Package 'scMET'

September 5, 2024

Type Package

Title Bayesian modelling of cell-to-cell DNA methylation heterogeneity

Version 1.6.0

Description High-throughput single-cell measurements of DNA methylomes can quantify methylation heterogeneity and uncover its role in gene regulation. However, technical limitations and sparse coverage can preclude this task. scMET is a hierarchical Bayesian model which overcomes sparsity, sharing information across cells and genomic features to robustly quantify genuine biological heterogeneity. scMET can identify highly variable features that drive epigenetic heterogeneity, and perform differential methylation and variability analyses. We illustrate how scMET facilitates the characterization of epigenetically distinct cell populations and how it enables the formulation of novel hypotheses on the epigenetic regulation of gene expression.

License GPL-3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.0

Biarch true

BugReports https://github.com/andreaskapou/scMET/issues

Depends R (>= 4.2.0)

Imports methods, Rcpp (>= 1.0.0), RcppParallel (>= 5.0.1), rstan (>= 2.21.3), rstantools (>= 2.1.0), VGAM, data.table, MASS, logitnorm, ggplot2, matrixStats, assertthat, viridis, coda, BiocStyle, cowplot, stats, SummarizedExperiment, SingleCellExperiment, Matrix, dplyr, S4Vectors

Suggests testthat, knitr, rmarkdown

LinkingTo BH (>= 1.66.0), Rcpp (>= 1.0.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.21.3), StanHeaders (>= 2.21.0.7)

SystemRequirements GNU make

biocViews ImmunoOncology, DNAMethylation, DifferentialMethylation, DifferentialExpression, GeneExpression, GeneRegulation, Epigenetics, Genetics, Clustering, FeatureExtraction, Regression, Bayesian, Sequencing, Coverage, SingleCell

VignetteBuilder knitr

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

git_branch RELEASE_3_19

git_last_commit f3a4cd8

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-09-04

Author Andreas C. Kapourani [aut, cre] (<https://orcid.org/0000-0003-2303-1953>), John Riddell [ctb]

Maintainer Andreas C. Kapourani <kapouranis.andreas@gmail.com>

Contents

scMET-package *scMET: Bayesian modelling of DNA methylation at single-cell resolution.*

Description

Package for analysing single-cell DNA methylation datasets. scMET performs feature selection, by identifying highly variable features, and also differential testing, based on mean but also more importantly on variability between two groups of cells.

Value

scMET main package documentation.

Author(s)

C.A.Kapourani <kapouranis.andreas@gmail.com>

References

Stan Development Team (2020). RStan: the R interface to Stan. R package version 2.19.3. https://mc-stan.org

See Also

scmet, scmet_differential, scmet_hvf

bb_mle *Beta binomial maximum likelihood estimation (BB MLE)*

Description

Maximum Likelihood Estimate (MLE) of Beta-Binomial (BB) model. Some details about this model can be found on the following tutorial https://rpubs.com/cakapourani/beta-binomial

Usage

```
bb_mle(x, w = NULL, n_starts = 10, lower_thresh = 0.001)
```

Arguments

x	An n x 2 data.table or matrix, where 1st column keeps total number of trials and 2nd column number of successes, n is the total number of samples.
w	Vector with initial values of alpha and beta, if NULL the method of moments is used to initialize them.
n_starts	Total number of restarts when optimisation fails.
lower_thresh	Threshold when to stop optimisation.

Value

A list with the following elements:

- `gamma`: The overdispersion parameter. This is the most important parameter, since it tells us if and how much overdispersion we observe in the data that cannot be explained by the Binomial model.
- `mu`: The mean parameter, i.e. success probability of the beta binomial.
- `alpha`: Alpha parameter, when taking the different parametrisation of the BB.
- `beta`: Beta parameter, when taking the different parametrisation of the BB.
- `is_conv`: Logical, whether or not the optimisation converged.
- `lrt`: The likelihood ratio test statistic, for testing whether the Binomial or the Beta-Binomial fit better the data.
- `chi2_test`: The p-value from the Chi-squared test obtained from the LRT statistics.
- `Z_score`: The Z score statistic proposed by Tarone (1979). Seems more stable than LRT, in test whether we have overdispersion in our data.
- `z_test`: The p-value obtain from the Z-score statistic.
- `bb_ll`: Beta binomial log likelihood (used internally to compute the LRT statistic and the BIC)
- `BIC_bb`: The Bayes Information Criterion for beta binomial model
- `bin_ll`: Binomial log likelihood (used internally to compute the LRT statistic and the BIC.)
- `BIC_bin|`: The Bayes Information Criterion for binomial model

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf

Examples

```
# Extract data from a single Feature
x <- scmet_dt$Y[Feature == "Feature_1", c("total_reads", "met_reads")]
fit_mle <- bb_mle(x)
```

create_design_matrix *Create design matrix*

Description

Generic function for crating a radial basis function (RBF) design matrix for input vector X.

