

# Package ‘CyTOFpower’

December 4, 2025

**Type** Package

**Title** Power analysis for CyTOF experiments

**Version** 1.17.0

**Description** This package is a tool to predict the power of CyTOF experiments in the context of differential state analyses. The package provides a shiny app with two options to predict the power of an experiment:  
i. generation of in-silico CyTOF data, using users input ii. browsing in a grid of parameters for which the power was already precomputed.

**biocViews** FlowCytometry, SingleCell, CellBiology, StatisticalMethod, Software

**License** LGPL-3

**Encoding** UTF-8

**Depends** R (>= 4.1)

**Imports** CytoGLMM, diffcyt, DT, dplyr, ggplot2, magrittr, methods, rlang, stats, shiny, shinyFeedback, shinyjs, shinyMatrix, SummarizedExperiment, tibble, tidy

**RoxygenNote** 7.1.2

**VignetteBuilder** knitr

**Suggests** testthat (>= 3.0.0), BiocStyle, knitr

**Config/testthat/edition** 3

**git\_url** <https://git.bioconductor.org/packages/CyTOFpower>

**git\_branch** devel

**git\_last\_commit** 7886a64

**git\_last\_commit\_date** 2025-10-29

**Repository** Bioconductor 3.23

**Date/Publication** 2025-12-04

**Author** Anne-Maud Ferreira [cre, aut] (ORCID:  
<<https://orcid.org/0000-0002-4749-746X>>),  
Catherine Blish [aut],  
Susan Holmes [aut]

**Maintainer** Anne-Maud Ferreira <[anne-maud.ferreira@stanford.edu](mailto:anne-maud.ferreira@stanford.edu)>

Contents

|                                                                           |           |
|---------------------------------------------------------------------------|-----------|
| compute_effectsize . . . . .                                              | 2         |
| compute_pwr . . . . .                                                     | 3         |
| compute_variance . . . . .                                                | 3         |
| CytoFpower . . . . .                                                      | 4         |
| function_apply_modelcomputations_modelchoice . . . . .                    | 4         |
| function_apply_onesimulation_withmarkerinfo . . . . .                     | 5         |
| function_check_nbDEmarkers . . . . .                                      | 6         |
| function_check_nbmarkers . . . . .                                        | 6         |
| function_compute_diffcyt_features . . . . .                               | 7         |
| function_create_mock_dataset_withmarkerinfo . . . . .                     | 7         |
| function_DEmarkers_sup_nbmarkers . . . . .                                | 8         |
| function_desigmat_contrast_diffcytDSlimma_fixedeffect . . . . .           | 9         |
| function_desigmat_contrast_diffcytDSlimma_randomeffect . . . . .          | 9         |
| function_extract_marker_names . . . . .                                   | 10        |
| function_formula_contrast_diffcytDSLMM_randomeffect . . . . .             | 10        |
| function_is_data_paired . . . . .                                         | 11        |
| function_names_DE_markers . . . . .                                       | 12        |
| function_run_bootstrapcytoGLMM . . . . .                                  | 12        |
| function_run_cytoGLMM . . . . .                                           | 13        |
| function_run_diffcytDSlimma . . . . .                                     | 13        |
| function_run_diffcytDSLMM . . . . .                                       | 14        |
| function_run_diffcyt_full_pipeline . . . . .                              | 15        |
| function_summary_results_models . . . . .                                 | 15        |
| function_to_compute_model_computation_onesimulation_modelchoice . . . . . | 16        |
| function_to_transform_data . . . . .                                      | 17        |
| function_value_onemarker . . . . .                                        | 17        |
| <b>Index</b>                                                              | <b>19</b> |

---

|                    |                            |
|--------------------|----------------------------|
| compute_effectsize | <i>Compute effect size</i> |
|--------------------|----------------------------|

---

Description

Compute effect size

Usage

compute\_effectsize(raw\_data\_lg)

Arguments

raw\_data\_lg      data.frame, cells values in long format.

Details

Compute observed Cohen’s effect size and observed fold change.

**Value**

data.frame, Cohen's effect size and fold change that were observed in the data for each marker.

---

compute\_pwr

*Compute power*

---

**Description**

Compute power

**Usage**

```
compute_pwr(model_values, alpha = 0.05)
```

**Arguments**

|              |                                     |
|--------------|-------------------------------------|
| model_values | data.frame, output of run_models(). |
| alpha        | numeric, significance level.        |

**Details**

Compute the power based on the model values.

**Value**

numeric, power.

