Package 'rpubchem'

January 27, 2015

Version 1.5.0.2

Date 2014-04-24

Title rpubchem - Interface to the PubChem Collection

Author Rajarshi Guha <rajarshi.guha@gmail.com>

Maintainer Rajarshi Guha <rajarshi.guha@gmail.com>

Depends R (>= 2.0.0)

Imports XML, car, RCurl, RJSONIO

License GPL (>= 2)

Description Access PubChem data (compounds, substance, assays). Structural information is provided in the form of SMILES strings. This package only provides access to a subset of the precalculated data stored by PubChem. Bio-assay data can be accessed to obtain descriptions as well as the actual data. It is also possible to search for assay ID's by keyword. Currently the main limitation is that only 1000 molecules can be downloaded at a time from the PubChem servers

NeedsCompilation no

Repository CRAN

Date/Publication 2014-04-24 15:23:55

R topics documented:

find.assay.id *Search for Assay ID's*

Description

PubChem allows one to obtain the ID's of bio-assays that match a search string. This function uses the Entrez interface to supply a query string and return the ID's of matching bio-assays

Usage

```
find.assay.id(query, quiet=TRUE)
```

Arguments

query	A character string containing the query
quiet	If FALSE the output is verbose

Value

A numeric vector containing the ID's that match the search query

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

get.assay.desc, get.assay

Examples

```
## Not run:
## find assay ID's related to yeast
aids <- find.assay.id('yeast')

## get the description of the first 10 assays
descs <- lapply( lapply(as.list(aids[1:10]), get.assay.desc), function(x)
x$assay.desc )

## End(Not run)
```

`get.aid.by.cid` *Get Assay ID Based on Compound Activity*

Description

This function allows you to identify PubChem assays in which a compound, specified by CID, has been tested in. The method uses the PubChem Power User Gateway (PUG) and as a result can be slow.

The function can be used to identify assays in which the CID is active, inactive, discrepant in, or simply the assays in which it has been tested.

However, since the query can be slow, it is useful have the raw data. In such a case, one can get an indicator matrix which can be processed to identify assays in which the compound are active, inactive, etc.

Usage

```
get.aid.by.cid(cid, type="raw", quiet=TRUE)
```

Arguments

`cid`	A single compound ID
`type`	What type of query should be performed. Valid values are 'active', 'inactive', 'discrepant', 'tested' or 'raw'.
`quiet`	If `FALSE`, output is verbose

Details

Since the method employs PUG, it is rather slow. In addition it can also fail intermittently. In general, as long as the CID is correctly specified, a pause between subsequent calls usually allows the query to successfully complete

Value

If the `type` argument was one of 'active', 'inactive', 'discrepant' or 'tested' a numeric vector of assay IDs. If 'raw' was specifiec, a binary matrix is returned with the number of rows equal to the number of assays the compound was tested in and 4 columns - aid, active, inactive, discrepant. If a row has a a 1 in columns 2 to 4, it indicates that the compound is either active, inactive or discrepant.

In case an invalid CID was specified or the query failed, `NULL` is returned.

Author(s)

Rajarshi Guha <rguha@indiana.edu>

See Also

`get.assay`

get.assay *Get a PubChem Bio-Assay*

Description

PubChem provides access to a number of bio-assays which are generally results obtained from High Throughput Screens (HTS). The number of observations in a given assay can be as high as 42000. This method allows one to obtain the assay data for a given assay ID. Assay ID's can be obtained using a text search using the `find.assay.id` function.

Usage

```
get.assay(aid, quiet=TRUE)
```

Arguments

`aid`	An assay ID
`quiet`	If `FALSE` the output is verbose

Details

The assay data are obtained for a variety of targets using a variety of techniques. As a result though each assay dataset contains a set of fixed fields, they can have additional fields.

Value

A data frame with the observations in the rows. The number of columns varies from assay to assay. Any assay will, however, have the following columns:

`PUBCHEM.SID` PubChem SID

`PUBCHEM.CID` PubChem CID

`PUBCHEM.ACTIVITY.OUTCOME`
Activity outcome

`PUBCHEM.ACTIVITY.SCORE`
Activity score, higher is more active

`PUBCHEM.ASSAYDATA.COMMENT`
Test result specific comment

The activity outcome field is provided as a numeric but is recoded as described in the PubChem documentation. The remaining fields are obtained by parsing the description file for the corresponding assay.

In addition to the usual attributes for a `data.frame` object this function adds some extra attributes:

- descriptionA short description of the assay
- commentsComments associated with the assay
- typesA named `list` where the names are the assay specific field names. Each element of the `list` is a 2-element vector containing the description of the field along with the units. In case the field is unitless the unit is NA

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

get.assay.desc, find.assay.id

---

get.assay.desc *Get An Assay Description*

Description

PubChem stores a number of pieces of information for each bio-assay. These include the description of the assay, related comments as well as type information (name, units, description) for the extra columns in the assay data.

This method accesses the description information and extracts a subset of that available.