Usage

```
create_design_matrix(L, X, c = 1.2)
```

Arguments

L	Total number of basis functions, including the bias term.
X	Vector of covariates
c	Scaling parameter for variance of RBFs

Value

A design matrix object H.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf

Examples

```
# Extract
H <- create_design_matrix(L = 4, X = scmet_dt$X)
```

sce_to_scmet *Convert from SingleCellExperiment to scmet object*

Description

Helper function that converts SCE objects to scmet objects that can be used as input to the scmet function. The structure of the SCE object to store single cell methylation data is the following. We create two sparse assays, `met` storing methylated CpGs and `total` storing total number of CpGs. Rows correspond to features and columns to cells, similar to scRNA-seq convention.To distinguish between a feature (in a cell) having zero methylated CpGs vs not having CpG coverage at all (missing value), we check if the corresponding entry in `total` is zero as well. The `rownames` and `colnames` slots should store the feature and cell names, respectively. Covariates `X` that might explain variability in mean (methylation) should be stored in `metadata(rowData(sce)X`.

Usage

```
sce_to_scmet(sce)
```

Arguments

sce	SummarizedExperiment object

Value

A named list containing the matrix Y (methylation data in format required by the `scmet` function) and the covariates X.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

`scmet`, `scmet_differential`, `scmet_hvf_lvf`

Examples

```
# Extract
sce <- scmet_to_sce(Y = scmet_dt$Y, X = scmet_dt$X)

df <- sce_to_scmet(sce)
```

---

scmet *Perform inference with scMET*

Description

Compute posterior of scMET model. This is the main function which infers model parameters and corrects for the mean-overdispersion relationship. The most important parameters the user should focus are `X`, `L`, `user_mcmc` and `iter`. Advanced users may want to optimise the model by changing the prior parameters. For small datasets, we recommend using MCMC implementation of scMET since it is more stable.

Usage

```
scmet(
  Y,
  X = NULL,
  L = 4,
  use_mcmc = FALSE,
  use_eb = TRUE,
  iter = 5000,
```

```
  algorithm = "meanfield",
  output_samples = 2000,
  chains = 4,
  m_wmu = rep(0, NCOL(X)),
  s_wmu = 2,
  s_mu = 1.5,
  m_wgamma = rep(0, L),
  s_wgamma = 2,
  a_sgamma = 2,
  b_sgamma = 3,
  rbf_c = 1,
  init_using_eb = TRUE,
  tol_rel_obj = 1e-04,
  n_cores = 2,
  lambda = 4,
  seed = sample.int(.Machine$integer.max, 1),
  ...
)
```

Arguments

Y	Observed data (methylated reads and total reads) for each feature and cell, in a long format data.table. That is it should have 4 named columns: (Feature, Cell, total_reads, met_reads).
X	Covariates which might explain variability in mean (methylation). If X = NULL, then we do not perform any correction on the mean estimates. NOTE that if X is provided, rownames of X should be the unique feature names in Y. If the dimensions or all feature names do not match, an error will be thrown.
L	Total number of basis function to fit the mean-overdispersion trend. For L = 1, this reduces to a model that does not correct for the mean-overdispersion relationship.
use_mcmc	Logical, whether to use the MCMC implementation for posterior inference. If FALSE, we run the VB implementation (default). For small datasets, we recommend using MCMC implementation since it is more stable.
use_eb	Logical, whether to use 'Empirical Bayes' for parameter initialization. If TRUE (default), it will intialise the m_wmu and m_wgamma parameters below.
iter	Total number of iterations, either MCMC or VB algorithm. NOTE: The STAN implementation of VB relies on black-box variational inference and potentially with relatively small sample sizes sometimes tends to 'search' around the local/global minima. We've seen that with larger sample sizes (thousands of cells), it tends to converge much faster, e.g. around 2-3k iterations.
algorithm	Stan algorithm to be used by Stan. If MCMC: Possible values are: "NUTS", "HMC". If VB: Possible values are: "meanfield" and "fullrank".
output_samples	If VB algorithm, the number of posterior samples to draw and save.
chains	Total number of chains.
m_wmu	Prior mean of regression coefficients for covariates X.

