Package 'cypress'

November 11, 2026

Type Package

Title Cell-Type-Specific Power Assessment

Version 1.8.0

Date 2024-05-27

License GPL-2 | GPL-3

Depends R(>= 4.4.0)

Imports stats, abind, sirt, MASS,TOAST, tibble, parallel, preprocessCore, SummarizedExperiment, TCA, PROPER, methods,dplyr, utils, RColorBrewer, graphics, edgeR, BiocParallel, checkmate, mvtnorm, DESeq2, rlang, e1071

Suggests knitr, rmarkdown, MatrixGenerics, htmltools, RUnit, BiocGenerics, BiocManager, BiocStyle, Biobase

Collate AllClasses.R AllGenerics.R CIBERSORT.R csRNA_seq_sim.R cypress_design.R cypress_wrapper.R cypressEmbedded.R metrics.R cypress-methods.R PlotTDR.R DE_arg.R TOAST_implement.R user.R util_datasim.R util_EST.R PlotFDC.R PlotPower.R

biocViews Software, GeneExpression, DataImport, RNASeq, Sequencing

Description CYPRESS is a cell-type-specific power tool. This package aims to perform power analysis for the cell-type-specific data. It calculates FDR, FDC, and power, under various study design parameters, including but not limited to sample size, and effect size. It takes the input of a SummarizeExperimental(SE) object with observed mixture data (feature by sample matrix), and the cell-type mixture proportions (sample by cell-type matrix). It can solve the cell-type mixture proportions from the reference free panel from TOAST and conduct tests to identify cell-type-specific differential expression (csDE) genes.

Encoding UTF-8

LazyData FALSE

VignetteBuilder knitr

URL https://github.com/renlyly/cypress

BugReports https://github.com/renlyly/cypress/issues

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

git_branch RELEASE_3_23

git_last_commit 7b2f22c

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Shilin Yu [aut, cre] (ORCID: <https://orcid.org/0000-0002-5494-1960>),
Guanqun Meng [aut],
Wen Tang [aut]

Maintainer Shilin Yu <sy597@georgetown.edu>

Contents

cypress-package 2
asd_nopropPower 3
asd_noprop_param 4
ASD_prop 5
cypress-class 5
cypress-methods 7
GSE60424Power 7
GSE60424_param 8
ibd_propPower 9
ibd_prop_param 10
plotFDC 11
plotPower 12
plotTDR 13
quickPower 14
simFromData 15
simFromParam 17

Index **19**

`cypress-package` *cypress: cell-type-specific power assessment*

Description

The **cypress** package is specifically designed to perform comprehensive cell-type-specific power assessment for differential expression using RNA-sequencing experiments. It accepts real Bulk RNAseq data as input for parameter estimation and simulation. The tool provides flexibility by allowing users to customize sample sizes, number of cell types, and effect sizes (log-fold change). Additionally, it computes statistical power, true discovery rate (TDR), and false discovery cost (FDC) under different scenarios as results.

Details

cypress is the first statistical tool to evaluate the power in cell-type-specific Differentially Expressed (csDE) genes detection experiments from a prospective way by letting researchers be flexible in tuning sample sizes, effect sizes, csDE genes percentage, total number of genetic features, type I error control, etc.

Value

cypress offers 3 options for simulation and power evaluation: `simFromData()` ,`simFromParam()` and `quickPower()`. If users have their own bulk RNA-seq count data, they can use the `simFromData()` function; otherwise, they can use `simFromParam()`, which uses one of the three sets of simulation parameters estimated from existing studies, to perform power evaluation under user-defined simulation settings. If users prefer to quickly examine the power evaluation results and do not want to run the simulation, they can use the `quickPower()` function to view results from our existing simulations. The output of these 3 functions is a S4 object including a list of simulation results under various experimental settings, including as statistical power, TDR, and FDC.

Once users have obtained an S4 object with a list of results output from either `simFromData()`, `simFromParam()` or `quickPower()`, they can use the following functions to generate basic evaluation plots: `plotPower()`, `plotTDR()` and `plotFDC()`.

