

# Package ‘PowerExplorer’

May 24, 2019

**Title** Power Estimation Tool for RNA-Seq and proteomics data

**Version** 1.4.0

**URL** <https://gitlab.utu.fi/CompBioMedNGSTools/PowerExplorer>

**Description** Estimate and predict power among groups and multiple sample sizes with simulated data, the simulations are operated based on distribution parameters estimated from the provided input dataset.

**biocViews** ImmunoOncology, RNASeq, Proteomics, DifferentialExpression, MultipleComparison, Sequencing, Coverage, ChIPSeq

**Depends** R (>= 3.5.0), SummarizedExperiment

**Imports** DESeq2, ROTS, vsn, stats, utils, methods, gridExtra, MASS, data.table, ggplot2, Biobase, S4Vectors, BiocParallel

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**License** Artistic-2.0

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1.9000

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

**git\_branch** RELEASE\_3\_9

**git\_last\_commit** a07a074

**git\_last\_commit\_date** 2019-05-02

**Date/Publication** 2019-05-23

**Author** Xu Qiao [aut, cre],  
Laura Elo [cph]

**Maintainer** Xu Qiao <xu.qiao@utu.fi>

## R topics documented:

estimatePower . . . . .	2
exampleObject . . . . .	3
exampleProteomicsData . . . . .	4
exampleRNASeqData . . . . .	4
listEstPwr . . . . .	5

listPredPwr . . . . .	5
plotEstPwr . . . . .	6
plotPredPwr . . . . .	7
PowerExplorerStorage-class . . . . .	7
predictPower . . . . .	8
show,PowerExplorerStorage-method . . . . .	9

## Index 10

---

estimatePower	<i>Estimate Power of the Actual Data</i>
---------------	--

---

### Description

Estimate power of comparison between each two groups based on the data simulated from estimated normal distributions of entries in the entire dataset

### Usage

```
estimatePower(inputObject, groupVec, isLogTransformed = FALSE,
  dataType = c("RNASeq", "Proteomics"), minLFC = 0.5, alpha = 0.05,
  ST = 100, seed = 123, enableROTS = FALSE, paraROTS = list(B = 1000, K
  = NULL, paired = FALSE, a1 = NULL, a2 = NULL, progress = FALSE),
  showProcess = FALSE, saveResultData = FALSE, parallel = FALSE,
  BPPARAM = bpparam())
```

### Arguments

inputObject	a numeric raw data matrix or SummarizedExperiment object
groupVec	a vector indicating the grouping of samples
isLogTransformed	logical; set to TRUE, if the input data is log transformed.
dataType	"RNASeq" or "Proteomics" indicates the data type of the input data matrix.
minLFC	the threshold for log <sub>2</sub> fold change, entries with lower LFC are not included in the power calculation, set to 0 if no threshold is needed.
alpha	controlled false positive rate.
ST	the number of simulations of abundance data generation and repeated times of statistical test for each entry (>=100 recommended).
seed	an integer seed for the random number generator.
enableROTS	logical; if TRUE, Reproducibility-Optimized Test Statistic (ROTS) will be used as the statistical model. used as the statistical model.
paraROTS	a list object containing additional parameters passed to ROTS (if enabled), see <a href="#">ROTS</a> .
showProcess	logical; if TRUE, show the detailed information of each simulation, used for debugging only.
saveResultData	logical; if TRUE, save the simulated data into RData with name pattern "simulated_Data_numRep_X_numSim_XXX_XXXXX.RData"
parallel	logical; if FALSE parallelization is disabled; if TRUE, parallelize calculations using <a href="#">BiocParallel</a> .
BPPARAM	an optional argument object passed <a href="#">bplapply</a> to indicate the registered cores, if parallel=TRUE.

**Value**

a list of power estimates grouped in comparisons between each two groups

**See Also**

[predictPower](#) predict power with increasing sample sizes

**Examples**

```
# Example 1: a random generated Proteomics dataset (10 DE, 100 non-DE)
# Note: Simulation times(ST) is specified as 10 for shorter example runtime,
# ST > 50 is recommended
data(exampleProteomicsData)
dataMatrix <- exampleProteomicsData$dataMatrix
groupVec <- exampleProteomicsData$groupVec

# Run estimation without LFC filtration
resObject <- estimatePower(dataMatrix, groupVec,
                           dataType="Proteomics",
                           isLogTransformed=FALSE,
                           minLFC=0, alpha=0.05,
                           ST=10, seed=123)
```

---

exampleObject

*An Example Predicted Power Object For Tests*

---

**Description**

This is an example PowerExplorerStorage resulted from an example run on a publically available RNA-Seq datase containing the gene expression in C57BL/6J and DBA/2J Mouse Striatum

**Usage**

```
data(exampleObject)
```

**Format**

An object of class PowerExplorerStorage with 12152 rows and 21 columns.

