

Package ‘SigOptR’

July 21, 2025

Title R API Wrapper for SigOpt

Version 0.0.1

Author Alexandra Johnson [aut, cre]

Maintainer Alexandra Johnson <alexandra@sigopt.com>

Description Interfaces with the 'SigOpt' API. More info at <<https://sigopt.com>>.

Imports http, jsonlite

Depends R (>= 3.2.0)

License MIT + file LICENSE

LazyData true

RoxygenNote 6.0.1.9000

NeedsCompilation no

Repository CRAN

Date/Publication 2017-03-10 08:50:09

Contents

create_experiment	2
create_observation	2
create_suggestion	3
fetch_experiment	4
franke	5
sigopt_api_token	5
sigopt_api_url	6
sigopt_api_user_agent	6
sigopt_auth	7
sigopt_check	7
sigopt_GET	8
sigopt_parse	8
sigopt_POST	9

Index	10
--------------	-----------

create_experiment	Create an experiment
-------------------	----------------------

Description

Create an experiment

Usage

```
create_experiment(body)
```

Arguments

body	POST body of create request
------	-----------------------------

Value

experiment created by SigOpt

Examples

```
env <- Sys.getenv("NOT_CRAN")
if (!identical(env, "true")) {
  0
} else {
  create_experiment(list(
    name="R test experiment",
    parameters=list(
      list(name="x1", type="double", bounds=list(min=0, max=100)),
      list(name="x2", type="double", bounds=list(min=0, max=100))
    )
  ))
}
```

create_observation	Create an observation for an experiment
--------------------	---

Description

Create an observation for an experiment

Usage

```
create_observation(experiment_id, body)
```

Arguments

experiment_id	the id of an experiment to create an observation for
body	POST body of create request

Value

observation created by SigOpt

Examples

```
env <- Sys.getenv("NOT_CRAN")
if (!identical(env, "true")) {
  0
} else {
  experiment <- create_experiment(list(
    name="R test experiment",
    parameters=list(
      list(name="x1", type="double", bounds=list(min=0, max=100)),
      list(name="x2", type="double", bounds=list(min=0, max=100))
    )
  ))
  suggestion <- create_suggestion(experiment$id)
  create_observation(experiment$id, list(suggestion=suggestion$id, value=99.08))
  create_observation(experiment$id, list(suggestion=suggestion$id, value=99.58, value_stddev=0.1))}
```

create_suggestion	<i>Create a suggestion for an experiment</i>
-------------------	--

Description

Create a suggestion for an experiment

Usage

```
create_suggestion(experiment_id, body = NULL)
```

Arguments

`experiment_id` the id of an experiment to create an suggestion for
`body` POST body of create request

Value

suggestion created by SigOpt

Examples

```
env <- Sys.getenv("NOT_CRAN")
if (!identical(env, "true")) {
  0
} else {
  experiment <- create_experiment(list(
    name="R test experiment",
    parameters=list(
```

```

    list(name="x1", type="double", bounds=list(min=0, max=100)),
    list(name="x2", type="double", bounds=list(min=0, max=100))
  )
})
create_suggestion(experiment$id)}

```

fetch_experiment

Fetch an experiment

Description

Fetch an experiment

Usage

```
fetch_experiment(experiment_id, body = NULL)
```

Arguments

experiment_id the id of an experiment to fetch
body Url params of GET request

Value

SigOpt experiment with id experiment_id

Examples

```

env <- Sys.getenv("NOT_CRAN")
if (!identical(env, "true")) {
  0
} else {
  experiment <- create_experiment(list(
    name="R test experiment",
    parameters=list(
      list(name="x1", type="double", bounds=list(min=0, max=100)),
      list(name="x2", type="double", bounds=list(min=0, max=100))
    )
  ))
  fetch_experiment(experiment$id)}

```

franke	<i>Franke function - http://www.sfu.ca/~ssurjano/franke2d.html</i>
--------	--

