Package 'perturbplan'

June 7, 2025

Title Experimental Design and Power Analysis for Perturb-Seq Experiments

Version 0.0.1

Description Experimental design and power analysis for perturb-seq experiments.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports dplyr, stats, rhdf5, minpack.lm, Rcpp, tibble, dtplyr, data.table, methods, utils

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

LinkingTo PoissonBinomial, Rcpp

NeedsCompilation yes

Author Ziang Niu [aut, cre], Yihui He [aut], Eugene Katsevich [aut]

Maintainer Ziang Niu <ziangniu@wharton.upenn.edu>

Depends R (>= 3.5.0)

Contents

`.compute_underspecified_power_efficient` *Internal function for efficient separated power computation using C++*

Description

Internal function for efficient separated power computation using C++

Internal function for efficient separated power computation using C++

Usage

```
.compute_underspecified_power_efficient(
  num_total_cells,
  library_size,
  MOI = 10,
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  multiple_testing_alpha = 0.05,
  multiple_testing_method = "BH",
  control_group = "complement",
  side = "left",
  num_pairs = 1000,
  fc_expression_df,
  expression_dispersion_curve,
  fc_output_grid,
  expr_output_grid,
  prop_non_null = 0.1
)

.compute_underspecified_power_efficient(
  num_total_cells,
```

```
  library_size,
  MOI = 10,
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  multiple_testing_alpha = 0.05,
  multiple_testing_method = "BH",
  control_group = "complement",
  side = "left",
  num_pairs = 1000,
  fc_expression_df,
  expression_dispersion_curve,
  fc_output_grid,
  expr_output_grid,
  prop_non_null = 0.1
)
```

Arguments

num_total_cells	Total number of cells
library_size	Library size (reads per cell)
MOI	Multiplicity of infection
num_targets	Number of targets
gRNAs_per_target	Number of gRNAs per target
non_targeting_gRNAs	Number of non-targeting gRNAs
multiple_testing_alpha	Alpha level for multiple testing
multiple_testing_method	Multiple testing method
control_group	Control group type
side	Test sidedness
num_pairs	Number of pairs
fc_expression_df	Data frame with fold change and expression info
expression_dispersion_curve	Function for expression-size relationship
fc_output_grid	Grid points for fold change curve
expr_output_grid	Grid points for expression curve
prop_non_null	Proportion of non-null hypotheses

Value

List with overall power and power curves

List with overall power and power curves

`.compute_underspecified_power_separated` *Internal function for separated power computation*

Description

Internal function for separated power computation

Internal function for separated power computation

Usage

```
.compute_underspecified_power_separated(
  num_total_cells,
  library_size,
  MOI = 10,
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  multiple_testing_alpha = 0.05,
  multiple_testing_method = "BH",
  control_group = "complement",
  side = "left",
  num_pairs = 1000,
  fc_expression_df,
  expression_dispersion_curve,
  fc_output_grid,
  expr_output_grid,
  prop_non_null = 0.1
)

.compute_underspecified_power_separated(
  num_total_cells,
  library_size,
  MOI = 10,
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  multiple_testing_alpha = 0.05,
  multiple_testing_method = "BH",
  control_group = "complement",
  side = "left",
  num_pairs = 1000,
  fc_expression_df,
  expression_dispersion_curve,
  fc_output_grid,
  expr_output_grid,
  prop_non_null = 0.1
)
```

Arguments

num_total_cells	Total number of cells
library_size	Library size (reads per cell)
MOI	Multiplicity of infection
num_targets	Number of targets
gRNAs_per_target	Number of gRNAs per target
non_targeting_gRNAs	Number of non-targeting gRNAs
multiple_testing_alpha	Alpha level for multiple testing
multiple_testing_method	Multiple testing method
control_group	Control group type
side	Test sidedness
num_pairs	Number of pairs
fc_expression_df	Data frame with fold change and expression info
expression_dispersion_curve	Function for expression-size relationship
fc_output_grid	Grid points for fold change curve
expr_output_grid	Grid points for expression curve
prop_non_null	Proportion of non-null hypotheses

Value

List with overall power and power curves

List with overall power and power curves

---

adjusted_cutoff *Compute the adjusted significance level with either BH or Bonferroni procedure.*

Description

Compute the adjusted significance level with either BH or Bonferroni procedure.

