Package 'DMRMark'

April 21, 2017

Type Package

Title DMR Detection by Non-Homogeneous Hidden Markov Model from Methylation Array Data

Version 1.1.1

Date 2017-02-25

Author Linghao SHEN <sl013@ie.cuhk.edu.hk>

Maintainer Linghao SHEN <sl013@ie.cuhk.edu.hk>

Depends MCMCpack, mvtnorm, ellipse

Description Perform differential analysis for methylation array data. Detect differentially methylated regions (DMRs) from array M-values. The core is a Non-homogeneous Hidden Markov Model for estimating spatial correlation and a novel Constrained Gaussian Mixture Model for modeling the M-value pairs of each individual locus.

License GPL

NeedsCompilation no

Repository CRAN

Date/Publication 2017-04-21 17:58:43 UTC

R topics documented:

DMRMark-package *DMR Detection by Non-Homogeneous Hidden Markov Model from Methylation Array Data*

Description

Perform differential analysis for methylation array data. DMRMark detects differentially methylated regions (DMRs) from array M-values. The core of DSS is a Non-homogeneous Hidden Markov Model for estimating spatial correlation and a novel Constrained Gaussian Mixture Model for modeling the M-value pairs of each individual locus.

DMRMark only works for two-group comparisons currently. We have the plan to extend the transition and response model that make then suitable for complex experimental designs in the future.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

Examples

```
# DMR detection performed on chr18 of a small BLCA dataset from TCGA
data(BLCA)

# Use a small subset
nprobe <- 500
# M-values
mv <- BLCA$mv[1:nprobe,]

# Distance between probes, L<0 indicates acorssing chromosomes
L = BLCA$distance[1:nprobe]

# Initialize new chain when probe distance too long
# or across different chromosomes
newChains <- which((L > 100000) | L < 0)
# The starting positions of new chains
starting <- c(1, newChains[-length(newChains)]+1)

# Run DMRMark with default options
set.seed(0)
par <- DMRMark(mv, L, starting)

# Get the posterior of being certain states
# Return the result of DMC for plotting by setting 'region=FALSE'
results <- DMRViterbi(mv, par, L, starting, region=FALSE)

# The MAP states being 3 or 4 indicate DMCs
isDMC <- (results$states > 2) + 0
mvScatter(mv, isDMC, nPlot=10000)
```

BLCA — *Single paired M-values of BLCA chr18 from TCGA*

Description

This data set contains one pair of M-values of BLCA chr18 from The Cancer Genome Atlas TCGA (TCGA). In addition, it contains the distance between each probes and the gold standard methylation status getting by the matched WGBS data (also from TCGA).

Usage

```
data(BLCA)
```

Format

A List with the following items, all items will length 5492 corresponding to the Illumina 450K probes on chr18:

mv A matrix with one pair of M-values

distance a numeric vector of the probe distance

truth a binary vector represents the WGBS methylation status (0-nDML, 1-DML)

Source

Data generated by the TCGA Research Network: <http://cancergenome.nih.gov/>

Examples

```
data(BLCA)
```

boundFinder — *Find a pair of reasonable distances of group means for hyper- and hypomethylation based on the quantile of two-group difference.*

Description

This function takes the M-values to produce the distance D to be the maximum value satisfies that the proportion of absolute values of two-group difference larger than D is at certain level. Due to the precision limitation, the D's for hyper- and hypomethylation are not necessarily the same. If the samples are not totally paired, than user should first call 'reformData' to process M-values.

Usage

```
boundFinder(mv, prop = 0.1)
```

Arguments

mv	The input M-values matrix. If the samples are not totally paired, than user MUST first call "reformData" to process M-values.
prop	The proportion that absolute values of two-group difference larger than D must be satisfied. Default value is 0.1

Details

The choices of 'prop' are not going to be too extreme or stringent, which will produce to dominated prior. This value should depend on the belief that around 'prop' proportion of loci are differentially methylated. In general 0.1 to 0.2 should be reasonable and and well-performed. Users may also choose different D's for two differentially methylation status freely, in this situation, values around 1.5 to 3 are recommended.

Users must ensure the M-values come from paired samples or has been processed by 'reformData' according to the experiment design.

