

Package ‘fitzRoy’

May 23, 2020

Title Easily Scrape and Process AFL Data

Version 0.3.2

Description An easy package for scraping and processing Australia Rules Football (AFL) data. 'fitzRoy' provides a range of functions for accessing publicly available data from 'AFL Tables' <<https://aftables.com>>, 'Footy Wire' <<https://www.footywire.com>> and 'The Squiggle' <<https://squiggle.com.au>>. Further functions allow for easy processing, cleaning and transformation of this data into formats that can be used for analysis.

License GPL-3

URL <https://jimmyday12.github.io/fitzRoy/>,
<https://github.com/jimmyday12/fitzRoy>

BugReports <https://github.com/jimmyday12/fitzRoy/issues>

Depends R (>= 3.3)

Imports dplyr, httr, jsonlite, lubridate, magrittr, purrr, readr,
rlang (>= 0.1.2), rvest, stringr (>= 1.3.0), tidyr (>= 1.0.0),
tidyselect, xml2, tibble, glue

Suggests covr, ggplot2, knitr, rmarkdown, testthat, roxygen2, elo,
spelling, curl

VignetteBuilder knitr

ByteCompile true

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Language en-US

NeedsCompilation no

Author James Day [cre, aut],
Robert Nguyen [aut],
Matthew Erbs [ctb],
Oscar Lane [aut],
Jason Zivkovic [ctb]

Maintainer James Day <jamesthomasday@gmail.com>

Repository CRAN

Date/Publication 2020-05-23 05:10:05 UTC

R topics documented:

calculate_round	2
convert_results	3
footywire_html	3
get_aftables_stats	4
get_aftables_urls	5
get_aflw_cookie	5
get_aflw_detailed_data	6
get_aflw_detailed_match_data	6
get_aflw_match_data	7
get_aflw_rounds	8
get_aflw_round_data	9
get_fixture	9
get_footywire_betting_odds	10
get_footywire_stats	11
get_match_data	11
get_match_results	12
get_score_progression_raw	12
get_squiggle_data	13
replace_teams	14
replace_venues	14
return_ladder	15
scrape_aftables_match	16
update_footywire_stats	16
Index	18

calculate_round	<i>Helper function for get_fixture, betting_data</i>
-----------------	--

Description

Work out round number of each game from day and week. Games from Wednesday through Tuesday go in same Round.

Usage

```
calculate_round(data_frame)
```

Arguments

`data_frame` A data frame with match-level data and a Date column

convert_results	<i>Convert AFL Men's results into long format</i>
-----------------	---

Description

convert_results returns a dataframe containing the results in long format.

Usage

```
convert_results(results)
```

Arguments

results A dataframe that has been returned from get_match_results

Details

The standard results returned by aflightables.com will be in wide format. This is useful for game based analysis but less so for team based ones. This function converts the data into long format for easier analysis.

Value

A data frame with match results where each row is a team-match combination

Examples

```
## Not run:  
results <- get_match_results()  
convert_results(results)  
  
## End(Not run)
```

footywire_html	<i>Helper function for get_footywire_stats</i>
----------------	--

Description

Helper function for get_footywire_stats

Usage

```
footywire_html(x, id)
```

Arguments

x	URL of the match
id	Match ID number

Value

A data frame with advanced player results

get_afltables_stats *Return afltables match stats*

Description

get_afltables_stats returns a data frame containing match stats for each game within the specified date range

Usage

```
get_afltables_stats(start_date = "1897-01-01", end_date = Sys.Date())
```

Arguments

start_date	character string for start date return to URLs from, in "dmy" or "ymd" format
end_date	optional, character string for end date to return URLs, in "dmy" or "ymd" format

Details

This function returns a data frame containing match stats for each game within the specified date range. The data from contains all stats on afltables match pages and returns 1 row per player.

The data for this function is hosted on github to avoid extensive scraping of historical data from afltables.com. This will be updated regularly.

