

# Package ‘DMCFB’

April 15, 2020

**Type** Package

**Title** Differentially Methylated Cytosines via a Bayesian Functional Approach

**Version** 1.0.0

**Author** Farhad Shokoohi

**Maintainer** Farhad Shokoohi <shokoohi@icloud.com>

**Description** DMCFB is a pipeline for identifying differentially methylated cytosines using a Bayesian functional regression model in bisulfite sequencing data. By using a functional regression data model, it tries to capture position-specific, group-specific and other covariates-specific methylation patterns as well as spatial correlation patterns and unknown underlying models of methylation data. It is robust and flexible with respect to the true underlying models and inclusion of any covariates, and the missing values are imputed using spatial correlation between positions and samples. A Bayesian approach is adopted for estimation and inference in the proposed method.

**Depends** R (>= 3.6.0), SummarizedExperiment, methods, S4Vectors, BiocParallel, GenomicRanges, IRanges

**Imports** utils, stats, speedglm, MASS, data.table, splines, arm, rtracklayer, benchmarkme, tibble, matrixStats, fastDummies, graphics

**Suggests** testthat, knitr, rmarkdown

**VignetteBuilder** knitr

**biocViews** DifferentialMethylation, Sequencing, Coverage, Bayesian, Regression

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**BugReports** <https://github.com/shokoohi/DMCFB/issues>

**RoxygenNote** 6.1.1

**git\_url** <https://git.bioconductor.org/packages/DMCFB>

**git\_branch** RELEASE\_3\_10

**git\_last\_commit** 834d392

**git\_last\_commit\_date** 2019-10-29

**Date/Publication** 2020-04-14

## R topics documented:

DMCFB-package . . . . .	2
BSDMC-class . . . . .	2
cBSDMC-method . . . . .	3
combine-method . . . . .	5
findDMCFB-method . . . . .	6
methLevels-method . . . . .	7
methReads-method . . . . .	8
params . . . . .	9
plotDMCFB-method . . . . .	11
readBismark-method . . . . .	12
totalReads-method . . . . .	13

<b>Index</b>	<b>14</b>
--------------	-----------

---

DMCFB-package	<i>Differentially Methylated cytosines using funtional Bayesian regression models</i>
---------------	---

---

### Description

DMCFB is a profiling tool for identifying differentially methylated cytosines using Functional Bayesian Model in bisulfite sequencing data.

### DMCFB methods

[findDMCFB](#), [plotDMCFB](#), [cBSDMC](#), [readBismark](#).

### BSDMC objects

[BSDMC-class](#)

---

BSDMC-class	<i>BSDMC object</i>
-------------	---------------------

---

### Description

The BSDMC object is an S4 class that represents differentially methylated CpG sites (DMCs) in BS-Seq Data.

### Arguments

methReads	The matrix methReads contains the number of methylated reads spanning a CpG-site. The rows represent the CpG sites in rowRanges and the columns represent the samples in colData.
totalReads	The matrix totalReads contains the number of reads spanning a CpG-site. The rows represent the CpG sites in rowRanges and the columns represent the samples in colData.
methLevels	The matrix methLevels contains the predicted methylation level spanning a CpG-site using Bayesian functional regression models. The rows represent the CpG sites in rowRanges and the columns represent the samples in colData.

**Value**

A [BSDMC-class](#) object

**Slots**

methReads An integer matrix  
 totalReads An integer matrix  
 methLevels A numeric matrix

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**See Also**

[RangedSummarizedExperiment-class](#) [GRanges-class](#)

**Examples**

```
nr <- 500
nc <- 16
metht <- matrix(as.integer(runif(nr * nc, 0, nr)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
meths <- matrix(as.integer(runif(nr * nc, 0, 10)), nr)
methl <- methc / metht
methv <- matrix((runif(nr * nc, 0.1, 0.5)), nr)
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc])
OBJ2 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, methStates = meths, methVars = methv, colData = cd1
)
OBJ2
```

---

cBSDMC-method

*cBSDMC method*


---

**Description**

Creates a [BSDMC-class](#) object

**Usage**

```
cBSDMC(methReads, totalReads, methLevels, rowRanges,
  colData = DataFrame(row.names = colnames(methReads)),
  metadata = list(), ...)
```