Usage

```
adjusted_cutoff(
  mean_list,
  sd_list,
  multiple_testing_alpha,
  multiple_testing_method,
  side,
  QC_prob
)
```

Arguments

`mean_list`	Asymptotic mean of test statistic
`sd_list`	Asymptotic sd of test statistic
`multiple_testing_alpha`	(Optional) A numeric value between 0 and 1 specifying the alpha level for multiple testing correction; defaults to 0.1
`multiple_testing_method`	(Optional) A character string specifying the multiple testing correction method to use, either "BH" or "bonferroni"; defaults to "BH"
`side`	(Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"
`QC_prob`	The probability of failing QC

Value

The adjusted significance level.

`BH_cutoff_bisection` *Benjamini–Hochberg cutoff with bisection search (C++ back-end)*

Description

Thin wrapper that validates inputs and forwards to the compiled routine.

Usage

```
BH_cutoff_bisection(mean_list, sd_list, side, multiple_testing_alpha, QC_prob)
```

Arguments

`mean_list`	Asymptotic mean of test statistic
`sd_list`	Asymptotic sd of test statistic
`side`	(Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"
`multiple_testing_alpha`	(Optional) A numeric value between 0 and 1 specifying the alpha level for multiple testing correction; defaults to 0.1
`QC_prob`	The probability of failing QC

Value

Adjusted cutoff/significance level.

`calculate_power_grid` *Calculate power grid for app heatmap visualization*

Description

This function provides power analysis functionality for the Shiny application. It creates a grid of cell/read combinations and computes power for each combination.

Usage

```
calculate_power_grid(
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  num_pairs = 1000,
  tpm_threshold = 10,
  fdr_target = 0.05,
  fc_mean = 0.85,
  fc_sd = 0.15,
  prop_non_null = 0.1,
  MOI = 10,
  biological_system = "K562",
  experimental_platform = "10x Chromium v3"
)
```

Arguments

`num_targets`	Number of targets
`gRNAs_per_target`	Number of gRNAs per target
`non_targeting_gRNAs`	Number of non-targeting gRNAs
`num_pairs`	Number of pairs analyzed
`tpm_threshold`	Minimum TPM threshold
`fdr_target`	FDR target level
`fc_mean`	Fold-change mean
`fc_sd`	Fold-change SD
`prop_non_null`	Proportion of non-null pairs
`MOI`	Multiplicity of infection
`biological_system`	Biological system
`experimental_platform`	Experimental platform

Value

List with power grid, cell/read sequences, and parameters

`compute_distribution_teststat` *Compute mean and sd of the score test statistic*

Description

Compute mean and sd of the score test statistic

Usage

```
compute_distribution_teststat(
  num_trt_cells,
  num_cntrl_cells,
  num_trt_cells_sq,
  expression_mean,
  expression_size,
  fold_change_mean,
  fold_change_sd
)
```

Arguments

`num_trt_cells`	Number of treatment cells in score test
`num_cntrl_cells`	Number of control cells in score test
`num_trt_cells_sq`	Squared number of control cells in score test
`expression_mean`	Mean gene expression
`expression_size`	Size parameter in NB distribution
`fold_change_mean`	A numeric value to use for mean effect size for all element-gene pairs
`fold_change_sd`	A numeric value to use for standard deviation of effect size for all element-gene pairs

Value

A list including mean and sd of the test statistic

`compute_power_grid_efficient` *Compute power grid using efficient C++ test statistic computation*

Description

This function is identical to compute_power_grid_separated() except it uses the C++ implementation compute_distribution_teststat_fixed_es_cpp() instead of the R function compute_test_stat_clean() for test statistic computation.

