Package ‘hockeyR’

October 12, 2022

Title Collect and Clean Hockey Stats
Version 1.3.0
Description Various functions to scrape hockey play-by-play data from the
<https://www.nhl.com/>. It also contains functions to scrape data from <https://www.hockey-reference.com/>,
including standings, player stats, and jersey number history.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
URL https://github.com/danmorse314/hockeyR
BugReports https://github.com/danmorse314/hockeyR/issues
Depends R (>= 3.5.0)
Imports dplyr, glue, httr, janitor, jsonlite, magrittr, lubridate,
purrr, rvest, stringr, tidyr, tidyselect, utils, zoo, stats,
xgboost, padr
Suggests ggimage, ggplot2, ggrepel, knitr, rmarkdown, scales, sportyR
VignetteBuilder knitr
NeedsCompilation no
Author Daniel Morse [aut, cre]
Maintainer Daniel Morse <danmorse8642@gmail.com>
Repository CRAN
Date/Publication 2022-10-12 09:42:32 UTC

R topics documented:

- calculate_individual
- calculate_on_ice
- calculate_toi
calculate_individual  Calculate individual player stats

Description

Calculate individual player stats

Usage

```r
calculate_individual(pbp, type = c("R", "P"), game_strength = "all")
```

Arguments

- **pbp**: A tibble of play-by-play data, typically returned from either `load_pbp` or `scrape_game`
- **type**: Season type to filter by; "R" for regular season and/or "P" for postseason
- **game_strength**: String or vector of strings defining strength state to filter by; ex c("3v5","4v5","3v4")
  returns stats for shorthanded strength
**Value**

A tibble containing individual shooting stats for all players in supplied pbp data. **These stats are individual player stats, not player on-ice stats.** For help with on-ice stats, please see [calculate_on_ice](#).

- **player_name**  String identifying player name
- **player_id**  Integer value of the NHL player ID
- **team**  String identifying player’s most recent team
- **gp**  Games Played
- **ixg**  Numeric expected goals
- **goals**  Numeric goals scored
- **assists**  Numeric total assists
- **points**  Numeric points scored
- **assists_primary**  Numeric primary assists
- **assists_secondary**  Numeric secondary assists
- **points_primary**  Numeric primary points
- **gax**  Numeric goals scored over expected
- **icf**  Numeric shot attempts (Corsi)
- **iff**  Numeric unblocked shot attempts (Fenwick)
- **isog**  Numeric shots on goal
- **sh_perc**  Numeric shooting percentage

If supplied play-by-play data includes shift change events (the default for [scrape_game](#); if using [load_pbp](#) user must set `shift_events` argument to `TRUE`) then the following rate stats will also be calculated:

- **toi**  String description of total time on ice in ‘minutes:seconds’
- **mean_toi**  String description of average time on ice over all supplied games, in ‘minutes:seconds’
- **toi_minutes**  Numeric total time on ice, in minutes
- **mean_toi_minutes**  Numeric average time on ice over all supplied games, in minutes
- **ixg_per60**  Numeric expected goals per 60 minutes
- **goals_per60**  Numeric goals scored per 60 minutes
- **assists_per60**  Numeric total assists per 60 minutes
- **points_per60**  Numeric points scored per 60 minutes
- **assists_primary_per60**  Numeric primary assists per 60 minutes
- **assists_secondary_per60**  Numeric secondary assists per 60 minutes
- **points_primary_per60**  Numeric primary points per 60 minutes
- **gax_per60**  Numeric goals scored over expected per 60 minutes
- **icf_per60**  Numeric shot attempts (Corsi) per 60 minutes
- **iff_per60**  Numeric unblocked shot attempts (Fenwick) per 60 minutes
- **isog_per60**  Numeric shots on goal per 60 minutes
Examples

```r
## Not run:
#load_pbp
pbp_2022 <- load_pbp(2022, shift_events = TRUE)
player_stats <- calculate_individual(pbp_2022, type = "R", game_strength = "5v5")
## End(Not run)
```

---

calculate_on_ice  
*Calculate player on-ice stats*

Description

Calculate player on-ice stats

Usage

```r
calculate_on_ice(pbp, type = c("R", "P"), game_strength = "all")
```

Arguments

- **pbp**: A tibble of play-by-play data, typically returned from either `load_pbp` or `scrape_game`
- **type**: Season type to filter by; "R" for regular season and/or "P" for postseason
- **game_strength**: String or vector of strings defining strength state to filter by; ex c("3v5","4v5","3v4") returns stats for shorthanded strength

