Package 'methyvim'

June 7, 2020

Title Targeted, Robust, and Model-free Differential Methylation
Analysis

Version 1.10.0

Description This package provides facilities for differential methylation
analysis based on variable importance measures (VIMs), a class of statistical
target parameters that arise in causal inference. The estimation and inference
procedures provided are nonparametric, relying on ensemble machine learning to
flexibly assess functional relationships among covariates and the outcome of
interest. These tools can be applied to differential methylation at the level
of CpG sites, to obtain valid statistical inference even after corrections for
multiple hypothesis testing.

Depends R (>= 3.4.0)

License file LICENSE

URL https://github.com/nhejazi/methyvim

BugReports https://github.com/nhejazi/methyvim/issues

Encoding UTF-8

Imports stats, cluster, methods, ggplot2, ggsci, gridExtra, superheat,
dplyr, gtools, tmle (>= 1.4.0.1), future, doFuture, S4Vectors,
BiocGenerics, BiocParallel, SummarizedExperiment, GenomeInfoDb,
bumphunter, IRanges, limma, minfi

Suggests testthat, knitr, rmarkdown, BiocStyle, SuperLearner, earth,
nnet, gam, arm, snow, parallel, BatchJobs, minfiData,
methyvimData

VignetteBuilder knitr

RoxygenNote 7.0.2

biocViews Clustering, DNAMethylation, DifferentialMethylation,
MethylationArray, MethylSeq

git_url https://git.bioconductor.org/packages/methyvim

git_branch RELEASE_3_11

git_last_commit 9bcf5ef

git_last_commit_date 2020-04-27

Date/Publication 2020-06-06

Author Nima Hejazi [aut, cre, cph] (<https://orcid.org/0000-0002-7127-2789>),
    Rachael Phillips [ctb] (<https://orcid.org/0000-0002-8474-591X>),
    Mark van der Laan [aut, ths] (<https://orcid.org/0000-0003-1432-5511>),
    Alan Hubbard [ctb, ths] (<https://orcid.org/0000-0002-3769-0127>)

Maintainer Nima Hejazi <nh@nimahejazi.org>

R topics documented:

---

fdr_msa   *FDR-MSA correction*

Description

Modified FDR Controlling Procedure for Multi-Stage Analyses (MJ van der Laan and C Tuglus, 2009, <doi:10.2202/1544-6115.1397>)

Usage

```
fdr_msa(pvals, total_obs)
```

Arguments

pvals	Numeric vector containing the p-values that result from any chosen statistical hypothesis testing procedure.
total_obs	Numeric indicating the total number of observations that would have been available for testing prior to the selection procedure employed in the multi-stage analysis performed.

Value

A `numeric` vector of corrected p-values, controlling the False Discovery Rate, using the method of Tuglus and van der Laan.

Examples

```
g <- 1e4
n <- 1e2
p <- abs(rnorm(n, mean = 1e-8, sd = 1e-2))
# treating the vector p as one of p-values, FDR-MSA may be applied
fdr_p <- fdr_msa(pvals = p, total_obs = g)
```

---

methyheat   *Heatmap for methytmle objects*

Description

Heatmap for methytmle objects

Usage

```
methyheat(x, ..., n_sites = 25, type = "raw")
```

Arguments

x   Object of class `methytmle` as produced by an appropriate call to `methyvim`.

...   Additional arguments passed to `superheat`. Consult the documentation of the
             superheat package for a list of options.

n_sites   Numeric indicating the number of CpG sites to be shown in the plot. If the
             number of sites analyzed is greater than this cutoff, sites to be displayed are
             chosen by ranking sites based on their raw (marginal) p-values.

type   Whether to plot the original data (M-values or Beta-values) for the set of top
             CpG sites or to plot the measurements after applying a transformation into in-
             fluence curve space (with respect to the target parameter of interest). The latter
             uses the fact that the parameters have asymptotically linear representations to
             obtain a rotation of the raw data into an alternative space; moreover, in this
             setting, the heatmap reduces to visualizing a supervised clustering procedure.

Value

Nothing. This function is called for its side-effect of outputting a heatmap to the graphics device.
The heatmap is constructed using the `superheat` package.

