

Package ‘Rnumerai’

August 25, 2022

Title Interface to the Numerai Machine Learning Tournament API

Version 3.0.0

Description Routines to interact with the Numerai Machine Learning Tournament API <<https://numer.ai>>. The functionality includes the ability to automatically download the current tournament data, submit predictions, and to get information for your user.

Depends R (>= 4.0)

License GPL-3

Encoding UTF-8

URL <https://github.com/Omni-Analytics-Group/Rnumerai>

BugReports <https://github.com/Omni-Analytics-Group/Rnumerai/issues>

RoxygenNote 7.2.1

Imports httr, lubridate, jsonlite, ghql

NeedsCompilation no

Author Omni Analytics Group [aut],
Eric Hare [cre],
Yogesh Bansal [ctb]

Maintainer Eric Hare <eric@omnianalytics.org>

Repository CRAN

Date/Publication 2022-08-25 09:12:38 UTC

R topics documented:

diagnostics	2
download_dataset	3
download_validation_data	3
get_account	4
get_api_key	5
get_competitions	5
get_current_round	6

get_leaderboard	6
get_models	7
get_public_id	7
list_datasets	8
round_model_performances	8
set_api_key	9
set_bio	9
set_link	10
set_public_id	11
set_stake_type	11
set_submission_webhook	12
stake_change	13
submission_status	13
ticker_universe	14
upload_diagnostics	14
upload_predictions	15
wallet_transactions	16

Index	17
--------------	-----------

diagnostics	<i>Fetch results of diagnostics run</i>
-------------	---

Description

Fetch results of diagnostics run

Usage

```
diagnostics(model_id, tournament = 8, diagnostics_id)
```

Arguments

model_id	Target model UUID
tournament	Tournament ID, 8 for Main, 11 for Signal
diagnostics_id	id returned by "upload_diagnostics"

Value

Diagnostic results

Examples

```
## Not run:
diagnostics(model_id = get_models()[["bayo"]], tournament=8, diagnostics_id = "")

## End(Not run)
```

download_dataset	<i>Download specified file for the given round.</i>
------------------	---

Description

Download specified file for the given round.

Usage

```
download_dataset(  
  filename = "v2/numerai_live_data.csv",  
  dest_path = NA,  
  round_num = NA  
)
```

Arguments

filename	File to be downloaded, defaults to live data
dest_path	File path where the file should be stored, Defaults to current directory with file-name parameter
round_num	Tournament round you are interested in, defaults to the current round.

Value

list of filenames

Examples

```
## Not run:  
download_dataset(round_num=328)  
  
## End(Not run)
```

download_validation_data	<i>Download CSV file with historical targets and ticker universe</i>
--------------------------	--

Description

Download CSV file with historical targets and ticker universe

Usage

```
download_validation_data(file_path = "signals_historical_targets.csv")
```

Arguments

file_path CSV file with predictions that will get uploaded

Value

List of currently accepted tickers

Examples

```
## Not run:  
download_validation_data(file_path = "signals_historical_targets.csv")  
  
## End(Not run)
```

get_account	<i>Get all information about your account</i>
-------------	---

Description

Get all information about your account

Usage

```
get_account()
```

Value

User information

Examples

```
## Not run:  
get_account()  
  
## End(Not run)
```

get_api_key *Gets the Numerai API key*

Description

Gets the Numerai API key

Usage

```
get_api_key()
```

Value

Your Numerai API key, if set

Examples

```
## Not run:  
get_api_key()  
  
## End(Not run)
```

get_competitions *Retrieves information about all rounds*

Description

Retrieves information about all rounds

Usage

```
get_competitions(tournament = 8)
```

Arguments

tournament Tournament ID, 8 for Main, 11 for Signal

Value

Rounds Information

Examples

```
## Not run:  
get_competitions()  
  
## End(Not run)
```

get_current_round *Get number of the current active round.*

Description

Get number of the current active round.