Author(s)

Shilin Yu <sy597@georgetown.edu> Guanqun Meng <gxm324@case.edu> Wen Tang <wxt175@case.edu>

`asd_nopropPower` *Power calculation results From ASD data*

Description

Pre-calculated power evaluation results from Autism Spectrum Disorder (ASD) study. The results can be used to create plots using plot functions (`plotFDC`, `plotPower`, `plotTDR`).

Usage

```
data('quickPowerASD')
```

Format

A S4 object.

`ct_TDR_bio_smry` Cell-type-specific target TDR.

`TDR_bio_smry` Average target TDR.

`ct_PWR_bio_smry` Cell-type-specific target power.

`PWR_bio_smry` Average target power.

`PWR_strata_ct_bio_smry` Cell type specific target power by gene expression stratification.

`PWR_strata_bio_smry` Average target power by gene expression stratification.

`ct_FDC_bio_smry` Cell type specific target FDC.

`FDC_bio_smry` Average target FDC.

Value

One S4 object.

Examples

```
data('quickPowerASD')
```

`asd_noprop_param` *Simulation parameters estimated from ASD study*

Description

An S4 object that stores parameter estimates associated with the Autism Spectrum Disorder (ASD) dataset. This object contains a variety of numerical vectors and matrices representing different statistical parameters used in the simulation.

Usage

```
data('quickParaASD')
```

Format

Simulation parameters for `simFromParam` function.

`health_lm_mean` A numeric vector containing the log-mean parameter estimates for each cell type from healthy samples.

`health_lm_mean_d` A matrix containing the variance-covariance estimates of log-mean parameters across cell types from healthy samples.

`lod_m` A numeric vector containing the log-dispersion parameter estimates for each cell type from healthy samples.

`lod_d` A matrix containing the variance-covariance estimates of log-dispersion parameters across cell types from healthy samples.

`health_alpha` A numeric vector of the estimated alpha parameter used to simulate cell type proportions for healthy samples.

`case_alpha` A numeric vector of the estimated alpha parameter used to simulate cell type proportions for case samples.

Value

One S4 object.

Examples

```
data('quickParaASD')
```

ASD_prop *SimFromData example raw input data*

Description

ASD_prop is an example of SummarizedExperiment (SE) object input for the simFromData function. It contains the following elements:

counts A gene expression value dataset from Autism Spectrum Disorder (ASD) study, in the form of raw read counts, 29674 genes by 48 samples, with 24 cases and 24 controls

colData Sample meta-data. The first column is the group status (i.e. case/ctrl), the second column is the subject ID. The remaining are the cell type proportions of all samples.

Usage

```
data(ASD_prop_se)
```

Format

SE object.

Value

One SE object.

Examples

```
data(ASD_prop_se)
```

cypress-class *Classes for cypress Package*

Description

The cypress_out and est_out classes are custom S4 classes in the **cypress** package. both classes are designed as a comprehensive container for various types of analysis results.

Value

Class for cypress.

cypress_out

Description: The `cypress_out` class is a S4 class in the **cypress** package. This class is customized to better present results and use for cypress plot functions.

Slots:

`ct_TDR_bio_smry` Cell type specific target TDR

`TDR_bio_smry` Target TDR

`ct_PWR_bio_smry` Cell type specific target power

`PWR_bio_smry` Target power

`PWR_strata_bio_smry` Target power by gene expression stratification

`PWR_strata_ct_bio_smry` Cell type specific target power by gene expression stratification

`ct_FDC_bio_smry` Cell type specific target FDC

`FDC_bio_smry` Target FDC.

est_out

Description: The `est_out` class is designed to output the parameter estimated results, providing a structured representation of results.

Slots:

`health_alpha` Control group proportion simulation parameter.

`case_alpha` Case group proportion simulation parameter.

`health_lmean_m` Mean of genetic distribution mean for each cell.

`health_lmean_d` Var/cov of genetic distribution mean among cell types

`lod_m` Mean of genetic distribution dispersion for each cell.

`lod_d` Var/cov of genetic distribution dispersion among cell types.

`sample_CT_prop` Matrix of sample Cell type proportions.

