Package ‘TCGAretriever’

December 17, 2019

Type Package
Title Retrieve Genomic and Clinical Data from TCGA
Version 1.5
Date 2019-12-17
Author Damiano Fantini
Maintainer Damiano Fantini <damiano.fantini@gmail.com>
Description The Cancer Genome Atlas (TCGA) is a program aimed at improving our understanding of Cancer Biology. Several TCGA Datasets are available online. 'TCGAretriever' helps accessing and downloading TCGA data hosted on 'cBioPortal' via its Web Interface (see <http://www.cbioportal.org/> for more information). 'TCGAretriever' is easy to use (get all the TCGA data you need with a few lines of code), enforces reliable data download (via 'httr'), and is suitable for downloading large volumes of data.

URL https://www.data-pulse.com/dev_site/TCGAretriever/
Depends R(>= 3.1)
Imports httr
Suggests graphics, utils, knitr, rmarkdown
VignetteBuilder knitr
Encoding UTF-8
License GPL-2
RoxygenNote 6.1.1
NeedsCompilation no
Repository CRAN
Date/Publication 2019-12-17 21:20:02 UTC

R topics documented:

  basic_tcga_query .................................................. 2
  expand_cases .......................................................... 3
  fetch_all_tcgadata .................................................. 3
  get_cancer_studies .................................................. 4
**basic_tcga_query**

get_cancer_types ........................................... 5
get_case_lists ............................................... 6
get_clinical_data ........................................... 7
get_ext_mutation ............................................ 8
get_genetic_profiles ....................................... 9
get_profile_data ............................................ 9
get_protein_data ........................................... 10
get_protein_info ............................................ 11
make_groups .................................................. 12

**Index**

**basic_tcga_query**  
TCGA Core Query Engine

**Description**

Core Function that queries the URL provided as argument (typically a cbioportal.org URL). The function halts until the content has been completely downloaded and returns a data frame.

**Usage**

```
basic_tcga_query(my_url)
```

**Arguments**

my_url  
string. Typically, a URL pointing to the cBioPortal API.

**Details**

This is a core function invoked by other functions in the package.

**Author(s)**

Damiano Fantini, <damiano.fantini@gmail.com>

**References**

- [https://www.data-pulse.com/dev_site/TCGAretriever/](https://www.data-pulse.com/dev_site/TCGAretriever/)
**expand_cases**

*Explode TCGA Case Identifiers from a TCGA Study*

**Description**

Each TCGA Study includes one or more "case lists". These are lists of sample/patient identifiers. All case lists of a study of interest are retrieved and the individual case identifiers are expanded and returned.

**Usage**

```r
expand_cases(csid = NULL)
```

**Arguments**

- `csid` string corresponding to a TCGA Cancer Study identifier

**Value**

list containing as many elements as TCGA case lists available for a given TCGA Study. Each element is a list containing two elements:

- a string corresponding to the Id of the case list as defined by TCGA
- character vector including all case IDs corresponding to the case list

**Examples**

```r
expand_cases("blca_tcga")
```

---

**fetch_all_tcgadata**

*Recursively Fetch All Data Included in a TCGA Study Subset*

**Description**

Recursively query TCGA to retrieve large volumes of data corresponding to a high number of genes (up to the entire genome). Data are returned as a data frame that can be easily manipulated for further analyses.

**Usage**

```r
fetch_all_tcgadata(case_id = NULL, gprofile_id = NULL, glist = NULL, mutations = FALSE)
```
get_cancer_studies

Retrieve a List of Cancer Studies Available at TCGA

Description

Retrieve information about the different TCGA studies that are available at cBioPortal. Information include a cancer_study_id, a name of the study and a description for each study.

Usage

get_cancer_studies()
get_cancer_types

Value

Data Frame including one study per row and three columns.

Author(s)

Damiano Fantini. <damiano.fantini@gmail.com>

References

- https://www.data-pulse.com/dev_site/TCGAretriever/

Examples

```r
all_studies <- get_cancer_studies()
message(paste("There are", nrow(all_studies), "studies currently available..."))
if(ncol(all_studies) >= 2) {
  head(all_studies[,1:2])
}
```

---

get_cancer_types  Retrieve a List of Cancer Types as Defined by the TCGA Guidelines

Description

Retrieve information about the different types of cancer that may be included in TCGA Studies. Information include Identifier and Cancer Name.