Usage

```
compute_power_grid_efficient(
  cells_reads_df,
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  num_pairs = 1000,
  tpm_threshold = 10,
  fdr_target = 0.05,
  fc_mean = 0.85,
  fc_sd = 0.15,
  prop_non_null = 0.1,
  MOI = 10,
  biological_system = "K562",
  experimental_platform = "10x Chromium v3",
  side = "left",
  control_group = "complement",
  B = 500,
  fc_curve_points = 10,
  expr_curve_points = 10
)
```

Arguments

`cells_reads_df`	Data frame with columns num_total_cells and reads_per_cell
`num_targets`	Number of targets to test
`gRNAs_per_target`	Number of gRNAs per target
`non_targeting_gRNAs`	Number of non-targeting gRNAs
`num_pairs`	Number of pairs for multiple testing
`tpm_threshold`	TPM threshold (currently unused)
`fdr_target`	Target false discovery rate
`fc_mean`	Mean fold change for effect size distribution
`fc_sd`	Standard deviation of fold change distribution
`prop_non_null`	Proportion of non-null hypotheses
`MOI`	Multiplicity of infection
`biological_system`	Biological system for baseline expression
`experimental_platform`	Experimental platform
`side`	Test sidedness ("left", "right", "both")
`control_group`	Control group type ("complement" or "nt_cells")
`B`	Number of Monte Carlo samples for integration
`fc_curve_points`	Number of points for fold change curve
`expr_curve_points`	Number of points for expression curve

Value

Data frame with power analysis results

`compute_power_grid_separated` *Compute power grid using separated Monte Carlo approach*

Description

This function separates Monte Carlo integration accuracy from output grid resolution, providing an efficient and flexible approach to power analysis.

Usage

```
compute_power_grid_separated(
  cells_reads_df,
  num_targets = 100,
  gRNAs_per_target = 4,
  non_targeting_gRNAs = 10,
  num_pairs = 1000,
  tpm_threshold = 10,
  fdr_target = 0.05,
  fc_mean = 0.85,
  fc_sd = 0.15,
  prop_non_null = 0.1,
  MOI = 10,
  biological_system = "K562",
  experimental_platform = "10x Chromium v3",
  side = "left",
  control_group = "complement",
  B = 500,
  fc_curve_points = 10,
  expr_curve_points = 10
)
```

Arguments

`cells_reads_df`	Data frame with columns num_total_cells and reads_per_cell
`num_targets`	Number of targets to test
`gRNAs_per_target`	Number of gRNAs per target
`non_targeting_gRNAs`	Number of non-targeting gRNAs
`num_pairs`	Number of pairs for multiple testing
`tpm_threshold`	TPM threshold (currently unused)
`fdr_target`	Target false discovery rate
`fc_mean`	Mean fold change for effect size distribution
`fc_sd`	Standard deviation of fold change distribution

prop_non_null Proportion of non-null hypotheses

MOI Multiplicity of infection

biological_system
Biological system for baseline expression

experimental_platform
Experimental platform

side Test sidedness ("left", "right", "both")

control_group Control group type ("complement" or "nt_cells")

B Number of Monte Carlo samples for integration

fc_curve_points
Number of points for fold change curve

expr_curve_points
Number of points for expression curve

Value

Data frame with power analysis results

`compute_power_posthoc` *Compute power for each perturbation-gene pair*

Description

Compute power for each perturbation-gene pair

Usage

```
compute_power_posthoc(
  discovery_pairs,
  cells_per_grna,
  baseline_expression_stats,
  control_group,
  fold_change_mean,
  fold_change_sd,
  num_total_cells = NULL,
  cutoff = NULL,
  n_nonzero_trt_thresh = 7L,
  n_nonzero_cntrl_thresh = 7L,
  side = "both",
  multiple_testing_method = "BH",
  multiple_testing_alpha = 0.1
)
```

Arguments

discovery_pairs
A data frame specifying which element-gene pairs to consider, with columns `grna_target` and `response_id`