Value

A tibble containing on-ice shot stats for all players in supplied pbp data. **These stats are for all players in which a player was on the ice, not individual stats.** For help with individual stats, please see `calculate_individual`. On-ice stats include:

- **player_name**: String identifying player name
- **player_id**: Integer value of the NHL player ID
- **team**: String identifying player’s most recent team
- **gp**: Games Played
- **cf**: Numeric shot attempts (Corsi) for
- **ca**: Numeric shot attempts (Corsi) against
- **cf_perc**: Numeric Corsi For % (CF%)
- **ff**: Numeric unblocked shot attempts (Fenwick) for
- **fa**: Numeric unblocked shot attempts (Fenwick) against
- **ff_perc**: Numeric Fenwick For % (FF%)
- **gf**: Numeric goals for
- **ga**: Numeric goals against
**calculate_toi**

A function to calculate individual skater time on ice for a provided play-by-play data set.

## Description

A function to calculate individual skater time on ice for a provided play-by-play data set.

## Usage

`calculate_toi(pbp)`

## Arguments

- **pbp**
  
  A tibble of play-by-play data, typically returned from either `load_pbp` or `scrape_game`
calculate_xg

Value
A tibble containing time on ice information for every skater in supplied pbp data

- **player_name** String identifying player name
- **player_id** Integer value of the NHL player ID
- **gp** Games Played
- **toi** String description of total time on ice in 'minutes:seconds'
- **mean_toi** String description of average time on ice over all supplied games, in 'minutes:seconds'
- **toi_minutes** Numeric total time on ice, in minutes
- **mean_toi_minutes** Numeric average time on ice over all supplied games, in minutes

Examples

```r
## Not run:
pbp_2022 <- load_pbp(2022)
skater_toi <- calculate_toi(pbp_2022)
## End(Not run)
```

calculate_xg

*Calculate hockeyR expected goals (xG)*

Description
Uses the hockeyR expected goals model to calculate xG for any pbp data frame generated by hockeyR

Usage
```
calculate_xg(pbp)
```

Arguments
- **pbp** A play-by-play data frame, previously returned by hockeyR::scrape_game

Value
The original supplied play-by-play data with a column for expected goals appended

Examples

```r
## Not run:
pbp <- load_pbp(2022) %>% dplyr::select(-xg)
pbp_preds <- calculate_xg(pbp)
## End(Not run)
```
**get_current_rosters**  
*Get current NHL rosters*

**Description**
Get current NHL rosters

**Usage**
```r
get_current_rosters()
```

**Value**
A tibble containing the current rosters for every team per the NHL:

- **player_id**  NHL player ID, as an integer
- **player**  Player name as a string
- **jersey_number**  Player jersey number, as an integer
- **position**  Abbreviated official player position, as a string
- **position_type**  Abbreviated position group: F, D, or G
- **team_id**  NHL integer team ID
- **full_team_name**  Full team name as a string
- **team_abbr**  Team abbreviation, as a string

**Examples**
```r
## Not run:
current_rosters <- get_current_rosters()

## End(Not run)
```

---

**get_draft_class**  
*Get draft classes*

**Description**
Get all selections for any single NHL draft class back to 1963

**Usage**
```r
get_draft_class(
  draft_year = as.numeric(format(Sys.Date() - 181, "%Y")),
  player_details = FALSE
)
```
get_game_ids

Arguments

draft_year Draft year to scrape
player_details If true, returns more detailed data on each prospect

Value

A tibble containing all selections for the supplied draft year.