Usage

get_cancer_types()

Value

A data.frame with one row per cancer type and two columns

Author(s)

Damiano Fantini. <damiano.fantini@gmail.com>

References

- https://www.data-pulse.com/dev_site/TCGAretriever/
Examples

```r
all_canc <- get_cancer_types()
message(paste("There are", nrow(all_canc), "types on cancer defined at TCGA..."))
head(all_canc)
```

**get_case_lists**

Retrieve All Case List Available for a Specific TCGA Study

**Description**

TCGA keeps track of which samples were analyzed by which technique within a given Study. Sample identifiers are organized in lists of cases (samples/patients) and are associated with a case_list identifier. The function retrieves information about the case lists available for a given TCGA Study.

**Usage**

```r
get_case_lists(csid = NULL)
```

**Arguments**

- `csid` String corresponding to the Identifier of the TCGA Study of Interest

**Value**

Data Frame including one row per case_list and five columns

**Author(s)**

Damiano Fantini, <damiano.fantini@gmail.com>

**References**

- [https://www.data-pulse.com/dev_site/TCGAretriever/](https://www.data-pulse.com/dev_site/TCGAretriever/)

**Examples**

```r
all_case_lists <- get_case_lists("blca_tcga")
if(ncol(all_case_lists) >= 3) {
  all_case_lists[,1:3]
}
```
**get_clinical_data**

**Retrieves Clinical Information from a TCGA Study**

**Description**

Retrieve Information about the Patients included in a TCGA Study of Interest. Each patient is associated with a `case_id`. Each `case_id` is accompanied by a set of clinical information that may include sex, age, therapeutic regimen, Tumor Staging, vital status and others. NA are allowed.

**Usage**

```r
get_clinical_data(case_id = NULL)
```

**Arguments**

- `case_id` string corresponding to the case_list identifier of a specific list of cases of interest

**Value**

data.frame including one row per patient/case/sample

**Author(s)**

Damiano Fantini, <damiano.fantini@gmail.com>

**References**

- [https://www.data-pulse.com/dev_site/TCGAretriever/](https://www.data-pulse.com/dev_site/TCGAretriever/)

**Examples**

```r
clinic_data <- get_clinical_data("blca_tcga_all")
if (nrow(clinic_data) >= 6 & ncol(clinic_data) >= 5) {
  clinic_data[1:6,1:5]
  hist(as.numeric(clinic_data$AGE),
       col = "darkorange",
       xlab = "Age",
       main = "Bladder Cancer, age of diagnosis")
}
```
get_ext_mutation

Retrieve Extended Information About DNA Mutations from TCGA

Description
Query TCGA for Data about DNA Sequence Variations (Mutations) identified by exome sequencing projects. The function will retrieve an extensive set of information for each mutation that was identified in the set of cases of interest. The function can only handle a limited number of query genes. For larger queries, use the fetch_all_tcgadata() function.

Usage
get_ext_mutation(case_id = NULL, gprofile_id = NULL, glist = NULL)

Arguments
- case_id: string corresponding to the Identifier of the case list of interest
- gprofile_id: string corresponding to the Identifier of the Genetic Profile of Interest
- glist: character vector including Gene Identifiers (ENTREZID or OFFICIAL_SYMBOL)

Value
data Frame including one row per mutation

Author(s)
Damiano Fantini, <damiano.fantini@gmail.com>

References
- https://www.data-pulse.com/dev_site/TCGAretriever/

Examples

```r
tp53_mutats <- get_ext_mutation("blca_tcgas_all", "blca_tcgas_mutations", "TP53")
if(ncol(tp53_mutats) >= 6 & nrow(tp53_mutats) >= 10){
  tp53_mutats[1:10,1:6]
}
```
**get_genetic_profiles**  
*Retrieve Genetic Profiles for a TCGA Study of Interest*

**Description**
Retrieve Information about all genetic profiles associated with a TCGA Study of interest. Each TCGA Study includes one or more kind of molecular analyses whose results are referred to as genetic profiles.

**Usage**
```r
get_genetic_profiles(csid = NULL)
```

**Arguments**
csid  
string corresponding to the cancer study id of interest

**Value**
data.frame including one row per genetic profile and six columns

**Author(s)**
Damiano Fantini, <damiano.fantini@gmail.com>

**References**
- [https://www.data-pulse.com/dev_site/TCGAretriever/](https://www.data-pulse.com/dev_site/TCGAretriever/)

**Examples**
```r
get_genetic_profiles("blca_tcga")
```

---

**get_profile_data**  
*Retrieve TCGA Data corresponding to a Specific Genetic Profile of Interest*

**Description**
Retrieve Data corresponding to a Genetic Profile of interest from a given TCGA Study. This function is the workhorse of the TCGAretriever package and can be used to fetch data concerning several genes at once. For larger queries, the use of the fetch_all_tcgadata() function is mandatory.
get_protein_data

Retrieve Protein Expression Data from a TCGA Study

Description

TCGA includes Information about Protein Expression measured by reverse-phase protein arrays. Antibody Information can be exported together with Expression Data. All expression data will be retrieved for all available protein targets.