Value

A two-value vector contains D_1 and D_2 for the group-mean difference of hypermethylation and hypomethylation respectively. Due to the precision limitation, D_1 may not necessarily equal to D_2.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

See Also

reformData to tackle unpaired data.

Examples

```
# Finding the 5% and 95% quantile of normal samples
set.seed(0)
mv <- cbind(rep(0,100000),rnorm(100000))
boundFinder(mv)

# Output matched the normal p-values
#     5.0%     94.9%
#-1.639578  1.639691
```

DMRMark *Gibbs Sampler to estimate model parameters*

Description

Given the M-values and probe distance, this function calls Gibbs Sampler for estimating the parameters of non-homogeneous hidden Markov model.

Usage

```
DMRMark(mv, L = rep(1, nrow(mv)), starting = NULL,
pd = NULL, initHeuristic = TRUE,
GSoptions = NULL)
```

Arguments

mv	The input M-values matrix, NA is not allowed.
L	A vector to specify the distance between each probes in bp. $L < 0$ represents change of chromosome. Default is $L = 1$ for all probes.
starting	A vector to specify the position to initial new chains. We suggest new chains should be initiated at least at starting of new chromosome. When it is null, new chains initiate at beginning and where $L > 100000 or $L < 0$.
pd	A design matrix, which can be generated by 'stats::model.matrix'. If the M-values are totally paired or single paired, just leave it to be NULL.
initHeuristic	If set to TRUE, heuristics will be used for faster computation, which rely on finding good initial value and then using less iterations. . This will mask GS controls parameters of 'GSoptions'. Recommended for getting some quick insight about new study. Default it TRUE.
GSoptions	List of prior parameters and GS control parameters. See MakeGSoptions.

Details

This function is the main functionality of this package. It takes the M-values and probe distance and calls Gibbs Sampler for estimating the parameters of non-homogeneous hidden Markov model. New chains will be initiated at positions specified in 'starting'. Depends on the scale of M-values, this function may take certain time to the GS. In this situation user may first set 'initHeuristic = TRUE' for a quick insight.

Value

The return value depends on 'GSoptions$track'. In default situation ('GSoptions$track = FALSE'), the return value is a list contains:

theta	A vector contains posterior means of non-DMC's control groups.
mu	A 2-by-2 matrix, each row corresponding to the paired posterior mean of DMCs.
sigma12	A vector contains posterior means of variance of non-DMC's control groups.

sigmaN	Single value, the posterior mean of variance of non-DMC's between-group difference.
Sigma34	An Array contains posterior means of DMC's Covariance.
charL	Posterior means of characteristic length.
init	The probabilities of the initial states of all chains. Sum up to 1. If 'GSoptions$track = TRUE', an additional dimension will be added to each item of the list, and along this dimension user can retrieve the sample from each iterations.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

See Also

See MakeGSoptions for different prior parameters and Gibbs Sampler control parameters. See DMRViterbi for interpreting the estimated parameters.

Examples

```
# DMRMark
# DMR detection performed on chr18 of a small BLCA dataset from TCGA
data(BLCA)

# Use a small subset
nprobe <- 500
# M-values
mv <- BLCA$mv[1:nprobe,]

# Distance between probes, L<0 indicates acorssing chromosomes
L = BLCA$distance[1:nprobe]

# Initialize new chain when probe distance too long
# or across different chromosomes
newChains <- which((L > 100000) | L < 0)
# The starting positions of new chains
starting <- c(1, newChains[-length(newChains)]+1)

# Run DMRMark with default options
pars <- DMRMark(mv, L, starting)
pars
```

DMRViterbi *Viterbi algorithm to estimate posterior probabilities of DMRs.*

Description

This function takes M-values and estimated parameters from 'DMRMark', then uses Viterbi algorithm for estimating states' posterior probabilities for each locus.

