Package 'enigma'

February 17, 2017

Title Client for the 'Enigma' 'API'

Description The company 'Enigma' (<http://enigma.io>) holds many public 'datasets' from governments, companies, universities, and organizations. 'Enigma' provides an 'API' for data, 'metadata', and statistics on each of the 'datasets'. 'enigma' is a client to interact with the 'Enigma' 'API', including getting the data and 'metadata' for 'datasets' in 'Enigma', as well as collecting statistics on 'datasets'. In addition, you can download a 'gzipped' 'csv' file of a 'dataset' if you want the whole 'dataset'. An 'API' key from 'Enigma' is required to use 'enigma'.

Version 0.3.0

License MIT + file LICENSE

LazyData true

URL https://github.com/ropengov/enigma

BugReports https://github.com/ropengov/enigma/issues

VignetteBuilder knitr

Imports utils, crul (>= 0.2.0), jsonlite, plyr, xml2, tibble

Suggests roxygen2 (>= 6.0.1), knitr, testthat, covr

RoxygenNote 6.0.1

NeedsCompilation no

Author Scott Chamberlain [aut, cre]

Maintainer Scott Chamberlain <myrmecocystus@gmail.com>

Repository CRAN

Date/Publication 2017-02-17 23:36:59

R topics documented:

enigma *enigma, an R client for Enigma.io*

Description

Enigma holds government data and provides a really nice set of APIs for data, metadata, and stats on each of the datasets. That is, you can request a dataset itself, metadata on the dataset, and summary statistics on the columns of each dataset.

package API

- enigma_data - Fetch and dataset, and filter on columns or rows.
- enigma_metadata - Get metadata on datasets.
- enigma_stats - Get columnwise statistics on datasets.
- enigma_fetch - Get gzipped csv of a dataset. Goes along with enigma_read
- rate_limit - Get columnwise statistics on datasets.

Authentication

An API key is required to use this package. You can supply your key in each function call, or store in your .Renviron file like ENIGMA_KEY=your key), or in your .Rprofile file as options(enigmaKey = "<your key>"), Obtain an API key by creating an account with Enigma at http://enigma.io, then obtain an API key from your account page.

enigma_data *Fetch a dataset from Enigma.*

Description

Fetch a dataset from Enigma.

Usage

```
enigma_data(dataset = NULL, limit = 500, select = NULL,
  conjunction = NULL, sort = NULL, page = NULL, where = NULL,
  search = NULL, key = NULL, ...)
```

Arguments

`dataset` Dataset name. Required.

`limit` (numeric) Number of rows of the dataset to return. Default (and max): 500

`select` (character) Vector of columns to be returned with each row. Default is to return all columns.

`conjunction` one of "and" or "or". Only applicable when more than one `search` or `where` parameter is provided. Default: "and"

`sort` (character) Sort rows by a particular column in a given direction. + denotes ascending order, - denotes descending. See examples.

`page` (numeric) Paginate row results and return the nth page of results. Pages are calculated based on the current limit, which defaults to 50.

`where` (character) Filter results with a SQL-style "where" clause. Only applies to numerical columns - use the `search` parameter for strings. Valid operators are >, < and =. Only one `where` clause per request is currently supported.

`search` (character) Filter results by only returning rows that match a search query. By default this searches the entire table for matching text. To search particular fields only, use the query format "@fieldname query". To match multiple queries, the | (or) operator can be used eg. "query1|query2".

`key` (character) Required. An Enigma API key. Supply in the function call, or store in your `.Renviron` file like `ENIGMA_KEY=your key`), or in your `.Rprofile` file as `options(enigmaKey = "<your key>")`, Obtain an API key by creating an account with Enigma at `http://enigma.io`, then obtain an API key from your account page.