Usage

get_protein_data(case_id = NULL, array_info = TRUE)

Arguments

case_id String corresponding to the Identifier of the Case List of Interest
array_info Logical. If TRUE, Antibody Information will also be exported
get_protein_info

Value

Data Frame with one gene (protein target) per row

Author(s)

Damiano Fantini, <damiano.fantini@gmail.com>

References

- https://www.data-pulse.com/dev_site/TCGAretriever/

Examples

# Protein Expression Only
blca_protein <- get_protein_data("blca_tcga_sequenced", FALSE)
if (nrow(blca_protein) > 10 & ncol(blca_protein) > 8) {
  blca_protein[1:8,1:8]
} else {
  message("Server may be down, please try again later...")
}

# Example including Antibody Information
blca_protein <- get_protein_data("blca_tcga_sequenced", TRUE)
if (nrow(blca_protein) > 10 & ncol(blca_protein) > 8) {
  blca_protein[1:8,1:8]
} else {
  message("Server may be down, please try again later...")
}

get_protein_info Retrieve Information on Antibodies Used for Protein Levels Determination

Description

Retrieve information on antibodies used by reverse-phase protein arrays (RPPA) to measure protein/phosphoprotein levels.

Usage

get_protein_info(csid = NULL, array_type = "protein_level",
glist = NULL)
Arguments

csid String corresponding to the Cancer Study Identifier
array_type String, c("protein_level", "phosphorylation"). Retrieve information about antibodies used for detecting total protein levels or phosphorylated levels of the protein product of the gene of interest
glist Character vector including one or more gene identifiers (ENTREZID or OFFICIAL_SYMBOL)

Value
data frame having one antibody per row and four columns

Author(s)

Damiano Fantini, <damiano.fantini@gmail.com>

References

- https://www.data-pulse.com/dev_site/TCGAretriever/

Examples

```r
info1 <- get_protein_info("blca_tcga", glist = c("TP53", "PTEN", "E2F1", "AKT1"))
if (nrow(info1) > 0) {
  message("Total protein levels information")
  info1
} else {
  message("Server may be down, please try again later...")
}
#
info2 <- get_protein_info("blca_tcga", "phosphorilation", c("TP53", "PTEN", "E2F1", "AKT1"))
if (nrow(info2) > 0) {
  message("Phospho-protein levels information")
  info2
} else {
  message("Server may be down, please try again later...")
}
```

make_groups Split Numeric Vectors in Groups

Description

Assign each element of a numeric vector to a group. Grouping is based on ranks: numeric values are sorted and then split in 2 or more groups. Values may be sorted in an increasing or decreasing fashion. The vector is returned in the original order. Labels may be assigned to each group.
**make_groups**

**Usage**

```r
def make_groups(num_vector, groups, group_labels = NULL, desc = FALSE)
```

**Arguments**

- `num_vector`: numeric vector. It includes the values to be assigned to the different groups.
- `groups`: integer. The number of groups that will be generated.
- `group_labels`: character vector. Labels for each group. Note that the length of `group_labels` has to be equal to the number of groups.
- `desc`: logical. If TRUE, the sorting is applied in a decreasing fashion.

**Value**

data.frame including the vector provided as argument in the original order ("value") and the grouping vector ("rank"). If labels are provided as an argument, group labels are also included in the data.frame ("labels").

**Author(s)**

Damiano Fantini, <damiano.fantini@gmail.com>

**References**

- [https://www.data-pulse.com/dev_site/TCGAretriever/](https://www.data-pulse.com/dev_site/TCGAretriever/)

**Examples**

```r
eexprs_geneX <- c(19.1, 18.4, 22.4, 15.5, 20.2, 17.4, 9.4, 12.4, 31.2, 33.2, 18.4, 22.1)
groups_num <- 3
groups_labels <- c("high", "med", "low")
make_groups(exprs_geneX, groups_num, groups_labels, desc = TRUE)
```
Index

*Topic internal
   basic_tcga_query, 2

basic_tcga_query, 2
expand_cases, 3
fetch_all_tcgadata, 3
get_cancer_studies, 4
get_cancer_types, 5
get_case_lists, 6
get_clinical_data, 7
get_ext_mutation, 8
get_genetic_profiles, 9
get_profile_data, 9
get_protein_data, 10
get_protein_info, 11
make_groups, 12