s_wmu	Prior standard deviation of regression coefficients for covariates X.
s_mu	Prior standard deviation for mean parameter mu.
m_wgamma	Prior mean of regression coefficients of the basis functions.
s_wgamma	Prior standard deviation of regression coefficients of the basis functions.
a_sgamma	Gamma prior (shape) for standard deviation for dispersion parameter gamma.
b_sgamma	Gamma prior (rate) for standard deviation for dispersion parameter gamma.
rbf_c	Scale parameter for empirically computing the variance of the RBFs.
init_using_eb	Logical, initial values of parameters for STAN posterior inference. Preferably this should be set always to TRUE, to lower the chances of VB/MCMC initialisations being far away from posterior mass.
tol_rel_obj	If VB algorithm, the convergence tolerance on the relative norm of the objective.
n_cores	Total number of cores.
lambda	The penalty term to fit the RBF coefficients for the mean-overdispersion trend when initialising hyper-parameter with EB.
seed	The seed for random number generation.
...	Additional parameters passed to Stan fitting functions.

Value

An object of class scmet_mcmc or scmet_vb with the following elements:

- posterior: A list of matrices containing the samples from the posterior. Each matrix corresponds to a different parameter returned from scMET.
- Y: The observed data Y.
- feature_names: A vector of feature names.
- theta_priors: A list with all prior parameter values, for reproducibility purposes.
- opts: A list of all additional parameters when running scMET. For reproducibility purposes.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet_differential, scmet_hvf_lvf

Examples

```
# Fit scMET (in practice 'iter' should be much larger)
obj <- scmet(Y = scmet_dt$Y, X = scmet_dt$X, L = 4, iter = 300)
```

scmet_differential *Differential testing using scMET*

Description

Function for performing differential methylation testing to identify differentially methylted (DM) and differentially variable (DV) features across two groups of pre-specified cell populations.

Usage

```
scmet_differential(
  obj_A,
  obj_B,
  psi_m = log(1.5),
  psi_e = log(1.5),
  psi_g = log(1.5),
  evidence_thresh_m = 0.8,
  evidence_thresh_e = 0.8,
  evidence_thresh_g = 0.8,
  efdr_m = 0.05,
  efdr_e = 0.05,
  efdr_g = 0.05,
  group_label_A = "GroupA",
  group_label_B = "GroupB",
  features_selected = NULL,
  filter_outlier_features = FALSE,
  outlier_m = 0.05,
  outlier_g = 0.05
)
```

Arguments

obj_A	The scMET posterior object for group A.
obj_B	The scMET posterior object for group B.
psi_m	Minimum log odds ratio tolerance threshold for detecting changes in overall methylation (positive real number). Default value: psi_m = log(1.5) (i.e. 50% increase).
psi_e	Minimum log odds ratio tolerance threshold for detecting changes in residual over-dispersion (positive real number).
psi_g	Minimum log odds ratio tolerance threshold for detecting changes in biological over-dispersion (positive real number).
evidence_thresh_m	Optional parameter. Posterior evidence probability threshold parameter alpha_{M} for detecting changes in overall methylation (between 0.6 and 1). If efdr_m = NULL, then threshold will be set to evidence_thresh_m. If a value for EFDR_M is

provided, the posterior probability threshold is chosen to achieve an EFDR equal to efdr_m and evidence_thresh_m defines a minimum probability threshold for this calibration (this avoids low values of evidence_thresh_m to be chosen by the EFDR calibration. Default value evidence_thresh_m = 0.8.

evidence_thresh_e
: Optional parameter. Posterior evidence probability threshold parameter alpha_{G} for detecting changes in cell-to-cell residual over-dispersion. Same usage as above.

evidence_thresh_g
: Optional parameter. Posterior evidence probability threshold parameter alpha_{G} for detecting changes in cell-to-cell biological over-dispersion. Same usage as above.

efdr_m
: Target for expected false discovery rate related to the comparison of means. If efdr_m = NULL, no calibration is performed, and alpha_{M} is set to evidence_thresh_m. Default value: efdr_m = 0.05.

efdr_e
: Target for expected false discovery rate related to the comparison of residual over-dispersions If efdr_e = NULL, no calibration is performed, and 'alpha_E" is set to evidence_thresh_e. Default value: efdr_e = 0.05.

