

Package ‘ecoengine’

August 17, 2017

Type Package

Title Programmatic Interface to the Web Service Methods Provided by UC Berkeley's Natural History Data

Version 1.11.0

Description The ecoengine ('ecoengine'; <<https://ecoengine.berkeley.edu/>>). provides access to more than 5 million georeferenced specimen records from the University of California, Berkeley's Natural History Museums.

License MIT + file LICENSE

URL <https://github.com/ropensci/ecoengine>

BugReports <https://github.com/ropensci/ecoengine/issues>

LazyData yes

Depends R (>= 3.1.1)

Imports magrittr, data.table, httr (>= 0.3), plyr, assertthat, dplyr, whisker, jsonlite, lubridate

Suggests testthat, knitr, ggplot2, rmarkdown

RoxygenNote 6.0.1

NeedsCompilation no

Author Karthik Ram [aut, cre]

Maintainer Karthik Ram <karthik.ram@gmail.com>

Repository CRAN

Date/Publication 2017-08-17 15:39:59 UTC

R topics documented:

california_counties	2
checklist_details	2
ecoengine	3
ee_about	3
ee_cbind	4

ee_checklists	5
ee_footprints	5
ee_observations	6
ee_pages	8
ee_paginator	9
ee_photos	9
ee_search	11
ee_search_obs	12
ee_sources	13
full_sensor_list	13
print.ecoengine	14
view_photos	14

Index 15

california_counties *california_counties*

Description

A data.frame containing list of California counties. Useful for most functions since the data is primarily state based.

Usage

```
data(california_counties)
```

Format

A data table with 58 rows and 3 variables. Contains county name, FIPS state code (06 in this case) and FIPS county code. See [Wikipedia](#) for more information.

checklist_details *Checklist details*

Description

Will return details on any checklist

Usage

```
checklist_details(list_name, ...)
```

Arguments

list_name	URL of a checklist
...	Additional arguments (currently not implemented)

Value

data.frame

See Also

[ee_checklists](#)

Examples

```
## Not run:
spiders <- ee_checklists(subject = "Spiders")
# Now retrieve all the details for each species on both lists
library(plyr)
spider_details <- ldply(spiders$url, checklist_details)

## End(Not run)
```

ecoengine

ecoengine

Description

The **Berkeley Ecoengine** provides an open API to a wealth of museum data contained in the **Berkeley natural history museums**. This R package provides a programmatic interface to this rich repository of data allowing for the data to be easily analyzed and visualized or brought to bear in other contexts. This vignette provides a brief overview of the package's capabilities. The current version of the package works with API version 0.0.3

Details

The **Berkeley Ecoengine** provides an open API to a wealth of museum data contained in the **Berkeley natural history museums**. This R package provides a programmatic interface to this rich repository of data allowing for the data to be easily analyzed and visualized or brought to bear in other contexts. This vignette provides a brief overview of the package's capabilities. The current version of the package works with API version 0.0.3

ee_about

About the Berkeley Ecoinformatics Engine

Description

Function returns the current status of fast-evolving API. Returns endpoints and category. Default return is a list but one can also request a nicely formatted data.frame by setting the `as.df` argument to TRUE.

Usage

```
ee_about(as.df = TRUE, type = NA)
```

Arguments

as.df	FALSE Returns a list unless this set to TRUE
type	The type of end point. Options include data, meta-data, and actions

Value

list

Examples

```
ee_about()
# set as.df = FALSE to return a list rather than a data.frame
ee_about(as.df = FALSE)
# You can also filter by methods by data, meta-data, and actions.
# ee_about(type = "data")
# ee_about(type = "meta-data")
# ee_about(type = "actions")
```

ee_cbind

ee_cbind

Description

Allows for combining split ecoengine calls (e.g. paginated calls) back into one single result object

Usage

```
ee_cbind(results)
```

Arguments

results	A list of objects of class ecoengine
---------	--------------------------------------

Examples

```
## Not run:
x1 <- ee_observations(genus = "Lynx", page = 1)
x2 <- ee_observations(genus = "Lynx", page = 2)
x12 <- ee_cbind(list(x1, x2))

## End(Not run)
```

ee_checklists	<i>Ecoengine checklists</i>
---------------	-----------------------------

Description

Retrieves existing checklists from the ecoengine database

Usage

```
ee_checklists(subject = NULL, foptions = list())
```

Arguments

subject	Enter one of the following subjects: Mammals, Mosses, Beetles, Spiders, Amphibians, Ants, Fungi, Lichen, Plants.
foptions	A list of additional arguments for httr. There is no reason to use this argument except for debugging purposes.