`cells_per_grna` A data frame specifying how many cells contain each gRNA, with columns `grna_id`, `grna_target`, and `num_cells`

`baseline_expression_stats`
A data frame specifying the baseline expression statistics for each gene, with columns `response_id`, `expression_mean`, and `expression_size`

`control_group` A character string specifying the control group, either "complement" or "nt_cells"

`fold_change_mean`
A numeric value to use for mean effect size for all element-gene pairs

`fold_change_sd` A numeric value to use for standard deviation of effect size for all element-gene pairs

`num_total_cells`
(Required only if control_group == "complement") A positive integer specifying the total number of cells in the experiment

`cutoff` (Optional) A numeric value between 0 and 1 to use as the p-value cutoff

`n_nonzero_trt_thresh`
(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7

`n_nonzero_cntrl_thresh`
(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7

`side` (Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"

`multiple_testing_method`
(Optional) A character string specifying the multiple testing correction method to use, either "BH" or "bonferroni"; defaults to "BH"

`multiple_testing_alpha`
(Optional) A numeric value between 0 and 1 specifying the alpha level for multiple testing correction; defaults to 0.1

Value

A list with two elements: `individual_power` (a data frame with columns `grna_target`, `response_id`, and `power`) and `expected_num_discoveries` (a numeric value)

`compute_power_posthoc_fixed_fc`
Compute power for each perturbation-gene pair with fixed fold change

---

Description

This function computes power for perturb-seq experiments with fixed (non-random) gRNA assignment. It uses C++ implementations for computational efficiency.

Usage

```
compute_power_posthoc_fixed_fc(
  discovery_pairs,
  cells_per_grna,
  baseline_expression_stats,
  control_group,
  fold_change,
  num_total_cells = NULL,
  cutoff = NULL,
  n_nonzero_trt_thresh = 7L,
  n_nonzero_cntrl_thresh = 7L,
  side = "both",
  multiple_testing_method = "BH",
  multiple_testing_alpha = 0.1
)
```

Arguments

`discovery_pairs`	A data frame specifying which element-gene pairs to consider, with columns `grna_target` and `response_id`; it can also have `grna_id` as a column but this is optional
`cells_per_grna`	A data frame specifying how many cells contain each gRNA, with columns `grna_id`, `grna_target`, and `num_cells`
`baseline_expression_stats`	A data frame specifying the baseline expression statistics for each gene, with columns `response_id`, `expression_mean`, and `expression_size`
`control_group`	A character string specifying the control group, either "complement" or "nt_cells"
`fold_change`	A numeric value or data frame to use for fixed effect size for all gRNA-gene pairs
`num_total_cells`	(Required only if control_group == "complement") A positive integer specifying the total number of cells in the experiment
`cutoff`	(Optional) A numeric value between 0 and 1 to use as the p-value cutoff
`n_nonzero_trt_thresh`	(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7
`n_nonzero_cntrl_thresh`	(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7
`side`	(Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"
`multiple_testing_method`	(Optional) A character string specifying the multiple testing correction method to use, either "BH" or "bonferroni"; defaults to "BH"
`multiple_testing_alpha`	(Optional) A numeric value between 0 and 1 specifying the alpha level for multiple testing correction; defaults to 0.1

Value

A list with two elements: `individual_power` (a data frame with columns `grna_target`, `response_id`, and `power`) and `expected_num_discoveries` (a numeric value)

`compute_QC` *Compute QC probability for each enhancer-gene pair*

Description

Compute QC probability for each enhancer-gene pair

Usage

```
compute_QC(
  fold_change_mean,
  expression_mean,
  expression_size,
  num_cntrl_cells,
  num_trt_cells,
  n_nonzero_trt_thresh,
  n_nonzero_cntrl_thresh
)
```

Arguments

`fold_change_mean`	A numeric value to use for mean effect size for all element-gene pairs
`expression_mean`	Mean gene expression
`expression_size`	Size parameter in NB distribution
`num_cntrl_cells`	Number of control cells in score test
`num_trt_cells`	Number of treatment cells in score test
`n_nonzero_trt_thresh`	(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7
`n_nonzero_cntrl_thresh`	(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7