efdr_g
: Target for expected false discovery rate related to the comparison of biological over-dispersions If efdr_g = NULL, no calibration is performed, and alpha_{G} is set to evidence_thresh_g. Default value: efdr_g = 0.05.

group_label_A
: Label assigned to group A.

group_label_B
: Label assigned to group B.

features_selected
: User defined list of selected features to perform differential analysis. Should be the same length as the total number of features, with TRUE for features included in the differential analysis, and FALSE for those excluded from further analysis.

filter_outlier_features
: Logical, whether to filter features that have either mean methylation levels mu or overdispersion gamma across both groups near the range edges, i.e. taking values near 0 or 1. This mostly is an issue due to taking the logit transformation which effectively makes small changes in actual space (0, 1) to look really large in transformed space (-Inf, Inf). In general we expect this will not remove many interesting features with biological information.

outlier_m
: Value of average mean methylation across both groups so a feature is considered as outlier. I.e. if set to 0.05, then will remove features with mu < 0.05 or mu > 1 - 0.05. Only used if filter_outlier_features = TRUE.

outlier_g
: Value of average overdispersion gamma across groups so a feature is considered as outlier. Same as outlier_m parameter above.

Value

An scmet_differential object which is a list containing the following elements:

- diff_mu_summary: A data.frame containing differential mean methylation output information per feature (rows), including posterior median parameters for each group and mu_LOR

containing the log odds-ratio between the groups. The `mu_tail_prob` column contains the posterior tail probability of a feature being called as DM. The `mu_diff_test` column informs the outcomes of the test.

- `diff_epsilon_summary`: Same as above, but for differential variability based on residual overdispersion.
- `diff_gamma_summary`: The same as above but for DV analysis based on overdispersion.
- `diff_mu_thresh`: Information about optimal posterior evidence threshold search for mean methylation mu.
- `diff_epsilon_thresh`: Same as above but for residual overdispersion epsilon..
- `diff_gamma_thresh`: Same as above but for overdispersion gamma.
- `opts`: The parameters used for testing. For reproducibility purposes.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_hvf_lvf

Examples

```
## Not run:
# Fit scMET for each group
fit_A <- scmet(Y = scmet_diff_dt$scmet_dt_A$Y,
X = scmet_diff_dt$scmet_dt_A$X, L = 4, iter = 50, seed = 12)
fit_B <- scmet(Y = scmet_diff_dt$scmet_dt_B$Y,
X = scmet_diff_dt$scmet_dt_B$X, L = 4, iter = 50, seed = 12)

# Run differential test
diff_obj <- scmet_differential(obj_A = fit_A, obj_B = fit_B)

## End(Not run)
```

---

scmet_diff_dt *Synthetic methylation data from two groups of cells*

Description

Small synthetic data for quick analysis, mostly useful for showing the differential analysis one can perform using scMET.

Usage

```
scmet_diff_dt
```

Format

An object of class `scmet_simulate_diff` of length 9.

Value

A list object with simulated data.

`scmet_dt` *Synthetic methylation data from a single population*

Description

Small synthetic data for quick analysis, mostly useful for performing feature selection and capturing mean-variance relationship with scMET.

Usage

```
scmet_dt
```

Format

An object of class `scmet_simulate` of length 5.

Value

A list object with simulated data.

`scmet_hvf` *Detect highly (or lowly) variable features with scMET*

Description

Function for calling features as highly (or lowly) variable within a datasert or cell population. This can be thought as a feature selection step, where the highly variable features (HVF) can be used for diverse downstream tasks, such as clustering or visualisation. Two approaches for identifying HVFs (or LVFs): (1) If we correct for mean-dispersion relationship, then we work directly on residual dispersions `epsilon`, and define a percentile threshold `delta_e`. This is the preferred option since the residual overdispersion is not confounded by mean methylation levels. (2) Work directly with the overdispersion parameter `gamma` and define an overdispersion contribution threshold `delta_g`, above (below) of which we call HVFs (LVFs).