`genename` Gene Name.

`samplename` Sample Name.

`CTname` Cell type names.

`dimensions_Z_hat_ary` dimensions for the Z hat array.

Author(s)

Shilin Yu <sy597@georgetown.edu>

Examples

```
data(quickParaGSE60424)
```

cypress-methods *Get a slot from cypress output*

Description

Accessor function for getting or replace slots. Show methods for cypress object.

Usage

```
getcypress(object, name)
setcypress(object, name, value)
```

Arguments

object	object from cypress.
name	name of the slot in cypress object.
value	value of the slot in cypress object.

Value

Methods for cypress.

Examples

```
data(quickPowerIBD)
getcypress(ibd_propPower, "ct_TDR_bio_smry")
```

GSE60424Power *Power calculation results from immune-related disease (IAD) study (GSE60424)*

Description

Pre-calculated power evaluation results from immune-related disease (IAD) study (GSE60424). The results can be used to create plots using plot functions (plotFDC, plotPower, plotTDR).

Usage

```
data('quickPowerGSE60424')
```

Format

A S4 object.

`ct_TDR_bio_smry` Cell-type-specific target TDR.

`TDR_bio_smry` Average target TDR.

`ct_PWR_bio_smry` Cell-type-specific target power.

`PWR_bio_smry` Average target power.

`PWR_strata_ct_bio_smry` Cell type specific target power by gene expression stratification.

`PWR_strata_bio_smry` Average target power by gene expression stratification.

`ct_FDC_bio_smry` Cell type specific target FDC.

`FDC_bio_smry` Average target FDC.

Value

One S4 object.

Examples

```
data('quickPowerGSE60424')
```

GSE60424_param *Simulation parameters estimated from immune-related disease (IAD) study (GSE60424)*

Description

An S4 object that stores simulation parameter estimated from the immune-related disease (IAD) study (GSE60424). This object contains a variety of numerical vectors and matrices representing different statistical parameters used in the simulation. The patients were drawn from healthy subjects from the immune-related diseases study.

Usage

```
data('quickParaGSE60424')
```

Format

Simulation parameters for `simFromParam` function.

`health_lm_mean` A numeric vector containing the log-mean parameter estimates for each cell type from healthy samples.

`health_lm_mean_d` A matrix containing the variance-covariance estimates of log-mean parameters across cell types from healthy samples.

`lod_m` A numeric vector containing the log-dispersion parameter estimates for each cell type from healthy samples.

`lod_d` A matrix containing the variance-covariance estimates of log-dispersion parameters across cell types from healthy samples.

`health_alpha` A numeric vector of the estimated alpha parameter used to simulate cell type proportions for healthy samples.

`case_alpha` A numeric vector of the estimated alpha parameter used to simulate cell type proportions for case samples.

Value

One S4 object.

Examples

```
data('quickParaGSE60424')
```

`ibd_propPower` *Power calculation results from pediatric inflammatory bowel disease (IBD) study (GSE57945)*

Description

Pre-calculated power evaluation results from pediatric inflammatory bowel disease (IBD) study (GSE57945). The results can be used to create plots using plot functions (`plotFDC`, `plotPower`, `plotTDR`).

Usage

```
data('quickPowerIBD')
```

Format

A S4 object.

`ct_TDR_bio_smry` Cell-type-specific target TDR.

`TDR_bio_smry` Average target TDR.

`ct_PWR_bio_smry` Cell-type-specific target power.

`PWR_bio_smry` Average target power.

`PWR_strata_ct_bio_smry` Cell type specific target power by gene expression stratification.

`PWR_strata_bio_smry` Average target power by gene expression stratification.

`ct_FDC_bio_smry` Cell type specific target FDC.

`FDC_bio_smry` Average target FDC.

Value

One S4 object.

Examples

```
data('quickPowerIBD')
```

`ibd_prop_param` *Simulation parameters estimated from pediatric inflammatory bowel disease (IBD) study (GSE57945)*

Description

An S4 object that stores simulation parameter estimated from the inflammatory bowel disease (IBD) study (GSE57945). This object contains a variety of numerical vectors and matrices representing different statistical parameters used in the simulation. The patients were drawn from healthy subjects from the immune-related diseases study.