Basic draft class variables:
• Draft year
• Round
• Overall Pick #
• Round Pick #
• Drafting team
• Player ID
• Player name
• Player link

Detailed draft class variables:
• Player birthdate
• Player birthplace & nationality
• Player height & weight
• Player shoots/catches
• Player position
• Player amateur team & league

Examples

## Not run:
draft_2022 <- get_draft_class(draft_year = 2022, player_details = TRUE)

## End(Not run)

get_game_ids Fetch game IDs for a single day or a full season

Description

Fetch game IDs for a single day or a full season

Usage

get_game_ids(season = NULL, day = as.Date(Sys.Date(), "%Y-%m-%d"))
get_game_info

Arguments

- **season**: An integer value denoting the end year of the season to scrape
- **day**: A day in the format of 'YYYY-MM-DD'

Value

A tibble containing game IDs and basic info for specified time frame

- **game_id**: Integer value of NHL game ID used in *scrape_game*
- **season_full**: String defining NHL season
- **date**: Date of game, as a string
- **game_time**: Scheduled start time (US/Eastern) of game, as a string
- **home_name**: Home team name, as a string
- **away_name**: Away team name, as a string
- **home_final_score**: Numeric final score for home team - will return 0 for games that haven’t started
- **away_final_score**: Numeric final score for away team - will return 0 for games that haven’t started
- **game_type**: String denoting type of game: "REG" or "POST"

Examples

```r
## Not run:
get_game_ids(season = 2020)
get_game_ids(day = "2015-03-12")
## End(Not run)
```

get_game_info

Gather basic game information

Description

Scrapes basic game info like date, venue, & information about the home and away teams for a given game

Usage

```r
game_id

Arguments

- **game_id**: Game ID to scrape (Can be found using get_game_ids function)

Value

A 1xN tibble containing N pieces of information about the specified game
get_game_rosters  
Scrape game day rosters for individual game

Description

Scrapes the game-day rosters for both teams in the given game ID

Usage

get_game_rosters(game_id)

Arguments

game_id  
Game ID to scrape (Can be found using get_game_ids function)

Value

A tibble containing player names, ids, and positions for both team rosters in a given game.

get_game_shifts  
Fetch game shift data

Description

A function to gather shift data from a given game. Shifts are turned into events to match the style of events in standard game pbp.

Portions of this code are modified versions of code from the NHL scraper by Evolving-Hockey, which in turn were modified from the NHL scraper by Manny Perry.

Usage

get_game_shifts(game_id)

Arguments

game_id  
Game ID to scrape

Value

A tibble containing each player change as an event

Examples

## Not run:
get_game_shifts(2020020561)

## End(Not run)
### get_goalie_stats_hr

_**Scrape goalie stats**_

**Description**

A function to scrape all goalie stats from a single season via hockey-reference.com

**Usage**

```r
get_goalie_stats_hr(season = as.numeric(format(Sys.Date() + 81, "%Y")))
```

**Arguments**

- `season` : Integer value denoting the end year of the season to scrape

**Value**

A tibble containing all goalie stats found on hockey-reference.com for the given season

**Examples**

```r
## Not run:
get_goalie_stats_hr(2022)
## End(Not run)
```

---

### get_jersey_players

_**Get Players by Jersey Number**_

**Description**

Get the name of every player to wear a specific jersey number in the NHL and the season in which they wore it.

**Usage**

```r
get_jersey_players(jersey)
```

**Arguments**

- `jersey` : An integer or a vector of integers between 0 & 99

**Value**

A tibble containing each player-season where a player wore the specified number
get_player_stats_hr

## Examples

```r
## Not run:
# get_jersey_players(69)
get_jersey_players(c(99, 66))
```

## Not run:
```
get_player_stats(“Wayne Gretzky”)
get_player_stats_hr(c(“Wayne Gretzky”, “Mario Lemieux”))
```

## End(Not run)

---

get_player_stats_hr  
*Get player counting stats from hockey-reference.com*

### Description

Get player counting stats from hockey-reference.com

### Usage

```r
get_player_stats_hr(player_name, season = "career", league = "NHL")
```

### Arguments

- `player_name`: A player name or vector of player names
- `season`: An integer value denoting the end year of the season(s) to scrape
- `league`: The league stats to scrape, either ‘NHL’ or ‘WHA’

### Value

A tibble containing goals, assists, and various other stats for the specified player(s) from hockey-reference.com

### Examples

