of the discovered modes.

Value

A matrix of antimodes values in the support of the `copre_result` density

---

autoplot.copre_result *Create a CopRe Result ggplot*

Description

Create a CopRe Result ggplot

Usage

```
autoplot.copre_result(x, ..., func = "density", confint = NULL)
```

Arguments

x	A copre_result object.
...	Additional arguments discarded from plot.
func	Either 'distribution', 'density', or 'gradient'.
confint	A decimal value indicating the confidence interval width (e.g. 0.95 for a 95% confidence interval). Defaults to NULL, in which case no confidence intervals will be drawn.

Value

A ggplot object.

---

autoplot.grideval_result

Create a ggplot of a grideval_result *Object*

Description

Create a ggplot of a grideval_result Object

Usage

```
autoplot.grideval_result(x, ..., confint = NULL)
```

Arguments

x	A grideval_result object.
...	Additional arguments discarded from plot.
confint	A decimal value indicating the confidence interval width (e.g. 0.95 for a 95 percent confidence interval). Defaults to NULL, in which case no confidence intervals will be drawn.

Value

A ggplot object.

autoplot.seqre_result *Create a SeqRe Result ggplot*

Description

Create a SeqRe Result ggplot

Usage

```
autoplot.seqre_result(x, ..., func = "density", confint = NULL)
```

Arguments

x	A seqre_result object.
...	Additional arguments discarded from plot.
func	Either 'distribution', 'density', or 'gradient'.
confint	A decimal value indicating the confidence interval width (e.g. 0.95 for a 95% confidence interval). Defaults to NULL, in which case no confidence intervals will be drawn.

Value

A ggplot object.

base_measure *Base Measure for Mixture Models*

Description

A structure for wrapping base measures as in Escobar and West (1995).

Usage

```
base_measure(idx, dim, pars, hpars, eval)
```

Arguments

idx	A unique index for the base measure.
dim	A dimension for the support of the base measure.
pars	A list of parameters used to generate mixture components.
hpars	A list of hyperparameters used to generate pars.
eval	An evaluation function taking phi, a list of mixture parameter matrices, grd, a grid vector, f, a character string indicating whether to calculate the gradient, density, or distribution function, and nthreads, a number of threads to utilize for parallel execution.

Value

A `base_measure` object for use in the sequence resampling scheme for mixtures.

References

- Escobar M. D., West, M. (1995) Bayesian Density Estimation and Inference Using Mixtures. Journal of the American Statistical Association. DOI: doi:10.1080/01621459.1995.10476550

See Also

`seqre()`

`copre` *Copula Resampling*

Description

A function that samples predictive distributions for univariate continuous data using the bivariate Gaussian copula.

Usage

```
copre(
  data,
  N,
  k,
  rho = 0.91,
  grd_res = 1000,
  nthreads = parallel::detectCores(),
  gpu = FALSE,
  gpu_path = NULL,
  gpu_odir = NULL,
  gpu_seed = 1234
)
```

Arguments

`data`	The data from which to sample predictive distributions.
`N`	The number of unobserved data points to resample for each chain.
`k`	The number of predictive distributions to sample.
`rho`	A scalar concentration parameter.
`grd_res`	The number of points on which to evaluate the predictive distribution.
`nthreads`	The number of threads to call for parallel execution.
`gpu`	A logical value indicating whether or not to use the CUDA implementation of the algorithm.

gpu_path	The path to the CUDA implementation source code.
gpu_odir	A directory to output the compiled CUDA code.
gpu_seed	A seed for the CUDA random variates.

Value

A copre_result object, whose underlying structure is a list which contains the following components:

References

Fong, E., Holmes, C., Walker, S. G. (2021). Martingale Posterior Distributions. arXiv. DOI: doi:10.48550/arxiv.2103.15671

Examples

```
res_cop <- copre(rnorm(50), 10, 10, nthreads = 1)
```

---

functional *Obtain Functionals from a CopRe Result*

Description

Obtain Functionals from a CopRe Result

Usage

```
functional(obj, f, ..., mean = FALSE)
```

Arguments

obj	A copre_result object.
f	A list of functions.
...	Additional arguments passed to f.
mean	A logical value indicating whether or not to obtain the functional from the pointwise mean of the sampled distributions or from each individually.

Value

The integral over the copre_result grid of the functions in the list multiplied by the density of each sample distribution in obj.

G_normls *Normal-Inverse-Gamma Base Measure for Location-Scale Normal Mixture Models.*

Description

Normal-Inverse-Gamma Base Measure for Location-Scale Normal Mixture Models.