Value

data.frame

Examples

```
# all_lists <- ee_checklists()
# mammals_list <- ee_checklists(subject = "Mammals")
# spiders <- ee_checklists(subject = "Spiders")
```

ee_footprints	<i>ee_footprints</i>
---------------	----------------------

Description

List of ecoengine footprints.

Usage

```
ee_footprints(foptions = list())
```

Arguments

foptions	A list of additional arguments for httr. There is no reason to use this argument except for debugging purposes.
----------	---

Value

data.frame

Examples

```
# footprints <- ee_footprints()
```

ee_observations	<i>Observations List</i>
-----------------	--------------------------

Description

API endpoint that represents a list of observations.

Usage

```
ee_observations(page = NULL, page_size = 1000, country = "United States",
  state_province = NULL, county = NULL, kingdom = NULL, phylum = NULL,
  order = NULL, class = NULL, family = NULL, genus = NULL,
  scientific_name = NULL, kingdom__exact = NULL, phylum__exact = NULL,
  order__exact = NULL, class__exact = NULL, family__exact = NULL,
  genus__exact = NULL, scientific_name__exact = NULL, remote_id = NULL,
  collection_code = NULL, source = NULL, min_date = NULL,
  max_date = NULL, georeferenced = FALSE, bbox = NULL, exclude = NULL,
  extra = NULL, quiet = FALSE, progress = FALSE, foptions = list())
```

Arguments

page	Page number
page_size	The number of observations per page returned by the query. The package default is 250 observations. The API default, however, is 10. Setting this number higher will require fewer calls (i.e. pages) but a large size may also fail or timeout.
country	country name
state_province	description needed.
county	California county. See <code>data(california_counties)</code>
kingdom	kingdom name
phylum	phylum name
order	order name
class	class name
family	family name
genus	genus name.
scientific_name	A full scientific name
kingdom__exact	exact kingdom name
phylum__exact	exact phylum name
order__exact	exact order name

class__exact	class name
family__exact	exact family name
genus__exact	exact genus name
scientific_name__exact	exact scientific name
remote_id	remote ID
collection_code	collections code
source	data source. See ee_sources
min_date	Lower date bound. Most be in format Y-m-D h:m:s. This package uses lubridate's ymd_hms function to format dates.
max_date	upper date bound. Most be in format Y-m-D h:m:s. This package uses lubridate's ymd_hms function to format dates.
georeferenced	Default is FALSE. Set to TRUE to return only georeferenced records.
bbox	Set a bounding box for your search. Use format bbox=-124,32,-114,42. Order is min Longitude , min Latitude , max Longitude , max Latitude. Use http://boundingbox.klokantech.com/ this website to quickly grab a bounding box (set format to csv on lower right)
exclude	Default is NULL. Pass a list of fields to exclude.
extra	Default is NULL. Pass a list of extra fields to be returned. Additional fields include: # "id", "record", "source", "remote_resource", "begin_date", "end_date", # "collection_code", "institution_code", "state_province", "county", # "last_modified", "original_id", "geometry", "coordinate_uncertainty_in_meters" # "md5", "scientific_name", "observation_type", "date_precision", "locality" # "earliest_period_or_lowest_system", "latest_period_or_highest_system", "kingdom" # "phylum", "class", "order", "family", "genus", "specific_epithet", # "infraspecific_epithet", "minimum_depth_in_meters", "maximum_depth_in_meters", # "maximum_elevation_in_meters", "minimum_elevation_in_meters", "catalog_number" # "preparations", "sex", "life_stage", "water_body", "country", "individual_count", "associated_resources"
quiet	Default is FALSE. Set to TRUE to suppress messages.
progress	Progress bar is set to TRUE by default. Use FALSE especially when calling functions inside documents.
foptions	A list of additional arguments for httr. There is no reason to use this argument except for debugging purposes.

Value

data.frame

Examples

```
# vulpes <- ee_observations(genus = "vulpes")
## Not run:
# pinus <- ee_observations(scientific_name = "Pinus", page_size = 100)
# lynx_data <- ee_observations(genus = "Lynx")
```

```

# Georeferenced data only
# lynx_data <- ee_observations(genus = "Lynx", georeferenced = TRUE)
# animalia <- ee_observations(kingdom = "Animalia")
# Artemisia <- ee_observations(scientific_name = "Artemisia douglasiana")
# asteraceae <- ee_observations(family = "asteraceae")
# vulpes <- ee_observations(genus = "vulpes")
# Anas <- ee_observations(scientific_name = "Anas cyanoptera", page = "all")
# loons <- ee_observations(scientific_name = "Gavia immer", page = "all")
# plantae <- ee_observations(kingdom = "plantae")
# chordata <- ee_observations(phylum = "chordata")
# Class is clss since the former is a reserved keyword in SQL.
# aves <- ee_observations(clss = "aves")
# You can also exclude fields using the exclude field
# aves <- ee_observations(clss = "aves", exclude = "remote_resource")
# aves <- ee_observations(clss = "aves", exclude = "source,remote_resource")
# or request additional fields using extra
# aves <- ee_observations(clss = "aves", extra = "kingdom,genus")
# aves <- ee_observations(clss = "aves", bbox = '-124,32,-114,42')
# aves <- ee_observations(clss = "aves", county = "Alameda county")

## End(Not run)

