Package 'betaclust'

November 23, 2022

Type Package

Title A Family of Beta Mixture Models for Clustering Beta-Valued DNA Methylation Data

Version 1.0.0

Description A family of novel beta mixture models (BMMs) to appositely model the beta-valued cytosine-guanine dinucleotide (CpG) sites, to objectively identify methylation state thresholds and to identify the differentially methylated CpG (DMC) sites using a model-based clustering approach. The family of BMMs employs different parameter constraints applicable to different study settings. The EM algorithm is used for parameter estimation, with a novel approximation during the M-step providing tractability and ensuring computational feasibility.

License GPL-3

Depends R (>= 3.5.0)

Imports foreach, doParallel, stats, utils, ggplot2, plotly, scales, devtools

Encoding UTF-8

NeedsCompilation yes

LazyData true

LazyDataCompression xz

RoxygenNote 7.2.2

Suggests rmarkdown, knitr

VignetteBuilder knitr

Maintainer Koyel Majumdar <koyel.majumdar@ucdconnect.ie>

Author Koyel Majumdar [aut, cre] (<https://orcid.org/0000-0003-4692-9002>), Thomas Brendan Murphy [aut] (<https://orcid.org/0000-0002-5668-7046>), Isobel Claire Gormley [aut] (<https://orcid.org/0000-0001-7713-681X>)

R topics documented:

betaclust 2
beta_k 4
beta_kn 5
beta_kr 7
ecdf.betaclust 8
em_aic 9
em_bic 9

em_icl 10
legacy.data 11
pca.methylation.data 12
plot.betaclust 13
summary.betaclust 14
threshold 15

Index **17**

`betaclust` *The betaclust wrapper function*

Description

A family of model-based clustering techniques to identify methylation states in beta-valued DNA methylation data.

Usage

```
betaclust(
  data,
  M = 3,
  N,
  R,
  model_names = "K..",
  model_selection = "BIC",
  seed = NULL
)
```

Arguments

`data`	A dataframe of dimension $C \times NR$ containing methylation values for C CpG sites from R samples collected from N patients. Samples are grouped together in the dataframe such that the columns are ordered as Sample1_Patient1, Sample1_Patient2, Sample2_Patient1, Sample2_Patient2, etc.
`M`	Number of methylation states to be identified in a DNA sample.
`N`	Number of patients in the study.
`R`	Number of samples collected from each patient for the study.
`model_names`	Models to run from the set of models, K.., KN. and K.R, default = K.. . See details.
`model_selection`	Information criterion used for model selection. Options are AIC, BIC or ICL (default = BIC).
`seed`	Seed to allow for reproducibility (default = NULL).

Details

This is a wrapper function which can be used to fit all three models (K.., KN., K.R) within a single function.

The K.. and KN. models are used to analyse a single DNA sample ($R = 1$) and cluster the C CpG sites into the K clusters which represent the different methylation states in a DNA sample. As each CpG site can belong to any of the $M = 3$ methylation states (hypomethylation, hemimethylation and hypermethylation), the default value for $K = M = 3$. The thresholds between methylation states are objectively inferred from the clustering solution.

The K.R model is used to analyse R independent samples collected from N patients, where each sample contains C CpG sites, and cluster the dataset into $K = M^R$ clusters to identify the differentially methylated CpG (DMC) sites between the R DNA samples.

Value

The function returns an object of the `betaclust` class which contains the following values:

- information_criterion - The information criterion used to select the optimal model.
- ic_output - The information criterion value calculated for each model.
- optimal_model - The model selected as optimal.
- function_call - The parameters passed as arguments to the function `betaclust`.
- K - The number of clusters identified using the beta mixture models.
- C - The number of CpG sites analysed using the beta mixture models.
- N - The number of patients analysed using the beta mixture models.
- R - The number of samples analysed using the beta mixture models.
- optimal_model_results - Information from the optimal model. Specifically,
  - cluster_size - The total number of CpG sites in each of the K clusters.
  - llk - A vector containing the log-likelihood value at each step of the EM algorithm.
  - alpha - This contains the first shape parameter for the beta mixture model.
  - delta - This contains the second shape parameter for the beta mixture model.
  - tau - The proportion of CpG sites in each cluster.
  - z - A matrix of dimension $C \times K$ containing the posterior probability of each CpG site belonging to each of the K clusters.
  - classification - The classification corresponding to z, i.e. map(z).
  - uncertainty - The uncertainty of each CpG site's clustering.
  - thresholds - Threshold points calculated under the K.. or the KN. model.

