Users' Guide
MultiPower

Sonia Tarazona (sotacam@eio.upv.es)

Version 2.0 (March, 2023)

Content

Introduction

The statistical power of a method, which is its ability to detect features with a true change between experimental groups, is determined by the within-group variability, the size of the effect to be detected, the significance level to be achieved and the number of replicates per experimental group (sample size). Evaluating the power in omic experiments is a challenging task because a statistical test per each omic feature is performed. This implies that the within-condition variability might be different for different features, and that the significance level must be adapted to take into account the multiple testing correction. Moreover, different omic platforms have varying levels of noise, dynamic ranges, etc., which may result in non-comparable performance of differential analysis methods. Consequently, in a multi-omic experiment, independently computing the statistical power for each omic might not be the best strategy if the omics are to be analysed in an integrative fashion. In this case, a joint power study for all the omics is more appropriate, since the lack of power for one of them can be compensated with the power of the others.

The MultiPower R method performs a joint power study in which the cost of the multi-omic experiment is minimized while both a minimum power for each omic must be independently achieved and an average power for all of them is required. The parameters required to compute power can be either estimated by MultiPower from multi-omic pilot or available data (recommended) or set by users. MultiPower considers the multiple testing correction by adjusting the significance level to control the False Discovery Rate (FDR). Normally distributed data , count data and binary data (0/1 or TRUE/FALSE) are accepted, and the optimal sample size for each omic can be computed either with the same or different sample sizes for each omic. In this last case, the cost of generating each omic sample needs to be also considered as an additional parameter in the power maximization problem. The MultiPower method is designed to help users not only in the design of the multi-omic experiment but also to assess if an already generated multi-omic data set provides enough power for differential expression and to highlight potential limitations. MultiPower can also help to filter out features with the lowest effects in order to improve the statistical power of the dataset.

Getting started

The MultiPower method is available as an R package from https://github.com/ConesaLab/MultiPower. As for other packages in GitHub, it can be installed from R with the following instructions:

```
> install.packages("devtools")
> library(devtools)
> devtools::install_github("ConesaLab/multipower")
```

However some additional libraries must be previously installed.

Required R libraries

These are the libraries needed from the CRAN repository that can be installed with `install.packages()` function and then loaded with `library()`.

- FDRsampsize
- lpSolve [*only if you plan to allow for different sample sizes in each omic modality*]

Optimal sample size estimation from available data

MultiPower method needs some input parameters for each omic data type in order to compute the statistical power for a two-group comparison. We recommend estimating such input parameters from available data (pilot data or data from previous studies). In that case, or in the case that MultiPower is applied to evaluate the limitations of an already generated multi-omic data set, this is the wrapper function to be used to estimate the optimal sample size for the multi-omic experiment:

```
MultiPower(data, groups, type, omicPower = 0.6, averagePower = 0.85,
fdr = 0.05, cost = 1, equalSize = TRUE, max.size = 200, omicCol = NULL,
powerPlots = TRUE, null.effect = 0)
```

The arguments of the `MultiPower()` function are described in detail in the next sections.

Input data

data: List with as many elements as omic data types. The names of the omics should be the names of the list. Each element in this list must be a raw count data matrix, and in this case MultiPower will take into account the library sizes to estimate power; a normally distributed data matrix which must have been already pre-processed and normalized; or a binary data matrix (with 0/1 or TRUE/FALSE values). In any case, for each one of these matrices, rows must correspond to omic features (genes, methylation sites, ChIP-seq peaks, etc.) and columns to observations (biological samples, patients, etc.).

	TCGA-02-0432-01	TCGA-08-0245-01	TCGA-28-1750-01	TCGA-06-1084-01	TCGA-02-0064-01
FSTL1	6.750974	9.081284	9.961373	9.801834	10.822028
AACS	6.392410	6.572695	7.102250	6.924443	6.586677
RPS11	11.010146	10.781873	10.583565	10.540609	10.749480
CREB3L1	4.569818	4.722297	4.472831	4.290404	4.712333
ELMO2	7.789996	6.224809	6.914030	6.155211	6.817825
PNMA1	9.914452	9.418119	9.029799	8.005483	9.677593

groups: List with as many elements as omic data types. The names of the omics should be the names of the list. Each element in this list must be a vector with a length equal to the number of observations for that omic in `data` argument. Each element of this vector must indicate the experimental group where each observation belongs. Only two experimental groups are allowed.

```
[1] "Proneural"   "Proneural"   "Mesenchymal" "Mesenchymal" "Mesenchymal" "Proneural"
```

type: Vector with length equal to the number of omic data types. Each element of this vector must be a 1, 2 or 3 to indicate whether the omic data are count data (1), continuous data approximately following a normal distribution (2) or binary data (3).