Usage

```
scmet_hvf(
  scmet_obj,
  delta_e = 0.9,
  delta_g = NULL,
  evidence_thresh = 0.8,
  efdr = 0.1
)

scmet_lvf(
  scmet_obj,
  delta_e = 0.1,
  delta_g = NULL,
  evidence_thresh = 0.8,
  efdr = 0.1
)
```

Arguments

`scmet_obj`	The scMET posterior object after performing inference, i.e. after calling `scmet` function.
`delta_e`	Percentile threshold for residual overdispersion to detect variable features (between 0 and 1). Default: 0.9 for HVF and 0.1 for LVF (top 10%). NOTE: This parameter should be used when correcting for mean-dispersion relationship.
`delta_g`	Overdispersion contribution threshold (between 0 and 1).
`evidence_thresh`	Optional parameter. Posterior evidence probability threshold parameter `alpha_{H}` (between 0.6 and 1).
`efdr`	Target for expected false discovery rate related to HVF/LVF detection (default = 0.1).

Value

The scMET posterior object with an additional element named `hvf` or `lvf` according to the analysis performed. This is a list object containing the following elements:

- `summary`: A data.frame containing HVF or LVF analysis output information per feature, including posterior medians for `mu`, `gamma`, and `epsilon`. The `tail_prob` column contains the posterior tail probability of a feature being called as HVF or LVF. The logical `is_variable` column informs whether the feature is called as variable or not.
- `evidence_thresh`: The optimal evidence threshold.
- `efdr`: The EFDR value.
- `efnr`: The EFNR value.
- `efdr_grid`: The EFDR values for the grid search.
- `efnr_grid`: The EFNR values for the grid search.
- `evidence_thresh_grid`: The grid where we searched for optimal evidence threshold.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential

Examples

```
# Fit scMET
obj <- scmet(Y = scmet_dt$Y, X = scmet_dt$X, L = 4, iter = 100)

# Run HVF analysis
obj <- scmet_hvf(scmet_obj = obj)

# Run LVF analysis
obj <- scmet_lvf(scmet_obj = obj)
```

---

scmet_plot_efdr_efnr_grid

Plot EFDR/EFNR grid

Description

Function for plotting the grid search performed to obtain the optimal posterior evidence threshold to achieve a specific EFDR.

Usage

```
scmet_plot_efdr_efnr_grid(obj, task = "hvf")
```

Arguments

obj	Either the scMET object after calling the scmet_hvf_lvf functions or the object from calling the scmet_differential function.
task	String. When calling variable features, i.e. output of scmet_hvf_lvf, it can be either "hvf" or "lvf". For differential analysis, i.e. output of scmet_differential, it can be either: (1) "diff_mu" for diff mean methylation, (2) "diff_epsilon" for residual overdispersion, or (3) "diff_gamma" for overdispersion analysis.

Value

A ggplot2 object.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf, scmet_plot_mean_var, scmet_plot_vf_tail_prob, scmet_plot_volcano, scmet_plot_ma

Examples

```
# Fit scMET
obj <- scmet(Y = scmet_dt$Y, X = scmet_dt$X, L = 4, iter = 100)
obj <- scmet_hvf(scmet_obj = obj, delta_e = 0.7)
scmet_plot_vf_tail_prob(obj = obj, task = "hvf")
```

---

scmet_plot_estimated_vs_true

Plot true versus inferred parameter estimated.

Description

Function for plotting true on x-axis and inferred parameter estimates on y-axis (either mean methylation or overdispersion). Along with posterior medians, the 80 high posterior density is shown as error bars. Wehn MLE estimates are provided, a plot showing the shrinkage introduced by scMET is shown as arrows.

Usage

```
scmet_plot_estimated_vs_true(
  obj,
  sim_dt,
  param = "mu",
  mle_fit = NULL,
  diff_feat_idx = NULL,
  hpd_thresh = 0.8,
  title = NULL,
  nfeatures = NULL
)
```

Arguments

obj	The scMET object after calling the scmet function.
sim_dt	The simulated data object. E.g. after calling the scmet_simulate function.
param	The parameter to plot posterior estimates, either "mu" or "gamma".
mle_fit	A three column matrix of beta-binomial maximum likelihood estimates. First column feature name, second column mean methylation and third column overdispersion estimates. Number of features should match the ones used by scMET.
diff_feat_idx	Vector with locations of features that were simulated to be differentially variable or methylated. This is stored in the object after calling the scmet_simulate_diff function.

hpd_thresh	The high posterior density threshold, as computed by the HPDinterval function.
title	Optional title, default NULL.
nfeatures	Optional parameter, denoting a subset of number of features to plot. Mostly to reduce over-plotting.