References

Silva, R., Moran, B., Russell, N.M., Fahey, C., Vlajnic, T., Manecksha, R.P., Finn, S.P., Brennan, D.J., Gallagher, W.M., Perry, A.S.: Evaluating liquid biopsies for methylomic profiling of prostate cancer. Epigenetics 15(6-7), 715-727 (2020). doi: 10.1080/15592294.2020.1712876.

Majumdar, K., Silva, R., Perry, A.S., Watson, R.W., Murphy, T.B., Gormley, I.C.: betaclust: a family of mixture models for beta valued DNA methylation data. arXiv [stat.ME] (2022). doi: 10.48550/ARXIV.2211.01938.

See Also

beta_k

beta_kn

beta_kr

pca.methylation.data

plot.betaclust

summary.betaclust

threshold

Examples

```
## Not run:
data(pca.methylation.data)
my.seed <- 190
M <- 3
N <- 4
R <- 2
data_output <- betaclust(pca.methylation.data[,2:9], M, N, R,
            model_names = c("K..","KN.","K.R"), model_selection = "BIC", seed = my.seed)

## End(Not run)
```

---

beta_k *Fit the K.. model*

Description

Fit the K.. model from the betaclust family of beta mixture models for DNA methylation data. The K.. model analyses a single DNA sample and identifies the thresholds between the different methylation states.

Usage

```
beta_k(data, M = 3, seed = NULL)
```

Arguments

data	A dataframe of dimension $C \times N$ containing methylation values for C CpG sites from $R = 1$ samples collected from N patients. Samples are grouped together in the dataframe such that the columns are ordered as Sample1_Patient1, Sample1_Patient2, etc.
M	Number of methylation states to be identified in a DNA sample.
seed	Seed to allow for reproducibility (default = NULL).

Details

The K.. model clusters each of the C CpG sites into one of K methylation states, based on data from N patients for one DNA sample (i.e. $R = 1$). As each CpG site can belong to any of the $M = 3$ methylation states (hypomethylated, hemimethylated or hypermethylated), the default value of $K = M = 3$. Under the K.. model the shape parameters of each cluster are constrained to be equal for each patient. The returned object from this function can be passed as an input parameter to the `threshold` function available in this package to calculate the thresholds between the methylation states.

Value

A list containing:

- cluster_size - The total number of CpG sites in each of the K clusters.
- llk - A vector containing the log-likelihood value at each step of the EM algorithm.
- alpha - The first shape parameter for the beta mixture model.
- delta - The second shape parameter for the beta mixture model.
- tau - The proportion of CpG sites in each cluster.
- z - A matrix of dimension $C \times K$ containing the posterior probability of each CpG site belonging to each of the K clusters.
- classification - The classification corresponding to z, i.e. map(z).
- uncertainty - The uncertainty of each CpG site's clustering.

See Also

`beta_kn`

`betaclust`

`threshold`

Examples

```
## Not run:
data(pca.methylation.data)
my.seed <- 190
M <- 3
data_output <- beta_k(pca.methylation.data[,2:5], M, seed = my.seed)
thresholds <- threshold(data_output, pca.methylation.data[,2:5], "K..")

## End(Not run)
```

---

beta_kn *Fit the KN. model*

Description

Fit the KN. model from the `betaclust` family of beta mixture models for DNA methylation data. The KN. model analyses a single DNA sample and identifies the thresholds between the different methylation states.

Usage

```
beta_kn(data, M = 3, seed = NULL)
```

Arguments

data	A dataframe of dimension $C \times N$ containing methylation values for C CpG sites from $R = 1$ samples collected from N patients. Samples are grouped together in the dataframe such that the columns are ordered as Sample1_Patient1, Sample1_Patient2, etc.
M	Number of methylation states to be identified in a DNA sample.
seed	Seed to allow for reproducibility (default = NULL).