Power parameters

omicPower: The minimum power that must be achieved for each omic. It must be a vector with a length equal to the number of omics. If it is a single number, this same number will be used for all the omics. By default, omicPower = 0.6.

averagePower: The minimum average power that must be globally achieved. By default, averagePower = 0.85.

fdr: False Discovery Rate level to be used. It is the significance level after multiple testing correction. By default, fdr = 0.05.

null.effect: Value of the effect size that corresponds to null hypothesis. E.g. if we aim to test H_0: $\mu_1 = \mu_2$, this is equivalent to H_0: $\mu_1 - \mu_2 = 0$, and the null.effect value would be 0 (default).

Other parameters

cost: The cost to generate a replicate (a sample) for each omic. It must be a vector with a length equal to the number of omics. If it is a single number, this same number will be used for all the omics. This argument will only be used when a different sample size per omic is allowed. By default, cost = 1 (which means that all the omics will be assumed to have the same cost).

equalSize: If TRUE (default), the same optimal sample size will be estimated for all the omics. If FALSE, omics are allowed to have different sample sizes.

max.size: Maximum allowed sample size. By default, max.size = 200.

omicCol: The color that will be used to plot each omic. It must be a vector with a length equal to the number of omics. If it is NULL (default), default colors are used.

powerPlots: If TRUE (default), power plots will be generated.

Interpretation of results

When applying `MultiPower()` function, the result is a list containing the following elements:

parameters: List with as many elements as omic data types. For each omic, each element of the list is another list containing the different parameters used to compute power, either estimated from the pilot data or provided by the user. The estimated parameters depend on the data type and are described next for each type of data.

Count (negative binomial) data (type = 1)

- type: Omic type (1, in this case) .
- logFC: Log2 of the fold-change between first group mean (first level of the group factor) and second group mean (second level of the group factor), per feature.
- pooledSD: Pooled standard deviation per group, per feature.
- CV: Coefficient of variation per feature, computed as the ratio of the pooled standard deviation and the mean counts (see below).
- delta: Difference between mean groups per feature.
- mu: Average of counts per feature across all samples (without taking the group into account).
- m: Number of features.
- d: Cohen's d per feature.

What is Cohen's d?

The Cohen's d is defined as Δ/σ, where Δ is the absolute difference of means for each experimental group (delta) and σ is the pooled standard deviation of both groups. The Cohen's d is not dependent on the scale of the data as it happens with Δ value. Cohen [1] and Sawilowsky [2] suggested the classification in the table below to aid users to associate it to the magnitude of effect size.

Cohen's d / Cohen's h	Effect size
0.01	Very small
0.2	Small
0.5	Medium
0.8	Large
1.2	Very large
2.0	Huge

Gaussian data (type = 2)

- type: Omic type (2, in this case) .
- delta: Difference between mean groups per feature.
- pooledSD: Pooled standard deviation per group, per feature.
- m: Number of features.
- d: Cohen's d per feature.

Binary data (type = 3)

- type: Omic type (3, in this case) .
- p1_p2: Difference between group proportions, where each proportion is computed as the number of 1/TRUE values in that group divided by the sample size in that group.
- pooledSD: Pooled standard deviation per group, per feature.
- p1: Proportion of 1/TRUE values in the first group (first level in group factor).
- delta: Difference between mean groups per feature.
- m: Number of features.
- d: Cohen's d per feature.

optimalSampleSize: List with the following elements:

n0: Sample size per group to achieve the minimum omic power (`omicPower`) for each omic.

n: Optimal sample size per group.

finalPower: Power at the optimal sample size for each omic.

fdr: See `fdr` parameter.

omicPower: See `omicPower` parameter.

averagePower: See `averagePower` parameter.

