Package ‘mmcmcBayes’

November 24, 2025

Type Package

Title Multistage MCMC Method for Detecting DMRs

Version 0.1.0

Description Implements a multi-stage MCMC Bayesian framework for detecting differentially methylated regions (DMRs) in epigenetic data. It uses Bayesian inference with Alpha-Skew Generalized Normal (ASGN) model and support Bayes Factor or Anderson-Darling Test for region selection. The methodology is based on Yang (2025) <https://www.proquest.com/docview/3218878972>.

License GPL-3

URL https://github.com/zyang1919/mmcmcBayes

BugReports https://github.com/zyang1919/mmcmcBayes/issues

Depends R (>= 4.0.0)

Imports MCMCpack (>= 1.4-0)

Suggests kSamples, testthat (>= 3.0.0), knitr, rmarkdown

Encoding UTF-8

LazyData true

LazyDataCompression xz

RoxygenNote 7.3.3

NeedsCompilation no

Author Zhexuan Yang [aut, cre],
Duchwan Ryu [aut],
Feng Luan [aut]

Maintainer Zhexuan Yang <zky5198@psu.edu>

Repository CRAN

Date/Publication 2025-11-24 08:50:13 UTC

Contents

`asgn_func` *ASGN Model Parameter Estimation using MCMC*

Description

Estimates parameters (alpha, mu, sigma2) for the ASGN model using Markov Chain Monte Carlo (MCMC) methods. This function can be used for both cancer and normal groups in the multi-stage DMR detection framework, and also as a standalone function for fitting skewed and potentially bimodal data.

Usage

```
asgn_func(
  data,
  priors = NULL,
  mcmc = list(nburn = 5000, niter = 10000, thin = 1),
  seed = NULL,
  return_mcmc = FALSE,
  return_summary = FALSE
)
```

Arguments

`data` A matrix or vector containing the mean methylation levels (M-values)

`priors` A list of prior parameters for alpha, mu, and sigma2. If NULL, weakly informative priors are automatically generated from the data.

`mcmc` A list of MCMC parameters:

- `nburn`: Number of burn-in iterations (default: 5000)
- `niter`: Number of sampling iterations (default: 10000)
- `thin`: Thinning interval (default: 1)

`seed` Random seed for reproducibility. If NULL, no seed is set.

`return_mcmc` Logical indicating whether to return full MCMC samples for diagnostic purposes (default: FALSE)

`return_summary` Logical indicating whether to return a summary data.frame with parameter estimates and 95% credible intervals (default: FALSE)

Details

This function implements a Metropolis-Hastings within Gibbs sampler to estimate the parameters of the ASGN distribution, which can model skewed and potentially bimodal data. The algorithm updates alpha and mu using Metropolis-Hastings steps and sigma2 using inverse Gamma sampling.

The ASGN (Alpha-skewed generalized normal) distribution is particularly useful for modeling methylation data that often exhibits skewness and bimodal issue.

Value

A list that may contain the following elements:

- `posteriors`: Vector of posterior means for alpha, mu, and sigma2 (always present)
- `mcmc_samples`: List containing full MCMC chains for alpha, mu, and sigma2 (only if return_mcmc = TRUE)
- `summary`: Data.frame containing parameter estimates with 95% credible intervals (only if return_summary = TRUE)

Author(s)

Zhexuan Yang, Duchwan Ryu, and Feng Luan

See Also

`mmcmcBayes` for the main DMR detection function, `traceplot_asgn` for MCMC diagnostic plots

Examples

```
# Generate sample data
set.seed(2021)
dt <- rgamma(1000, shape = 2, rate = 1)
dt <- as.matrix(dt, ncol=1)

result <- asgn_func(dt, return_mcmc = TRUE, return_summary = TRUE)
print(result$summary)
```

`cancer_demo` *Cancer Methylation Demo Data*

Description

A demonstration dataset containing methylation M-values for cancer samples. Used for testing and examples in the mmcmcBayes package.

Usage

```
cancer_demo
```

Format

A data frame with CpG sites and methylation values.

Source

The first 5000 CpG sites of Chromosome 6 of 450K dataset.

Examples

```
data(cancer_demo)
head(cancer_demo)
```

compare_dmrs *Compare Differentially Methylated Regions (DMRs) from Two Methods*

Description

Identifies and analyzes overlapping regions between two sets of differentially methylated regions (DMRs) detected by different methods. Computes overlap percentages to assess consistency between detection approaches.

Usage

```
compare_dmrs(rst1, rst2)
```

Arguments

`rst1`	A data frame containing the first set of DMR results. Must contain columns: ‘Chromosome‘, ‘Start_CpG‘, and ‘End_CpG‘.
`rst2`	A data frame containing the second set of DMR results. Must contain columns: ‘Chromosome‘, ‘Start_CpG‘, and ‘End_CpG‘.

Details

This function performs comparison of genomic regions between two DMR detection results. It identifies both partial and complete overlaps between regions and calculates overlap percentage by total region size.

Value

A data frame with the following columns:

`Chromosome`	Chromosome name of the overlapping region
`Start_CpG_Method1`	Start CpG site from the first method
`End_CpG_Method1`	End CpG site from the first method

`Start_CpG_Method2`
: Start CpG site from the second method

`End_CpG_Method2`
: End CpG site from the second method

`Overlap_Percentage`
: Percentage of overlap

Returns NULL if no overlaps are found.

