Package 'RTCGA'

October 25, 2015

Title The Cancer Genome Atlas Data Integration

Version 1.0.0

Date 2015-08-05

Author Marcin Kosinski <m.p.kosinski@gmail.com>, Przemyslaw Biecek <przemyslaw.biecek@gmail.com>

Maintainer Marcin Kosinski <m.p.kosinski@gmail.com>

Description The Cancer Genome Atlas (TCGA) Data Portal provides a platform for researchers to search, download, and analyze data sets generated by TCGA. It contains clinical information, genomic characterization data, and high level sequence analysis of the tumor genomes. The key is to understand genomics to improve cancer care. RTCGA package offers download and integration of the variety and volume of TCGA data using patient barcode key, what enables easier data possession. This may have an benefcial infuence on impact on development of science and improvement of patients' treatment. Furthermore, RTCGA package transforms TCGA data to tidy form which is convenient to use.

BugReports https://github.com/MarcinKosinski/RTCGA/issues

License GPL-2

LazyLoad yes

LazyData yes

Depends R (>= 3.2.0), knitr

Imports XML, assertthat, stringi, rvest, data.table, magrittr, xml2

Suggests testthat, pander

Repository Bioconductor

biocViews Software

VignetteBuilder knitr

NeedsCompilation no

R topics documented:

RTCGA-package *The Caner Genome Atlas data integration*

Description

The Cancer Genome Atlas (TCGA) Data Portal provides a platform for researchers to search, download, and analyze data sets generated by TCGA. It contains clinical information, genomic characterization data, and high level sequence analysis of the tumor genomes. The key is to understand genomics to improve cancer care. RTCGA package offers download and integration of the variety and volume of TCGA data using patient barcode key, what enables easier data possession. This may have an benefcial infuence on impact on development of science and improvement of patients' treatment. Furthermore, RTCGA package transforms TCGA data to form which is convenient to use in R statistical package. Those data transformations can be a part of statistical analysis pipeline which can be more reproducible with RTCGA

Details

For more detailed information visit **RTCGA** `wiki` on Github.

Author(s)

Marcin Kosinski [aut, cre] < `m.p.kosinski@gmail.com` >
Przemyslaw Biecek [aut] < `przemyslaw.biecek@gmail.com` >

See Also

Other RTCGA: `checkTCGA`; `datasetsTCGA`; `downloadTCGA`; `infoTCGA`; `readTCGA`

Examples

```
## Not run:
browseVignettes('RTCGA')

## End(Not run)
```

checkTCGA *Information about datasets from TCGA project*

Description

The checkTCGA function let's to check

- DataSets: TCGA datasets' names for current release date and cohort.
- Dates: TCGA datasets' dates of release.

Usage

```
checkTCGA(what, cancerType, date = NULL)
```

Arguments

what	One of DataSets or Dates.
cancerType	A character of length 1 containing abbreviation (Cohort code - http://gdac.broadinstitute.org/) of types of cancers to check for.
date	A NULL or character specifying from which date informations should be checked. By default (date = NULL) the newest available date is used. All available dates can be checked on http://gdac.broadinstitute.org/runs/ or by using checkTCGA('Dates') function. Required format 'YYYY-MM-DD'.

Details

- If what='DataSets' enables to check TCGA datasets' names for current release date and cohort.
- If what='Dates' enables to check dates of TCGA datasets' releases.

Value

- If what='DataSets' a vector of available datasets' names to pass to the downloadTCGA function.
- If what='Dates' a vector of available dates to pass to the downloadTCGA function.

See Also

Other RTCGA: RTCGA, RTCGA-package; datasetsTCGA; downloadTCGA; infoTCGA; readTCGA