Value

A ggplot2 object.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_simulate_diff, scmet_simulate, scmet_plot_mean_var, scmet_plot_vf_tail_prob, scmet_plot_efdr_efnr_grid, scmet_plot_volcano, scmet_plot_ma

Examples

```
# Fit scMET
obj <- scmet(Y = scmet_dt$Y, X = scmet_dt$X, L = 4, iter = 100)
scmet_plot_estimated_vs_true(obj = obj, sim_dt = scmet_dt, param = "mu")

# BB MLE fit to compare with scMET
mle_fit <- scmet_dt$Y[, bb_mle(cbind(total_reads, met_reads))
[c("mu", "gamma")], by = c("Feature")]
scmet_plot_estimated_vs_true(obj = obj, sim_dt = scmet_dt, param = "mu",
mle_fit = mle_fit)
```

---

scmet_plot_ma *MA plot for differential analysis*

Description

Function showing MA plots for differential analysis. The y-axis shows difference between measurements across two groups and the x-axis shows the average measurements across the two groups.

Usage

```
scmet_plot_ma(
  diff_obj,
  task = "diff_epsilon",
  x = "mu",
  xlab = NULL,
  ylab = NULL,
  title = NULL,
  nfeatures = NULL
)
```

Arguments

diff_obj	The differential scMET object after calling the scmet_differential function.
task	The differential test to plot. For differential mean methylation: diff_mu that plots the LOR(mu_A, mu_B) on y-axis. For differential variability: either (1) diff_epsilon that plots the change (epsilon_A - epsilon_B), or (2) diff_gamma that plots the LOR(gamma_A, gamma_B) on y-axis.
x	The average parameter across the two populations to plot on the x-axis. Can be either mu, epsilon or gamma. When task = epsilon, x can be either mu or epsilon. When task = gamma, x can be either mu or gamma. When task = mu, x can be only mu.
xlab	Optional x-axis label.
ylab	Optional y-axis label.
title	Optional title, default NULL.
nfeatures	Optional parameter, denoting a subset of number of features to plot (only for non-differential features). Mostly to reduce over-plotting.

Value

A ggplot2 object.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf, scmet_plot_mean_var, scmet_plot_vf_tail_prob, scmet_plot_efdr_efnr_grid, scmet_plot_volcano

Examples

```
## Not run:
# Fit scMET for each group
fit_A <- scmet(Y = scmet_diff_dt$scmet_dt_A$Y,
X = scmet_diff_dt$scmet_dt_A$X, L = 4, iter = 100, seed = 12)
fit_B <- scmet(Y = scmet_diff_dt$scmet_dt_B$Y,
X = scmet_diff_dt$scmet_dt_B$X, L = 4, iter = 100, seed = 12)

# Run differential test
diff_obj <- scmet_differential(obj_A = fit_A, obj_B = fit_B)
# Create volcano plot
scmet_plot_ma(diff_obj, task = "diff_epsilon")

## End(Not run)
```

scmet_plot_mean_var *Plotting mean-variability relationship*

Description

Function for plotting mean methylation on x-axis and variability on y-axis (either overdispersion or residual overdispersion). If HVF/LVF analysis is performed, points will be also coloured accordingly.

Usage

```
scmet_plot_mean_var(
  obj,
  y = "gamma",
  task = NULL,
  show_fit = TRUE,
  title = NULL,
  nfeatures = NULL,
  n = 80
)
```

Arguments

obj	The scMET object after calling the scmet_hvf_lvf function.
y	The parameter to plot on the y-axis. Values can be gamma (default) or epsilon.
task	If NULL (default) the mean-variability relationship is plotted. If set to "hvf" or "lvf", points are coloured according the HVF/LVF analysis task.
show_fit	Logical, whether to show the fitted mean-overdispersion trend. Applicable only when y = gamma and task = NULL.
title	Optional title, default NULL.
nfeatures	Optional parameter, denoting a subset of number of features to plot. Mostly to reduce over-plotting. When task = hvf or lvf, the subsampling is performed on the features that are not called as HVF or LVF (i.e. not interesting features).
n	Optional integer denoting the number of grid points to colour them by density. Used by kde2d function. Used only when task = NULL.

Value

A ggplot2 object.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf, scmet_plot_vf_tail_prob, scmet_plot_efdr_efnr_grid, scmet_plot_volcano, scmet_plot_ma

Examples

```
# Fit scMET
obj <- scmet(Y = scmet_dt$Y, X = scmet_dt$X, L = 4, iter = 100)
scmet_plot_mean_var(obj = obj, y = "gamma")
```

---

scmet_plot_vf_tail_prob

Plot tail probabilities for variable feature analysis

Description

Function for plotting the tail probabilities associated with the HVF/LVF analysis. The tail probabilities are plotted on the y-axis, and the user can choose which parameter can be plotted on the x-axis, using the x parameter.