## S4 method for signature 'matrix,matrix,matrix,GRanges'

```
cBSDMC(methReads, totalReads,
  methLevels, rowRanges, colData = DataFrame(row.names =
  colnames(methReads)), metadata = list(), ...)
```

**Arguments**

methReads	The matrix methReads contains the number of methylated reads spanning a CpG-site. The rows represent the CpG sites in rowRanges and the columns represent the samples in colData.
totalReads	The matrix totalReads contains the number of reads spanning a CpG-site. The rows represent the CpG sites in rowRanges and the columns represent the samples in colData.
methLevels	The matrix methLevels contains the predicted methylation level spanning a CpG-site using Bayesian functional regression models. The rows represent the CpG sites in rowRanges and the columns represent the samples in colData.
rowRanges	A <a href="#">GRanges</a> or <a href="#">GRangesList</a> object describing the ranges of interest. Names, if present, become the row names of the <a href="#">SummarizedExperiment</a> object. The length of the <a href="#">GRanges</a> or <a href="#">GRangesList</a> must equal the number of rows of the matrices in assays. If rowRanges is missing, a <a href="#">SummarizedExperiment</a> instance is returned.
colData	Object of class 'DataFrame' containing information on variable values of the samples
metadata	A list of storing MCMC samples or DMCs
...	other possible parameters

**Details**

The rows of a BSDMC object represent ranges (in genomic coordinates) of interest. The ranges of interest are described by a [GRanges](#) or a [GRangesList](#) object, accessible using the `rowRanges` function. The [GRanges](#) and [GRangesList](#) classes contains sequence (e.g., chromosome) name, genomic coordinates, and strand information. Each range can be annotated with additional data; this data might be used to describe the range or to summarize results (e.g., statistics of differential abundance) relevant to the range. Rows may or may not have row names; they often will not.

**Value**

A [BSDMC-class](#)

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**Examples**

```
set.seed(1980)
nr <- 150
nc <- 8
metht <- matrix(as.integer(runif(nr * nc, 0, 100)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
meths <- matrix(as.integer(runif(nr * nc, 0, 10)), nr)
methl <- methc / metht
methv <- matrix((runif(nr * nc, 0.1, 0.5)), nr)
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(
  Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc]
```

```

)
OBJ2 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, methStates = meths, methVars = methv, colData = cd1
)
OBJ2

```

---

combine-method

*combine method*


---

## Description

combine two [BSDMC-class](#) or two [BSDMC-class](#)

## Usage

```

combine(obj1, obj2)

## S4 method for signature 'BSDMC,BSDMC'
combine(obj1, obj2)

```

## Arguments

```

obj1          A BSDMC-class
obj2          A BSDMC-class

```

## Value

A [BSDMC-class](#) or [BSDMC-class](#)

## Author(s)

Farhad Shokoohi <shokoohi@icloud.com>

## Examples

```

set.seed(1980)
nr <- 150
nc <- 8
metht <- matrix(as.integer(runif(nr * nc * 2, 0, nr)), nr)
methc <- matrix(
  rbinom(n = nr * nc, c(metht), prob = runif(nr * nc * 2)),
  nr, nc * 2
)
methl <- methc / metht
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(Group = rep("G1", each = nc), row.names = LETTERS[1:nc])
OBJ1 <- cBSDMC(
  rowRanges = r1, methReads = methc[, 1:nc], totalReads = metht[, 1:nc],
  methLevels = methl[, 1:nc], colData = cd1
)
cd2 <- DataFrame(
  Group = rep("G2", each = nc),

```

```

    row.names = LETTERS[nc + 1:nc]
  )
  OBJ2 <- cBSDMC(
    rowRanges = r1, methReads = methc[, nc + 1:nc], totalReads =
      metht[, nc + 1:nc], methLevels = methl[, nc + 1:nc], colData = cd2
  )
  OBJ3 <- combine(OBJ1, OBJ2)
  OBJ3