Usage

```
data('quickParaIBD')
```

Format

Simulation parameters for `simFromParam` function.

`health_lm_mean` A numeric vector containing the log-mean parameter estimates for each cell type from healthy samples.

`health_lm_mean_d` A matrix containing the variance-covariance estimates of log-mean parameters across cell types from healthy samples.

`lod_m` A numeric vector containing the log-dispersion parameter estimates for each cell type from healthy samples.

`lod_d` A matrix containing the variance-covariance estimates of log-dispersion parameters across cell types from healthy samples.

`health_alpha` A numeric vector of the estimated alpha parameter used to simulate cell type proportions for healthy samples.

`case_alpha` A numeric vector of the estimated alpha parameter used to simulate cell type proportions for case samples.

Value

One S4 object.

Examples

```
data('quickParaIBD')
```

plotFDC *plotFDC: Generate FDC results figure*

Description

Plot false discovery cost results. This function plots false discovery cost results in a 2x1 panel. The illustration of each plot from left to right:

1: False discovery cost(FDC) by effect size, each line represents cell type. Sample size to be fixed at 10 if sample_size=10.

2: False discovery cost(FDC) by top effect size, each line represents sample size. FDC was the average value across cell types.

Arguments

simulation_results	A list of results produced by power evaluation functions.
sample_size	A numerical value indicating which sample size to be fixed. For example, 10 means when plotting the relationship between FDC and effect size, we fixed the scenario of sample size at 10. Default is 10.

Value

This function does not return a value. It generates a two-panel plot visualizing the false discovery cost (FDC) results. The first panel shows the FDC by effect size for each cell type at a fixed sample size (default is 10). The second panel illustrates the FDC by the top effect sizes, with each line representing a different sample size, averaged across cell types.

Author(s)

Wen Tang <wxt175@case.edu> Shilin Yu <sy597@georgetown.edu>

Examples

```
 data(quickPowerGSE60424)
### Plot power results
 plotFDC(GSE60424Power,sample_size=10)
```

plotPower *plotPower: Generate statistical power results figure*

Description

This function plots all statistical power measurements in a 2x3 panel. The illustration of each plot from left to right and from up to bottom is as follows:

1: Statistical power by effect size, each line represents sample size. Statistical power was the average value across cell types.

2: Statistical power by effect size, each line represents cell type. Sample size is fixed at 10 if sample_size=10.

3: Statistical power by sample size, each line represents cell type. Effect size is fixed at 1 if effect.size=1.

4: Statistical power by strata, each line represents cell type. Sample size is fixed at 10 and effect size is fixed at 1 if sample_size=10 and effect.size=1.

5: Statistical power by strata, each line represents sample size. Statistical power was the average value across cell types and effect size is fixed at 1 if effect.size=1.

6: Statistical power by strata, each line represents effect size. Statistical power was the average value across cell types and sample size is fixed at 10 if sample_size=10.

Arguments

`simulation_results`
: A list of results produced by power evaluation functions.

`effect.size`
: A numerical value indicating which effect size is to be fixed. For example, 1 means when plotting the relationship between power and strata, we fixed the scenario of log fold change at 1. The default is 1.

`sample_size`
: A numerical value indicating which sample size to be fixed. For example, 10 means when plotting the relationship between power and strata, we fixed the scenario of sample size at 10. The default is 10.

Value

This function generates a 2x3 panel plot visualizing various statistical power measurements, but does not return a programmable value. Each panel displays power metrics under different conditions such as effect size, sample size, and stratification, with lines representing either sample size, cell type, or effect size.

Author(s)

Wen Tang <wxt175@case.edu> Shilin Yu <sy597@georgetown.edu>

Examples

```
data(quickPowerGSE60424)
### Plot power results
plotPower(GSE60424Power,effect.size=1,sample_size=10)
```

---

plotTDR *plotTDR: Generate TDR results figure*

Description

This function plots all true discovery rate measurements in a 2x2 panel. The illustration of each plot is as follows:

1: True discovery rate(TDR) by top-rank genes, each line represents cell type. Sample size to be fixed at 10 and effect size to be fixed at 1 if sample_size=10 and effect.size=1.

