

# Package: phantusLite (via r-universe)

June 21, 2024

**Type** Package

**Title** Loading and annotation RNA-seq counts matrices

**Version** 1.3.0

**Description** PhantusLite – a lightweight package with helper functions of general interest extracted from phantus package. In particular it simplifies working with public RNA-seq datasets from GEO by providing access to the remote HSDS repository with the precomputed gene counts from ARCHS4 and DEE2 projects.

**Depends** R (>= 4.2)

**Imports** data.table, rhdf5client(>= 1.25.1), httr, stringr, stats, utils, Biobase, methods

**biocViews** GeneExpression, Transcriptomics, RNASeq

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.1

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown, BiocStyle, rhdf5, GEOquery

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**URL** <https://github.com/ctlab/phantusLite/>

**BugReports** <https://github.com/ctlab/phantusLite/issues>

**Repository** <https://bioc.r-universe.dev>

**RemoteUrl** <https://github.com/bioc/phantusLite>

**RemoteRef** HEAD

**RemoteSha** 236c8d2b767da0b39bb379474ec2d0974d2d33d7

## Contents

getHSDSFileList . . . . .	2
inferCondition . . . . .	3
loadCountsFromH5FileHSDS . . . . .	3
loadCountsFromHSDS . . . . .	4
readGct . . . . .	5
writeGct . . . . .	5
<b>Index</b>	<b>7</b>

---

getHSDSFileList	<i>Returns list of all HDF5-files on HSDS-server</i>
-----------------	--

---

## Description

Returns list of all HDF5-files on HSDS-server

## Usage

```
getHSDSFileList(
  url = "https://alserglab.wustl.edu/hsds/?domain=/counts",
  directory = NULL
)
```

## Arguments

url	containing url of the server and root domain.
directory	containing name of the directory

## Value

List of all HDF5-files on the server or all files of the collection

## Examples

```
url <- 'https://alserglab.wustl.edu/hsds/?domain=/counts'
getHSDSFileList(url)
```

---

inferCondition	<i>Adds condition to the annotation.</i>
----------------	--

---

**Description**

Adds condition to the annotation.

**Usage**

```
inferCondition(es)
```

**Arguments**

es                   contains ExpressionSet object

**Value**

Annotated ExpressionSet with conditions and replicates

**Examples**

```
ess <- GEOquery::getGEO("GSE143903")
es <- ess[[1]]
es <- inferCondition(es)
es$condition # contains inferred groups
es$replicate # contains inferred replicate numbers
```

---

```
loadCountsFromH5FileHSDS
```

*Load count matrix from remote HDF5-file*

---

**Description**

Load count matrix from remote HDF5-file

**Usage**

```
loadCountsFromH5FileHSDS(  
  es,  
  url = "https://alserglab.wustl.edu/hsds/?domain=/counts",  
  file,  
  sampleIndexes = NULL  
)
```

**Arguments**

es                    containing ExpressionSet loaded from GEO. Contains empty expression matrix.  
 url                    containing url of the server and root domain.  
 file                    containing name of the file (relative to the root domain)  
 sampleIndexes        containing sample indexes list

**Value**

ExpressionSet object with loaded count matrix

**Examples**

```

ess <- GEOquery::getGEO("GSE53053")
es <- ess[[1]]
url <- 'https://alserglab.wustl.edu/hsds/?domain=/counts'
file <- "/dee2/mmusculus_star_matrix_20240409.h5"
es <- loadCountsFromH5FileHSDS(es, url, file)

```

---

loadCountsFromHSDS      *Load count matrix from HDF5-files.*

---

**Description**

Load count matrix from HDF5-files.

**Usage**

```

loadCountsFromHSDS(
  es,
  url = "https://alserglab.wustl.edu/hsds/?domain=/counts"
)

```

**Arguments**

es                    containing ExpressionSet loaded from GEO. Contains empty expression matrix.  
 url                    containing url of the server and root domain.

**Value**

ExpressionSet with loaded count matrix

**Examples**

```

ess <- GEOquery::getGEO("GSE85653")
es <- ess[[1]]
url <- 'https://alserglab.wustl.edu/hsds/?domain=/counts'
es <- loadCountsFromHSDS(es, url)

```

---

readGct	<i>Reads ExpressionSet from a GCT file.</i>
---------	---

---

**Description**

Only versions 1.2 and 1.3 are supported.

**Usage**

```
readGct(gct)
```

**Arguments**

gct	Path to gct file
-----	------------------

**Value**

ExpressionSet object

**Examples**

```
es <- readGct(system.file("extdata/testdata/gct/test.gct", package="phantasusLite"))
```

---

writeGct	<i>Saves ExpressionSet to a GCT file (version 1.3).</i>
----------	---

---

**Description**

Saves ExpressionSet to a GCT file (version 1.3).

**Usage**

```
writeGct(es, file, gzip = FALSE)
```

**Arguments**

es	ExpressionSet object to save
file	Path to output gct file
gzip	Whether to gzip apply gzip-compression for the output file#'

**Value**

Result of the closing file (as in 'close()' function')

**Examples**

```
es <- readGct(system.file("extdata/testdata/gct/test.gct", package="phantasusLite"))
out <- tempfile(fileext = ".gct.gz")
writeGct(es, out, gzip=TRUE)
```

# Index

`getHSDSFileList`, [2](#)

`inferCondition`, [3](#)

`loadCountsFromH5FileHSDS`, [3](#)

`loadCountsFromHSDS`, [4](#)

`readGct`, [5](#)

`writeGct`, [5](#)