Value

a data table containing player stats for each game between start date and end date

Examples

```
#
## Not run:
# Gets all data
get_afltables_stats()
# Specify a date range
get_afltables_stats("01/01/2018", end_date = "01/04/2018")

## End(Not run)
```

get_afltables_urls *Return match URLs for specified dates*

Description

get_afltables_urls returns a character vector containing match URLs for the specified date range

Usage

```
get_afltables_urls(start_date, end_date = Sys.Date())
```

Arguments

start_date character string for start date return to URLs from, in "dmy" or "ymd" format
end_date optional, character string for end date to return URLs, in "dmy" or "ymd" format

Details

This function returns match URLs for the specified date range. This will typically be used to pass to scrape_afltables_match to return player match results.

Value

a character vector of match URL's between start_date and end_date

Examples

```
## Not run:  
get_afltables_urls("01/01/2018", end_date = "01/04/2018")  
  
## End(Not run)
```

get_aflw_cookie *Get AFL Stats cookie (internal function)*

Description

Gets a cookie from <http://www.afl.com.au/womens/matches/stats> to authenticate further requests.

Usage

```
get_aflw_cookie()
```

Value

token code

Examples

```
## Not run:  
cookie <- get_aflw_cookie()  
  
## End(Not run)
```

```
get_aflw_detailed_data  
    Get detailed AFLW data
```

Description

Get detailed AFLW data

Usage

```
get_aflw_detailed_data(matchids)
```

Arguments

matchids vector of match IDs, like those returned by get_aflw_match_data()

Value

Dataframe with detailed match data. Each row is a match.

Examples

```
## Not run:  
get_aflw_detailed_data(c("CD_M20172640101", "CD_M20172640102"))  
  
## End(Not run)
```

```
get_aflw_detailed_match_data  
    Get detailed womens match data (internal function)
```

Description

Gets detailed match data for a given match. Requires the match, round, and competition IDs, which are given in the tables produced by get_aflw_round_data()

Usage

```
get_aflw_detailed_match_data(matchid, roundid, competitionid, cookie)
```

Arguments

<code>matchid</code>	matchid from <code>get_match_data()</code>
<code>roundid</code>	roundid from <code>get_match_data()</code>
<code>competitionid</code>	competitionid from <code>get_match_data()</code>
<code>cookie</code>	cookie from <code>get_womens_cookie()</code>

Value

Dataframe with detailed match data (wide)

Examples

```
## Not run:  
get_aflw_detailed_match_data(  
  "CD_M20172640101",  
  "CD_R201726401", "CD_S2017264", get_aflw_cookie()  
)  
  
## End(Not run)
```

`get_aflw_match_data` *Get AFLW match data*

Description

Retrieves AFLW match data for all available matches. Sources data from <http://www.afl.com.au/womens/matches/stats>

Usage

```
get_aflw_match_data(start_year = 2017)
```

Arguments

`start_year` optional, integer for start year to return match data onwards from

Value

a data frame of data for all available AFLW matches

Examples

```
## Not run:  
# All data  
get_aflw_match_data()  
  
# 2018 data onward  
get_aflw_match_data(start_year = 2018)  
  
## End(Not run)
```

get_aflw_rounds	<i>Get rounds (internal function)</i>
-----------------	---------------------------------------

Description

Returns data frame for available round data. Includes the rounds played, as well as identifiers to make further requests, importantly the roundId.

Usage

```
get_aflw_rounds(cookie)
```

Arguments

cookie a cookie produced by get_aflw_cookie()

Value

A dataframe with information about each round

Examples

```
## Not run:  
get_aflw_rounds(get_aflw_cookie())  
  
## End(Not run)
```

get_aflw_round_data *Get match data (internal function)*

Description

For a given round ID, get the data for each match played in that round. Use the column roundId in the dataframe created by the get_rounds() function to specify matches to fetch.

Usage

```
get_aflw_round_data(roundid, cookie)
```

Arguments

roundid	a round ID string
cookie	a cookie produced by get_womens_cookie()

Value

a dataframe containing match data

Examples

```
## Not run:  
get_aflw_round_data("CD_R201826401", get_aflw_cookie())  
  
## End(Not run)
```

get_fixture *Get upcoming fixture from <https://www.footywire.com>*

Description

get_fixture returns a dataframe containing upcoming AFL Men's season fixture.