Description

Franke function - <http://www.sfu.ca/~ssurjano/franke2d.html>

Usage

```
franke(x, y)
```

Arguments

x	First dimension
y	Second dimension

Value

The franke function evaluated at x,y

Examples

```
franke(0,1)
```

sigopt_api_token	<i>Get the SigOpt API token from the SIGOPT_API_TOKEN environment variable or user input</i>
------------------	--

Description

Get the SigOpt API token from the SIGOPT_API_TOKEN environment variable or user input

Usage

```
sigopt_api_token(force = FALSE)
```

Arguments

force	force entry of SigOpt API token, even if present
-------	--

Value

SigOpt API token

sigopt_api_url	<i>Get the SigOpt API url from the SIGOPT_API_URL environment variable or use default Most users will be ok with the default value</i>
----------------	--

Description

Get the SigOpt API url from the SIGOPT_API_URL environment variable or use default Most users will be ok with the default value

Usage

```
sigopt_api_url()
```

Value

Base url for SigOpt API requests

See Also

[sigopt_GET](#) and [sigopt_POST](#), which perform the HTTP requests

sigopt_api_user_agent	<i>User agent for current version of SigOpt R API Client</i>
-----------------------	--

Description

User agent for current version of SigOpt R API Client

Usage

```
sigopt_api_user_agent()
```

Value

User agent

See Also

[sigopt_GET](#) and [sigopt_POST](#), which perform the HTTP requests

sigopt_auth	Create authentication for SigOpt API, using HTTP Basic Auth
-------------	---

Description

Create authentication for SigOpt API, using HTTP Basic Auth

Usage

```
sigopt_auth(api_token = sigopt_api_token())
```

Arguments

api_token	SigOpt API token
-----------	------------------

Value

http basic authentication with api_token as username and no password

sigopt_check	Check content returned by the SigOpt API
--------------	--

Description

Check content returned by the SigOpt API

Usage

```
sigopt_check(req)
```

Arguments

req	result of request to the SigOpt API
-----	-------------------------------------

Value

invisible(), stops if there is an error status code

See Also

[sigopt_GET](#) and [sigopt_POST](#), which call this function interally

`sigopt_GET`*GET request to SigOpt API path, with json encoded body*

Description

GET request to SigOpt API path, with json encoded body

Usage

```
sigopt_GET(path, query = NULL, api_token = sigopt_api_token())
```

Arguments

<code>path</code>	path of SigOpt API url to POST to
<code>query</code>	list of query parameters to be url-encoded
<code>api_token</code>	SigOpt api token

Value

result of request to SigOpt API, needs to be JSON decoded

See Also

[sigopt_parse](#), which parses the result of this function

`sigopt_parse`*Parse content returned by the SigOpt API*

Description

Parse content returned by the SigOpt API

Usage

```
sigopt_parse(req)
```

Arguments

<code>req</code>	result of request to the SigOpt API
------------------	-------------------------------------

Value

json decoding of request object

See Also

[sigopt_GET](#) and [sigopt_POST](#), which call this function interally

`sigopt_POST`*POST request to SigOpt API path, with json encoded body*

Description

POST request to SigOpt API path, with json encoded body

Usage

```
sigopt_POST(path, body, api_token = sigopt_api_token())
```

Arguments

<code>path</code>	path of SigOpt API url to POST to
<code>body</code>	POST body, will be json-encoded
<code>api_token</code>	SigOpt api token

Value

result of request to SigOpt API, needs to be JSON decoded

See Also

[sigopt_parse](#), which parses the result of this function

Index

`create_experiment`, [2](#)
`create_observation`, [2](#)
`create_suggestion`, [3](#)

`fetch_experiment`, [4](#)
`franke`, [5](#)

`sigopt_api_token`, [5](#)
`sigopt_api_url`, [6](#)
`sigopt_api_user_agent`, [6](#)
`sigopt_auth`, [7](#)
`sigopt_check`, [7](#)
`sigopt_GET`, [6–8](#), [8](#)
`sigopt_parse`, [8](#), [8](#), [9](#)
`sigopt_POST`, [6–8](#), [9](#)