```

---

 findDMCFB-method

*findDMCFB method*


---

## Description

DMC identification via Bayesian functional regression models

## Usage

```
findDMCFB(object, bwa, bwb, nBurn, nMC, nThin, alpha, sdv, nCores, pSize,
  sfiles)
```

```
## S4 method for signature 'BSDMC'
findDMCFB(object, bwa, bwb, nBurn, nMC, nThin, alpha,
  sdv, nCores, pSize, sfiles)
```

## Arguments

object	A <a href="#">BSDMC-class</a> object
bwa	An integer value specifying the band-width size of B-spline basis matrix for a natural cubic spline for the group-specific effects of the Bayesian functional regression model
bwb	An integer value specifying the band-width size of B-spline basis matrix for a natural cubic spline for the individual-specific effects of the Bayesian functional regression model
nBurn	An integer value specifying the number of burn-in samples
nMC	An integer value specifying the number of MCMC samples after burn-in
nThin	An integer value specifying the thinning number in MCMC
alpha	A numeric value specifying the level of $\alpha$ in credible interval $(1 - \alpha)\%$
sdv	An double value specifying the standard deviation of priors
nCores	An integer value specifying the number of machine cores for parallel computing
pSize	An integer value specifying the number of cytosines in a region to be used in a Bayesian functional regression model for DMC detection
sfiles	A logical value indicating whether files to be saved or not.

## Value

[BSDMC-class](#) object

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**Examples**

```
set.seed(1980)
nr <- 1000
nc <- 4
metht <- matrix(as.integer(runif(nr * nc, 0, 100)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
methl <- methc / metht
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(
  Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc]
)
OBJ1 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, colData = cd1
)
OBJ2 <- findDMCFB(OBJ1,
  bwa = 10, bwB = 10, nBurn = 50, nMC = 50, nThin = 1,
  alpha = 0.05, nCores = 2, pSize = 500, sfiles = FALSE
)
OBJ2
```

---

methLevels-method      *methLevels method*

---

**Description**

Returns methLevels stored in [BSDMC-class](#)

Assigns methLevels to [BSDMC-class](#)

**Usage**

```
methLevels(object)

methLevels(object) <- value

## S4 method for signature 'BSDMC'
methLevels(object)

## S4 replacement method for signature 'BSDMC,matrix'
methLevels(object) <- value
```

**Arguments**

object            A [BSDMC-class](#) object

value             An integer matrix

**Value**

A matrix

A [BSDMC-class](#) object

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**Examples**

```
nr <- 150
nc <- 8
metht <- matrix(as.integer(runif(nr * nc, 0, 100)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
methl <- methc / metht
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(
  Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc]
)
OBJ1 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, colData = cd1
)
methLevels(OBJ1)
methLevels(OBJ1) <- methl
```

---

methReads-method

*methReads method*

---

**Description**

Returns methReads stored in [BSDMC-class](#)

Assigns methReads to [BSDMC-class](#)

**Usage**

```
methReads(object)
```

```
methReads(object) <- value
```

```
## S4 method for signature 'BSDMC'
methReads(object)
```

```
## S4 replacement method for signature 'BSDMC,matrix'
methReads(object) <- value
```

**Arguments**

object            A [BSDMC-class](#) object

value            An integer matrix



**Value**

A matrix  
 A [BSDMC-class](#) object

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**Examples**

```
nr <- 150
nc <- 8
metht <- matrix(as.integer(runif(nr * nc, 0, 100)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
methl <- methc / metht
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(
  Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc]
)
OBJ1 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, colData = cd1
)
methReads(OBJ1)
methReads(OBJ1) <- methc
```

---

 params

*params*


---

**Description**

parameters name and their descriptions

**Arguments**

methReads	The matrix <code>methReads</code> contains the number of methylated reads spanning a CpG-site. The rows represent the CpG sites in <code>rowRanges</code> and the columns represent the samples in <code>colData</code> .
totalReads	The matrix <code>totalReads</code> contains the number of reads spanning a CpG-site. The rows represent the CpG sites in <code>rowRanges</code> and the columns represent the samples in <code>colData</code> .
methLevels	The matrix <code>methLevels</code> contains the predicted methylation level spanning a CpG-site using Bayesian functional regression models. The rows represent the CpG sites in <code>rowRanges</code> and the columns represent the samples in <code>colData</code> .
rowRanges	A <a href="#">GRanges</a> or <a href="#">GRangesList</a> object describing the ranges of interest. Names, if present, become the row names of the <a href="#">SummarizedExperiment</a> object. The length of the <a href="#">GRanges</a> or <a href="#">GRangesList</a> must equal the number of rows of the matrices in assays. If <code>rowRanges</code> is missing, a <a href="#">SummarizedExperiment</a> instance is returned.