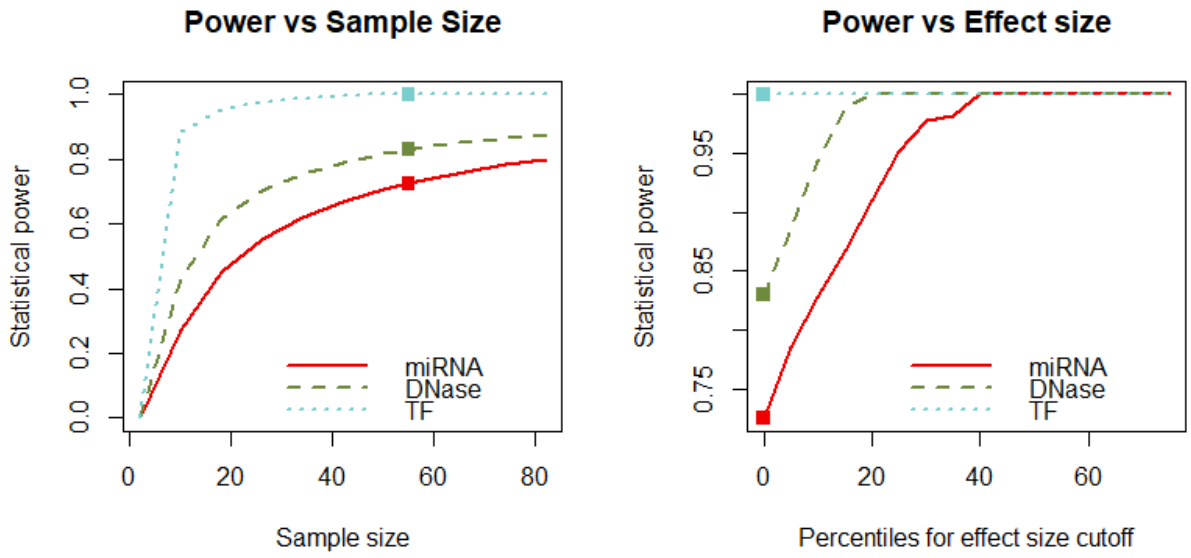
cost: See `cost` parameter.

summary: Table summarizing MultiPower results (see an example below). The columns are: the omic data names (omic), the omic data types (type), the number of omic features for each omic (numFeat), the minimum and maximum observed Cohen's d values (minCohenD and maxCohenD), the minimum power to be achieved for each omic (minPower), the average power to be achieved in the multi-omic experiment (averPower), the cost per omic (cost), the minimum

sample size needed for each omic to achieve minPower (minSampleSize), the optimal sample size per group (optSampleSize), and the power at the optimal sample size (power).

omic	type	numFeat	minCohenD	maxCohenD	minPower	averPower	cost	minSampleSize	optSampleSize	power
miRNA	1	106	0.03	4.38	0.6	0.85	1	32	55	0.7251
DNase	2	9846	0.00	5.29	0.6	0.85	1	17	55	0.8295
TF	2	130	0.08	17.03	0.6	0.85	1	5	55	1.0000

data2plot: Data generated to create the power plots that are also returned by the function (see example below).



These plots display the statistical power curve for each omic at different sample sizes (left) or at different cutoffs to filter low effect size features (right). The squares represent the values for the optimal sample size. The effect sizes used for the plot on the right are Cohen's d (h) values. We considered as cutoffs to filter out features with low Cohen's d (h) different percentiles from the minimum Cohen's d (P_0) where no filter is applied (all features are used to compute power) to the 75th percentile (P_{75}), where only the 25% of features with the highest effect size are included in the study.

When optimal sample size is out of the budget

Sometimes, the optimal sample size computed by MultiPower can be prohibitive in terms of cost. What options do we have then if we do not want to sacrifice power? We can allow for a different number of replicates per omic (see next section) since the omics with higher power might require a lower number or replicates and then the cost of the experiment can be reduced at the expense of not having the ideal scenario to perform an integration analysis. A second option could be to accept a higher FDR, which would increase the number of false positives in the statistical analysis. And finally, the most recommendable option is to filter out features with low effect size since they are the less likely features to present significant changes between conditions or to be used in real practice by clinicians, biologists, etc. In this case, MultiPower package has the function `postMultiPower()` to find out which Cohen's d threshold should be used to filter out omic features with low effect size to maintain the desired power while setting the sample size under budget. These are the arguments of the function:

optResults: Object returned by MultiPower() function.

max.size: Maximum sample size allowed by the user. It will be used to determine the effect size that can be detected (by default, 5).

omicCol: The color that will be used to plot each omic. It must be a vector with length equal to the number of omics. If it is NULL (default), default colors are used.