Examples

```
## Not run:

##############################

# names for current release date and cohort
checkTCGA('DataSets', 'BRCA' )
checkTCGA('DataSets', 'OV', tail(checkTCGA('Dates'))[1] )
#checkTCGA('DataSets', 'OV', checkTCGA('Dates')[5] ) # error
```

```
# dates of TCGA datasets' releases.
checkTCGA('Dates')

#############################

# TCGA datasets' names availability for
# current release date and cancer type.

releaseDate <- '2015-06-01'
cancerTypes <- c('OV', 'BRCA')

cancerTypes %>% sapply(function(element){
  grep(x = checkTCGA('DataSets', element, releaseDate),
      pattern = 'humanmethylation450', value = TRUE) %>%
       as.vector()
       })

#############################

# TCGA genes' names and availability
# in Merge_rnaseqv2__... dataset
dir.create('data2')
sapply( cancerTypes, function(element){
tryCatch({
    downloadTCGA( cancerTypes = element,
                  dataSet = paste0('rnaseqv2__illuminahiseq_rnaseqv2__unc',
                  '_edu__Level_3__RSEM_genes_normalized__data.Level'),
                  destDir = 'data2',
                  date = releaseDate )},
    error = function(cond){
        cat('Error: Maybe there weren't rnaseq data for ', element, ' cancer.\n')
    }
)
})

# Paths to rna-seq data

sapply( cohorts, function( element ){
folder <- grep( paste0( '(_',element,'\\.', '|','_',element,'-FFPE)', '.*rnaseqv2'),
               list.files('data2/'), value = TRUE)
file <- grep( paste0(element, '.*rnaseqv2'), list.files( paste0( 'data2/',folder ) ),
              value = TRUE)
path <- paste0( 'data2/', folder, '/', file )
assign( value = path, x = paste0(element, '.rnaseq.path'), envir = .GlobalEnv)
})

rnaseqDir <- 'OV.rnaseq'


fread(rnaseqDir, select = c(1),
      data.table = FALSE,
      colClasses = 'character')[-1, 1]


## End(Not run)
```

datasetsTCGA *RTCGA.data - The Family of R Packages with Data from The Cancer Genome Atlas Study*

Description

Snapshots of the clinical, mutations, cnvs and rnaseq datasets from the 2015-06-01 release date are included in the RTCGA.data family that contains 4 packages:

- **RTCGA.rnaseq**
- **RTCGA.clinical**
- **RTCGA.mutations**
- **RTCGA.cnv**

Details

For more detailed information visit **RTCGA.data** website.

Author(s)

Marcin Kosinski [aut, cre] < m.p.kosinski@gmail.com >
Przemyslaw Biecek [aut] < przemyslaw.biecek@gmail.com >

See Also

Other RTCGA: RTCGA, RTCGA-package; checkTCGA; downloadTCGA; infoTCGA; readTCGA

Examples

```
# installation of packages containing snapshots
# of TCGA project's datasets

## Not run:
source('http://bioconductor.org/biocLite.R')
biocLite(RTCGA.clinical)
biocLite(RTCGA.mutations)
biocLite(RTCGA.rnaseq)
biocLite(RTCGA.cnv)

# use cases and examples + more data info
browseVignettes('RTCGA')

## End(Not run)
```

`downloadTCGA` *Download TCGA data*

Description

Enables to download TCGA data from specified dates of releases of concrete Cohorts of cancer types. Pass a name of required dataset to the `dataSet` parameter. By default the Merged Clinical dataSet is downloaded (value `dataSet = 'Merge_Clinical.Level_1'`) from the newest available date of the release.

Usage

```
downloadTCGA(cancerTypes, dataSet = "Merge_Clinical.Level_1", destDir,
  date = NULL, untarFile = TRUE, removeTar = TRUE)
```

Arguments

`cancerTypes` A character vector containing abbreviations (Cohort code) of types of cancers to download from http://gdac.broadinstitute.org/. For easy access from R check details below.

`dataSet` A part of the name of dataSet to be downloaded from http://gdac.broadinstitute.org/runs/. By default the Merged Clinical dataSet is downloaded (value `dataSet = 'Merge_Clinical.Level_1` Available datasets' names can be checked using checkTCGA function.

`destDir` A character specifying a directory into which `dataSets` will be downloaded.

`date` A `NULL` or character specifying from which date `dataSets` should be downloaded. By default (`date = NULL`) the newest available date is used. All available dates can be checked on http://gdac.broadinstitute.org/runs/ or by using checkTCGA function. Required format `'YYYY-MM-DD'`.

`untarFile` Logical - should the downloaded file be untarred. Default is `TRUE`.

`removeTar` Logical - should the downloaded `.tar` file be removed after untarring. Default is `TRUE`.

Details

All cohort names can be checked using: `sub( x = names( infoTCGA() ), '-counts', '' )`.

Value

No values. It only downloads files.