Usage

```
G_normls(mu = 0, tau = 1, s = 1, S = 1, a = NULL, A = NULL, w = NULL, W = NULL)
```

Arguments

mu	The mean parameter.
tau	The variance scaling parameter.
s	The primary shape parameter for the Inverse-Gamma component.
S	The secondary shape parameter for the Inverse-Gamma component.
a	The prior mean parameter for mu.
A	The prior variance for mu.
w	The prior primary shape parameter for tau.
W	The prior secondary shape parameter for tau.

Value

A base_measure object for use in the sequence resampling scheme for mixtures.

See Also

base_measure(), seqre()

gibbsmix *Marginal Gibbs-type Mixture Model Sampler*

Description

A function that samples marginal mixture densities via a marginal Gibbs sampler.

Usage

```
gibbsmix(data, k, b_msr, s_msr, burn = 1000, thin = 150)
```

Arguments

data	The data from which to sample predictive distributions.
k	The number of predictive samples to draw.
b_msr	A base_measure object.
s_msr	A seq_measure object.
burn	The number of initial sampling iterations to discard, will be truncated if a non-integer.
thin	The number of sampling iterations to discard between records, will be truncated if a non-integer.

Value

A seqre_result object.

See Also

seqre(), seq_measure(), base_measure()

---

length.grideval_result *Length*

Description

Length

Usage

```
## S3 method for class 'grideval_result'
length(x)
```

Arguments

x	A grideval_result object.

Value

The number of samples k in obj.

modes *Mode Extractor*

Description

Extracts the modes from a copre_result or seqre_result object.

Usage

```
modes(obj, mean = FALSE, grd = NULL, idx = FALSE, anti = FALSE)

## S3 method for class 'seqre_result'
modes(obj, mean = FALSE, grd = NULL, idx = FALSE, anti = FALSE)

## S3 method for class 'grideval_result'
modes(obj, mean = FALSE, grd = NULL, idx = FALSE, anti = FALSE)

n_modes(obj, mean = FALSE, grd = NULL, anti = FALSE)
```

Arguments

obj	A copre_result or seqre_result object.
mean	A logical value indicating whether to count the modes of the mean density of each of the individual sampled density.
grd	For seqre_result, a grid on which to evaluate the object.
idx	A logical value indicating whether to also return the index within grd of the discovered modes.
anti	A logical value indicating whether to extract true modes or anti-modes (i.e. local minima of the density function).

Value

A matrix of modes values in the support of the copre_result density

Methods (by class)

- modes(seqre_result): Mode-counting method for seqre_result objects.
- modes(grideval_result): Mode-counting method for grideval_result objects.

Functions

- n_modes(): Counts the modes from a copre_result or seqre_result object.

moment *Obtain Moments from a CopRe or SeqRe Result*

Description

Obtain Moments from a CopRe or SeqRe Result

Usage

```
moment(obj, mom, cntrl = TRUE, grd = NULL)

## S3 method for class 'seqre_result'
moment(obj, mom, cntrl = TRUE, grd = NULL)

## S3 method for class 'grideval_result'
moment(obj, mom, cntrl = TRUE, grd = NULL)
```

Arguments

obj	A copre_result or seqre_result object.
mom	A numeric scalar indicating the moment to calculate.
cntrl	A logical value indicating whether the moment should be central or not. Defaults to TRUE.
grd	A numeric vector of grid values on which the density function samples in obj should be calculated for trapezoidal integration.

Value

A vector of moment values for each sampled distribution in obj.

Methods (by class)

- moment(seqre_result): Moment calculation method for seqre_result objects.
- moment(grideval_result): Moment calculation method for grideval_result objects.

plot.copre_result *Create a CopRe Result Plot*

Description

Create a CopRe Result Plot

Usage

```
## S3 method for class 'copre_result'
plot(x, ..., func = "density", confint = NULL, use_ggplot = TRUE)
```

Arguments

x	A copre_result object.
...	Additional arguments discarded from plot.
func	Either 'distribution', 'density', or 'gradient'.
confint	A decimal value indicating the confidence interval width (e.g. 0.95 for a 95% confidence interval). Defaults to NULL, in which case no confidence intervals will be drawn.
use_ggplot	A logical value indicating whether to use ggplot2 instead of the base plot function.

Value

None.

---

plot.grideval_result *Create a Plot of a* grideval_result *Object*

Description

Create a Plot of a grideval_result Object

Usage

```
## S3 method for class 'grideval_result'
plot(x, ..., confint = NULL, use_ggplot = TRUE)
```

Arguments

x	A grideval_result object.
...	Additional arguments discarded from plot.
confint	A decimal value indicating the confidence interval width (e.g. 0.95 for a 95 percent confidence interval). Defaults to NULL, in which case no confidence intervals will be drawn.
use_ggplot	A logical value indicating whether to use ggplot2 instead of the base plot function.