Details

The KN. model clusters each of the C CpG sites into one of K methylation states, based on data from N patients for one DNA sample (i.e. $R = 1$). As each CpG site can belong to any of the $M = 3$ methylation states (hypomethylated, hemimethylated or hypermethylated), the default value of $K = M = 3$. The KN. model differs from the K.. model as it is less parsimonious, allowing cluster and patient-specific shape parameters. The returned object can be passed as an input parameter to the threshold function available in this package to calculate the thresholds between the methylation states.

Value

A list containing:

- cluster_size - The total number of CpG sites in each of the K clusters.
- llk - A vector containing the log-likelihood value at each step of the EM algorithm.
- alpha - The first shape parameter for the beta mixture model.
- delta - The second shape parameter for the mixture model.
- tau - The proportion of CpG sites in each cluster.
- z - A matrix of dimension $C \times K$ containing the posterior probability of each CpG site belonging to each of the K clusters.
- classification - The classification corresponding to z, i.e. map(z).
- uncertainty - The uncertainty of each CpG site's clustering.

See Also

beta_k

betaclust

threshold

Examples

```
## Not run:
data(pca.methylation.data)
my.seed <- 190
M <- 3
data_output <- beta_kn(pca.methylation.data[,2:5], M, seed = my.seed)
thresholds <- threshold(data_output, pca.methylation.data[,2:5], "KN.")

## End(Not run)
```

beta_kr *Fit the K.R Model*

Description

A beta mixture model for identifying differentially methylated CpG sites between R DNA samples collected from N patients.

Usage

```
beta_kr(data, M = 3, N, R, seed = NULL)
```

Arguments

data	A dataframe of dimension $C \times NR$ containing methylation values for C CpG sites from R samples collected from N patients. Samples are grouped together in the dataframe such that the columns are ordered as Sample1_Patient1, Sample1_Patient2, Sample2_Patient1, Sample2_Patient2, etc.
M	Number of methylation states to be identified.
N	Number of patients in the study.
R	Number of samples collected from each patient for study.
seed	Seed to allow for reproducibility (default = NULL).

Details

The K.R model allows identification of the differentially methylated CpG sites between the R DNA samples collected from each of N patients. As each CpG site in a DNA sample can belong to one of M methylation states, there can be $K = M^R$ methylation state changes between R DNA samples. The shape parameters vary for each DNA sample but are constrained to be equal for each patient. An initial clustering using k-means is performed to identify K clusters. The resulting clustering solution is provided as starting values to the Expectation-Maximisation algorithm. A digamma approximation is used to obtain the maximised parameters in the M-step.

Value

A list containing:

- cluster_size - The total number of CpG sites in each of the K clusters.
- llk - A vector containing the log-likelihood value at each step of the EM algorithm.
- alpha - The first shape parameter for the beta mixture model.
- delta - The second shape parameter for the beta mixture model.
- tau - The proportion of CpG sites in each cluster.
- z - A matrix of dimension $C \times K$ containing the posterior probability of each CpG site belonging to each of the K clusters.
- classification - The classification corresponding to z, i.e. map(z).
- uncertainty - The uncertainty of each CpG site's clustering.

See Also

betaclust

Examples

```
## Not run:
data(pca.methylation.data)
my.seed <- 190
M <- 3
N <- 4
R <- 2
data_output = beta_kr(pca.methylation.data[,2:5], M, N, R, seed = my.seed)

## End(Not run)
```

---

ecdf.betaclust *The empirical cumulative distribution function plot*

Description

An empirical cumulative distribution function (ECDF) plot for a betaclust object.

Usage

```
ecdf.betaclust(x, R = 2, sample_name = NULL, title = NULL)
```

Arguments

x	A dataframe containing methylation values of identified differentially methylated regions related to a gene. Samples are grouped together in the dataframe such that the columns are ordered as Sample1_Patient1, Sample1_Patient2, Sample2_Patient1, Sample2_Patient2, etc.
R	Number of tissue samples from which DNA methylation data are collected (default R = 2).
sample_name	The order in which the samples are grouped in the dataframe. If no value is specified then default values of sample names, e.g. Sample 1, Sample 2, etc are used (default = NULL).
title	The title that the user wants to display on the graph. The default is "NULL".