---

compute\_variance

*Compute variance*

---

**Description**

Compute variance

**Usage**

```
compute_variance(raw_data_lg)
```

**Arguments**

|             |                                          |
|-------------|------------------------------------------|
| raw_data_lg | data.frame, cells values in long format. |
|-------------|------------------------------------------|

**Details**

Compute the observed variance in the data.

**Value**

numeric, observed variance mean.

---

CyTOFpower

*CyTOFpower: R-package to predict the power of a CyTOF experiment*


---

### Description

This package provides functions and a shiny app to predict the power of a CyTOF experiment.

Interactive shiny app to predict the power of a CyTOF experiment.

### Usage

```
CyTOFpower()
```

### Value

Interactive shiny app.

### Examples

```
# Launch the shiny app
if (interactive()) {
  CyTOFpower()
}
```

---

function\_apply\_modelcomputations\_modelchoice

*Run DS tests for one simulation.*


---

### Description

Run DS tests for one simulation.

### Usage

```
function_apply_modelcomputations_modelchoice(
  list_combined_output,
  model = c("cytoglm", "cytoglm", "testDS_limma_random", "testDS_limma_fixed",
    "testDS_lmm")
)
```

### Arguments

|                      |                                                                                               |
|----------------------|-----------------------------------------------------------------------------------------------|
| list_combined_output | list, of simulated data (output of the function_apply_onesimulation_withmarkerinfo function). |
| model                | vector, name(s) of models to test.                                                            |

**Details**

Wrapper to run the models through the different simulations.

**Value**

data.frame of results for each simulation.

---

function\_apply\_onesimulation\_withmarkerinfo

*One simulation with markers NB information.*

---

**Description**

One simulation with markers NB information.

**Usage**

```
function_apply_onesimulation_withmarkerinfo(variation)
```

**Arguments**

|           |                                                                                                                                 |
|-----------|---------------------------------------------------------------------------------------------------------------------------------|
| variation | list, list of list containing the different input parameter variations @describeIn function_create_mock_dataset_withmarkerinfo. |
|-----------|---------------------------------------------------------------------------------------------------------------------------------|

**Details**

Apply one simulation for a given parameter combination number times (iterations), when we have prior information about the markers distribution parameters (mean and dispersion of the negative binomial).

**Value**

list of simulated data. Each list member contains 5 slots: - variation: the variation of input parameters which has been used in input; - df\_info: experimental information (donor IDs, group IDs, samples IDs); - DE\_markers\_names: name of the differentially expressed markers; - ls\_mock\_data: list of data.frames, each data.frame being one simulation of the cell values using the input parameters provided (list is length nb\_sim).

function\_check\_nbDEmarkers

*Number of DE markers greater than 1.*

---

**Description**

Number of DE markers greater than 1.

**Usage**

```
function_check_nbDEmarkers(nb_DEmarker)
```

**Arguments**

nb\_DEmarker      numeric, number of differentially expressed markers.

**Details**

Function to check that the number of DE markers is greater than 1.

**Value**

error message

---

function\_check\_nbmarkers

*Number of markers greater than 2.*

---

**Description**

Number of markers greater than 2.

**Usage**

```
function_check_nbmarkers(nb_marker)
```

**Arguments**

nb\_marker      numeric, total number of markers.

**Details**

Function to check that the number of markers is greater than 2.

**Value**

error message if the total number of markers is lower than 3.

---

```
function_compute_diffcyt_features
```

*Compute cells counts and medians.*

---

### Description

Compute cells counts and medians.

### Usage

```
function_compute_diffcyt_features(mock_flowset)
```

### Arguments

mock\_flowset      data.frame, cell values for each marker.

### Details

Function to calculate features for diffcyt package models.

### Value

list with cell counts and medians for each markers.

---

```
function_create_mock_dataset_withmarkerinfo
```

*Compute simulated cell values for one simulation with markers NB information.*

---

### Description

Compute simulated cell values for one simulation with markers NB information.

### Usage

```
function_create_mock_dataset_withmarkerinfo(variation)
```

### Arguments

variation      list, list of data.frames containing the different variable input paramaters to generate the data: - marker\_name: name of the marker (character); - nb\_donor: number of donors; - rho: fold change; - subject\_effect: standard deviation for the normal distribution from which the donor's means will be drawn; - mu0: mean of the negative binomial for the gamma distribution from which the means of the different donor will be drawn; - dispersion: dispersion of the negative binomial from which the DE marker's cell values will be drawn; - nb\_cell\_per\_sample: number of cells per sample.

**Details**

Function to compute the simulated cell values using a combination of variable parameters, when we have prior information about the markers distribution parameters (mean and dispersion of the negative binomial).