Usage

```
DMRViterbi(mv, pars, L = rep(1, nrow(mv)), starting = NULL,
pd = NULL, region = TRUE,
orderBy = c("max", "mean", "median", "min"), VitP = NULL)
```

Arguments

mv	The input M-values matrix, NA is not allowed.
pars	The list of model parameters. Getting by calling 'DMRMark'.
L	A vector to specify the distance between each probes in bp. $L < 0$ represents change of chromosome. Default is $L = 1$ for all probes.
starting	A vector to specify the position to initial new chains. We suggest new chains should be initiated at least at starting of new chromosome. When it is null, new chains initiate at beginning and where $L > 100000 or $L < 0$.
pd	A design matrix, which can be generated by 'stats::model.matrix'. If the M-values are totally paired or single paired, just leave it to be NULL.
region	If set to TRUE, this function returns the regions formed by Viterbi posterior states. Otherwise, it returns posterior probabilities and states for individual loci. Default is TRUE.
orderBy	Only enabled when 'region = TRUE'. Order the regions by which statistics? Choice include 'max', 'mean', 'median' and 'min', which orders the regions by the maximum, geometric mean, median or minimum of the posterior probabilities in each region respectively. Default is 'max'.
VitP	Only enabled when 'region = FALSE'. The minimum posterior probabilities required to be the DMC states. A locus with DMC's posterior probability lower than 'VitP' will in the non-DMC states with highest probabilities. When set to NULL, simply return the MAP states. Default is NULL.

Value

If 'region = FALSE', the return value is a list contains:

states	The MAP methylation status satisfies the 'VitP'.
deltas	The matrix with each row corresponds to the posterior probabilities of each locus in which states.

If 'region = TRUE', the return value is a dataframe with the following fields:

begin	Beginning of each region. In probe index.
ending	Ending of each region. In probe index.
MAP_state	The MAP state of each region.

minVP	The minimum Viterbi posterior probability of the MAP state in each region
meanVP	The geometric mean of Viterbi posterior probability of the MAP state in each region
maxVP	The maximum Viterbi posterior probability of the MAP state in each region
midVP	The median Viterbi posterior probability of the MAP state in each region

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

See Also

See DMRMark about model parameter estimation

Examples

```
# DMRViterbi
# DMR detection performed on chr18 of a small BLCA dataset from TCGA
data(BLCA)

# Use a small subset
nprobe <- 500
# M-values
mv <- BLCA$mv[1:nprobe,]

# Distance between probes, L<0 indicates acorssing chromosomes
L = BLCA$distance[1:nprobe]

# Initialize new chain when probe distance too long
# or across different chromosomes
newChains <- which((L > 100000) | L < 0)
# The starting positions of new chains
starting <- c(1, newChains[-length(newChains)]+1)

# Run DMRMark with default options
pars <- DMRMark(mv, L, starting)
# Get the posterior of being certain states
results <- DMRViterbi(mv, pars, L, starting)
head(results)
```

FullSample *Function implementing Gibbs Sampler with old-version interface*

Description

This function implements Gibbs Sampler for estimating model parameters, but with the old version interface. This function remains callable just for backward compatibility, and not going to be used by new users.

Details

This function not going to be used by new users.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

`MakeGSoptions` *Encapsulate prior parameters and Gibbs Sampler (GS) control parameters*

Description

This function encapsulate prior parameters and Gibbs Sampler control parameters. All parameters with initial values. The encapsulation is for easy initiating, managing and passing of parameters.

Usage

```
MakeGSoptions(pi0 = c(100, 100, 5, 5),
  cmu0 = c(11.5, 11.5, 8, 8),
  theta0 = c(-3, 2),
  mu0 = matrix(c(-2, 2, 2, -2), 2, byrow = TRUE),
  kappa0 = c(50, 50, 5, 5),
  nu0 = rep(4, 2),
  A0 = array(rep(c(2, 0.8, 0.8, 4), 2),
  dim = c(2, 2, 2)),
  alpha12N = rep(40, 3),
  beta12N = rep(60, 3),
  D_mu = rep(-2, 2),
  chi_alpha = 0.2, #This and above for priors
  burnin = 500, #This and below for Gibbs Sampler Control
  nsamples = 100,
  sampleSep = 10,
  onHMM = TRUE,
  track = FALSE,
  verbose = FALSE)
```

Arguments

`pi0`	Length-4 vector, the concentration of Dirichlet distribution. Prior of initial states.
`cmu0`	Single value, the mean of Normal distribution. Prior of characteristic length.
`theta0`	Length-2 vector, each value is the mean of a Normal distributions. Priors for means of control groups of two non-differentially methylated CpG sites (non-DMCs) responses.