Details

This function plots the ECDF of the differentially methylated CpG sites identified using the K.R model for all patient samples. The plot visualises the methylation state changes between the different DNA samples for each patient.

Value

The ECDF plot for the selected CpG sites for all patients and their DNA samples.

See Also

betaclust

beta_kr

em_aic *Akaike Information Criterion*

Description

Compute the AIC value for the optimal model.

Usage

```
em_aic(llk, C, M, N, R, model_name = "K..")
```

Arguments

`llk`	Log-likelihood value.
`C`	Number of CpG sites.
`M`	Number of methylation states identified in a DNA sample.
`N`	Number of patients.
`R`	Number of DNA samples collected from each patient.
`model_name`	Fitted mixture model. Options are "K..", "KN." and/or "K.R" (default = "K..").

Details

Computes the AIC for a specified model given the log-likelihood, the dimension of the data, and the model names.

Value

The AIC value for the selected model.

See Also

`em_bic`

`em_icl`

em_bic *Bayesian Information Criterion*

Description

Compute the BIC value for the optimal model.

Usage

```
em_bic(llk, C, M, N, R, model_name = "K..")
```

Arguments

`llk`	Log-likelihood value.
`C`	Number of CpG sites.
`M`	Number of methylation states identified in a DNA sample.
`N`	Number of patients.
`R`	Number of DNA samples collected from each patient.
`model_name`	Fitted mixture model. Options are "K..", "KN." and/or "K.R" (default = "K..").

Details

Computes the BIC for a specified model given the log-likelihood, the dimension of the data, and the model names.

Value

The BIC value for the selected model.

See Also

`em_aic`

`em_icl`

`em_icl` *Integrated Complete-data Likelihood (ICL) Criterion*

Description

Compute the ICL value for the optimal model.

Usage

```
em_icl(llk, C, M, N, R, model_name = "K..", z)
```

Arguments

`llk`	Log-likelihood value.
`C`	Number of CpG sites.
`M`	Number of methylation states identified in a DNA sample.
`N`	Number of patients.
`R`	Number of DNA samples collected from each patient.
`model_name`	Fitted mixture model. Options are "K..", "KN." and/or "K.R" (default = "K..").
`z`	A matrix of posterior probabilities of cluster membership, stored as z in the object from `beta_k`, `beta_kn` and `beta_kr` functions.

Details

Computes the ICL for a specified model given the log-likelihood, the dimension of the data, and the model names. This criterion penalises the BIC by including an entropy term favouring well separated clusters.

Value

The ICL value for the selected model.

See Also

em_aic

em_bic

---

legacy.data *MethylationEPIC manifest data.*

Description

A dataset containing a subset of the manifest data from the Illumina MethylationEPIC beadchip array. A subset of the complete dataset has been uploaded in the package for testing purpose. The complete dataset is available on GitHub.

Usage

```
data(legacy.data)
```

Format

A data frame with 10,081 rows and 6 columns.

- IlmnID: The unique identifier from the Illumina CG database, i.e. the probe ID.
- Genome_Build: The genome build referenced by the Infinium MethylationEPIC manifest.
- CHR: The chromosome containing the CpG (Genome_Build = 37).
- MAPINFO: The chromosomal coordinates of the CpG sites.
- UCSC_RefGene_Name: The target gene name(s), from the UCSC database. Note: multiple listings of the same gene name indicate splice variants.
- UCSC_CpG_Islands_Name: The chromosomal coordinates of the CpG Island from UCSC.

See Also

pca.methylation.data

`pca.methylation.data` *DNA methylation data from patients with prostate cancer*

Description

A dataset containing pre-processed beta methylation values from $R = 2$ samples, collected from $N = 4$ patients with prostate cancer.

Usage

```
data(pca.methylation.data)
```

Format

A data frame with 10,068 rows and 9 columns. The data contain no missing values.