Usage

```
get.assay.desc(aid)
```

Arguments

aid	A valid assay ID. This can be obtained using find.assay.id if not already known

Value

A list object with the following named components

assay.desc	A short description of the assay
assay.comments	A list of comments for the assay
types	A matrix with 3 columns. The first column is the name of the assay specific columns. The second column contains the descriptions of each assay specific column. The final column lists the units for each of the assay specific columns. In case an assay column is unitless, the value of the unit for that column is NA

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

find.assay.id, get.assay

`get.assay.summary` *Get a PubChem Bio-Assay Summary*

Description

Obtain the assay summary for a given assay id.

Usage

```
get.assay.summary(aid)
```

Arguments

`aid`	An assay ID

Details

The Pubchem assay summary has a number of sections, with each section seperated into chunks. The method will concatenate all chunks for a given section.

Value

A list with three elements

- Comment
- Protocol
- Description

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

`get.assay`, `get.assay.desc`, `find.assay.id`

get.cid *Get PubChem Compound Information*

Description

The PubChem compound collection stores a variety of information for each molecule. These include canonical SMILES, molecular properties, substance associations, synonyms etc.

This function will extract a subset of the molecular property information for one or more compound ID's

Usage

```
get.cid(cid, quiet=TRUE, from.file=FALSE)
```

Arguments

cid	A vector of one or more compound ID's
quiet	If FALSE, output is verbose
from.file	If TRUE then the first argument is considered to be the name of a file containing the XML data. If FALSE the first argument must be a sequence of compound ID's and the data will be downloaded from the PubChem FTP site

Details

Processing a large number of compound ID's can take a long time. For large numbers of CID's the resultant XML file can be many megabytes. This may take a long time to download. After download it takes approximate 20 sec to process a 23MB data file.

It should also be noted that the data files are downloaded using the R interface to Curl. In addition, the PubChem servers do not allow very large query URL's. This limits the number of compound ID's that can be directly pulled of the PubChem servers to about 1000

Value

A data.frame with 9 columns:

CID	The compound ID
IUPACName	The IUPAC name of the compound
CanonicalSmiles	The canonical SMILES for the compound
MolecularWeight	Molecular weight
TotalFormalCharge	The formal charge
MolecularFormula	The molecular formula

TPSA	Topological polar surface area
HeavyAtomCount	Heavy atom count
FormalCharge	Total formal charge
HydrogenBondDonor	Hydrogen bond donor count
HydrogenBondAcceptor	Hydrogen bond acceptor count

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

get.assay, get.sid, get.sid.list

---

get.sid *Get PubChem Substance Information*

Description

The PubChem substance collection stores a variety of information for each molecule. These include canonical SMILES, molecular properties, substance associations, synonyms etc.

This function will extract a subset of the molecular property information for one or more compound ID's

Usage

```
get.sid(sid, quiet=TRUE, from.file=FALSE)
```

Arguments

sid	A vector of one or more compound ID's
quiet	If FALSE, output is verbose
from.file	If TRUE then the first argument is considered to be the name of a file containing the XML data. If FALSE the first argument must be a sequence of compound ID's and the data will be downloaded from the PubChem FTP site

Details

Processing a large number of substance ID's can take a long time. For large numbers of SID's the resultant XML file can be many megabytes. This may take a long time to download. After download it takes approximate 20 sec to process a 23MB data file.

It should also be noted that the data files are downloaded using the R interface to Curl. In addition, the PubChem servers do not allow very large query URL's. This limits the number of substance ID's that can be directly pulled of the PubChem servers to about 1000

Value

A `data.frame` with 9 columns:

`SID`	The substance ID
`IUPACName`	The IUPAC name of the compound
`CanonicalSmiles`	The canonical SMILES for the compound
`MolecularWeight`	Molecular weight
`TotalFormalCharge`	The formal charge
`MolecularFormula`	The molecular formula
`TPSA`	Topological polar surface area
`HeavyAtomCount`	Heavy atom count
`FormalCharge`	Total formal charge
`HydrogenBondDonor`	Hydrogen bond donor count
`HydrogenBondAcceptor`	Hydrogen bond acceptor count

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

`get.assay`, `get.cid`, `get.sid.list`

`get.sid.list` *Get PubChem Substance ID's Associated With A Compound*

Description

Each unique compound is associated with a number of substances. Given a CID it is possible to determine the associated substance ID's.

Usage

```
get.sid.list(cid, quiet=TRUE, from.file=FALSE)
```

Arguments

cid	A vector of one or more compound ID's
quiet	If FALSE, output is verbose
from.file	If TRUE then the first argument is considered to be the name of a file containing the XML data. If FALSE the first argument must be a sequence of compound ID's and the data will be downloaded from the PubChem FTP site

Details

Processing a large number of compound ID's can take a long time. For large numbers of CID's the resultant XML file can be many megabytes. This may take a long time to download. After download it takes approximate 60 sec to process a 23MB data file.

It should also be noted that the data files are downloaded using the R interface to Curl. In addition, the PubChem servers do not allow very large query URL's. This limits the number of compound ID's that can be directly pulled of the PubChem servers to about 1000

Value

A list object, the name of each element is the CID and the value is the SID's that are associated with the compound in question.

Author(s)

Rajarshi Guha <rajarshi.guha@gmail.com>

See Also

get.cid, get.assay

Index