Usage

```
scmet_plot_vf_tail_prob(
  obj,
  x = "mu",
  task = "hvf",
  title = NULL,
  nfeatures = NULL
)
```

Arguments

obj	The scMET object after calling the scmet_hvf_lvf function.
x	The parameter to plot on the x-axis. Values can be mu (default), epsilon or gamma.
task	The task for identifying variable, either "hvf" or "lvf".
title	Optional title, default NULL.
nfeatures	Optional parameter, denoting a subset of number of features to plot (only for non HVF/LVF features). Mostly to reduce over-plotting.

Value

A ggplot2 object.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf, scmet_plot_mean_var, scmet_plot_efdr_efnr_grid, scmet_plot_volcano, scmet_plot_ma

Examples

```
# Fit scMET
obj <- scmet(Y = scmet_dt$Y, X = scmet_dt$X, L = 4, iter = 100)
obj <- scmet_hvf(scmet_obj = obj, delta_e = 0.7)
scmet_plot_vf_tail_prob(obj = obj, x = "mu")
```

---

scmet_plot_volcano *Volcano plot for differential analysis*

Description

Function showing volcano plots for differential analysis. The posterior tail probabilities are ploteted on the y-axis, and depending on the differential test to plot the effect size will be plotted on the x-axis. For differential variability (DV) analysis we recommend using the `epsilon` parameter.

Usage

```
scmet_plot_volcano(
  diff_obj,
  task = "diff_epsilon",
  xlab = NULL,
  ylab = "Posterior tail probability",
  title = NULL,
  nfeatures = NULL
)
```

Arguments

`diff_obj`	The differential scMET object after calling the `scmet_differential` function.
`task`	The differential test to plot. For differential mean methylation: `diff_mu` that plots the LOR(mu_A, mu_B) on x-axis. For differential variability: either (1) `diff_epsilon` that plots the change (epsilon_A - epsilon_B), or (2) `diff_gamma` that plots the LOR(gamma_A, gamma_B) on x-axis.
`xlab`	Optional x-axis label.
`ylab`	Optional y-axis label.
`title`	Optional title, default NULL.
`nfeatures`	Optional parameter, denoting a subset of number of features to plot (only for non-differential features). Mostly to reduce over-plotting.

Value

A ggplot2 object.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf, scmet_plot_mean_var, scmet_plot_vf_tail_prob, scmet_plot_efdr_efnr_grid, scmet_plot_ma

Examples

```
## Not run:
# Fit scMET for each group
fit_A <- scmet(Y = scmet_diff_dt$scmet_dt_A$Y,
X = scmet_diff_dt$scmet_dt_A$X, L = 4, iter = 100, seed = 12)
fit_B <- scmet(Y = scmet_diff_dt$scmet_dt_B$Y,
X = scmet_diff_dt$scmet_dt_B$X, L = 4, iter = 100, seed = 12)

# Run differential test
diff_obj <- scmet_differential(obj_A = fit_A, obj_B = fit_B)
# Create volcano plot
scmet_plot_volcano(diff_obj, task = "diff_epsilon")

## End(Not run)
```

---

scmet_simulate *Simulate methylation data from scMET.*

Description

General function for simulating datasets with diverse proprties. This for instance include, adding covariates X that explain differences in mean methylation levels. Or also defining the trend for the mean - overdispersion relationship.

Usage

```
scmet_simulate(
  N_feat = 100,
  N_cells = 50,
  N_cpgs = 15,
  L = 4,
  X = NULL,
  w_mu = c(-0.5, -1.5),
  s_mu = 1,
```

```
  w_gamma = NULL,
  s_gamma = 0.3,
  rbf_c = 1,
  cells_range = c(0.4, 0.8),
  cpgs_range = c(0.4, 0.8)
)
```

Arguments

N_feat	Total number of features (genomics regions).
N_cells	Maximum number of cells.
N_cpgs	Maximum number of CpGs per cell and feature.
L	Total number of radial basis functions (RBFs) to fit the mean-overdispersion trend. For L = 1, this reduces to a model that does not correct for the mean-overdispersion relationship.
X	Covariates which might explain variability in mean (methylation). If X = NULL, a 2-dim matrix will be generated, first column containing intercept term (all values = 1), and second colunn random generated covariates.
w_mu	Regression coefficients for covariates X. Should match number of columns of X.
s_mu	Standard deviation for mean parameter mu.
w_gamma	Regression coefficients of the basis functions. Should match the value of L. If NULL, random coefficients will be generated.
s_gamma	Standard deviation of dispersion parameter gamma.
rbf_c	Scale parameter for empirically computing the variance of the RBFs.
cells_range	Range (betwen 0 and 1) to randomly (sub)sample the number of cells per feature.
cpgs_range	Range (betwen 0 and 1) to randomly (sub)sample the number of CpGs per cell and feature.