The `postMultiPower()` function returns a list with the following elements:

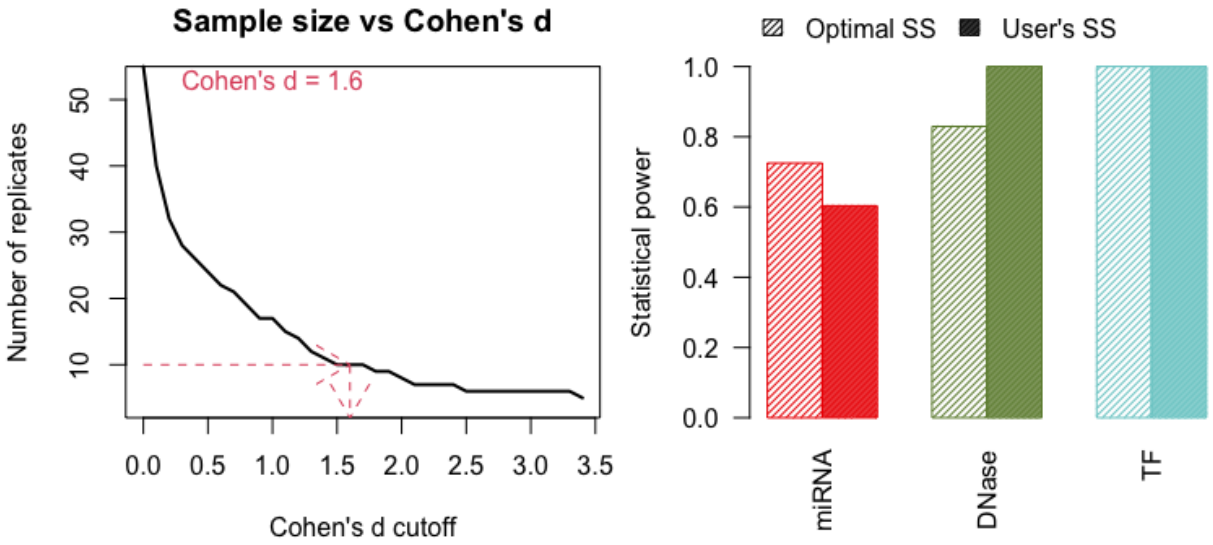
SampleSize: Matrix containing the optimal sample size (per group) for each omic data type (in columns) and for different values of Cohen's d or percentiles (in rows).

Power: Matrix containing the statistical power at the optimal sample size for each omic data type (in columns) and for different values of Cohen's d or percentiles (in rows).

NumFeat: Matrix containing the number of features selected in each omic data type (in columns) for different values of Cohen's d or percentiles (in rows).

d: Values of Cohen's d for which the optimal sample size has been estimated.

In addition, two plots are displayed (see an example below) that summarize these results.



In this example, equal sample size for all omics was required. That is why we have a unique line in the left plot. The maximum sample size per group was set to 10 and the cutoff for Cohen's d was 1.6. This means that to obtain the desired power with a sample size of 10 replicates per group, you need to filter out omic features with Cohen's d below 1.6. The right plot compares the effective power at the optimal sample size and with the power at the selected sample size (n=10).

If you wish to filter out omic features with Cohen's d (or h) below a given threshold (e.g. the value 1.6 obtained before), you can apply the function `CohenFilter()` in the package which returns the filtered multi-omics dataset. The following arguments are required:

data: The original data given to MultiPower function.

d: Cutoff value for Cohen's d (or h). A scalar (and then the same cutoff is applied to all the omics) or a vector with as many values as omic data types in the data object.

parameters: List of parameters for each omic estimated by MultiPower function. They can be retrieved from MultiPower output as output@parameters.

Optimal sample size estimation without pilot or previous data

When multi-omic pilot data sets are not available, users must set the values for the input parameters and then use the `optimalRep()` function, which requires the following arguments:

parameters: List with as many elements as omic data types. For each omic, each element of this list is another list containing the different parameters needed to compute power which, in this case, must be set by the user. See next section for more details.

omicPower: The minimum power that must be achieved for each omic. It must be a vector with a length equal to the number of omics. If it is a single number, this same number will be used for all the omics. By default, omicPower = 0.6.

averagePower: The minimum average power that must be globally achieved. By default, averagePower = 0.85.

fdr: False Discovery Rate level to be used. It is the significance level after multiple testing corrections. By default, fdr = 0.05.

cost: The cost to generate a replicate (a sample) for each omic. It must be a vector with a length equal to the number of omics. If it is a single number, this same number will be used for all the omics. This argument will only be used when a different sample size per omic is allowed. By default, cost = 1 (which means that all the omics will be assumed to have the same cost).

equalSize: If TRUE (default), the same optimal sample size will be estimated for all the omics. If FALSE, omics are allowed to have different sample sizes.

max.size: Maximum allowed sample size. By default, max.size = 200.