- IlmnID: The unique identifier from the Illumina CG database, i.e. the probe ID.
- Benign_Patient_1: Methylation values from benign prostate tissue from patient 1.
- Benign_Patient_2: Methylation values from benign prostate tissue from patient 2.
- Benign_Patient_3: Methylation values from benign prostate tissue from patient 3.
- Benign_Patient_4: Methylation values from benign prostate tissue from patient 4.
- Tumour_Patient_1: Methylation values from tumor prostate tissue from patient 1.
- Tumour_Patient_2: Methylation values from tumor prostate tissue from patient 2.
- Tumour_Patient_3: Methylation values from tumor prostate tissue from patient 3.
- Tumour_Patient_4: Methylation values from tumor prostate tissue from patient 4.

Details

The raw methylation array data was first quality controlled and preprocessed using the RnBeads package. The array data were then normalized and and probes located outside of CpG sites and on the sex chromosome were filtered out. The CpG sites with missing values were removed from the resulting dataset. A subset of the complete dataset has been uploaded in the package for testing purposes. The complete dataset is available on GitHub.

References

Mueller F, Scherer M, Assenov Y, Lutsik P, Walter J, Lengauer T, Bock C (2019). "RnBeads 2.0: comprehensive analysis of DNA methylation data." Genome Biology, 20(55). doi: 10.1186/s13059-019-1664-9, https://rnbeads.org.

Assenov Y, Mueller F, Lutsik P, Walter J, Lengauer T, Bock C (2014). "Comprehensive Analysis of DNA Methylation Data with RnBeads." Nature Methods, 11(11), 1138–1140. doi: 10.1038/nmeth.3115, https://rnbeads.org.

See Also

`legacy.data`

plot.betaclust *Plots for visualizing the betaclust class object*

Description

Visualise a betaclust clustering solution by plotting the fitted and kernel density estimates, the uncertainty and the information criterion.

Usage

```
## S3 method for class 'betaclust'
plot(
  x,
  what = "fitted density",
  plot_type = "ggplot",
  data = NULL,
  sample_name = NULL,
  title = NULL,
  patient_number = 1,
  threshold = FALSE,
  scale_param = "free_y",
  ...
)
```

Arguments

x	A betaclust object.
what	The different plots that can be obtained are either "fitted density","kernel density","uncertainty" or "information criterion" (default = "fitted density").
plot_type	The plot type to be displayed are either "ggplot" or "plotly" (default = "ggplot").
data	A dataframe of dimension $C \times NR$ containing methylation values for C CpG sites from R samples collected from N patients which was passed as an argument to the betaclust function. The data is not required as an input when generating "uncertainty" or "information criterion" plots and the default has been set as "NULL". The data needs to be passed as an argument to this function when generating either "fitted density" or "kernel density" plots.
sample_name	The names of DNA samples in the dataset analysed by the K.R model. If no value is passed then default values of sample names, e.g. Sample 1, Sample 2, etc are used as legend text (default = NULL).
title	The title that the user wants to display. If no title is to be displayed the default is "NULL".
patient_number	The column number representing the patient in the patient-wise ordered dataset selected for visualizing the clustering solution of the K.. or KN. model (default = 1).
threshold	The "TRUE" option displays the threshold points in the graph for the K.. and the KN. model (default = "FALSE").

scale_param	The position scales can be fixed or allowed to vary between different panels generated for the density estimate plots for visualizing the K.R clustering solution. Options are "fixed", "free_y","free_x" or "free" (default = "free_y"). The option "fixed" results in the x and y scales being fixed across all panels, "free" varies the x and y scales across the panels, "free_x" fixes the y scale and lets the x scale vary across all panels and "free_y" fixes the x scale and lets the y scale vary across all panels.
...	Other graphics parameters.

Details

The fitted density estimates can be visualized under the optimal clustering solution by specifying what = "fitted density" and kernel density estimates under the optimal clustering solution by specifying what = "kernel density".

The threshold inferred can be visualized by specifying threshold = TRUE. The KN. model calculates different pairs of threshold points for each patient as the shape parameters are allowed to vary for each patient. So the patient for whom the threshold needs to be displayed can be specified by inputting the column number representing the patient in the patient-wise ordered dataset in the parameter patient_number.