```r
## Not run:
# get_player_stats("Wayne Gretzky")
get_player_stats_hr(c("Wayne Gretzky", "Mario Lemieux"))
```

## End(Not run)
**get_rosters**

Get team rosters from hockey-reference.com

---

**Description**

Get the latest roster for any team from hockey-reference.com. You may enter either the team abbreviation or the full team name. Seasons must be 4-digit integers denoting the end-year of the regular season desired (ie 2021-22 season should be 2022)

Please note that this uses the hockey-reference.com team abbreviations, the oddest of which is Vegas being 'VEG' instead of 'VGK'. If you are unsure of the team abbreviation, enter the full team name instead, or check the full team abbreviations data and filter to your desired season:

```r
filter(team_abbr_yearly, season_short == {season})
```

**Usage**

```r
get_rosters(
  team = "all",
  season = as.numeric(format(Sys.Date() + 184, "%Y")),
  include_stats = FALSE
)
```

**Arguments**

- **team** A character vector of team names or abbreviations
- **season** An integer value denoting the end year of the season to scrape
- **include_stats** Set to TRUE to return player counting stats for the season

**Value**

A tibble containing the latest roster for the specified team(s) in the specified season

**Examples**

```r
## Not run:
get_rosters("SEA", 2022)
## End(Not run)
```
get_skater_stats_hr  Scrape skater stats

Description
A function to scrape all skater stats from a single season via hockey-reference.com

Usage
get_skater_stats_hr(season = as.numeric(format(Sys.Date() + 81, "%Y")))

Arguments
season  Integer value denoting the end year of the season to scrape

Value
A tibble containing all skater stats found on hockey-reference.com for the given season

Examples
## Not run:
get_skater_stats_hr(2022)
## End(Not run)

get_standings  Load team standings

Description
Get full regular season standings for given year(s), including win-loss record and goals for and against

Usage
get_standings(seasons = as.numeric(format(Sys.Date() + 78, "%Y")))

Arguments
seasons  End year of seasons to pull

Value
A tibble containing team records and stats for given year
get_team_records

Examples

```r
## Not run:
get_standings(2022)

## End(Not run)
```

---

get_team_records  

Get team records from hockey-reference.com

Description

Get team records from hockey-reference.com

Usage

```r
get_team_records(
  season = as.numeric(format(Sys.Date() + 184, "%Y")),
  league = "NHL",
  include_records = TRUE
)
```

Arguments

- `season`: An integer value denoting the end year of the season(s) to scrape
- `league`: The league stats to scrape, either ‘NHL’ or ‘WHA’ or get both with c(‘NHL’,’WHA’)
- `include_records`: Option to exclude records from the function, used to gather full team names & abbreviations for every season since 1918

Value

A tibble containing full team names & win-loss records for teams in all desired seasons

Examples

```r
## Not run:
get_team_records(2021)

## End(Not run)
```
get_team_rosters  
*Get current team roster*

**Description**

Get current team roster

**Usage**

```r
get_team_rosters(team)
```

**Arguments**

- `team`  
  A team name or abbreviation, as a string - or the NHL integer team ID

**Value**

A tibble containing the current official team roster per NHL.com

**Examples**

```r
## Not run:
get_team_rosters("SEA")

## End(Not run)
```

---

load_pbp  
*Load season play-by-play*

**Description**

Load season play-by-play

**Usage**

```r
load_pbp(
  season = as.numeric(substr(Sys.Date() + 184, 1, 4)),
  shift_events = FALSE
)
```
scrape_day

Arguments

season  An integer value or vector of values denoting the end year of the season(s) to scrape. `load_pbp` also accepts character strings with more explicit definitions of the season to scrape: '2020-2021', '2020-21', '2020_21' are also acceptable. The default value is the current season, switching to the next year on July 1st when the new league year begins.

shift_events  Logical value; when set to FALSE this function returns a smaller dataset that excludes specifically shift change events

Value

A tibble containing all play-by-play data for a given season(s) in the same format as the output of `scrape_game`

Examples

```r
## Not run:
pbp <- load_pbp(2021)
## End(Not run)
```

scrape_day  Scrape play-by-play for single day’s games

Description

Scrape play-by-play for single day’s games

Usage

```r
scrape_day(day = as.Date(Sys.Date(), "%Y-%m-%d"))
```

Arguments

day  A day in the format of 'YYYY-MM-DD'; defaults to system date.

Value

A tibble containing all play-by-play data for a given day in the same format as the output of `scrape_game`

Examples