Value

Probability of a enhancer-gene pair being filtered due to QC

`compute_test_stat_clean` *Helper function that returns named vector directly (cleaner than original)*

Description

Helper function that returns named vector directly (cleaner than original)

Usage

```
compute_test_stat_clean(
  num_trt_cells,
  num_cntrl_cells,
  expression_mean,
  expression_size,
  fold_change_mean
)
```

Arguments

`num_trt_cells`	Number of treatment cells
`num_cntrl_cells`	Number of control cells
`expression_mean`	Mean expression level
`expression_size`	Size parameter for negative binomial
`fold_change_mean`	Mean fold change

Value

Named vector with mean and sd of test statistic

`compute_zero_prob` *Compute probability mass of NB distribution at zero*

Description

Compute probability mass of NB distribution at zero

Usage

```
compute_zero_prob(fold_change_mean, expression_mean, expression_size)
```

Arguments

`fold_change_mean`
: A numeric value to use for mean effect size for all element-gene pairs

`expression_mean`
: Mean gene expression

`expression_size`
: Size parameter in NB distribution

Value

Probability of a NB variable being 0

`FDP_estimate` *FDP estimate based on rejection probability.*

Description

FDP estimate based on rejection probability.

Usage

```
FDP_estimate(mean_list, sd_list, side, cutoff, QC_prob)
```

Arguments

`mean_list`	Asymptotic mean of test statistic
`sd_list`	Asymptotic sd of test statistic
`side`	(Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"
`cutoff`	(Optional) A numeric value between 0 and 1 to use as the p-value cutoff
`QC_prob`	The probability of failing QC

Value

FDP estimate.

input_check_library_computation *Input checking function for library_computation*

Description

Input checking function for library_computation

Usage

```
input_check_library_computation(
  QC_data = NULL,
  downsample_ratio = NULL,
  D2_rough = NULL
)
```

Arguments

QC_data	The QC'd data coming from the function obtain_qc_h5_data.
downsample_ratio	The ratio of the size of the downsampled dataset to the one of the original dataset.
D2_rough	The rough estimate of D2 in the S-M curve model.

input_check_posthoc *Input checking function for compute_power_posthoc*

Description

Input checking function for compute_power_posthoc

Usage

```
input_check_posthoc(
  discovery_pairs = NULL,
  cells_per_grna = NULL,
  baseline_expression_stats = NULL,
  control_group = NULL,
  fold_change_mean = NULL,
  fold_change_sd = NULL,
  num_total_cells = NULL,
  cutoff = NULL,
  n_nonzero_trt_thresh = NULL,
  n_nonzero_cntrl_thresh = NULL,
  side = NULL,
  multiple_testing_method = NULL,
  multiple_testing_alpha = NULL
)
```

Arguments

`discovery_pairs`	A data frame specifying which element-gene pairs to consider, with columns `grna_target` and `response_id`
`cells_per_grna`	A data frame specifying how many cells contain each gRNA, with columns `grna_id`, `grna_target`, and `num_cells`
`baseline_expression_stats`	A data frame specifying the baseline expression statistics for each gene, with columns `response_id`, `expression_mean`, and `expression_size`
`control_group`	A character string specifying the control group, either "complement" or "nt_cells"
`fold_change_mean`	A numeric value to use for mean effect size for all element-gene pairs
`fold_change_sd`	A numeric value to use for standard deviation of effect size for all element-gene pairs
`num_total_cells`	(Required only if control_group == "complement") A positive integer specifying the total number of cells in the experiment
`cutoff`	(Optional) A numeric value between 0 and 1 to use as the p-value cutoff
`n_nonzero_trt_thresh`	(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7
`n_nonzero_cntrl_thresh`	(Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7
`side`	(Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"
`multiple_testing_method`	(Optional) A character string specifying the multiple testing correction method to use, either "BH" or "bonferroni"; defaults to "BH"
`multiple_testing_alpha`	(Optional) A numeric value between 0 and 1 specifying the alpha level for multiple testing correction; defaults to 0.1