**Value**

list with 4 slots: - df\_info: data.frame of experimental information; - DEmarkers\_names: vector of DE marker names; - raw\_data: data.frame of raw cell values; - data: data.frame of transformed cell values.

---

```
function_DEmarkers_sup_nbmarkers
```

*Check on the number of DE markers.*

---

**Description**

Check on the number of DE markers.

**Usage**

```
function_DEmarkers_sup_nbmarkers(nb_markers, nb_DEmarker)
```

**Arguments**

nb\_markers      numeric, total number of markers.

nb\_DEmarker     numeric, number of differentially expressed markers.

**Details**

Function to check that number of DE markers greater than number of markers.

**Value**

error message if the number of DE markers is greater than the total number of markers.

---

```
function_desigmat_contrast_diffcytDSLimma_fixedeffect
```

*Design and contrast matrices for diffcyt-DS-limma with fixed effect.*

---

**Description**

Design and contrast matrices for diffcyt-DS-limma with fixed effect.

**Usage**

```
function_desigmat_contrast_diffcytDSLimma_fixedeffect(df_experiment_info)
```

**Arguments**

df\_experiment\_info

data.frame, information about the experiment (donor IDs, group IDs and sample IDs).

**Details**

Function to create the design matrix and contrast for the diffcyt-DS-limma model with fixed effect.

**Value**

list with 3 slots: - design\_matrix: the design matrix; - contrast: the contrast matrix; - effect: the specification of the "fixed" effect.

---

```
function_desigmat_contrast_diffcytDSLimma_randomeffect
```

*Design and contrast matrices for diffcyt-DS-limma with random effect.*

---

**Description**

Design and contrast matrices for diffcyt-DS-limma with random effect.

**Usage**

```
function_desigmat_contrast_diffcytDSLimma_randomeffect(df_experiment_info)
```

**Arguments**

df\_experiment\_info

data.frame, information about the experiment (donor IDs, group IDs and sample IDs).

**Details**

Function to create the design matrix and contrast for the diffcyt-DS-limma model with random effect from the diffcyt package.

**Value**

list with 3 slots: - design\_matrix: the design matrix; - contrast: the contrast matrix; - effect: the specification of the "random" effect.

---

function\_extract\_marker\_names

*Extract marker names.*

---

**Description**

Extract marker names.

**Usage**

```
function_extract_marker_names(mock_dataset)
```

**Arguments**

mock\_dataset      data.frame, containing the cell values for each marker.

**Details**

Function to extract the marker names.

**Value**

vector of marker names.

---

function\_formula\_contrast\_diffcytDSLMM\_randomeffect

*Formula and contrast matrix for diffcyt-DS-LMM with random effect.*

---

**Description**

Formula and contrast matrix for diffcyt-DS-LMM with random effect.

**Usage**

```
function_formula_contrast_diffcytDSLMM_randomeffect(df_experiment_info)
```

**Arguments**

df\_experiment\_info  
data.frame, information about the experiment (donor IDs, group IDs and sample IDs).

**Details**

Function to create formula and contrast for diffcyt-DS-LMM with random effect.

**Value**

list with 3 slots: - formula: the formula for the model; - contrast: the contrast matrix.

---

function\_is\_data\_paired  
*Data paired.*

---

**Description**

Data paired.

**Usage**

```
function_is_data_paired(data)
```

**Arguments**

data                      data.frame, experimental information containing (sample IDS, donor IDS).

**Details**

Function to check if the data is paired.

**Value**

logical, TRUE is the data are paired - FALSE if the data are not paired

---

`function_names_DE_markers`*Generate name of the DE markers.*

---

**Description**

Generate name of the DE markers.

**Usage**

```
function_names_DE_markers(total_nb_marker, nb_DE_marker)
```

**Arguments**

|                              |                                   |
|------------------------------|-----------------------------------|
| <code>total_nb_marker</code> | numeric, total number of markers. |
| <code>nb_DE_marker</code>    | numeric, number of DE markers.    |

**Details**

Function to generate the name of the DE markers.

**Value**

vector of marker names.

---

`function_run_bootstrapcytoGLMM`*Run cytoglm.*

---

**Description**

Run cytoglm.

**Usage**

```
function_run_bootstrapcytoGLMM(mock_dataset, nb_bootstrap = 500)
```

**Arguments**

|                           |                                                                               |
|---------------------------|-------------------------------------------------------------------------------|
| <code>mock_dataset</code> | data.frame, cell values for each marker.                                      |
| <code>nb_bootstrap</code> | numeric, number of bootstrap (by defaults <code>nb_bootstrap = 1000</code> ). |

**Details**

Function to run the Generalized Linear Model with Bootstrap, from the CytoGLMM package.