```

ee_pages

ee_pages - Returns total number of pages for any ecoengine request

Description

ee_pages - Returns total number of pages for any ecoengine request

Usage

```
ee_pages(ee, page_size = 1000)
```

Arguments

ee	Object of class ecoengine
page_size	Default page size. Currently set to 1000 package wide.

Value

integer

ee_paginator	<i>ecoengine paginator</i>
--------------	----------------------------

Description

Takes a page range and total number of observations to return the right sequence of pages that need to be crawled.

Usage

```
ee_paginator(page, total_obs, page_size = 1000)
```

Arguments

page	requested page number or page range. Can also be "all"
total_obs	Total number of records available for any search query
page_size	Default is 25. Set higher if needed.

Examples

```
## Not run:
ee_paginator(1, 100)
ee_paginator("all", 100)
ee_paginator(1:2, 1000)
ee_paginator(1:4, 3800)
# This will return an error since there are only 4 pages per 100 observations
ee_paginator(1:5, 10000)

## End(Not run)
```

ee_photos	<i>ee_photos</i>
-----------	------------------

Description

Search the photos methods in the ecoengine API.

Usage

```
ee_photos(page = NULL, state_province = NULL, county = NULL,
  genus = NULL, scientific_name = NULL, authors = NULL,
  remote_id = NULL, collection_code = NULL, source = NULL,
  min_date = NULL, max_date = NULL, related_type = NULL, related = NULL,
  page_size = 1000, quiet = FALSE, georeferenced = FALSE,
  progress = FALSE, other_catalog_numbers = NULL, foptions = list())
```

Arguments

page	Page number
state_province	Need to describe these parameters
county	California counties. Package include a full list of counties. To load dataset <code>data(california_counties)</code>
genus	genus name
scientific_name	scientific name
authors	author name
remote_id	remote id
collection_code	Type of collection. Can be CalAcademy, Private, VTM, CDFA, CalFlora Others TBA
source	data source. See ee_sources
min_date	Lower date bound. Most be in format Y-m-D h:m:s. This package uses lubridate's <code>ymd_hms</code> function to format dates.
max_date	upper date bound. Most be in format Y-m-D h:m:s. This package uses lubridate's <code>ymd_hms</code> function to format dates.
related_type	Need to describe these parameters
related	Need to describe these parameters
page_size	The number of observations per page returned by the query. The package default is 250 observations. The API default, however, is 10. Setting this number higher will require fewer calls (i.e. pages) but a large size may also fail or timeout.
quiet	Default is FALSE. Set to TRUE to suppress messages.
georeferenced	Default is FALSE. Set to TRUE to filter by photos that have geo data.
progress	Progress bar is set to TRUE by default. Use FALSE especially when calling functions inside documents.
other_catalog_numbers	Need to describe these parameters
foptions	A list of additional arguments for <code>httr</code> . There is no reason to use this argument except for debugging purposes.

See Also

related: [ee_photos](#) [california_counties](#)

Examples

```
# Request all photos. This request will paginate.
# merced <- ee_photos(county = "Merced County")
  ee_photos(page_size = 10)
# Search by collection code. See notes above on options
# ee_photos(collection_code = "CalAcademy")
# ee_photos(collection_code = "VTM")
```

```

# ee_photos(collection_code = "CalFlora")
# ee_photos(collection_code = "CDFA")
# Search by county.
# sc_county <- ee_photos(county = "Santa Clara County")
# merced <- ee_photos(county = "Merced County")
# merced <- ee_photos(county = "Merced County", page = "all")
# The package also contains a full list of counties
data(california_counties)
# alameda <- ee_photos(county = california_counties[1, 1])
# alameda$data
# You can also get all the data for Alameda county with one request
# alameda <- ee_photos(county = "Alameda county", page = "all")
# Spidering through the rest of the counties can easily be automated.
# Or by author
# charles_results <- ee_photos(author = "Charles Webber", page = 1:2)
# You can also request all pages in a single call by using ee_photos()
# In this example below, there are 6 pages of results (52 result items).
# Function will return all at once.
# racoons <- ee_photos(scientific_name = "Procyon lotor", page = "all")

```

ee_search

Ecoengine search

Description

Search across the entire ecoengine database.