`input_check_power_function` *Input checking function for power_function*

Description

Input checking function for power_function

Usage

```
input_check_power_function(
  recovery_rate = NULL,
  num_total_reads = NULL,
  mapping_efficiency = NULL,
  cells_per_grna = NULL,
```

```
  baseline_relative_expression_stats = NULL,
  fold_change_mean = NULL,
  fold_change_sd = NULL,
  num_planned_cells = NULL,
  control_group = NULL,
  umi_per_cell = NULL,
  umi_variation = NULL,
  side = NULL,
  multiple_testing_method = NULL,
  multiple_testing_alpha = NULL,
  cutoff = NULL,
  discovery_pairs = NULL,
  n_nonzero_trt_thresh = NULL,
  n_nonzero_cntrl_thresh = NULL
)
```

Arguments

`recovery_rate`	A numeric value (between 0 and 1) indicating the fraction of cells that survive and are captured after library preparation.
`num_total_reads`	A numeric value specifying the total number of reads generated by sequencing. This is used to estimate the `library_size`.
`mapping_efficiency`	A numeric value (between 0 and 1) indicating the fraction of reads that successfully map to the transcriptome.
`cells_per_grna`	A data frame specifying the number of cells per gRNA, with columns `grna_id`, `grna_target`, and `num_cells`.
`baseline_relative_expression_stats`	A data frame specifying the relative expression levels for each gene, with columns `response_id` and `relative_expression`.
`fold_change_mean`	A numeric value indicating the mean fold change effect size for all gRNA-gene pairs (or a data frame with `grna_target` and `response_id` columns for per-pair values).
`fold_change_sd`	A numeric value indicating the standard deviation of the fold change effect size for all gRNA-gene pairs (or a data frame with `grna_target` and `response_id` columns for per-pair values).
`num_planned_cells`	A numeric value indicating the total planned number of cells before losses in library preparation.
`control_group`	A character string specifying the control group, either `"complement"` or `"nt_cells"`. This is passed to `compute_power_posthoc`.
`umi_per_cell`	A numeric value specifying the theoretical saturation level (in UMIs) for each cell.
`umi_variation`	A numeric value controlling how overdispersion in UMIs per read is modeled.
`side`	(Optional) A character string specifying the side of the test, either `"left"`, `"right"`, or `"both"`. Defaults to `"both"`.
`multiple_testing_method`	(Optional) A character string specifying the multiple testing correction method, either `"BH"` or `"bonferroni"`. Defaults to `"BH"`.

multiple_testing_alpha
: (Optional) A numeric value (between 0 and 1) specifying the alpha level for multiple testing correction. Defaults to 0.1.

cutoff
: (Optional) A numeric value between 0 and 1 to use as the p-value cutoff. If NULL, the function determines it automatically using the specified multiple_testing_method and multiple_testing_alpha.

discovery_pairs
: A data frame specifying which gRNA-gene pairs to consider, with columns grna_target and response_id.

n_nonzero_trt_thresh
: (Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7.

n_nonzero_cntrl_thresh
: (Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7.

---

launch_app *Launch the PerturbPlan Shiny App*

Description

Launch the PerturbPlan Shiny App

Usage

```
launch_app()
```

Value

None

---

library_computation *Fit the S-M curve between # mapped reads and # ovserved UMIs.*

Description

Fit the S-M curve between # mapped reads and # ovserved UMIs.

Usage

```
library_computation(QC_data, downsample_ratio = 0.7, D2_rough = 0.3)
```

Arguments

QC_data
: The QC'd data coming from the function obtain_qc_h5_data.

downsample_ratio
: The ratio of the size of the downsampled dataset to the one of the original dataset.

D2_rough
: The rough estimate of D2 in the S-M curve model.

Value

A fitted S-M curve model in the form of a nlsLM object.