`...` Named curl options passed on to `HttpClient`

Value

A list with items:

- success - a boolean if query was successful or not
- datapath - the dataset path (this is not a file path on your machine)
- info - a list of length 6 with:
  - rows_limit - rows limit
  - total_results - total items found (likely more than was returned)
  - total_pages - total pages found (see also `current_page`)
  - current_page - page returned (see also `total_pages`)
  - calls_remaining - number of requests remaining
  - seconds_remaining - seconds remaining before your rate limit resets
- result - a data.frame/tibble - columns depend on the data source returned

References

`https://app.enigma.io/api#data`

Examples

```
## Not run:
# After obtaining an API key from Enigma's website, pass in your key to
# the function call or set in your options (see above instructions for the
# key parameter) If you pass in your key to the function call use the
# key parameter

# White house visitor list
enigma_data(dataset='us.gov.whitehouse.visitor-list', limit = 10)

# White house visitor list - selecting three columns
enigma_data(dataset='us.gov.whitehouse.visitor-list',
  select=c('namelast','visitee_namelast', 'last_updatedby'))

# White house visitor list - sort by last name
enigma_data(dataset='us.gov.whitehouse.visitor-list', sort='+namelast',
  limit = 10)

# White house visitor list - get rows where total_people > 5
enigma_data(dataset='us.gov.whitehouse.visitor-list',
  where='total_people > 5', limit = 10)

# White house visitor list - search for Vitale in full name field
## remove the 2nd at symbol before running
# enigma_data(dataset='us.gov.whitehouse.visitor-list',
#   search='@namefull=Vitale')

# White house visitor list - search for SOPHIA in first name field
## remove the 2nd at symbol before running
# enigma_data(dataset='us.gov.whitehouse.visitor-list',
#   search='@namefirst=SOPHIA')

# Domestic Market Flight Statistics (Final Destination)
dataset='us.gov.dot.rita.trans-stats.air-carrier-statistics.t100d-market-all-carrier'
enigma_data(dataset=dataset, limit = 10)

# Search for 'casa' in the Mexico's Social Security Department dataset
enigma_data('gov.mx.imss.compras.main', search='casa',
  select=c('proveedor','fecha'), limit = 10)

## End(Not run)
```

enigma_fetch *Download a gzipped csv file of a dataset.*

Description

Download a gzipped csv file of a dataset.

Usage

```
enigma_fetch(dataset = NULL, select = NULL, search = NULL, where = NULL,
  conjunction = NULL, sort = NULL, path = NULL, key = NULL,
  poll_sleep = 0, ...)

enigma_read(input)
```

Arguments

`dataset` Dataset name. Required.

`select` (character) Vector of columns to be returned with each row. Default is to return all columns.

`search` (character) Filter results by only returning rows that match a search query. By default this searches the entire table for matching text. To search particular fields only, use the query format "@fieldname query". To match multiple queries, the | (or) operator can be used eg. "query1|query2".

`where` (character) Filter results with a SQL-style "where" clause. Only applies to numerical columns - use the `search` parameter for strings. Valid operators are >, < and =. Only one `where` clause per request is currently supported.

`conjunction` one of "and" or "or". Only applicable when more than one `search` or `where` parameter is provided. Default: "and"

`sort` (character) Sort rows by a particular column in a given direction. + denotes ascending order, - denotes descending. See examples.

`path` File name and path of output zip file. Defaults to write a zip file to your home directory with name of the dataset, and file extension `.csv.gz`.

`key` (character) Required. An Enigma API key. Supply in the function call, or store in your `.Renviron` file like `ENIGMA_KEY=your key`), or in your `.Rprofile` file as `options(enigmaKey = "<your key>")`, Obtain an API key by creating an account with Enigma at `http://enigma.io`, then obtain an API key from your account page.

`poll_sleep` (integer) Time to sleep between polling events to fetch data. For very large datasets, it could take a while to be ready. By default, we poll continuously. If you are requesting a large dataset and/or have not much left on your allowed requests with Enigam (see `rate_limit`) you may want to insert some sleep time between pollings.