Value

A ggplot object if ggplot2 is used, else none.

plot.seqre_result *Create a SeqRe Result Plot*

Description

Create a SeqRe Result Plot

Usage

```
## S3 method for class 'seqre_result'
plot(x, ..., func = "density", confint = NULL, use_ggplot = TRUE)
```

Arguments

x	A seqre_result object.
...	Additional arguments discarded from plot.
func	Either 'distribution', 'density', or 'gradient'.
confint	A decimal value indicating the confidence interval width (e.g. 0.95 for a 95% confidence interval). Defaults to NULL, in which case no confidence intervals will be drawn.
use_ggplot	A logical value indicating whether to use ggplot2 instead of the base plot function.

Value

None.

register_autoplot_s3_methods *Register* autoplot *methods to* ggplot2

Description

Register autoplot methods to ggplot2

Usage

```
register_autoplot_s3_methods()
```

Value

None

register_s3_method *Register S3 Methods from External Packages*

Description

https://github.com/tidyverse/hms/blob/master/R/zzz.R

Usage

```
register_s3_method(pkg, generic, class, fun = NULL)
```

Arguments

pkg	Package name.
generic	Generic function name.
class	Class name.
fun	Optional custom function name.

Value

None

seq_measure *Sequence Measure for Species Sampling Models*

Description

Sequence Measure for Species Sampling Models

Usage

```
seq_measure(idx, pars, hpars, Pn, Po)
```

Arguments

idx	A unique index for the sequence measure.
pars	A list of parameters used in Pn and Po to generate a sequence.
hpars	A list of hyperparameters used to generate pars.
Pn	A function on a sequence length n and a number of unique values k that returns the probability of the next member in the sequence having a new value.
Po	A function on a sequence length n, a number of unique values k, and the number of values equal to j, kj, that returns the probability of the next member in the sequence having the value j.

Value

A seq_measure object for use in the exchangeable sequence resampling scheme for mixtures.

See Also

seqre()

Sq_dirichlet *Dirichlet Sequence Measure.*

Description

Dirichlet Sequence Measure.

Usage

```
Sq_dirichlet(alpha = 1, c = NULL, C = NULL)
```

Arguments

alpha	The concentration parameter for the Dirichlet process. Must be greater than 0.
c	The prior primary shape parameter for alpha.
C	The prior secondary shape parameter for alpha.

Value

A seq_measure object for use in the exchangeable sequence resampling scheme for mixtures.

See Also

seq_measure(), seqre()

Sq_gnedin0 *Collapsed Gnedin Process Sequence Measure.*

Description

Collapsed Gnedin Process Sequence Measure.

Usage

```
Sq_gnedin0(gamma)
```

Arguments

gamma	The gamma parameter for the Gnedin process with xi set to 0. Bounded to [0, 1].

Value

A seq_measure object for use in the exchangeable sequence resampling scheme for mixtures.

See Also

seq_measure(), seqre()

Sq_pitmanyor *Pitman-Yor Sequence Measure.*

Description

Pitman-Yor Sequence Measure.

Usage

```
Sq_pitmanyor(d, alpha = 1, m = 1L)
```

Arguments

d	The discount parameter for the Pitman-Yor process. Must be less than 1.
alpha	The concentration parameter for the Pitman-Yor process. Must be greater than -sigma if sigma is in [0, 1), else ignored.
m	A positive integer used to set theta = m * abs(sigma) if sigma is negative.

Value

A seq_measure object for use in the exchangeable sequence resampling scheme for mixtures.

See Also

seq_measure(), seqre()

Index

∗ **internals**
    copre-package, 2
[[.grideval_result *($.grideval_result)*, 4
[[.seqreresult, 3
$.grideval_result, 4

antimodes, 5
autoplot.copre_result, 5
autoplot.grideval_result, 6
autoplot.seqre_result, 7

base_measure, 7
base_measure(), *10, 11*

copre, 8
copre-package, 2

functional, 9

G_normls, 10
gibbsmix, 10
gibbsmix(), *3*
grideval *($.grideval_result)*, 4

length.grideval_result, 11

modes, 12
moment, 13

n_modes *(modes)*, 12

plot.copre_result, 13
plot.grideval_result, 14
plot.seqre_result, 15

register_autoplot_s3_methods, 15
register_s3_method, 16

seq_measure, 16
seq_measure(), *11*, *17, 18*
seqre *([[.seqreresult)*, 3
seqre(), *8*, *10, 11*, *17, 18*
Sq_dirichlet, 17
Sq_gnedin0, 17
Sq_pitmanyor, 18