`library_estimation` *Compute the average total UMI per cell and UMI variation parameters.*

Description

Compute the average total UMI per cell and UMI variation parameters.

Usage

```
library_estimation(QC_data, downsample_ratio = 0.7, D2_rough = 0.3)
```

Arguments

`QC_data`	The QC'd data coming from the function obtain_qc_h5_data.
`downsample_ratio`	The ratio of the size of the downsampled dataset to the one of the original dataset.
`D2_rough`	The rough estimate of D2 in the S-M curve model.

Value

Named vector of average total UMI per cell and UMI variation.

`obtain_expression_information` *Obtain gene-level expression and filtering information based on TPM.*

Description

Obtain gene-level expression and filtering information based on TPM.

Usage

```
obtain_expression_information(
  TPM_thres = 10,
  response_matrix,
  cell_covariates = NULL
)
```

Arguments

`TPM_thres`	TPM threshold for gene filtering (default = 10).
`response_matrix`	Expression matrix (genes × cells).
`cell_covariates`	Optional cell-level covariates. If NULL, uses log-library size.

Value

A data frame of genes with relative expression and estimated size (theta).

`obtain_mapping_efficiency` *Compute mapping efficiency from QC'd molecule info and Cell Ranger metrics.*

Description

Compute mapping efficiency from QC'd molecule info and Cell Ranger metrics.

Usage

```
obtain_mapping_efficiency(QC_data, path_to_metrics_summary)
```

Arguments

`QC_data`	A data frame from `obtain_qc_h5_data()`, must contain a `num_reads` column.
`path_to_metrics_summary`	Path to the folder containing Cell Ranger's `metrics_summary.csv`.

Value

A numeric value representing the mapping efficiency (mapped_reads / total_reads).

`obtain_qc_h5_data` *Obtain a data frame with information of all QC'd reads for library estimation.*

Description

Obtain a data frame with information of all QC'd reads for library estimation.

Usage

```
obtain_qc_h5_data(path_to_h5_file)
```

Arguments

`path_to_h5_file`	The path to the outs folder of the cellranger output.

Value

A data frame with columns `num_reads`, `UMI_id`, `cell_id`, and `response_id`

`obtain_qc_response_data` *Load and QC gene expression matrix from Cell Ranger .mtx format.*

Description

This function reads a sparse expression matrix from a Cell Ranger output directory (in `.mtx` format) and performs quality control by removing genes with missing, empty, or duplicated names.

Usage

```
obtain_qc_response_data(path_to_gene_expression)
```

Arguments

`path_to_gene_expression`	Character. Path to the folder containing `matrix.mtx.gz`, `features.tsv.gz`, and `barcodes.tsv.gz`.

Value

A sparse gene-by-cell expression matrix of class `dgCMatrix`, where only genes with valid and unique names are retained. Row names are set to gene symbols.

`obtain_random_pairs` *Generate random gRNA–gene discovery pairs for control or simulation.*

Description

Generate random gRNA–gene discovery pairs for control or simulation.

Usage

```
obtain_random_pairs(num_targets, pairs_per_target, gene_info)
```

Arguments

`num_targets`	Integer. Number of pseudo gRNAs to simulate.
`pairs_per_target`	Integer. Number of genes to assign per gRNA.
`gene_info`	Data frame. Must contain a column named `response_id` with gene names.

Value

A data frame with columns `response_id` and `grna_id`, each row representing a pseudo discovery pair.

power_function *Compute approximate power of a CRISPR screen*

Description

This function computes the approximate power of detecting an effect (gene perturbation) in a CRISPR screen given various experimental and sequencing parameters. Internally, it calculates an average library size using provided parameters and estimates baseline expression levels. The function then calls `compute_power_posthoc` to obtain power estimates and the expected number of discoveries.