Interactive plots can also be produced using plot_type = "plotly". The uncertainty in the clustering solution can be plotted using what = "uncertainty". The information criterion values for all fitted models can be plotted using what = "information criterion".

See Also

betaclust

Examples

```
## Not run:
M <- 3
N <- 4
R <- 2
data_output <- betaclust(pca.methylation.data[,2:9], M, N, R,
            model_names = c("K..","KN.","K.R"), model_selection = "BIC", seed = my.seed)
plot(data_output, what = "fitted density", plot_type = "ggplot", data=pca.methylation.data[,2:9])
plot(data_output, what = "kernel density", plot_type = "ggplot", data=pca.methylation.data[,2:9])
plot(data_output, what = "kernel density", plot_type = "plotly", data=pca.methylation.data[,2:9])
plot(data_output, what = "uncertainty", plot_type = "ggplot")
plot(data_output, what = "information criterion", plot_type = "ggplot")
## End(Not run)
```

---

summary.betaclust *Summarizing the beta mixture model fits*

Description

Summary method for a betaclust object containing the results under the optimal model selected.

Usage

```
## S3 method for class 'betaclust'
summary(object, ...)
```

Arguments

object	A betaclust object.
...	Further arguments passed to or from other methods.

Value

An object of class summary.betaclust which contains the following list:

- C - The number of CpG sites analysed using the beta mixture models.
- N - The number of patients analysed using the beta mixture models.
- R - The number of samples analysed using the beta mixture models.
- K - The number of methylation states in R DNA samples.
- modelName - The optimal model selected.
- loglik - The log-likelihood value for the selected optimal model.
- information_criterion - The information criterion used to select the optimal model.
- ic_output - This stores the information criterion value calculated for each model.
- classification - The total number of CpG sites in each cluster.
- prop_data - The proportion of CpG sites in each cluster.

See Also

betaclust

Examples

```
## Not run:
M <- 3
N <- 4
R <- 2
data_output <- betaclust(pca.methylation.data[,2:9], M, N, R,
            model_names=c("K..","KN.","K.R"), model_selection="BIC", seed=my.seed)
summary(data_output)
## End(Not run)
```

---

threshold *Thresholds under the K.. and the KN. models*

Description

An objective method to calculate the threshold points for the clustering solution of the K.. and the KN. models.

Usage

```
threshold(object, data, model_name)
```

Arguments

object	A beta_k or beta_kn object.
data	A dataframe of dimension $C \times NR$ containing methylation values for C CpG sites from R samples collected from N patients which was passed as an argument to the betaclust function.
model_name	The name of the model for which the thresholds need to be calculated.

Details

As the K.. model constrains the shape parameters to be equal for all patients, a single pair of threshold points are calculated for all patients. The KN. model allows patient-specific shape parameters which results in a pair of threshold points for each patient based on the shape parameters for that patient. The first threshold point means that any CpG site with beta value less than this value is likely to be hypomethylated. The second threshold point means that any CpG site with beta value greater than this is highly likely to be hypermethylated. A CpG site with beta value lying between the two threshold points is likely to be hemimethylated.

Value

thresholds - The threshold points calculated for the selected model. A vector containing two threshold points are returned for the K.. model whereas a matrix containing two threshold points for each patient is returned for the KN. model.

See Also

beta_k

beta_kn

betaclust

Index

∗ **datasets**
legacy.data, 11
pca.methylation.data, 12

beta_k, *4*, 4, *6*, *10*, *16*
beta_kn, *4, 5*, 5, *10*, *16*
beta_kr, *4*, 7, *8*, *10*
betaclust, 2, *3–6*, *8*, *13–16*

ecdf.betaclust, 8
em_aic, 9, *10, 11*
em_bic, *9*, 9, *11*
em_icl, *9, 10*, 10

legacy.data, 11, *12*

pca.methylation.data, *4*, *11*, 12
plot.betaclust, *4*, 13

summary.betaclust, *4*, 14, *15*

threshold, *4–6*, 15