Usage

```
get_current_round()
```

Value

Number of the current active round

Examples

```
## Not run:  
get_current_round()  
  
## End(Not run)
```

get_leaderboard *Get the current leaderboard*

Description

Get the current leaderboard

Usage

```
get_leaderboard(tournament = 8)
```

Arguments

tournament Tournament ID, 8 for Main, 11 for Signal

Value

Leaderboard entries

Examples

```
## Not run:  
get_leaderboard()  
  
## End(Not run)
```

get_models	<i>Get mapping of account model names to model ids for convenience</i>
------------	--

Description

Get mapping of account model names to model ids for convenience

Usage

```
get_models(tournament = 8)
```

Arguments

tournament Tournament ID, 8 for Main, 11 for Signal

Value

modelname -> model_id list

Examples

```
## Not run:  
get_models()  
  
## End(Not run)
```

get_public_id	<i>Gets the Numerai Public ID</i>
---------------	-----------------------------------

Description

Gets the Numerai Public ID

Usage

```
get_public_id()
```

Value

Your Numerai Public ID, if set

Examples

```
## Not run:  
get_public_id()  
  
## End(Not run)
```

list_datasets *List of available data files*

Description

List of available data files

Usage

```
list_datasets(round_num = NA)
```

Arguments

round_num Tournament round you are interested in, defaults to the current round.

Value

list of filenames

Examples

```
## Not run:  
list_datasets(round_num=328)  
  
## End(Not run)
```

round_model_performances
Fetch round model performance of any user

Description

Fetch round model performance of any user

Usage

```
round_model_performances(username, tournament = 8)
```

Arguments

username User Name
tournament Tournament ID, 8 for Main, 11 for Signal

Value

list of round model performance entries

Examples

```
## Not run:  
round_model_performances(username = "bayo")  
  
## End(Not run)
```

set_api_key	<i>Sets the Numerai API key</i>
-------------	---------------------------------

Description

Sets the Numerai API key

Usage

```
set_api_key(key)
```

Arguments

key	The Numerai API key
-----	---------------------

Value

A boolean TRUE if the key was successfully set

Examples

```
## Not run:  
set_api_key("abcdefghijklmnop")  
  
## End(Not run)
```

set_bio	<i>Set bio field for a model id</i>
---------	-------------------------------------

Description

Set bio field for a model id

Usage

```
set_bio(model_id, bio)
```

Arguments

model_id	Target model UUID
bio	Bio to change

Value

If the bio was changed successfully

Examples

```
## Not run:  
set_bio(model_id = get_models()[["bayo"]], bio = "This Model Rocks")  
  
## End(Not run)
```

set_link	<i>Set link field for a model id</i>
----------	--------------------------------------

Description

Set link field for a model id

Usage

```
set_link(model_id, link, link_text)
```

Arguments

model_id	Target model UUID
link	URL
link_text	URL Text

Value

If the link was changed successfully

Examples

```
## Not run:  
set_link(model_id = get_models()[["bayo"]], link = "https://www.google.com", link_text = "Google")  
  
## End(Not run)
```

set_public_id	<i>Sets the Numerai Public ID</i>
---------------	-----------------------------------

Description

Sets the Numerai Public ID

Usage

```
set_public_id(id)
```

Arguments

id	The Numerai Public ID
----	-----------------------

Value

A boolean TRUE if the ID was successfully set

Examples

```
## Not run:  
set_public_id("abcdefghijklmnop")  
  
## End(Not run)
```

set_stake_type	<i>Change stake type by model</i>
----------------	-----------------------------------

Description

Change stake type by model

Usage

```
set_stake_type(  
  model_id,  
  corr_multiplier = 0,  
  tc_multiplier = 0,  
  take_profit = FALSE,  
  tournament = 8  
)
```

Arguments

model_id	Target model UUID
corr_multiplier	Multiplier of correlation for returns (Integer)
tc_multiplier	Multiplier of TC for returns (Numeric)
take_profit	TRUE/FALSE Determines whether payouts are returned to user wallet or automatically staked to next round.
tournament	Tournament ID, 8 for Main, 11 for Signal