colData	Object of class 'DataFrame' containing information on variable values of the samples
metadata	A list of storing MCMC samples or DMCs
object	A <a href="#">BSDMC-class</a> object
value	An integer matrix
name	A character list
obj1	A <a href="#">BSDMC-class</a>
obj2	A <a href="#">BSDMC-class</a>
files	A character list
file	A character
nCores	An integer value specifying the number of machine cores for parallel computing
pSize	An integer value specifying the number of cytosines in a region to be used in a Bayesian functional regression model for DMC detection
bwa	An integer value specifying the band-width size of B-spline basis matrix for a natural cubic spline for the group-specific effects of the Bayesian functional regression model
bwb	An integer value specifying the band-width size of B-spline basis matrix for a natural cubic spline for the individual-specific effects of the Bayesian functional regression model
nBurn	An integer value specifying the number of burn-in samples
nThin	An integer value specifying the thinning number in MCMC
nMC	An integer value specifying the number of MCMC samples after burn-in
sdv	An double value specifying the standard deviation of priors
alpha	A numeric value specifying the level of $\alpha$ in credible interval $(1 - \alpha)\%$
col	A character vector indicating which colors to alternate.
sfiles	A logical value indicating whether files to be saved or not.
region	An integer vector of length two specifying which subset of the object to be plotted
nSplit	A integer value specifying the number of subsets must be done for plotting the results of DMC identification
parList	A list specifying plots parameters, see <a href="#">par</a>
...	other possible parameters

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

---

plotDMCFB-method	<i>plotDMCFB method</i>
------------------	-------------------------

---

**Description**

Plotting the results of DMC identification stored in a [BSDMC-class](#) object

**Usage**

```
plotDMCFB(object, region, nSplit, parList)
```

```
## S4 method for signature 'BSDMC'
plotDMCFB(object, region, nSplit, parList)
```

**Arguments**

object	A <a href="#">BSDMC-class</a> object
region	An integer vector of length two specifying which subset of the object to be plotted
nSplit	A integer value specifying the number of subsets must be done for plotting the results of DMC identification
parList	A list specifying plots parameters, see <a href="#">par</a>

**Value**

Plot

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**Examples**

```
set.seed(1980)
nr <- 1000
nc <- 4
metht <- matrix(as.integer(runif(nr * nc, 0, 100)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
methl <- methc / metht
r1 <- GRanges(rep("chr1", nr), IRanges(1:nc, width = 1), strand = "*")
names(r1) <- 1:nc
cd1 <- DataFrame(
  Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc]
)
OBJ1 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, colData = cd1
)
OBJ2 <- findDMCFB(OBJ1,
  bwa = 10, bwb = 10, nBurn = 50, nMC = 50, nThin = 1,
  alpha = 0.05, nCores = 2, pSize = 500, sfiles = FALSE
```

```
)
plotDMCFB(OBJ2)
```

---

```
readBismark-method    readBismark method
```

---

### Description

reads BS-Seq data

### Usage

```
readBismark(files, colData)

## S4 method for signature 'character,DataFrame'
readBismark(files, colData)

## S4 method for signature 'character,data.frame'
readBismark(files, colData)

## S4 method for signature 'character,character'
readBismark(files, colData)
```

### Arguments

files	A character list
colData	Object of class 'DataFrame' containing information on variable values of the samples

### Value

A `BSDMC-class` object

### Author(s)

Farhad Shokoohi <shokoohi@icloud.com>

### Examples

```
fn <- list.files(system.file("extdata", package = "DMCHMM"))
fn.f <- list.files(system.file("extdata", package = "DMCHMM"),
  full.names = TRUE
)
OBJ <- readBismark(fn.f, fn)
cdOBJ <- DataFrame(Cell = factor(c("BC", "TC", "Mono"),
  labels = c("BC", "TC", "Mono")
), row.names = c("BCU1568", "BCU173", "BCU551"))
colData(OBJ) <- cdOBJ
OBJ
```