Value

A simulated dataset and additional information for reproducibility purposes.

Examples

```
sim <- scmet_simulate(N_feat = 150, N_cells = 50, N_cpgs = 15, L = 4)
```

scmet_simulate_diff *Simulate differential methylation data from scMET.*

Description

General function for simulating two methylation datasets for performing differential methylation analysis. Differential analysis can be either performed in detecting changes in mean or variability of methylation patterns between the two groups. Similar to scmet_simulate, the function allows inclusion of covariates X that explain differences in mean methylation levels. Or also defining the trend for the mean - overdispersion relationship.

Usage

```
scmet_simulate_diff(
  N_feat = 100,
  N_cells = 50,
  N_cpgs = 15,
  L = 4,
  diff_feat_prcg_mu = 0,
  diff_feat_prcg_gamma = 0.2,
  OR_change_mu = 3,
  OR_change_gamma = 3,
  X = NULL,
  w_mu = c(-0.5, -1.5),
  s_mu = 1,
  w_gamma = NULL,
  s_gamma = 0.3,
  rbf_c = 1,
  cells_range = c(0.4, 0.8),
  cpgs_range = c(0.4, 0.8)
)
```

Arguments

N_feat	Total number of features (genomics regions).
N_cells	Maximum number of cells.
N_cpgs	Maximum number of CpGs per cell and feature.
L	Total number of radial basis functions (RBFs) to fit the mean-overdispersion trend. For L = 1, this reduces to a model that does not correct for the mean-overdispersion relationship.
diff_feat_prcg_mu	Percentage of features (betwen 0 and 1) that show differential mean methylation between the two groups.
diff_feat_prcg_gamma	Percentage of features (betwen 0 and 1) that show differential variability between the two groups.

OR_change_mu	Effect size change (in terms of odds ratio) of mean methylation between the two groups.
OR_change_gamma	Effect size change (in terms of odds ratio) of methylation variability between the two groups.
X	Covariates which might explain variability in mean (methylation). If X = NULL, a 2-dim matrix will be generated, first column containing intercept term (all values = 1), and second colunn random generated covariates.
w_mu	Regression coefficients for covariates X. Should match number of columns of X.
s_mu	Standard deviation for mean parameter mu.
w_gamma	Regression coefficients of the basis functions. Should match the value of L. If NULL, random coefficients will be generated.
s_gamma	Standard deviation of dispersion parameter gamma.
rbf_c	Scale parameter for empirically computing the variance of the RBFs.
cells_range	Range (betwen 0 and 1) to randomly (sub)sample the number of cells per feature.
cpgs_range	Range (betwen 0 and 1) to randomly (sub)sample the number of CpGs per cell and feature.

Value

Methylation data from two cell populations/conditions.

Examples

```
sim_diff <- scmet_simulate_diff(N_feat = 150, N_cells = 100, N_cpgs = 15, L = 4)
```

---

scmet_to_sce *Convert from scmet to SingleCellExperiment object.*

Description

Helper function that converts an scmet to SCE object. The structure of the SCE object to store single cell methylation data is the following. We create two assays, `met` storing methylated CpGs and `total` storing total number of CpGs. Rows correspond to features and columns to cells, similar to scRNA-seq convention. The `rownames` and `colnames` slots should store the feature and cell names, respectively. Covariates `X` that might explain variability in mean (methylation) should be stored in `metadata(rowData(sce)$X`.

Usage

```
scmet_to_sce(Y, X = NULL)
```

Arguments

Y	Methylation data in data.table format.
X	(Optional) Matrix of covariates.

Value

An SCE object with the structure described above.

Author(s)

C.A.Kapourani <C.A.Kapourani@ed.ac.uk>

See Also

scmet, scmet_differential, scmet_hvf_lvf

Examples

```
# Extract
sce <- scmet_to_sce(Y = scmet_dt$Y, X = scmet_dt$X)
```

Index