Usage

```
ee_search(query = NULL, foptions = list())
```

Arguments

query	search term
foptions	A list of additional arguments for httr. There is no reason to use this argument except for debugging purposes.

Examples

```

## Not run:
# lynx_results <- ee_search(query = "genus:Lynx")

## End(Not run)

```

`ee_search_obs`*Search observations*

Description

A powerful way to search through the observations.

Usage

```
ee_search_obs(query = NULL, page = NULL, page_size = 1000,  
              quiet = FALSE, progress = TRUE, foptions = list())
```

Arguments

<code>query</code>	= The search term
<code>page</code>	Page number
<code>page_size</code>	The number of observations per page returned by the query. The package default is 250 observations. The API default, however, is 10. Setting this number higher will require fewer calls (i.e. pages) but a large size may also fail or timeout.
<code>quiet</code>	Default is FALSE. Set to TRUE to suppress messages.
<code>progress</code>	Progress bar is set to TRUE by default. Use FALSE especially when calling functions inside documents.
<code>foptions</code>	A list of additional arguments for <code>httr</code> . There is no reason to use this argument except for debugging purposes.

Value

`data.frame`

See Also

[ee_search](#))

Examples

```
# general_lynx_query <- ee_search_obs(query = "Lynx")  
## Not run:  
# lynx_data <- ee_search_obs(query = "genus:Lynx")  
# all_lynx_data <- ee_search_obs(query = "Lynx", page = "all")  
  
## End(Not run)
```

ee_sources	<i>Ecoengine data sources</i>
------------	-------------------------------

Description

Returns a full list of data sources supported by the ecoengine

Usage

```
ee_sources(foptions = list())
```

Arguments

`foptions` A list of additional arguments for `httr`. There is no reason to use this argument except for debugging purposes.

Value

`data.frame`

Examples

```
# source_list <- ee_sources()
```

full_sensor_list	<i>full_sensor_list</i>
------------------	-------------------------

Description

A `data.frame` containing all the sensors currently available through the engine. This list can be updated anytime by calling the `ee_sensors()` function.

Usage

```
data(full_sensor_list)
```

Format

A data table with 40 rows and 13 variables. Contains `units`, `variable`, `data_url`, `source`, `record`, `site_code`, `method_name`, `begin_date`, `end_date`, `station_name`, `geojson.type`, `geojson.coordinates1`, `geojson.coordinates2`

print.ecoengine	<i>Print a summary for an ecoengine object</i>
-----------------	--

Description

Print a summary for an ecoengine object

Usage

```
## S3 method for class 'ecoengine'  
print(x, ...)
```

Arguments

x	An object of class ecoengine
...	additional arguments

view_photos	<i>Browse photo queries in your default browser.</i>
-------------	--

Description

Browse photo queries in your default browser.

Usage

```
view_photos(input = NULL, output = NULL, browse = TRUE)
```

Arguments

input	Input, usually output from a call to ee_photos
output	Path and file name for output file. If NULL, a temp file is used.
browse	Browse file in your default browse immediately after file creation. If FALSE, the file is written, but not opened.

Examples

```
# view_photos(ee_photos())  
# Pictures of racoons  
# view_photos(ee_photos(scientific_name = "Procyon lotor", quiet = TRUE))  
# or the California Condor  
# view_photos(ee_photos(scientific_name = "Gymnogyps californianus", quiet = TRUE))
```

Index

*Topic **datasets**

- california_counties, [2](#)
- full_sensor_list, [13](#)

*Topic **search**

- ee_search, [11](#)
- ee_search_obs, [12](#)

- california_counties, [2](#), [10](#)
- checklist_details, [2](#)

- ecoengine, [3](#)
- ecoengine-package (ecoengine), [3](#)
- ee_about, [3](#)
- ee_cbind, [4](#)
- ee_checklists, [3](#), [5](#)
- ee_footprints, [5](#)
- ee_observations, [6](#)
- ee_pages, [8](#)
- ee_paginator, [9](#)
- ee_photos, [9](#), [10](#), [14](#)
- ee_search, [11](#), [12](#)
- ee_search_obs, [12](#)
- ee_sources, [7](#), [10](#), [13](#)

- full_sensor_list, [13](#)

- print.ecoengine, [14](#)

- view_photos, [14](#)