---

totalReads-method	<i>totalReads method</i>
-------------------	--------------------------

---

**Description**

Returns totalReads stored in [BSDMC-class](#)

Assigns totalReads to [BSDMC-class](#)

**Usage**

```
totalReads(object)
```

```
totalReads(object) <- value
```

```
## S4 method for signature 'BSDMC'
```

```
totalReads(object)
```

```
## S4 replacement method for signature 'BSDMC,matrix'
```

```
totalReads(object) <- value
```

**Arguments**

object            A [BSDMC-class](#) object

value            An integer matrix

**Value**

A matrix

A [BSDMC-class](#) object

**Author(s)**

Farhad Shokoohi <shokoohi@icloud.com>

**Examples**

```
nr <- 150
nc <- 8
metht <- matrix(as.integer(runif(nr * nc, 0, 100)), nr)
methc <- matrix(rbinom(n = nr * nc, c(metht), prob = runif(nr * nc)), nr, nc)
methl <- methc / metht
r1 <- GRanges(rep("chr1", nr), IRanges(1:nr, width = 1), strand = "*")
names(r1) <- 1:nr
cd1 <- DataFrame(
  Group = rep(c("G1", "G2"), each = nc / 2),
  row.names = LETTERS[1:nc]
)
OBJ1 <- cBSDMC(
  rowRanges = r1, methReads = methc, totalReads = metht,
  methLevels = methl, colData = cd1
)
totalReads(OBJ1)
totalReads(OBJ1) <- metht
```

# Index

## \*Topic **object**

- BSDMC-class, [2](#)
- BSDMC (BSDMC-class), [2](#)
- BSDMC-class, [2](#)
- cBSDMC, [2](#)
- cBSDMC (cBSDMC-method), [3](#)
- cBSDMC, matrix, matrix, matrix, GRanges-method (cBSDMC-method), [3](#)
- cBSDMC-method, [3](#)
- combine (combine-method), [5](#)
- combine, BSDMC, BSDMC-method (combine-method), [5](#)
- combine-method, [5](#)
- DMCFB (DMCFB-package), [2](#)
- DMCFB-package, [2](#)
- findDMCFB, [2](#)
- findDMCFB (findDMCFB-method), [6](#)
- findDMCFB, BSDMC-method (findDMCFB-method), [6](#)
- findDMCFB-method, [6](#)
- GRanges, [4](#), [9](#)
- GRangesList, [4](#), [9](#)
- methLevels (methLevels-method), [7](#)
- methLevels, BSDMC-method (methLevels-method), [7](#)
- methLevels-method, [7](#)
- methLevels<- (methLevels-method), [7](#)
- methLevels<-, BSDMC, matrix-method (methLevels-method), [7](#)
- methReads (methReads-method), [8](#)
- methReads, BSDMC-method (methReads-method), [8](#)
- methReads-method, [8](#)
- methReads<- (methReads-method), [8](#)
- methReads<-, BSDMC, matrix-method (methReads-method), [8](#)
- par, [10](#), [11](#)
- params, [9](#)
- plotDMCFB, [2](#)
- plotDMCFB (plotDMCFB-method), [11](#)
- plotDMCFB, BSDMC-method (plotDMCFB-method), [11](#)
- plotDMCFB-method, [11](#)
- readBismark, [2](#)
- readBismark (readBismark-method), [12](#)
- readBismark, character, character-method (readBismark-method), [12](#)
- readBismark, character, data.frame-method (readBismark-method), [12](#)
- readBismark, character, DataFrame-method (readBismark-method), [12](#)
- readBismark-method, [12](#)
- SummarizedExperiment, [4](#), [9](#)
- totalReads (totalReads-method), [13](#)
- totalReads, BSDMC-method (totalReads-method), [13](#)
- totalReads-method, [13](#)
- totalReads<- (totalReads-method), [13](#)
- totalReads<- , BSDMC, matrix-method (totalReads-method), [13](#)