`...` Named curl options passed on to `HttpClient`

`input` The output from `enigma_fetch` or a path to a file d ownloaded from Enigma.io

Details

Note that `enigma_fetch` downloads the file, and gives back a path to the file. In a separte function, `enigma_read`, you can read in the data. `enigma_fetch` doesn't read in data in case the file is very large which may make your R session crash or slow down significantly.

This function makes a request to ask Enigma to get a download ready. We then poll the provided URL from Enigma until it is ready. Once ready we fetch it and write it to disk.

If file exists already, we overwrite it.

Value

A (character) path to the file on your machine

References

https://app.enigma.io/api#exporting

Examples

```
## Not run:
## After obtaining an API key from Enigma's website, pass in your key to
## the function call or set in your options (see above instructions for the
## key parameter) If you pass in your key to the function call use the
## key parameter

# Fetch a dataset
res <- enigma_fetch(dataset = 'edu.umd.start.gtd')
enigma_read(res)

# Use the select parameter to limit fields returned
res <- enigma_fetch(dataset = 'edu.umd.start.gtd',
   select = c("country_txt", "resolution", "attacktype1"))
enigma_read(res)

# Use the search parameter to query entire table or particular fields
res <- enigma_fetch(dataset = 'edu.umd.start.gtd', search = "armed")
enigma_read(res)

# Use the search parameter to query entire table or particular fields
res <- enigma_fetch(dataset = 'edu.umd.start.gtd',
   where = "nkill > 0", select = c("country_txt", "attacktype1", "nkill"))
enigma_read(res)

## End(Not run)
```

enigma_metadata *Search for metadata on a dataset from Enigma.*

Description

Search for metadata on a dataset from Enigma.

Usage

```
enigma_metadata(dataset = NULL, key = NULL, ...)
```

Arguments

`dataset` Dataset name. Required.

`key` (character) Required. An Enigma API key. Supply in the function call, or store in your `.Renviron` file like `ENIGMA_KEY=your key`), or in your `.Rprofile` file as `options(enigmaKey = "<your key>")`, Obtain an API key by creating an account with Enigma at `http://enigma.io`, then obtain an API key from your account page.

`...` Named curl options passed on to `HttpClient`

Details

Notice when you run the examples that the format of output is different for the "parent nodes" vs. the "table nodes". Where the parent nodes have ouput$meta slots for paths, immediate nodes and children tables, while the table nodes have ouput$meta slots for info, table, ancestor datapaths, database boundary datapath, database boundary label, database boundary tables, and paths, and an additional slot for description of table column attributes.

Value

A list of length 3:

- success - a boolean if query was successful or not
- datapath - the dataset path (this is not a file path on your machine)
- info - a list of length 6 with:
  - paths - paths under the dataset
  - immediate_notes - children nodes
  - children_tables - children tables

References

`https://app.enigma.io/api#metadata`

Examples

```
## Not run:
## After obtaining an API key from Enigma's website, pass in your key to the
## function call or set in your options (see above instructions for the
## key parameter) If you pass in your key to the function call use the key
## parameter

## UCLA Ethnic power relations dataset
enigma_metadata(dataset = 'edu.ucla.epr')

# Table node response attributes
## US white house visitor list table
enigma_metadata(dataset = 'us.gov.whitehouse.visitor-list')

## End(Not run)
```

enigma_stats *Get statistics on columns of a dataset from Enigma.*

Description

Get statistics on columns of a dataset from Enigma.

Usage

```
enigma_stats(dataset = NULL, select, conjunction = NULL, operation = NULL,
  by = NULL, of = NULL, limit = 500, search = NULL, where = NULL,
  sort = NULL, page = NULL, key = NULL, ...)
```

Arguments

`dataset` Dataset name. Required.

`select` (character) Column to get statistics on. Required.

`conjunction` one of "and" or "or". Only applicable when more than one `search` or `where` parameter is provided. Default: "and"

`operation` (character) Operation to run on a given column. For a numerical column, valid operations are sum, avg, stddev, variance, max, min and frequency. For a date column, valid operations are max, min and frequency. For all other columns, the only valid operation is frequency. Defaults to all available operations based on the column's type.