Setting input parameters

When no datasets are available, users must create an R list containing the input parameters required to estimate power for each omic. These parameters depend on the type of omic and their descriptions can be found in previous sections. Here you can find some advice to create parameters for each omic data type:

Type 1 (count data)

1. Generate vectors of possible values for group means, e.g. mu1 and mu2, with as many values as features in that omic.
2. Generate vectors of possible values for group standard deviations, e.g. s1 and s2, with as many values as features in that omic.
3. Compute **logFC** as `log2(mu2/mu1)`.
4. Estimate **mu** as `(mu2+mu1)/2`.
5. Estimate **pooledSD** as `sqrt((s1^2+ s2^2)/2)`.
6. Compute **CV** as `pooledSD/mu`.
7. Compute **delta** as `abs(mu2-mu1)`.
8. Let **M** be the length of any of the previous vectors, e.g. mu1.
9. Compute **d** as `delta/pooledSD`.

Type 2 (gaussian data)

1. Generate vectors of possible values for group means, e.g. mu1 and mu2, with as many values as features in that omic.
2. Generate vectors of possible values for group standard deviations, e.g. s1 and s2, with as many values as features in that omic.
3. Estimate **pooledSD** as `sqrt((s1^2+ s2^2)/2)`.
4. Compute **delta** as `abs(mu2-mu1)`.
5. Let **M** be the length of any of the previous vectors, e.g. mu1.
6. Compute **d** as `delta/pooledSD`.

Type 3 (binary data)

1. Generate vectors of possible values for group proportions, e.g. p1 and p2, with as many values as features in that omic.
2. Compute **p1_p2** as `p1-p2`.
3. Let **M** be the length of any of the previous vectors, e.g. p1.
4. Compute **d** as `abs(2*asin(sqrt(p1)) - 2*asin(sqrt(p2)))`.

Estimating the optimal sample size

Once the parameters object has been created (e.g. myparameters), the following R code shows an example on how to run the `optimalRep()` function, how to retrieve the results, and how to summarize them in a table:

```
optimalSS = optimalRep(parameters = myparameters, omicPower = 0.6,
                       averagePower = 0.8, fdr = 0.05,
                       cost = 1, equalSize = TRUE, max.size = 30)

optimalSS$n  # optimal sample size

powerSummary(parameters = myparameters, optimalSampleSize = optimalSS)

plotData = powerPlot(parameters = myparameters, optimalSampleSize = optimalSS,
                     omicCol = NULL)
```

MultiPower for more than two groups

When more than two experimental groups are to be compared, and pilot or public data are available for all these groups, the `MultiGroupPower()` function can be used to perform such multiple comparisons. This function will apply MultiPower for all the pairwise comparisons required by the user or for all the possible pairwise comparisons if no comparisons are specified. The individual MultiPower results for each comparison are provided in a list and, in addition, also a global summary is displayed. In this summary, the optimal sample size is computed as the maximum sample size required in all the comparisons to reach the desired power. The `MultiGroupPower()` function can be used as follows:

```
MultiGroupPower(data, groups, type, comparisons = NULL, omicPower = 0.6,
averagePower = 0.85, fdr = 0.05, cost = 1, equalSize = TRUE, max.size = 200,
omicCol = NULL, powerPlots = FALSE, summaryPlot = TRUE)
```

Most of the arguments of the `MultiGroupPower()` function were already described in detail in the MultiPower section. Next, we describe those that are specific of `MultiGroupPower` or that may have a different format.

comparisons: Pairwise comparisons to be done between groups. If NULL (default option), the function will generate all the possible comparisons between the groups that are available for all omics. If users wish to indicate the comparisons to be done, they must provide a matrix with two rows and as many columns as comparisons. Each column must be a two-element vector with the

two groups to be compared. An easy way to generate this matrix is using the `combn()` function that returns a matrix with all the possible comparisons. Users can then remove the columns of the comparisons that are not interesting for them.

powerPlots: If TRUE (FALSE is the default), power plots will be generated for each individual comparison as in MultiPower function.

summaryPlot: If TRUE (default), summary plots for sample size and power will be generated including the results for all comparisons and the global result, that is, the maximum sample size for all comparisons ("optimal" sample size) and the corresponding statistical power for each omic.