2: True discovery rate(TDR) by top rank genes, each line represents effect size. TDR was the average value across cell types and sample size is fixed at 10 if sample_size=10.

3: True discovery rate(TDR) by top rank genes, each line represents sample size. TDR was the average value across cell types and effect size is fixed at 1 if effect.size=1.

4: True discovery rate(TDR) by effect size, each line represents sample size. TDR was calculated under the scenario of top rank gene equals 350.

Arguments

`simulation_results`	A list of results produced by power evaluation functions.
`effect.size`	A numerical value indicating which effect size is to be fixed. For example, 1 means when plotting the relationship between TDR and top rank genes for cell types or sample size, we fixed the scenario of log2 fold change at 1. The default is 1.
`sample_size`	A numerical value indicating which sample size to be fixed. For example, 10 means when plotting the relationship between TDR and top rank genes for cell types or effect size, we fixed the scenario of sample size at 10. The default is 10.

Value

This function creates a 2x2 panel plot showcasing various true discovery rate (TDR) measurements but does not return any values for further programmatic use. Each panel displays TDR analyses based on top rank genes, with lines representing different cell types, effect sizes, or sample sizes under specific conditions.

Author(s)

Wen Tang <wxt175@case.edu> Shilin Yu <sy597@georgetown.edu>

Examples

```
 data(quickPowerGSE60424)
### Plot power results
 plotTDR(GSE60424Power,effect.size=1,sample_size=10)
```

`quickPower` *Obtain pre-calculated results from three available datasets*

Description

This function quickly outputs pre-calculated power evaluation results from three datasets: (IAD, IBD, and ASD). The obtained results can be used to create plots from plot functions.

Usage

```
quickPower(data = "IAD")
```

Arguments

`data`	A character string specifying the dataset to be retrieved. Options are 'IAD', 'IBD', and 'ASD'.

Details

- `IAD`: Whole transcriptome signatures of 6 immune cell subsets. The patients were drawn from subjects with a range of immune-related diseases.
- `IBD`: Inflammatory Bowel Disease
- `ASD`: Autism Spectrum Disorder.

Value

`ct_TDR_bio_smry`
: Cell-type-specific target TDR.

`TDR_bio_smry`
: Average target TDR.

`ct_PWR_bio_smry`
: Cell-type-specific target power.

`PWR_bio_smry`
: Average target power.

`PWR_strata_ct_bio_smry`
: Cell type specific target power by gene expression stratification.

`PWR_strata_bio_smry`
: Average target power by gene expression stratification.

`ct_FDC_bio_smry`
: Cell type specific target FDC.

`FDC_bio_smry`
: Average target FDC.

Author(s)

Shilin Yu <sy597@georgetown.edu> Guanqun Meng <gxm324@case.edu>

Examples

```
# library(cypress)
Quick_power <- quickPower(data = "IAD")
```

---

simFromData *Power calculation results on input data*

Description

This function conducts simulations with various user-defined study design parameters, Users will need to provide SE object bulk data for parameter estimation purposes.

Usage

```
simFromData(INPUTdata = NULL, CT_index = NULL, CT_unk = FALSE,
            n_sim = 3, n_gene = 30000, DE_pct = 0.05,
            ss_group_set = c(10, 20, 50, 100),
            lfc_set = c(0, 0.5, 1, 1.5, 2),
            lfc_target = 0.5, fdr_thred = 0.1,
            DEmethod = "TOAST",BPPARAM=bpparam())
```

Arguments

INPUTdata	The input SE (SummarizedExperiment) object should contain a count matrix, study design, and an optional cell type proportion matrix. The study design should have a column named 'disease', where the control by 1 and the case is indicated by 2. If provided, the cell type proportion matrix should sum to 1 for each sample. The cell type proportion matrix is optional, the CT_unk should be True if the user did not provide this matrix
CT_index	Column index for cell types proportion matrix in Coldata, the input can also be a single number (>3) when the CT_unk is T.
CT_unk	Logical flag indicating whether unknown cell types are present. Defaults to FALSE.
n_sim	The total number of iterations users wish to conduct. Default to 3. In simulation results, it is set to 20.
n_gene	Total number of genetic features users with to conduct. Default to 30000. Must be greater than or equal to 1000.
DE_pct	Percentage of DEG on each cell type. Default to 0.05.
ss_group_set	Sample sizes per group users wish to simulate. The length should be less than or equal to 5. Default to 10,20,50.