mu0	2-by-2 matrix, each row is the means of a bivariate Normal distributions. Priors for means of two DMCs responses
kappa0	Length-4 vector, each value is the prior observation number of Normal-Inverse-Gamma (NIG) or Normal-Inverse-Wishart (NIW) depends on the corresponding state.
nu0	Length-2 vector, each value is the degree of freedom of an IW distribution. Priors for covariance of DMC responses.
A0	2-by-2-by-2 array, each 2-by-2 matrix along the third dimension is the scale matrix of an IW distribution. Priors for covariance of DMC responses.
alpha12N	Length-3 vector, each value is the shape of an IG distribution. Priors for variance of non-DMC responses.
beta12N	Length-3 vector, each value is the rate of an IG distribution. Priors for variance of non-DMC responses.
D_mu	Length-2 vector, each value is the minimum distance between two group means of DMCs. Prior for truncating the means of bivariate normals of DMC's responses.
chi_alpha	p-value of chi-square distribution with 2 degrees of freedom. Prior for truncating the covariant matrices of bivariate normals of DMC's responses.
burnin	Number of iterations for burn-in. Gibbs Sampler control parameter. Default is 500.
nsamples	Number of samples to compute the point estimators. Gibbs Sampler control parameter. Default is 100.
sampleSep	Only keep every 'sampleSep'-th samples to estimate point estimators. Gibbs Sampler control parameter. Default is 10.
onHMM	Set to FALSE will disable HMM, and reduce to simple clustering of Mixture Model. Gibbs Sampler control parameter. Default is TRUE.
track	Set to TRUE will make DMRMark return all samples from the beginning of burn-in to the end of sampling instead of point estimators. Useful for inspecting convergence. Please know well about this issue before you decide to set it to TRUE. Gibbs Sampler control parameter. Default is TRUE.
verbose	Set to TRUE to show the details when running the Gibbs Sampler. Gibbs Sampler control parameter. Default is FALSE.

Value

Simply a list with all items are the same with input. Just an encapsulation.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

See Also

DMRMark

Examples

```
# MakeGSoptions
opts <- MakeGSoptions()
```

`mvScatter` *Visualize the distributions of M-value pairs from differentially methylated CpG sites (DMC) or non-DMCs*

Description

Given the M-values, True DMCs and optional the experiment design, plot the scatter plot of M-values. DMCs are marked by red daggers and non-DMCs by green circles.

Usage

```
mvScatter(mv, isDMC, pd = NULL, nPlot = 5000)
```

Arguments

`mv`	The input M-values matrix, NA is not allowed.
`isDMC`	A binary vector corresponding to each row of 'mv'. 0 indicates non-DMC and 1 for DMC.
`pd`	A design matrix, which can be generated by 'stats::model.matrix'. If the M-values are totally paired or single paired, just leave it to be NULL.
`nPlot`	The maximum number of loci to be plotted. Using too large value will lead to messy scatter and long execution time. Default is 5000.

Value

This function only generates a figure and has no return value.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

Examples

```
# mvScatter
data(BLCA)
mvScatter(BLCA$mv, BLCA$truth, nPlot = 10000)
```

reformData *Reform M-values into a two-column matrix.*

Description

Reform M-values into a matrix with two columns representing matched control and case groups. It concatenate M-values pair-by-pair based on the design matrix.

Usage

```
reformData(mv, pd=NULL)
```

Arguments

mv	The input M-values matrix, NA is not allowed.
pd	A design matrix, which can be generated by 'stats::model.matrix'. If the M-values are totally paired or single paired, just leave it to be NULL.

Value

A matrix with two columns representing matched control and case groups. If a sample has no paired sample in another group (say group B), then the values in group B will be represented by NA.

Author(s)

Linghao SHEN <sl013@ie.cuhk.edu.hk>

Examples

```
# Assume the values come from Tumor is 10 larger than those from Normal.

# The case with totally paired data
mv1 <- matrix(1:20,5)
reformData(mv1)

# The case with One more sample from Tumour group
# The second Tumour sample is the extra one
mv2 <- matrix(1:25,5)
mv2[,2] <- mv2[,2] + 5
patient <- factor(c(1,3,1:3))
type = c(rep("Normal",2),rep("Tumour",3))
pd <- model.matrix(~patient + type + 0)
reformData(mv2, pd)
```

Index