`by` (character) Compound operation to run on a given pair of columns. Valid compound operations are sum and avg. When running a compound operation query, the `of` parameter is required (see below).

`of` (character) Numerical column to compare against when running a compound operation. Required when using the `by` parameter. Must be a numerical column.

`limit` (numeric) Limit the number of frequency, compound sum, or compound average results returned. Max: 500; Default: 500.

`search` (character) Filter results by only returning rows that match a search query. By default this searches the entire table for matching text. To search particular fields only, use the query format "@fieldname query". To match multiple queries, the | (or) operator can be used eg. "query1|query2".

`where` (character) Filter results with a SQL-style "where" clause. Only applies to numerical columns - use the `search` parameter for strings. Valid operators are >, < and =. Only one `where` clause per request is currently supported.

`sort` (character) Sort frequency, compound sum, or compound average results in a given direction. + denotes ascending order, - denotes descending

`page` (numeric) Paginate frequency, compound sum, or compound average results and return the nth page of results. Pages are calculated based on the current limit, which defaults to 500.

key (character) Required. An Enigma API key. Supply in the function call, or store in your .Renviron file like ENIGMA_KEY=your key), or in your .Rprofile file as options(enigmaKey = "<your key>"), Obtain an API key by creating an account with Enigma at http://enigma.io, then obtain an API key from your account page.

... Named curl options passed on to HttpClient

Value

A list with items:

- success - a boolean if query was successful or not
- datapath - the dataset path (this is not a file path on your machine)
- info - a list of length 6 with:
  - column - a list of information on the variable you requested stats on
  - operations - a list of the operations you requested
  - rows_limit - rows limit
  - total_results - total items found (likely more than was returned)
  - total_pages - total pages found (see also current_page)
  - current_page - page returned (see also total_pages)
  - calls_remaining - number of requests remaining
  - seconds_remaining - seconds remaining before your rate limit resets
- result - a named list of objects - depends on the data source returned

References

https://app.enigma.io/api#stats

Examples

```
## Not run:
# After obtaining an API key from Enigma's website, pass in your key to
# the function call or set in your options (see above instructions for the
# key parameter) If you pass in your key to the function call use the
# key parameter

# stats on a varchar column
x <- 'gov.mx.imss.compras.main'
enigma_stats(x, select='provider_id', limit = 10)

# stats on a numeric column
enigma_stats(x, select='serialid', limit = 10)

# stats on a date column
pakistan <- 'gov.pk.secp.business-registry.all-entities'
enigma_metadata(dataset=pakistan)
enigma_stats(dataset=pakistan, select='registration_date', limit = 10)
```

```
# stats on a date column, by the average of a numeric column
aust <- 'gov.au.government-spending.federal-contracts'
enigma_metadata(dataset=aust)
enigma_stats(dataset=aust, select='contractstart', by='avg', of='value',
  limit = 10)

# Get frequency of distances traveled
## get columns for the air carrier dataset
dset <- 'us.gov.dot.rita.trans-stats.air-carrier-statistics.t100d-market-all-carrier'
enigma_metadata(dset)$columns$table[,c(1:4)]
enigma_stats(dset, select='distance', limit = 10)

## End(Not run)
```

rate_limit *Get rate limit data.*

Description

Get rate limit data.

Usage

```
rate_limit(key = NULL, ...)
```

Arguments

key	(character) Required. An Enigma API key. Supply in the function call, or store in your .Renviron file like ENIGMA_KEY=your key), or in your .Rprofile file as options(enigmaKey = "<your key>"), Obtain an API key by creating an account with Enigma at http://enigma.io, then obtain an API key from your account page.
...	Named curl options passed on to HttpClient

Examples

```
## Not run:
rate_limit()

## End(Not run)
```

Index