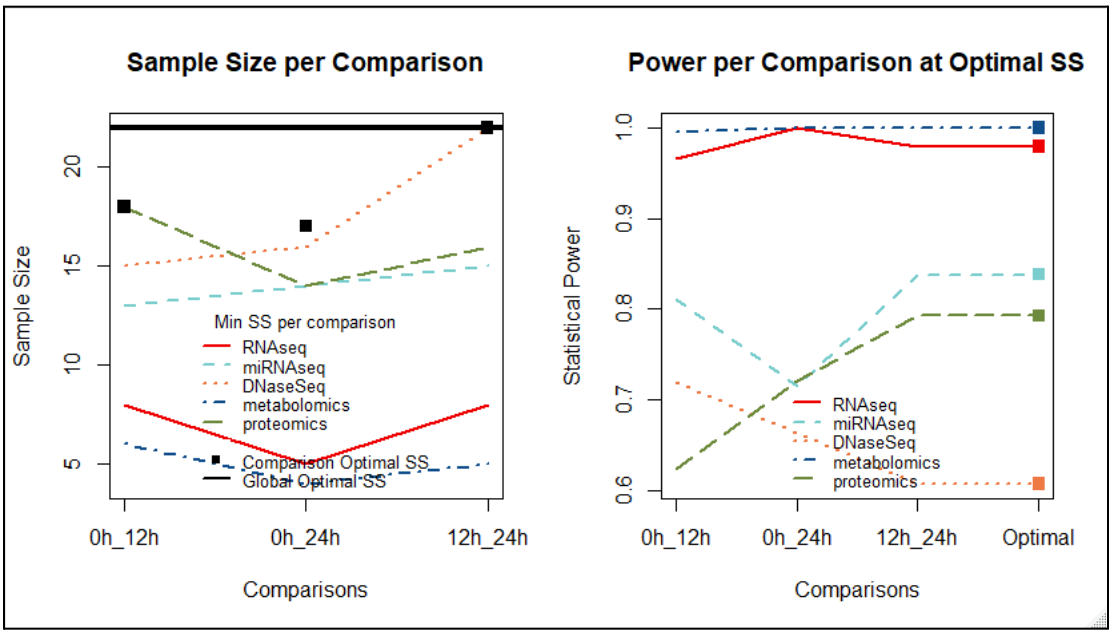
Interpretation of results

When applying `MultiGroupPower()` function, the result is a list containing as many elements (lists) as the number of comparisons and an additional element with the global summary of the results. The list obtained for each individual comparison is the same as the one returned by the MultiPower function (see the corresponding section), while the GlobalSummary element contains the following information:

```
==============================
Global summary
==============================
       omic type numFeat minCohenD maxCohenD minPower averPower cost minSampleSize optSampleSize  power
miRNA miRNA    1     106      0.01      1.85      0.6       0.8    1        17-185           185 0.6007
DNase DNase    2    9846      0.00      2.36      0.6       0.8    1          7-32           185 0.9347
TF       TF    2     130      0.02      1.52      0.6       0.8    1          5-30           185 0.9505
```

This summary table is very similar to the summary table of an individual pairwise comparison. The only difference is that the minimum sample size needed for each omic to achieve minPower (minSampleSize) contains a range of values when the value was different at each comparison.

In addition to this summary table, the following summary plots are also generated when the summaryPlot parameter is set to TRUE:



The sample size plot (left plot) is slightly different when equal or different sample size is allowed for each group. When equal sample size is required (example above), the color lines show the minimum sample size required for each omic at each comparison, while the black square is the optimal sample size at each comparison and the black line the global optimal sample size (the maximum of them). The power plot displays the statistical power for each omic at the optimal sample size. This information is given for each comparison and for the global optimal sample size (squares).

These plots can also be generated from the object returned by `MultiGroupPower()` function by using the `MultiCompaPlot()` function.

How to cite MultiPower

Tarazona, S., Balzano-Nogueira, L., Gómez-Cabrero, D., Schmidt, A., Imhof, A., Hankemeier, T., ... & Conesa, A. (2020). Harmonization of quality metrics and power calculation in multi-omic studies. Nature communications, 11(1), 3092.

References

[1] Cohen, Jacob. Statistical power analysis for the behavioral sciences. Hilsdale. NJ: Lawrence Earlbaum Associates 2 (1988).
[2] Sawilowsky, S. S. (2009). New effect size rules of thumb. Journal of Modern Applied Statistical Methods, 8(2), 597 – 599.