Author(s)

Zhexuan Yang, Duchwan Ryu, and Feng Luan

See Also

Related functions in this package: `mmcmcBayes` for DMR detection using multi-stage MCMC, `asgn_func` for parameter estimation with ASGN distribution

Examples

```
# Create sample DMR results
dmr_method1 <- data.frame(
  Chromosome = c("chr1", "chr1", "chr2"),
  Start_CpG = c("cg0001", "cg0050", "cg0100"),
  End_CpG = c("cg0020", "cg0070", "cg0150")
)

dmr_method2 <- data.frame(
  Chromosome = c("chr1", "chr2", "chr2"),
  Start_CpG = c("cg0005", "cg0120", "cg0090"),
  End_CpG = c("cg0025", "cg0160", "cg0110")
)

# Compare overlapping regions
overlaps <- compare_dmrs(dmr_method1, dmr_method2)
```

`mmcmcBayes` *Multi-stage MCMC Bayesian Method for DMR Detection*

Description

This function implements a multi-stage MCMC Bayesian method for detecting differentially methylated regions (DMRs) between cancer and normal groups. It uses the ASGN model for parameter estimation and provides both Bayes Factor and p-value based testing.

Usage

```
mmcmcBayes(
  cancer_data,
  normal_data,
  stage = 1,
  max_stages = 3,
  num_splits = 10,
  test = "BF",
  mcmc = NULL,
  priors_cancer = NULL,
  priors_normal = NULL,
  bf_thresholds = NULL,
  pvalue_thresholds = NULL,
  return_mcmc = FALSE
)
```

Arguments

`cancer_data` A matrix of methylation data for the cancer group (rows: regions, columns: samples)

`normal_data` A matrix of methylation data for the normal group (rows: regions, columns: samples)

`stage` The starting stage for multi-stage analysis (default: 1)

`max_stages` Maximum number of stages (default: 3)

`num_splits` Number of splits for the data in each stage (default: 10)

`test` Type of test to use: "BF" for Bayes Factor, "pvalue" for p-value (default: "BF")

`mcmc` A list of MCMC parameters (default: NULL, uses function defaults)

`priors_cancer` Prior parameters for the cancer group (default: NULL, uses function defaults)

`priors_normal` Prior parameters for the normal group (default: NULL, uses function defaults)

`bf_thresholds` Bayes Factor thresholds for each stage (default: NULL, uses function defaults)

`pvalue_thresholds`
p-value thresholds for each stage (default: NULL, uses function defaults)

`return_mcmc` Logical indicating whether to return MCMC samples for diagnostic purposes (default: FALSE)

Details

This function implements a multistage MCMC Bayesian approach for DMR detection. It recursively splits genomic regions and applies Bayesian testing. This function supports both Bayes Factor and Anderson-Darling tests for significance assessment. The algorithm begins by analyzing entire chromosomal regions, then recursively splits significant regions into smaller sub-regions for analysis, stopping when either maximum stages reached or no significant differences are detected.

Value

A list containing DMR detection results and, if requested, MCMC samples.

Returns NULL if no significant DMRs are detected.

Author(s)

Zhexuan Yang, Duchwan Ryu, and Feng Luan

See Also

Helper functions in this package: `asgn_func` for parameter estimation, `traceplot_asgn` for MCMC diagnostics, `compare_dmrs` for result comparison

Examples

```
# Load the datasets
data(cancer_demo)
data(normal_demo)

priors=list(alpha = 1,mu = 1,sigma2 = 1)

mcmc = list(nburn = 5000, niter = 10000, thin = 5)

set.seed(2021)
rst <- mmcmcBayes(cancer_demo, normal_demo,
                  stage = 1,max_stages = 2,num_splits = 5,
                  test = "BF", priors_cancer = NULL, priors_normal = NULL,
                  bf_thresholds = list(stage1 = 10, stage2 = 10.3, stage3 = 10.3),
                  return_mcmc = TRUE)
print(rst$dmrs)
```

`normal_demo` *Normal Methylation Demo Data*

Description

A demonstration dataset containing methylation M-values for normal samples. Used for testing and examples in the mmcmcBayes package.

Usage

```
normal_demo
```

Format

A data frame with CpG sites and methylation values.

Source

The first 5000 CpG sites of Chromosome 6 of 450K dataset.

Examples

```
data(normal_demo)
head(normal_demo)
```

---

traceplot_asgn *Create Traceplots for MCMC Parameters*

Description

Creates traceplots for MCMC parameters (alpha, mu, sigma2) to assess convergence. Users are responsible for setting up their preferred plot layout using par().

Usage

```
traceplot_asgn(mcmc_samples, param = "all", main = NULL)
```

Arguments

mcmc_samples	List containing MCMC samples for alpha, mu, and sigma2
param	Parameter to plot: "alpha", "mu", "sigma2", or "all" (default: "all")
main	Main title for the plot (optional)

Value

No return value, creates base R plots

Examples

```
# Load the datasets
data(cancer_demo)
data(normal_demo)
rst <- mmcmcBayes(cancer_demo, normal_demo,
                 stage = 1,max_stages = 2,num_splits = 5,
                 test = "BF", priors_cancer = NULL, priors_normal = NULL,
                 bf_thresholds = list(stage1 = 10, stage2 = 10.3, stage3 = 10.3),
                 return_mcmc = TRUE)

traceplot_asgn(rst$mcmc_samples$current_stage$cancer, param = "alpha", main = "Cancer Alpha")
```

Index