Usage

```
power_function(
  recovery_rate,
  num_total_reads,
  mapping_efficiency,
  cells_per_grna,
  baseline_relative_expression_stats,
  fold_change_mean,
  fold_change_sd,
  num_planned_cells,
  control_group,
  umi_per_cell,
  umi_variation,
  side = "both",
  multiple_testing_method = "BH",
  multiple_testing_alpha = 0.1,
  cutoff = NULL,
  discovery_pairs,
  n_nonzero_trt_thresh = 7L,
  n_nonzero_cntrl_thresh = 7L
)
```

Arguments

`recovery_rate`	A numeric value (between 0 and 1) indicating the fraction of cells that survive and are captured after library preparation.
`num_total_reads`	A numeric value specifying the total number of reads generated by sequencing. This is used to estimate the `library_size`.
`mapping_efficiency`	A numeric value (between 0 and 1) indicating the fraction of reads that successfully map to the transcriptome.
`cells_per_grna`	A data frame specifying the number of cells per gRNA, with columns `grna_id`, `grna_target`, and `num_cells`.
`baseline_relative_expression_stats`	A data frame specifying the relative expression levels for each gene, with columns `response_id` and `relative_expression`.

`fold_change_mean`
: A numeric value indicating the mean fold change effect size for all gRNA-gene pairs (or a data frame with `grna_target` and `response_id` columns for per-pair values).

`fold_change_sd`
: A numeric value indicating the standard deviation of the fold change effect size for all gRNA-gene pairs (or a data frame with `grna_target` and `response_id` columns for per-pair values).

`num_planned_cells`
: A numeric value indicating the total planned number of cells before losses in library preparation.

`control_group`
: A character string specifying the control group, either `"complement"` or `"nt_cells"`. This is passed to `compute_power_posthoc`.

`umi_per_cell`
: A numeric value specifying the theoretical saturation level (in UMIs) for each cell.

`umi_variation`
: A numeric value controlling how overdispersion in UMIs per read is modeled.

`side`
: (Optional) A character string specifying the side of the test, either `"left"`, `"right"`, or `"both"`. Defaults to `"both"`.

`multiple_testing_method`
: (Optional) A character string specifying the multiple testing correction method, either `"BH"` or `"bonferroni"`. Defaults to `"BH"`.

`multiple_testing_alpha`
: (Optional) A numeric value (between 0 and 1) specifying the alpha level for multiple testing correction. Defaults to 0.1.

`cutoff`
: (Optional) A numeric value between 0 and 1 to use as the p-value cutoff. If `NULL`, the function determines it automatically using the specified `multiple_testing_method` and `multiple_testing_alpha`.

`discovery_pairs`
: A data frame specifying which gRNA-gene pairs to consider, with columns `grna_target` and `response_id`.

`n_nonzero_trt_thresh`
: (Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7.

`n_nonzero_cntrl_thresh`
: (Optional) An integer specifying the sceptre QC parameter of the same name; defaults to 7.

Value

A list with two elements:

- `individual_power`: A data frame with columns `grna_target`, `response_id`, and `power`, giving the power for each pair.
- `expected_num_discoveries`: A numeric value indicating the expected total number of discoveries.

rejection_computation *Compute the rejection probability.*

Description

Compute the rejection probability.

Usage

```
rejection_computation(mean_list, sd_list, side, cutoff)
```

Arguments

mean_list	Asymptotic mean of test statistic
sd_list	Asymptotic sd of test statistic
side	(Optional) A character string specifying the side of the test, either "left", "right", or "both"; defaults to "both"
cutoff	(Optional) A numeric value between 0 and 1 to use as the p-value cutoff

Value

The rejection probablity.

summary_h5_data *Obtain the summary statistics of the QC'd data.*

Description

Obtain the summary statistics of the QC'd data.

Usage

```
summary_h5_data(QC_data)
```

Arguments

QC_data	The QC'd data coming from the function obtain_qc_h5_data

Value

A named vector with the number of cells and average reads per cell.

var_nb *Variance of NB distribution*

Description

Variance of NB distribution

Usage

```
var_nb(mean, size)
```

Arguments

mean	mean gene expression.
size	size parameter.

Value

variance of NB distribution.

Index