`lfc_set` Effect sizes users wish to simulate. The length should be less than or equal to 5. Default to 0,0.5,1,1.5.

`lfc_target` Target effect size, should be greater than or equal to 0. The absolute LFC lower than this values will be treated as None-DEGs. Default to 0.5

`fdr_thred` Adjusted p value threshold. The parameter value should be within the range (0, 1). Default to 0.1

`DEmethod` Differential expression (DE) methods available include 'TOAST', 'DESeq2', and 'CeDAR'. The default method is 'TOAST'

`BPPARAM` An instance of `BiocParallelParam` class, e.g. MulticoreParam, SnowParam, SerialParam, to facilitate parallel computing. If using Unix, MulticoreParam is recommended. Customized options within `BiocParallelParam` class is allowed. If not specified, the default back-end is retrieve.

Details

One `SummarizedExperiment` object containing the following elements:

`counts` A gene expression value dataset

`colData` Sample meta-data. The first column is the group status (i.e. case/ctrl) named as 'disease', and the second column is the subject ID. The remaining are the cell type proportions of all samples. The user could also input the Column index for cell types proportion matrix in Coldata. Example: CT_index= 3:8

Value

`ct_TDR_bio_smry`
Cell-type-specific target TDR.

`TDR_bio_smry` Average target TDR.

`ct_PWR_bio_smry`
Cell-type-specific target power.

`PWR_bio_smry` Average target power.

`PWR_strata_ct_bio_smry`
Cell type specific target power by gene expression stratification.

`PWR_strata_bio_smry`
Average target power by gene expression stratification.

`ct_FDC_bio_smry`
Cell type specific target FDC.

`FDC_bio_smry` Average target FDC.

Author(s)

Shilin Yu <sy597@georgetown.edu> Guanqun Meng <gxm324@case.edu>

Examples

```
data(ASD_prop_se)
result <- simFromData(INPUTdata = ASD_prop, CT_index = (seq_len(6) + 2),
CT_unk = FALSE, n_sim = 2,n_gene = 1000, DE_pct = 0.05,
 ss_group_set = c(8,10),  lfc_set = c(1, 1.5))
```

`simFromParam` *Power calculation results on pre-calculated parameters*

Description

This function conducts simulations with various user-defined study design parameters, including but not limited to sample size, and log fold change.)

Usage

```
simFromParam(n_sim = 3, n_gene = 30000, DE_pct = 0.05,
             ss_group_set = c(10, 20, 50, 100),
             lfc_set = c(0, 0.5, 1, 1.5, 2),
             sim_param = "IAD",
             lfc_target = 0.5, fdr_thred = 0.1,
             DEmethod = "TOAST", BPPARAM=bpparam())
```

Arguments

`n_sim`	The total number of iterations users wish to conduct. Default to 3. In simulation results, it is set to 20.
`n_gene`	Total number of genetic features users with to conduct. Default to 30000. Must be greater than or equal to 1000.
`DE_pct`	Percentage of DEG on each cell type. Default to 0.05.
`ss_group_set`	Sample sizes per group users wish to simulate. The length should be less than or equal to 5. Default to 10,20,50.
`lfc_set`	effect sizes users wish to simulate. The length should be less than or equal to 5. Default to 0,0.5,1,1.5.
`sim_param`	Users specify which embedded simulation parameters they wish to use. By default set to 'IAD', which is a cell line specific data. Other options include 'IBD' data, and 'ASD' data
`lfc_target`	Target effect size, should be greater than or equal to 0. The absolute LFC lower than this value will be treated as None-DEGs. Default to 0.5
`fdr_thred`	Adjusted p value threshold. The parameter value should be within the range (0, 1). Default to 0.1
`DEmethod`	Differential expression (DE) methods available include 'TOAST', 'DESeq2', and 'CeDAR'. The default method is 'TOAST'

BPPARAM An instance of BiocParallelParam class, e.g. MulticoreParam, SnowParam, SerialParam, to facilitate parallel computing. If using Unix, MulticoreParam is recommended. Customized options within BiocParallelParam class is allowed. If not specified, the default back-end is retrieve.