See Also

Other RTCGA: `RTCGA`, `RTCGA-package`; `checkTCGA`; `datasetsTCGA`; `infoTCGA`; `readTCGA`

Examples

```
## Not run:
dir.create( 'hre')

downloadTCGA( cancerTypes = 'ACC', dataSet = 'miR_gene_expression',
destDir = 'hre', date =  tail( checkTCGA('Dates'), 2 )[1] )


downloadTCGA( cancerTypes = c('BRCA', 'OV'), destDir = 'hre',
 date = tail( checkTCGA('Dates'), 2 )[1] )

## End(Not run)
```

---

infoTCGA *Information about cohorts from TCGA project*

Description

Function restores codes and counts for each cohort from TCGA project.

Usage

```
infoTCGA()
```

Value

A list with a tabular information from http://gdac.broadinstitute.org/.

See Also

Other RTCGA: RTCGA, RTCGA-package; checkTCGA; datasetsTCGA; downloadTCGA; readTCGA

Examples

```
infoTCGA()

(cohorts <- infoTCGA() %>%
rownames() %>%
   sub('-counts', '', x=.))
```

---

readTCGA *Read TCGA data to the tidy format*

Description

readTCGA function allows to read unzipped files:

- clinical data - Merge_Clinical.Level_1
- rnaseq data (genes' expressions) - Mutation_Packager_Calls.Level
- genes' mutations data - rnaseqv2__illuminahiseq_rnaseqv2

from TCGA project. Those files can be easily downloaded with downloadTCGA function. See examples.

Usage

```
readTCGA(path, dataType, ...)
```

Arguments

`path` If `dataType = 'clinical'` a directory to a `cancerType.clin.merged.txt` file. If `dataType = 'mutations'` a directory to the unzziped folder `Mutation_Packager_Calls.Lev` containing `.maf` files. If `dataType = 'rnaseq'` a directory to the uzziped file `rnaseqv2__illuminahiseq_rnaseqv2__unc_edu__Level_3__RSEM_genes_normalized__data.` See examples.

`dataType` One of `'clinical', 'rnaseq', 'mutations'` depending on which type of data users is trying to read in the tidy format.

`...` Further arguments passed to the as.data.frame.

Details

All cohort names can be checked using: `sub( x = names( infoTCGA() ), '-counts', '')`.

Value

An output:

- If `dataType = 'clinical'` a `data.frame` with clinical data.
- If `dataType = 'rnaseq'` a `data.frame` with rnaseq data.
- If `dataType = 'mutations'` a `data.frame` with mutations data.

See Also

Other RTCGA: `RTCGA`, `RTCGA-package`; `checkTCGA`; `datasetsTCGA`; `downloadTCGA`; `infoTCGA`

Examples

```
## Not run:

##############
##### clinical
##############

dir.create('hre')

# downloading clinical data
downloadTCGA( cancerTypes = c('BRCA', 'OV'), destDir = 'hre' )


# reading datasets
sapply( c('BRCA', 'OV'), function( element ){
folder <- grep( paste0( '(_',element,'\\.', '|','_',element,'-FFPE)', '.*Clinical'),
      list.files('hre/'),value = TRUE)
path <- paste0( 'hre/', folder, '/', element, '.clin.merged.txt')
assign( value = readTCGA( path, 'clinical' ),
         x = paste0(element, '.clin.data'), envir = .GlobalEnv)
         })

##############
```

```
##### rnaseq
##############

dir.create('data2')

# downloading rnaseq data
downloadTCGA( cancerTypes = 'BRCA',
dataSet = 'rnaseqv2__illuminahiseq_rnaseqv2__unc_edu__Level_3__RSEM_genes_normalized__data.Level',
destDir = 'data2' )

# shortening paths and directories
list.files( 'data2/') %>%
    file.path( 'data2', .) %>%
    file.rename( to = substr(.,start=1,stop=50))

# reading data
list.files( 'data2/') %>%
    file.path( 'data2', .) -> folder

folder %>%
    list.files %>%
    file.path( folder, .) %>%
    grep( pattern = 'illuminahiseq', x = ., value = TRUE) -> pathRNA
readTCGA( path = pathRNA, dataType = 'rnaseq' ) -> my_data


##############
##### mutations
##############

dir.create('data3')


downloadTCGA( cancerTypes = 'OV',
              dataSet = 'Mutation_Packager_Calls.Level',
              destDir = 'data3' )

# reading data
list.files( 'data3/') %>%
    file.path( 'data3', .) -> folder

readTCGA(folder, 'mutations') -> mut_file


## End(Not run)
```

Index