```r
## Not run:
pbp_day <- scrape_day(day = "2015-01-06")
## End(Not run)
```
scrape_game  Scrape game play-by-play

Description
Scrapes play-by-play data for a specified game ID.

Usage
scrape_game(game_id)

Arguments

game_id  Game ID to scrape

Value
A tibble containing event-based play-by-play data for an individual NHL game. The resulting data will have columns for:

- `xg` Numeric expected goal value for unblocked shot events
- `event` String defining the event
- `event_type` String with alternate event definition; in all caps
- `secondary_type` String defining secondary event type
- `event_team` String defining the primary team involved in the event
- `event_team_type` String indicator of event team type: home or away
- `description` String detailed description of event
- `period` Integer value of the game period
- `period_seconds` Numeric value of the seconds into the period of the event
- `period_seconds_remaining` Numeric value of the seconds remaining in the period
- `game_seconds` Numeric value of the seconds into the game of the event
- `game_seconds_remaining` Numeric value of the seconds remaining in the game; negative for overtime periods
- `home_score` Integer value of the home team score after the event
- `away_score` Integer value of the away team score after the event
- `event_player_1_name` String name of the primary event player
- `event_player_1_type` String indicator for the role of event_player_1
- `event_player_2_name` String name of the secondary event player
- `event_player_2_type` String indicator for the role of event_player_2
- `event_player_3_name` String name of the tertiary event player
- `event_player_3_type` String indicator for the role of event_player_3
**event_goalie_name**  String name of the goalie involved in the event

**strength_code**  String indicator for game strength: EV, SH, or PP

**strength**  String name for game strength: Even, Shorthanded, or Power Play

**strength_state**  String name for detailed game strength in the form of '(event team skaters)v(opponent skaters)'

**penalty_minutes**  Integer value of the penalty minutes on penalty events

**penalty_severity**  String name for penalty severity: Minor or Major

**num_on**  Integer value of the number of skaters substituted on during a shift change event

**players_on**  String of player names substituted on during a shift change event

**num_off**  Integer value of the number of skaters substituted off during a shift change event

**players_off**  String of player names substituted off during a shift change event

**extra_attacker**  Logical indicator of whether or not the event team had their goalie pulled

**x**  Numeric x-coordinate of event in feet, with origin at center ice

**y**  Numeric y-coordinate of event in feet, with origin at center ice

**x_fixed**  Numeric transformed x-coordinate of event in feet, where the home team always shoots to the right, away team to the left

**y_fixed**  Numeric transformed y-coordinate of event in feet, where the home team always shoots to the right, away team to the left

**shot_distance**  Numeric distance (in feet) to center of net for unblocked shot events

**shot_angle**  Numeric angle (in degrees) to center of net for unlocked shot events

**home_skaters**  Numeric value for number of skaters on the ice for the home team, excluding the goalie

**away_skaters**  Numeric value for number of skaters on the ice for the away team, excluding the goalie

**home_on_1**  String name of home team player on ice

**home_on_2**  String name of home team player on ice

**home_on_3**  String name of home team player on ice

**home_on_4**  String name of home team player on ice

**home_on_5**  String name of home team player on ice

**home_on_6**  String name of home team player on ice

**home_on_7**  String name of home team player on ice

**away_on_1**  String name of away team player on ice

**away_on_2**  String name of away team player on ice

**away_on_3**  String name of away team player on ice

**away_on_4**  String name of away team player on ice

**away_on_5**  String name of away team player on ice

**away_on_6**  String name of away team player on ice

**away_on_7**  String name of away team player on ice
home_goalie  String name of home goalie on ice
away_goalie  String name of away goalie on ice
game_id  Integer value of assigned game ID
event_idx  Numeric index for event
event_id  Numeric id for event – more specified than event_idx
event_player_1_id  Integer value of the player ID for the primary event player
event_player_1_link  String value of the NHL.com player link for the primary event player
event_player_1_season_total  Integer value for the total events for the primary event player this season
event_player_2_id  Integer value of the player ID for the secondary event player
event_player_2_link  String value of the NHL.com player link for the secondary event player
event_player_2_season_total  Integer value for the total events for the secondary event player this season
event_player_3_id  Integer value of the player ID for the tertiary event player
event_player_3_link  String value of the NHL.com player link for the tertiary event player
event_player_3_season_total  Integer value for the total events for the tertiary event player this season
event_goalie_id  Integer value of the player ID for the event goalie
event_goalie_link  String value of the NHL.com player link for the event goalie
event_goalie_type  String indicator for the role of the event_goalie
game_winning_goal  Logical indicator of whether or not the goal scored was the game-winning goal
empty_net  Logical indicator of whether or not the goal scored was on an empty net
period_type  String name of period type: REGULAR, OVERTIME, or SHOOTOUT
ordinal_num  String name of the ordinal period: 1st, 2nd, 3rd, 4th...
period_time  String value of the time into the period of the event
period_time_remaining  String value of the time remaining in the period
date_time  String value of the real-world timestamp of the event
event_team_id  Integer value of the NHL ID of event_team
event_team_link  String value of the NHL.com team link for the event_team
event_team_abbr  String value of the 3-letter NHL abbreviation for the event_team
home_final  Integer value of the final score for the home team
away_final  Integer value of the final score for the away team
season  String value of the official NHL season
season_type  String indicator of season type: R, or P
game_date  Date of game
game_start  Date time of start of game in US/Eastern time zone
game_end  Date time of end of game in US/Eastern time zone
game_length  Period value of length of game, in hours:minutes
game_state  String indicator of state of game
detailed_state  String indicator of detailed game state
venue_id  Integer value of the NHL ID for the venue
venue_name  String name of the game venue
venue_link  String value of the NHL.com link for the venue
home_name  String name of the home team
home_abbreviation  String value of the 3-letter NHL abbreviation of the home team
home_division_name  String value of the name of the NHL division of the home team
home_conference_name  String value of the name of the NHL conference of the home team
home_id  Integer value of the NHL ID of the home team
away_name  String name of the away team
away_abbreviation  String value of the 3-letter NHL abbreviation of the away team
away_division_name  String value of the name of the NHL division of the away team
away_conference_name  String value of the name of the NHL conference of the away team
away_id  Integer value of the NHL ID of the away team

Examples