Value

Confirmation that payout selection has been updated

Examples

```
## Not run:
set_stake_type(model_id = get_models()[["bayo"]], corr_multiplier=1, tc_multiplier=3, tournament=8)

## End(Not run)
```

set_submission_webhook

Set a model's submission webhook used in Numerai Compute

Description

Set a model's submission webhook used in Numerai Compute

Usage

```
set_submission_webhook(model_id, webhook)
```

Arguments

model_id	Target model UUID
webhook	The compute webhook to trigger this model

Value

Confirmation that your webhook has been set

Examples

```
## Not run:
set_submission_webhook(model_id = get_models()[["bayo"]], webhook = "..")

## End(Not run)
```

stake_change	<i>Change stake by 'value' NMR</i>
--------------	------------------------------------

Description

Change stake by 'value' NMR

Usage

```
stake_change(nmr, action = "decrease", model_id, tournament = 8)
```

Arguments

nmr	amount of NMR you want to increase/decrease
action	'increase' or 'decrease'
model_id	Target model UUID
tournament	Tournament ID, 8 for Main, 11 for Signal

Value

Stake change information

Examples

```
## Not run:
stake_change(nmr=10,action="decrease",model_id = get_models()[["bayo"]],tournament=8)

## End(Not run)
```

submission_status	<i>Submission status of the last submission associated with the account</i>
-------------------	---

Description

Submission status of the last submission associated with the account

Usage

```
submission_status(model_id, tournament = 8)
```

Arguments

model_id	Target model UUID
tournament	Tournament ID, 8 for Main, 11 for Signal

Value

Submission status and stats

Examples

```
## Not run:  
model_id = get_models()[["bayo"]]  
submission_status(model_id = model_id, tournament=8)  
  
## End(Not run)
```

ticker_universe	<i>Fetch universe of accepted tickers</i>
-----------------	---

Description

Fetch universe of accepted tickers

Usage

```
ticker_universe()
```

Value

List of currently accepted tickers

Examples

```
## Not run:  
ticker_universe()  
  
## End(Not run)
```

upload_diagnostics	<i>Upload predictions to diagnostics from file.</i>
--------------------	---

Description

Upload predictions to diagnostics from file.

Usage

```
upload_diagnostics(file_path = NA, model_id, df = NA, tournament = 8)
```

Arguments

file_path	CSV file with predictions that will get uploaded
model_id	Target model UUID (required for accounts with multiple models)
df	DataFrame to upload, if given both df and file_path, df will be uploaded.
tournament	Tournament ID, 8 for Main, 11 for Signal

Value

diagnostics_id

Examples

```
## Not run:
upload_diagnostics(file_path = "prediction.csv", model_id = get_models()[["bayo"]])

## End(Not run)
```

upload_predictions	<i>Upload predictions from file.</i>
--------------------	--------------------------------------

Description

Upload predictions from file.

Usage

```
upload_predictions(file_path = NA, model_id, df = NA, tournament = 8)
```

Arguments

file_path	CSV file with predictions that will get uploaded
model_id	Target model UUID (required for accounts with multiple models)
df	DataFrame to upload, if given both df and file_path, df will be uploaded.
tournament	Tournament ID, 8 for Main, 11 for Signal

Value

submission_id

Examples

```
## Not run:
upload_predictions(file_path = "prediction.csv", model_id = get_models()[["bayo"]])

## End(Not run)
```

wallet_transactions *Get all transactions in your wallet.*

Description

Get all transactions in your wallet.

Usage

```
wallet_transactions()
```

Value

Wallet Tx's Data Frame

Examples

```
## Not run:  
wallet_transactions()  
  
## End(Not run)
```


Index

diagnostics, [2](#)
download_dataset, [3](#)
download_validation_data, [3](#)

get_account, [4](#)
get_api_key, [5](#)
get_competitions, [5](#)
get_current_round, [6](#)
get_leaderboard, [6](#)
get_models, [7](#)
get_public_id, [7](#)

list_datasets, [8](#)

round_model_performances, [8](#)

set_api_key, [9](#)
set_bio, [9](#)
set_link, [10](#)
set_public_id, [11](#)
set_stake_type, [11](#)
set_submission_webhook, [12](#)
stake_change, [13](#)
submission_status, [13](#)

ticker_universe, [14](#)

upload_diagnostics, [14](#)
upload_predictions, [15](#)

wallet_transactions, [16](#)