Examples

```
suppressMessages(library(SummarizedExperiment))
library(methyvimData)
data(grsExample)
var_int <- as.numeric(colData(grsExample)[, 1])
# TMLE procedure for the ATE parameter over M-values with Limma filtering
methyvim_out_ate <- suppressWarnings(
  methyvim(
    data_grs = grsExample, sites_comp = 25, var_int = var_int,
    vim = "ate", type = "Mval", filter = "limma", filter_cutoff = 0.1,
    parallel = FALSE, tmle_type = "glm"
  )
)
methyheat(methyvim_out_ate, type = "raw")
```

---

methyvim   *Differential Methylation Statistics with Variable Importance Measures*

Description

Computes the Targeted Minimum Loss Estimate of a specified statistical target parameter, formally defined within models from causal inference. The variable importance measures currently supported are the Average Treatment Effect (ATE) and a Nonparametric Variable Importance Measure (NPVI, formally defined by Chambaz, Neuvial, and van der Laan <doi:10.1214/12-EJS703>).

Usage

```
methyvim(
  data_grs,
  var_int,
  vim = c("ate", "rr", "npvi"),
  type = c("Beta", "Mval"),
  filter = c("limma"),
  filter_cutoff = 0.05,
  window_bp = 1000,
  corr_max = 0.75,
  obs_per_covar = 20,
  sites_comp = NULL,
  parallel = TRUE,
  future_param = NULL,
  bppar_type = NULL,
  return_ic = FALSE,
  shrink_ic = FALSE,
  tmle_type = c("glm", "sl"),
 tmle_args = list(g_lib = c("SL.mean", "SL.glm", "SL.bayesglm", "SL.gam"), Q_lib =
  c("SL.mean", "SL.glm", "SL.gam", "SL.earth"), cv_folds = 5, npvi_cutoff = 0.25,
    npvi_descr = NULL),
  tmle_backend = c("tmle", "drtmle", "tmle.npvi")
)
```

Arguments

data_grs   An object of class `GenomicRatioSet`, containing standard data structures for
                DNA Methylation experiments. Consult the documentation of **minfi** to construct
                such objects.

var_int   A numeric vector containing subject-level measurements of the variable of in-
                terest. The length of this vector must match the number of subjects exactly. If
                argument vim is set to "ate" or "rr", then the variable of interest is treated as an
                exposure, and the variable must be binary in such cases. If setting vim to target
                parameters assessing continuous treatment effects, then the variable need not be
                binary of course.

vim   Character indicating the variable importance measure to be used in the estima-
                tion procedure. Currently supported options are the ATE for discretized expo-
                sures and NPVI for continuous exposures. ATE and RR are the appropriate
                choices when the underlying scientific question is of the effect of an exposure

on methylation, while NPVI (and other continuous treatment parameters) ought to be used when the effect of methylation on an outcome is sought.

type   Character indicating the particular measure of DNA methylation to be used as the observed data in the estimation procedure, either Beta values or M-values. The data are accessed via [getBeta]() or [getM]().

filter   Character indicating the model to be implemented when screening the data_grs object for CpG sites. The only currently supported option is "limma".

filter_cutoff   Numeric indicating the p-value cutoff that defines which sites pass through the filter.

window_bp   Numeric indicating the maximum genomic distance (in base pairs) between two sites for them to be considered neighboring sites.

corr_max   Numeric indicating the maximum correlation that a neighboring site can have with the target site.

obs_per_covar   Numeric indicating the number of observations needed for for covariate included in W for downstream analysis. This ensures the data is sufficient to control for the covariates.

sites_comp   A numeric indicating the maximum number of sites for which a variable importance measure is to be estimated post-screening. This is not typically useful in scientific settings, but may be useful when a large number of CpG sites pass the initial screening phase.

parallel   Logical indicating whether parallelization ought to be used. See the documentation of set_parallel for more information, as this argument is passed directly to that internal function.

future_param   Character indicating the type of parallelization to be used from the list available via the **future** package. See the documentation for set_parallel for more information, as this argument is passed directly to that internal function.

bppar_type   Character specifying the type of backend to be used for parallelization via **Bioc-Parallel**. See the documentation for set_parallel for more information, as this argument is passed directly to that internal function.

return_ic   Logical indicating whether an influence curve estimate should be returned for each site that passed through the filter.

shrink_ic   Logical indicating whether limma should be applied to reduce the variance in the ic based estimates in return_ic.

tmle_type   Character indicating the general class of regression models to be used in fitting the propensity score and outcome regressions. This is generally a shorthand and is overridden by tmle_args if that argument is changed from its default values.

tmle_args   List giving several key arguments to be passed to one of [tmle]() or [tmle.npvi](), depending on the particular variable importance measure specified. This overrides tmle_type, which itself provides sensible defaults. Consider changing this away from default settings only if you have sufficient experience with software and the underlying theory for Targeted Learning. For more information, consider consulting the documentation of the **tmle** and **tmle.npvi** packages.

tmle_backend   A character indicating the package to be used in the estimation procedure. The user should only set this parameter if they have sufficient familiarity with the backend packages used for estimation. Current choices include **tmle**, **drtmle**, and **tmle.npvi**.

