

Package ‘methcon5’

May 8, 2026

Title Identify and Rank CpG DNA Methylation Conservation Along the Human Genome

Version 0.1.0

Description Identify and rank CpG DNA methylation conservation along the human genome. Specifically it includes bootstrapping methods to provide ranking which should adjust for the differences in length as without it short regions tend to get higher conservation scores.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

Depends R (>= 2.10)

Imports magrittr, dplyr, purrr, rlang

Suggests ggplot2, testthat (>= 2.1.0), covr

URL <https://github.com/EmilHvitfeldt/methcon5>

BugReports <https://github.com/EmilHvitfeldt/methcon5/issues>

NeedsCompilation no

Author Emil Hvitfeldt [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-0679-1945>>)

Maintainer Emil Hvitfeldt <emilhhvitfeldt@gmail.com>

Repository CRAN

Date/Publication 2019-12-20 13:50:02 UTC

Contents

fake_methylation	2
meth_aggregate	2
meth_bootstrap	3
Index	5

fake_methylation	<i>Simple simulated methylation dataset</i>
------------------	---

Description

Simple simulated methylation dataset

Usage

```
fake_methylation
```

Format

A data frame with 2771 rows and 3 variables: gene, cons_level and meth.

Details

This dataset is for example use only. It contains 500 genes identified by gene each with one of 3 types of conservation levels "low", "medium" and "high". The methylation values are independent randomly distributed within each gene. Thus no spacial correlation is assumed.

meth_aggregate	<i>Calculate region wise summary statistics</i>
----------------	---

Description

Will take a data.frame and apply a function ('fun') to 'value' within the groups defined by the 'id' column.

Usage

```
meth_aggregate(data, id, value, fun = mean, ...)
```

Arguments

data	a data.frame.
id	variable name, to be aggregated around.
value	variable name, contains the value to take mean over. Must be a single column.
fun	function, summary statistic function to be calculated. Defaults to 'mean'.
...	Additional arguments for the function given to the argument fun.

Details

Please note the ordering of the data will matter depending on the choice of aggregation function.

Value

A methcon object. Contains the aggregated data along with original data.frame and variable selections.

Examples

```
meth_aggregate(fake_methylation, id = gene, value = meth, fun = mean)

meth_aggregate(fake_methylation, id = gene, value = meth, fun = var)

# custom functions can be used as well
mean_diff <- function(x) {
  mean(diff(x))
}

meth_aggregate(fake_methylation, id = gene, value = meth, fun = mean_diff)
```

meth_bootstrap	<i>Bootstrapped randomly samples values</i>
----------------	---

Description

"perm_v1" (the default method) will sample the variables the rows independently. "perm_v2" will sample regions of same size while allowing overlap between different regions. "perm_v3" will sample regions under the constraint that all sampled regions are contained in the region they are sampled in.

Usage

```
meth_bootstrap(data, reps, method = c("perm_v1", "perm_v2", "perm_v3"))
```

Arguments

data	a methcon data.frame output from 'meth_bootstrap'.
reps	Number of reps, defaults to 1000.
method	Character, determining which method to use. See details for information about methods. Defaults to "perm_v1".

Details

Note that you can apply 'meth_bootstrap' multiple times to get values for different methods.

Value

A methcon object. Contains the aggregated data along with original data.frame and variable selections and bootstrapped values.

Examples

```
# Note that you likely want to do more than 10 repetitions.
# rep = 10 was chosen to have the examples run fast.

fake_methylation %>%
  meth_aggregate(id = gene, value = meth, fun = mean) %>%
  meth_bootstrap(10)

fake_methylation %>%
  meth_aggregate(id = gene, value = meth, fun = mean) %>%
  meth_bootstrap(10, method = "perm_v2")

# Get multiple bootstraps
fake_methylation %>%
  meth_aggregate(id = gene, value = meth, fun = mean) %>%
  meth_bootstrap(10, method = "perm_v1") %>%
  meth_bootstrap(10, method = "perm_v2") %>%
  meth_bootstrap(10, method = "perm_v3")
```

Index

* **datasets**

fake_methylation, [2](#)

fake_methylation, [2](#)

meth_aggregate, [2](#)

meth_bootstrap, [3](#)