Details

- GSE60424: Immune-related disease (IAD) study. Whole transcriptome signatures of 6 immune cell subsets. The patients were drawn from subjects with a range of immune-related diseases.
- IBD: data in pediatric inflammatory bowel disease(IBD) study
- ASD: data in a large autism spectrum disorder (ASD) study

Value

ct_TDR_bio_smry
: Cell-type-specific target TDR.

TDR_bio_smry
: Average target TDR.

ct_PWR_bio_smry
: Cell-type-specific target power.

PWR_bio_smry
: Average target power.

PWR_strata_ct_bio_smry
: Cell type specific target power by gene expression stratification.

PWR_strata_bio_smry
: Average target power by gene expression stratification.

ct_FDC_bio_smry
: Cell type specific target FDC.

FDC_bio_smry
: Average target FDC.

Author(s)

Shilin Yu <sy597@georgetown.edu> Guanqun Meng <gxm324@case.edu>

Examples

```
data(quickParaGSE60424)
result <- simFromParam(sim_param="IAD",n_sim = 2,DE_pct = 0.05,n_gene = 1000,
                         ss_group_set = c(8, 10),
                         lfc_set = c(1, 1.5),
                         lfc_target = 0.5, fdr_thred = 0.1)
```

Index

∗ **datasets**
    asd_noprop_param, 4
    asd_nopropPower, 3
    ASD_prop, 5
    GSE60424_param, 8
    GSE60424Power, 7
    ibd_prop_param, 10
    ibd_propPower, 9
∗ **package**
    cypress-package, 2

AllClasses *(*cypress-class*)*, 5
asd_noprop_param, 4
asd_nopropPower, 3
asd_param *(*asd_noprop_param*)*, 4
ASD_prop, 5
ASD_prop_se *(*ASD_prop*)*, 5
asdPower *(*asd_nopropPower*)*, 3

cypress *(*cypress-package*)*, 2
cypress-class, 5
cypress-methods, 7
cypress-package, 2
cypress_out-class *(*cypress-class*)*, 5

est_out-class *(*cypress-class*)*, 5

getcypress *(*cypress-methods*)*, 7
getcypress,cypress_out-method
        *(*cypress-methods*)*, 7
getcypress,design_in-method
        *(*cypress-methods*)*, 7
getcypress,est_out-method
        *(*cypress-methods*)*, 7
GSE60424_param, 8
GSE60424Power, 7

iad_param *(*GSE60424Power*)*, 7
ibd_param *(*ibd_prop_param*)*, 10
ibd_prop_param, 10
ibd_propPower, 9
ibdPower *(*ibd_propPower*)*, 9

plotFDC, 11
plotPower, 12
plotTDR, 13

quickParaASD *(*asd_noprop_param*)*, 4
quickParaGSE60424 *(*GSE60424_param*)*, 8
quickParaIBD *(*ibd_prop_param*)*, 10
quickPower, 14
quickPowerASD *(*asd_nopropPower*)*, 3
quickPowerGSE60424 *(*GSE60424Power*)*, 7
quickPowerIBD *(*ibd_propPower*)*, 9

setcypress *(*cypress-methods*)*, 7
setcypress,cypress_out-method
        *(*cypress-methods*)*, 7
setcypress,design_in-method
        *(*cypress-methods*)*, 7
setcypress,est_out-method
        *(*cypress-methods*)*, 7
show,cypress_out-method
        *(*cypress-methods*)*, 7
show,est_out-method *(*cypress-methods*)*, 7
simFromData, 15
simFromParam, 17