Value

An object of class `methytmle`, with all unique slots filled in, in particular, including indices of CpG sites that pass screening, cluster of neighboring CpG sites, and a matrix of the results of the estimation procedure performed for the given variable importance measure. Optionally, estimates of the propensity score and outcome regressions, as well as the original data rotated into influence curve space may be returned, if so requested.

Examples

```
library(methyvimData)
suppressMessages(library(SummarizedExperiment))
data(grsExample)
var_int <- colData(grsExample)[, 1]
# TMLE procedure for the ATE parameter over M-values with Limma filtering
methyvim_out_ate <- suppressWarnings(
  methyvim(
    data_grs = grsExample, sites_comp = 1, var_int = var_int,
    vim = "ate", type = "Mval", filter = "limma", filter_cutoff = 0.05,
    parallel = FALSE, tmle_type = "sl"
  )
)
```

| methyvolc | *Volcano plot for methytmle objects* |

Description

Volcano plot for methytmle objects

Usage

```
methyvolc(x, param_bound = 2, pval_bound = 0.2)
```

Arguments

x	Object of class `methytmle` as produced by an appropriate call to `methyvim`.
param_bound	Numeric for a threshold indicating the magnitude of the size of the effect considered to be interesting. This is used to assign groupings and colors to individual CpG sites.
pval_bound	Numeric for a threshold indicating the magnitude of p-values deemed to be interesting. This is used to assign groupings and colors to individual CpG sites.

Value

Object of class `ggplot` containing a volcano plot of the estimated effect size on the x-axis and the -log10(p-value) on the y-axis. The volcano plot is used to detect possibly false positive cases, where a test statistic is significant due to low variance.

Examples

```
suppressMessages(library(SummarizedExperiment))
library(methyvimData)
data(grsExample)
var_int <- as.numeric(colData(grsExample)[, 1])
# TMLE procedure for the ATE parameter over M-values with Limma filtering
methyvim_out_ate <- suppressWarnings(
  methyvim(
    data_grs = grsExample, sites_comp = 25, var_int = var_int,
    vim = "ate", type = "Mval", filter = "limma", filter_cutoff = 0.1,
    parallel = FALSE, tmle_type = "glm"
  )
)
methyvolc(methyvim_out_ate)
```

---

plot.methytmle   *Plot p-values of methytmle objects*

Description

Plot p-values of methytmle objects

Usage

```
## S3 method for class 'methytmle'
plot(x, ..., type = "both")
```

Arguments

x	Object of class `methytmle` as produced by an appropriate call to `methyvim`.
...	Additional arguments passed `plot` as necessary.
type	The type of plot to build: one of side-by-side histograms (type "both") comparing raw p-values to FDR-adjusted p-values (using the FDR-MSA correction) or either of these two histogram separately. Set this argument to "raw_pvals" for a histogram of the raw p-values, and to "fdr_pvals" for a histogram of the FDR-corrected p-values.

Value

Object of class `ggplot` containing a histogram or side-by-side histograms of the raw (marginal) and corrected p-values, with the latter computed automatically using the method of Tuglus and van der Laan.

Examples

```
suppressMessages(library(SummarizedExperiment))
library(methyvimData)
data(grsExample)
var_int <- as.numeric(colData(grsExample)[, 1])
# TMLE procedure for the ATE parameter over M-values with Limma filtering
methyvim_out_ate <- suppressWarnings(
  methyvim(
```

```
      data_grs = grsExample, sites_comp = 25, var_int = var_int,
      vim = "ate", type = "Mval", filter = "limma", filter_cutoff = 0.1,
      parallel = FALSE, tmle_type = "glm"
    )
  )
  plot(methyvim_out_ate)
```

---

wrap_in_try   *Wrap a Function in a Try Statement*

Description

Function factory that generates versions of functions wrapped in `try`. Originally found in and borrowed from package `origami`.

Usage

```
wrap_in_try(fun, ...)
```

Arguments

fun	A `function` to be wrapped in a `try` statement.
...	Additional arguments passed to the previous argument `fun`.

Index