```r
## Not run:
pbp <- scrape_game(2020020420)

## End(Not run)
```

---

scrape_season  Scrape full season play-by-play

Description

Scrape full season play-by-play

Usage

```r
scrape_season(season, type = "all")
```

Arguments

- **season**: An integer value denoting the end year of the season to scrape
- **type**: A character vector of the game types to include: REG, POST, or "all"

Value

A tibble containing all play-by-play data for a given season in the same format as the output of 
`scrape_game`
team_logos_colors

Examples

```r
## Not run:
# scrape all regular season & postseason games for the 2016-2017 season
pbp_2016_2017 <- scrape_season(2017, type = "REG")

## End(Not run)
```

team_logos_colors Team logos & colors

Description

A dataset containing the full team names, abbreviations, colors & logos for all 32 NHL teams (plus some defunct teams)

Usage

team_logos_colors

Format

A data frame with 35 rows and 12 variables:

- `full_team_name` full team name
- `team_abbr` NHL.com team abbreviation
- `team_nick` lowercase, no spaces team nickname
- `division` current NHL division
- `conference` current NHL conference
- `team_logo_espn` primary team logo from ESPN.com
- `team_color1` current primary team color
- `team_color2` current secondary team color
- `team_logo_alternate` alternate or throwback logo
- `team_color_alt1` alternate logo primary color
- `team_color_alt2` alternate logo secondary color
- `status` active or inactive

Source

https://www.espn.com/nhl/teams
https://www.sportslogos.net/teams/list_by_league/1/National_Hockey_League/NHL/logos/
https://teamcolorcodes.com/nhl-team-color-codes/
Index

* datasets
  team_logos_colors, 22

calculate_individual, 2, 4
calculate_on_ice, 3, 4
calculate_toi, 5
calculate_xg, 6

get_current_rosters, 7
get_draft_class, 7
get_game_ids, 8
get_game_info, 9
get_game_rosters, 10
get_game_shifts, 10
get_goalie_stats_hr, 11
get_jersey_players, 11
get_player_stats_hr, 12
get_rosters, 13
get_skater_stats_hr, 14
get_standings, 14
get_team_records, 15
get_team_rosters, 16

load_pbp, 2–5, 16

scrape_day, 17
scrape_game, 2–5, 9, 17, 18, 21
scrape_season, 21

team_logos_colors, 22