**Value**

list of 3 slots: - model\_fit: fit of the model; - plot: plot of the effects; - results\_summary: data.frame containing the results of the models for each marker.

---

function\_run\_cytoGLMM    *Run cytoglmm*

---

**Description**

Run cytoglmm

**Usage**

```
function_run_cytoGLMM(mock_dataset)
```

**Arguments**

mock\_dataset    data.frame, cell values for each marker.

**Details**

Function to run GLMM model from CytoGLMM package.

**Value**

list of 3 slots: - model\_fit: fit of the model; - plot: plot of the effects; - results\_summary: data.frame containing the results of the models for each marker.

---

function\_run\_diffcytDSlimma  
                              *Run diffcyt-DS-limma model.*

---

**Description**

Run diffcyt-DS-limma model.

**Usage**

```
function_run_diffcytDSlimma(  
  ls_desigmat_contrast,  
  df_experiment_info,  
  ls_features  
)
```

**Arguments**

`ls_designmat_contrast` list, with design and contrast matrices.

`df_experiment_info` data.frame, information about the experiment (donor IDs, group IDs and sample IDs).

`ls_features` list, with cell counts and medians for each markers.

**Details**

Function to run diffcyt-DS-limma model from the diffcyt package.

**Value**

list with 2 slots: - `model_fit`: the model fit; - `result_summary`: results of the model for each marker.

---

`function_run_diffcytDSLMM`  
*Run diffcyt-DS-LMM model.*

---

**Description**

Run diffcyt-DS-LMM model.

**Usage**

```
function_run_diffcytDSLMM(ls_form_contrast, df_experiment_info, ls_features)
```

**Arguments**

`ls_form_contrast` list, with formula and constrat matrix.

`df_experiment_info` data.frame, information about the experiment (donor IDs, group IDs and sample IDs).

`ls_features` list, with cell counts and medians for each markers.

**Details**

Function to run diffcyt-DS-LMM model from the diffcyt package.

**Value**

list with 2 slots: - `model_fit`: the model fit; - `result_summary`: results of the model for each marker.

---

```
function_run_diffcyt_full_pipeline
```

*Run diffcyt pipeline*

---

**Description**

Run diffcyt pipeline

**Usage**

```
function_run_diffcyt_full_pipeline(  
  onevariation,  
  model = c("limma", "LMM"),  
  effect = c("random", "fixed")  
)
```

**Arguments**

|              |                                                                                             |
|--------------|---------------------------------------------------------------------------------------------|
| onevariation | list, of simulated data (output of the function_wrapper_apply_simulation_nbtimes function). |
| model        | character, model to run: "limma" or "LMM".                                                  |
| effect       | character, effect: "random" or "fixed".                                                     |

**Details**

Function to run diffcyt pipeline. We do not used the diffcyt function directly because of the limma model run differently with the effects.

**Value**

list with 2 slots: - model\_fit: the model fit; - result\_summary: results of the model for each marker.

---

```
function_summary_results_models
```

*Summarize data and results.*

---

**Description**

Summarize data and results.

**Usage**

```
function_summary_results_models(summary_from_model, package)
```

**Arguments**

summary\_from\_model      list, output from the functions running the models.

package                  character, package used to run the test: "CytoGLMM" or "diffcyt".

**Details**

Function to do a summary of the tested data and model's results for the CytoGLMM and diffcyt packages.

**Value**

data.frame of results for each simulation.

---

function\_to\_compute\_model\_computation\_onesimulation\_modelchoice  
*Run DS tests for one simulation.*

---

**Description**

Run DS tests for one simulation.

**Usage**

```
function_to_compute_model_computation_onesimulation_modelchoice(
  onevariation,
  model
)
```

**Arguments**

onevariation      list, of simulated data (output of the function\_wrapper\_apply\_simulation\_nbtimes function).

model              vector, name(s) of models to test.

**Details**

Function to run the DS tests through the different models.

**Value**

data.frame of results for each simulation all models combined.

---

```
function_to_transform_data
    arcsinh transformation.
```

---

**Description**

arcsinh transformation.

**Usage**

```
function_to_transform_data(data, cofactor = 5)
```

**Arguments**

data                      data.frame, cell values to transform.  
 cofactor                numeric, co-factor used in the arcsinh (by default cofactor = 5).