Usage

```
get_fixture(season = lubridate::year(Sys.Date()), convert_date = FALSE)
```

Arguments

season	Season to return, in yyyy format
convert_date	logical, if TRUE, converts date column to date format instead of date time.

Details

The dataframe contains the home and away team as well as venue.

Value

Returns a data frame containing the date, teams and venue of each game

Examples

```
## Not run:  
get_fixture(2018)  
  
## End(Not run)
```

```
get_footywire_betting_odds
```

Get AFL match betting odds from <https://www.footywire.com>

Description

get_footywire_betting_odds returns a data frame containing betting odds and basic match info for Men's AFL matches.

Usage

```
get_footywire_betting_odds(  
  start_season = "2010",  
  end_season = lubridate::year(Sys.Date())  
)
```

Arguments

start_season	First season to return, in yyyy format. Earliest season with data available is 2010.
end_season	Last season to return, in yyyy format

Details

The data frame contains the home and away team as well as venue.

Value

Returns a data frame containing betting odds and basic match info

Examples

```
## Not run:  
get_footywire_betting_odds(2012, 2018)  
  
## End(Not run)
```

get_footywire_stats *Scrape footywire player statistics.*

Description

get_footywire_stats returns a dataframe containing player match stats from footywire from 2010 onwards.

Usage

```
get_footywire_stats(ids)
```

Arguments

ids A vector containing match id's to return. Can be a single value or vector of values.

Details

The dataframe contains both basic and advanced player statistics from each match specified in the match_id input. To find match ID, find the relevant matches on <https://www.footywire.com>

Value

Returns a data frame containing player match stats for each match ID

Examples

```
## Not run:  
get_footywire_stats(ids = 5000:5100)  
  
## End(Not run)
```

get_match_data *Helper function for get_footywire_stats*

Description

Helper function for get_footywire_stats

Usage

```
get_match_data(id)
```

Arguments

id A match id from aflightables

`get_match_results` *Get basic match results from aftables.com*

Description

`get_match_results` returns a dataframe containing all match results from 1897-current

Usage

```
get_match_results()
```

Details

The dataframe contains information about the Date, teams involved, scores and venue. It comes from aftables 'big lists' section. This is a limited dataset but is very fast to access. It generally is updated on the day after the last game

Value

Returns a data frame containing a line for each match

Examples

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

`get_score_progression_raw`
Get raw score progression data

Description

`get_score_progression_raw` returns a dataframe raw, unprocessed scoring progression data from aftables.

Usage

```
get_score_progression_raw()
```

Details

The data is unprocessed and unstructured but is a starting point for analysis. It only exists for 2010 to 2017.

Value

Returns a data frame containing raw score progression data

Examples

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

get_squiggle_data	<i>Access Squiggle data using the squiggle API service.</i>
-------------------	---

Description

Use `get_squiggle_data` to access the [Squiggle API](https://api.squiggle.com.au). See instructions at api.squiggle.com.au.

Usage

```
get_squiggle_data(  
  query = c("sources", "games", "tips", "ladder", "standings"),  
  ...  
)
```

Arguments

query	A text string. The main query to use with the API. Must be one of sources, games, tips, ladder or standings
...	(optional) An optional argument provided to the Squiggle API . See details for more info.

Details

The optional arguments to `squiggle` can be one of the following.

#'

- year: an integer specifying the year to return data from, e.g. year = 2018
- round: an integer specifying the round to return data from, e.g. round = 12
- game: an integer specifying the game ID to return data from, e.g. game = 10
- source: an integer specifying the ID of the source to return data from, e.g. source = 1

For full instructions, see api.squiggle.com.au

Value

A dataframe, with the resultant data that matches the query specified in `query`, as well as any optional filters.