**Examples**

```
data(exampleObject)

show(exampleObject)
plotEstPwr(exampleObject)
```

---

exampleProteomicsData *Randomly Generated Proteomics Dataset*

---

### Description

This is a randomly generated proteomics dataset with 130 protein entries (rows) and 15 samples (columns) in 3 sample groups A, B and C, the log<sub>2</sub> fold change (LFC) between group B and A is specified as 1, between C and B is also 1, thus the LFC is 2 between C and A.

### Usage

```
data(exampleProteomicsData)
```

### Format

An list contains "dataMatrix" and "groupVec"

### Examples

```
data(exampleProteomicsData)
head(exampleProteomicsData$dataMatrix)
```

---

exampleRNASeqData *Randomly Generated RNASeq Dataset*

---

### Description

This is a randomly generated RNASeq dataset with 130 gene entries (rows) and 15 samples (columns) in 3 sample groups A, B and C, the log<sub>2</sub> fold change (LFC) between group B and A is specified as 1, between C and B is also 1, thus the LFC is 2 between C and A.

### Usage

```
data(exampleRNASeqData)
```

### Format

An list contains "dataMatrix" and "groupVec"

### Examples

```
data(exampleRNASeqData)
head(exampleRNASeqData$dataMatrix)
```

---

listEstPwr	<i>List Estimated Power</i>
------------	-----------------------------

---

**Description**

show a top-list of power in numerical order, or list the power selected genes/proteins.

**Usage**

```
listEstPwr(inputObject, decreasing = TRUE, top = 20, selected = NA)
```

**Arguments**

inputObject	the input inputObject.
decreasing	logical; TRUE, decreasing order; FALSE, increasing order.
top	the number of genes/proteins in the top list
selected	default as NA; specify as a list of geneID or protein ID to show power of a list of interested records.

**Value**

a top list of power / power of a list of interested genes or proteins

**Examples**

```
data(exampleObject)
# show 10 top genes with high power (decreasing order)
listEstPwr(exampleObject, decreasing = TRUE, top = 10)
# show a list of interested genes
listEstPwr(exampleObject,
            selected = c("ENSMUSG00000000303",
                        "ENSMUSG000000087272",
                        "ENSMUSG000000089921"))
```

---

listPredPwr	<i>List Predicted Power</i>
-------------	-----------------------------

---

**Description**

show a top-list of predicted power in numerical order, or list the power selected genes/proteins.

**Usage**

```
listPredPwr(inputObject, decreasing = TRUE, top = 20, selected = NA)
```

**Arguments**

inputObject	the input inputObject.
decreasing	logical; TRUE, decreasing order; FALSE, increasing order.
top	the number of genes/proteins in the top list
selected	default as NA; specify as a list of geneID or protein ID to show power of a list of interested records.

**Value**

a top list of power / power of a list of interested genes or proteins for each sample size

**Examples**

```
data(exampleObject)
# show 10 top genes with high power (decreasing order)
listPredPwr(exampleObject, decreasing = TRUE, top = 10)

# show a list of interested genes
listPredPwr(exampleObject,
             selected = c("ENSMUSG00000000303",
                          "ENSMUSG000000087272",
                          "ENSMUSG000000089921"))
```

---

plotEstPwr

*Plot A Summary of Estimated Power*

---

**Description**

Produce a plot to summary the power estimated by function [estimatePower](#), the plot function is called in [estimatePower](#), but using it manually is possible

**Usage**

```
plotEstPwr(inputObject)
```

**Arguments**

inputObject	a result container object <a href="#">PowerExplorerStorage</a> returned from <a href="#">estimatePower</a> .
-------------	--

**Value**

plot(s) of the summarised information on the estimated power

**Examples**

```
data(exampleObject)
plotEstPwr(exampleObject)
```

---

plotPredPwr

*Observe Predicted Power Within Different LFC Scales*


---

### Description

With a complete power list and LFC list returned from `predictPower`, power estimates can be observed dynamically within specified LFC ranges.