**Details**

Function to transform the data with the recommended transformation for cyTOF data: arcsinh with cofactor equals to 5.

**Value**

data.frame of transformed data.

---

```
function_value_onemarker
    Compute simulated cell values for one marker with markers NB infor-  

    mations
```

---

**Description**

Compute simulated cell values for one marker with markers NB informations

**Usage**

```
function_value_onemarker(
  marker_name,
  mu0,
  dispersion,
  subject_effect,
  nb_donor,
  nb_cell_per_sample,
  rho = 1
)
```

**Arguments**

|                    |                                                                                                                                        |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| marker_name        | character, name of the marker.                                                                                                         |
| $\mu_0$            | numeric, general donor mean from which the individual $\mu_{0i}$ will be drawn.                                                        |
| dispersion         | numeric, dispersion of the markers.                                                                                                    |
| subject_effect     | numeric, standard deviation for the normal distribution from which the donor's means will be drawn (by default subject_effect = 0.01). |
| nb_donor           | numeric, number of donors.                                                                                                             |
| nb_cell_per_sample | numeric, number of cells per sample (by default nb_cell_per_sample = 500).                                                             |
| rho                | numeric, fold change.                                                                                                                  |

**Details**

Function to generate value for one marker with a mean and dispersion specified for the negative binomiale.

**Value**

data.frame of cell values.

# Index

## \* internal

compute\_effectsize, [2](#)  
 compute\_pwr, [3](#)  
 compute\_variance, [3](#)  
 function\_apply\_modelcomputations\_modelchoice, [4](#)  
 function\_apply\_onesimulation\_withmarkerinfo, [5](#)  
 function\_check\_nbDEmarkers, [6](#)  
 function\_check\_nbmarkers, [6](#)  
 function\_compute\_diffcyt\_features, [7](#)  
 function\_create\_mock\_dataset\_withmarkerinfo, [7](#)  
 function\_DEmarkers\_sup\_nbmarkers, [8](#)  
 function\_designmat\_contrast\_diffcytDSlimma\_fixedeffect, [9](#)  
 function\_designmat\_contrast\_diffcytDSlimma\_ranomeffect, [9](#)  
 function\_extract\_marker\_names, [10](#)  
 function\_formula\_contrast\_diffcytDSLMM\_ranomeffect, [10](#)  
 function\_is\_data\_paired, [11](#)  
 function\_names\_DE\_markers, [12](#)  
 function\_run\_bootstrapcytoGLMM, [12](#)  
 function\_run\_cytoGLMM, [13](#)  
 function\_run\_diffcyt\_full\_pipeline, [15](#)  
 function\_run\_diffcytDSlimma, [13](#)  
 function\_run\_diffcytDSLMM, [14](#)  
 function\_summary\_results\_models, [15](#)  
 function\_to\_compute\_model\_computation\_onesimulation\_modelchoice, [16](#)  
 function\_to\_transform\_data, [17](#)  
 function\_value\_onemarker, [17](#)  
 compute\_variance, [3](#)  
 CyTOFpower, [4](#)  
 function\_apply\_modelcomputations\_modelchoice, [4](#)  
 function\_apply\_onesimulation\_withmarkerinfo, [5](#)  
 function\_check\_nbDEmarkers, [6](#)  
 function\_check\_nbmarkers, [6](#)  
 function\_compute\_diffcyt\_features, [7](#)  
 function\_create\_mock\_dataset\_withmarkerinfo, [7](#)  
 function\_DEmarkers\_sup\_nbmarkers, [8](#)  
 function\_designmat\_contrast\_diffcytDSlimma\_fixedeffect, [9](#)  
 function\_designmat\_contrast\_diffcytDSlimma\_ranomeffect, [9](#)  
 function\_extract\_marker\_names, [10](#)  
 function\_formula\_contrast\_diffcytDSLMM\_ranomeffect, [10](#)  
 function\_is\_data\_paired, [11](#)  
 function\_names\_DE\_markers, [12](#)  
 function\_run\_bootstrapcytoGLMM, [12](#)  
 function\_run\_cytoGLMM, [13](#)  
 function\_run\_diffcyt\_full\_pipeline, [15](#)  
 function\_run\_diffcytDSlimma, [13](#)  
 function\_run\_diffcytDSLMM, [14](#)  
 function\_summary\_results\_models, [15](#)  
 function\_to\_compute\_model\_computation\_onesimulation\_modelchoice, [16](#)  
 function\_to\_transform\_data, [17](#)  
 function\_value\_onemarker, [17](#)  
 compute\_effectsize, [2](#)  
 compute\_pwr, [3](#)