Examples

```
## Not run:
# Return a list of the sources, with ID's
sources <- get_squiggle_data("sources")

# Get tips for Round 1, 2018
tips <- get_squiggle_data(query = "tips", round = 1, year = 2018)

# Get tips from Squiggle 2019
squiggle <- get_squiggle_data(query = "tips", source = 1, year = 2019)

## End(Not run)
```

replace_teams	<i>Internal function to ensure names match between different sources and also name changes. This gets applied to any web scraper</i>
---------------	--

Description

Internal function to ensure names match between different sources and also name changes. This gets applied to any web scraper

Usage

```
replace_teams(team)
```

Arguments

team	Team name
------	-----------

replace_venues	<i>Internal function to ensure venue names match between different sources and also name changes across time. This gets applied to any web scraper, transforming all of them to AFL Tables naming conventions.</i>
----------------	--

Description

Internal function to ensure venue names match between different sources and also name changes across time. This gets applied to any web scraper, transforming all of them to AFL Tables naming conventions.

Usage

```
replace_venues(venue)
```

Arguments

venue	Venue name
-------	------------

return_ladder	<i>Recreate the ladder for every or any given round and/or season</i>
---------------	---

Description

return_ladder returns a dataframe containing the ladder for either all seasons and rounds since 1987, or individual rounds/seasons

Usage

```
return_ladder(match_results_df = NA, season_round = NA, season = NA)
```

Arguments

match_results_df	A dataframe that has been returned from get_match_results. If empty get_match_results will execute first
season_round	An integer of the round or vector of integers for multiple rounds. If empty, all rounds returned
season	An integer of the season or vector of integers for multiple seasons. If empty, all seasons returned

Details

The dataframe contains information about the Round, Season, Points For/Against, Ladder Position. It can either take in a data frame created using get_match_results, or if match_results_df is unspecified, will extract all games using get_match_results. Will only allow selecting rounds of the premiership season, not finals.

Value

Returns a data frame containing a line for each team's ladder position at each round of a season

Examples

```
## Not run:
return_ladder()
return_ladder(match_results_df=get_match_results_df, season_round=23, season=1990:2019)
return_ladder(season_round = 10, season = 2019)

## End(Not run)
```

`scrape_afltables_match`*Return afltables player match stats*

Description

`scrape_afltables_match` returns a character vector containing match URLs for the specified date range

Usage

```
scrape_afltables_match(match_urls)
```

Arguments

`match_urls` A list of URL's for matches to scrape data from

Details

This function returns the full afltables.com match stats for each player and each game specified in `match_urls`. It is useful to use the helper function `get_afltables_urls` to return these or simply navigate to afltables.com and find the match of interest.

Value

data table of afltables.com match results, with a row per player per match.

Examples

```
## Not run:
scrape_afltables_match("https://afltables.com/afl/stats/games/2018/071120180602.html")
scrape_afltables_match(get_afltables_urls("01/06/2018", "01/07/2018"))

## End(Not run)
```

`update_footywire_stats`*Update the included footywire stats data to the specified date.*

Description

`update_footywire_stats` returns a dataframe containing player match stats from **footywire**

Usage

```
update_footywire_stats(check_existing = TRUE)
```


Arguments

`check_existing` A logical specifying if we should check against existing dataset. Defaults to TRUE. Making it false will download all data from all history which will take some time.

Details

The dataframe contains both basic and advanced player statistics from each match from 2010 to the specified end date.

This function utilised the included ID's dataset to map known ID's. It looks for any new data that isn't already loaded and proceeds to download it.

Value

Returns a data frame containing player match stats for each match ID

Examples

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

Index

calculate_round, [2](#)
convert_results, [3](#)

footywire_html, [3](#)

get_afltables_stats, [4](#)
get_afltables_urls, [5](#)
get_aflw_cookie, [5](#)
get_aflw_detailed_data, [6](#)
get_aflw_detailed_match_data, [6](#)
get_aflw_match_data, [7](#)
get_aflw_round_data, [9](#)
get_aflw_rounds, [8](#)
get_fixture, [9](#)
get_footywire_betting_odds, [10](#)
get_footywire_stats, [11](#)
get_match_data, [11](#)
get_match_results, [12](#)
get_score_progression_raw, [12](#)
get_squiggle_data, [13](#)

replace_teams, [14](#)
replace_venues, [14](#)
return_ladder, [15](#)

scrape_afltables_match, [16](#)

update_footywire_stats, [16](#)