### Usage

```
plotPredPwr(inputObject, minLFC, maxLFC, LFCscale = 1)
```

### Arguments

inputObject	a result container object <code>PowerExplorerStorage</code>
minLFC	default as 0, the left edge of the LFC range within which genes will be included in the graph.
maxLFC	default as NA (the maximum value in data will be used) the right edge of the LFC range within which genes will be included in the graph.
LFCscale	the size of each unit when segmenting predicted power by LFC

### Value

plot(s) of power density under multiple sample sizes

### Examples

```
# load an example object containing
# predicted power under different sample sizes
data(exampleObject)
plotPredPwr(exampleObject)
plotPredPwr(exampleObject)
#It is possible to observe power trend in different scales and ranges of LFCs
plotPredPwr(exampleObject, minLFC=0, maxLFC=2, LFCscale=0.5)
```

---

PowerExplorerStorage-class

*PowerExplorer Object*


---

### Description

An extended `SummarizedExperiment` object to contain input `dataMatrix`, grouping information, estimated power, predicted power, fold change estimates and other estimation parameters.

---

predictPower                      *Estimate Power Under Increasing Sample Sizes*

---

### Description

Similar to [estimatePower](#), power estimations are performed under multiple increasing sample sizes

### Usage

```
predictPower(inputObject, groupVec, isLogTransformed = FALSE,
             dataType = c("RNASeq", "Proteomics"), enableROTS = FALSE,
             paraROTS = list(B = 1000, K = NULL, paired = FALSE, a1 = NULL, a2 = NULL,
                             progress = FALSE), minLFC = 0.5, rangeSimNumRep = NA, alpha = 0.05,
             ST = 100, seed = 123, parallel = FALSE, BPPARAM = bpparam(),
             showProcess = FALSE, saveResultData = FALSE)
```

### Arguments

inputObject	a numeric raw Proteomics abundance data matrix, in which rows correspond to proteins and columns correspond to samples.
groupVec	a vector indicating the grouping of samples
isLogTransformed	logical; logical; set to TRUE, if the input data is log transformed.
dataType	"RNASeq" or "Proteomics" indicates the data type of the input data matrix.
enableROTS	logical; if TRUE, Reproducibility-Optimized Test Statistic (ROTS) will be used as the statistical model.
paraROTS	a list object containing additional parameters passed to ROTS (if enabled), see <a href="#">ROTS</a> .
minLFC	LFC threshold
rangeSimNumRep	a vector of sample sizes under which power will be estimated
alpha	controlled false positive rate.
ST	the number of simulations of abundance data generation and repeated times of statistical test for each protein ( $\geq 100$ recommended).
seed	an integer seed for the random number generator.
parallel	logical; if FALSE parallelization is disabled; if TRUE, parallelize calculations using <a href="#">BiocParallel</a> .
BPPARAM	an optional argument object passed <a href="#">bplapply</a> to indicate the registered cores, if parallel=TRUE.
showProcess	logical; if TRUE, show the detailed information of each simulation, used for debugging only.
saveResultData	logical; if TRUE, save the simulated data into RData with name pattern "simulated_Data_numRep_X_numSim_XXX_XXXXXX.RData".

### Value

a list of power predictions for each sample size, grouped in comparisons between each two groups



**See Also**

[estimatePower](#) estimate power based on actual data

**Examples**

```
# Example 1: a random generated Proteomics dataset (10 DE, 100 non-DE)
data(exampleProteomicsData)
dataMatrix <- exampleProteomicsData$dataMatrix
groupVec <- exampleProteomicsData$groupVec

# Run estimation
# Note: Simulation times(ST) is specified as 5 for shorter example runtime
#       For better performance, ST > 50 is recommended
predictedPower <- predictPower(dataMatrix, groupVec,
                               isLogTransformed=FALSE,
                               dataType="Proteomics",
                               minLFC=0,
                               rangeSimNumRep=c(5, 10, 15),
                               alpha=0.05, ST=5, seed=123)
```

---

show,PowerExplorerStorage-method

*show method for PowerExplorerStorage*

---

**Description**

show method for PowerExplorerStorage

**Usage**

```
## S4 method for signature 'PowerExplorerStorage'
show(object)
```

**Arguments**

object            a PowerExplorerStorage object as input

**Value**

a summary of input PowerExplorerStorage object

**Methods (by class)**

- PowerExplorerStorage: method for PowerExplorerStorage objects

**Examples**

```
data(exampleObject)
show(exampleObject)
```

# Index

## \*Topic **datasets**

- exampleObject, 3
- exampleProteomicsData, 4
- exampleRNASeqData, 4

BiocParallel, 2, 8  
bplapply, 2, 8

estimatePower, 2, 6, 8, 9  
exampleObject, 3  
exampleProteomicsData, 4  
exampleRNASeqData, 4

listEstPwr, 5  
listPredPwr, 5

plotEstPwr, 6  
plotPredPwr, 7  
PowerExplorerStorage, 6, 7  
PowerExplorerStorage  
(PowerExplorerStorage-class), 7  
PowerExplorerStorage-class, 7  
predictPower, 3, 7, 8

ROTS, 2, 8

show, PowerExplorerStorage-method, 9