fastRhockey: A Package For Women's Hockey Data

Author

Abstract

The boxscore and roster data scraped through fastRhockey allows for the aggregation of summary statistics and enable fantasy PHF hockey to be played. The processed play-by-play data fastRhockey delivers contains every event that occurs within a PHF game, the primary/secondary players, as well as player-on-ice data for goals. fastRhockey was designed to increase access to women's hockey data, and the data returned by the package enables the women's hockey analytics space to grow, replicating and reproducing work done regarding men's hockey, as well as novel research specific to the PHF. fastRhockey was developed as a part of the SportsDataverse universe, and was merged with hockeyR to include NHL data and capabilities under the fastRhockey umbrella, with plans to continue building out women's hockey functions as the women's hockey landscape grows.

1 Introduction

The data landscape around the Premier Hockey Federation (PHF), formerly the National Women's Hockey League (NWHL) has been uneven and rocky. The league itself has almost continuously undergone changes year-to-year, which have been reflected in its data.

At times, there have been scrapers and packages intended to create access to PHF data, which at times, was detailed enough to include the location data of various events. However, with the transition from the NWHL to PHF, and the accompanying website changes, that API access and detail of data was lost and previous scrapers were broken.

`fastRhockey` was developed to recreate access to PHF data given those changes. As the PHF grows, with games being played on ESPN, it is crucial that there is a publicly (and easily) available data source for both team and public analysts.

While the data that is tracked and provided through the API that `fastRhockey` accesses is not always consistent and clean thanks to extraneous factors on the PHF side of things, `fastRhockey` is a women's hockey R package created for the long haul and to be maintained as the PHF grows and the demand for women's hockey data continues to grow.

In addition to the PHF data and functions available through `fastRhockey`, functions to access the NHL API are available through `fastRhockey`. The following paper will briefly discuss the NHL API access and functions, but they were not the primary motivation for creating the `fastRhockey` paper.

`fastRhockey` can be found on GitHub in its own repository as well as downloaded from CRAN using `install.packages("fastRhockey")`.

2 Structure of fastRhockey

`fastRhockey` was developed entirely as an R package that pulls live data while additionally storing the data in a `fastRhockey-data` repository. `fastRhockey` contains a multitude of functions, but the ones that return actionable data being with either the `phf_` or `load_` prefix.

Additionally, the package contains plenty of helper and additional functions that are designed to work with the data and to parse it into actionable play-by-play data. Many of the functions begin with the `process_phf...` prefix and are designed to clean and prepare the data in the wrapper functions that one uses to pull the data. While the code for these functions was written as part of the `fastRhockey` package, they are not necessary or expected to be used by anyone doing PHF analysis.

The functions that `fastRhockey` is dependent on are common R packages, for example, `rvest` and `jsonlite` are relied upon for accessing the API and parsing the returned data, as well as the `stringr` package, used for the parsing of text play-by-play data.

3 Installing fastRhockey

`fastRhockey` has been submitted to CRAN and can be downloaded from CRAN using the following method:

```
install.packages("fastRhockey")
```

Additionally, the development branch of the package can be downloaded using `devtools` if one prefers that method:

```
if (!requireNamespace('devtools', quietly = TRUE)){
  install.packages('devtools')
}
devtools::install_github(repo = "sportsdataverse/fastRhockey")
```

3.1 Data Source

The PHF API is accessible with an API Key that can be set for each individual person using the `getOption` function from `base R`. A unique API key can be found through the PHF website if one is curious, but it has not caused any issues yet.

3.2 Analysis

Given the way that `fastRhockey` was set up and developed, an aspiring analyst can start at the very beginning of their questions and use the data returned from each function in subsequent functions.

Let's start with finding the clinching game of the 2022 Isobel Cup to explore the functionality of `fastRhockey`. An exploration begins with the `phf_schedule()` function, which takes a season input as an argument.

```
phf_schedule(season = 2022) %>%
  dplyr::filter(game_type == 'Playoffs') %>%
  dplyr::filter(home_team == 'Connecticut Whale' &
                  away_team == 'Boston Pride') %>%
  dplyr::select(game_id, date_group, facility, attendance,
                home_team, away_team,
                home_score, away_score,
                winner)
```

```
##   game_id date_group                facility attendance         home_team
## 1  514869 2022-03-28 AdventHealth Center Ice         NA Connecticut Whale
##      away_team home_score away_score       winner
## 1 Boston Pride          2          4 Boston Pride
```

Most notably, we're able to extract the `game_id` of the game, as well as the important data from the game, namely the winner and teams involved. The returned dataframe contains plenty more columns, most of which are metadata and ID columns.

```
phf_schedule(season = 2021) %>%
  colnames()
```

```
##  [1] "type"               "id"                 "league_id"
##  [4] "season_id"          "tournament_id"      "game_id"
##  [7] "number"             "datetime"           "datetime_tz"
## [10] "time_zone"          "time_zone_abbr"     "updated_at"
## [13] "created_at"         "home_team_id"       "home_team"
## [16] "home_team_short"    "home_team_logo_url" "away_team_id"
## [19] "away_team"          "away_team_short"    "away_team_logo_url"
## [22] "home_division_id"   "home_division"      "away_division_id"
## [25] "away_division"      "home_score"         "away_score"
## [28] "facility_id"        "facility"           "facility_address"
## [31] "rink_id"            "rink"               "game_type"
## [34] "notes"              "status"             "overtime"
## [37] "shootout"           "allow_players"      "tickets_url"
## [40] "watch_live_url"     "external_url"       "has_play_by_play"
## [43] "highlight_color"    "attendance"         "date_group"
## [46] "winner"
```

Now we know that the Boston Pride defeated the Connecticut Whale by a score of 4-2 on March 28th, 2022 at AdventHealth Center Ice for the Isobel Cup title, where the `game_id = 514869`.

Let's take that game ID and look at how the two teams fared statistically.

```
phf_team_box(game_id = 514869) %>%
  dplyr::select(team, winner, total_scoring, total_shots,
                successful_power_play, penalty_minutes,
                faceoff_percent, period_1_scoring,
                period_2_scoring, period_3_scoring)
```

```
## # A tibble: 2 x 10
##   team  winner total_scoring total_shots successful_power_play penalty_minutes
##   <chr> <lgl>          <int>       <int>                 <dbl>           <dbl>
## 1 bos   TRUE               4          30                     0               7
## 2 ctw   FALSE              2          34                     0               4
## # ... with 4 more variables: faceoff_percent <dbl>, period_1_scoring <int>,
## #   period_2_scoring <int>, period_3_scoring <int>
```

A very quick query and we have the story of the game. Boston won the game by turning on the jets in the third period, scoring three of their four goals in the final period.

The Pride additionally won more faceoffs than the Whale; prior women's hockey research has indicated that winning faceoffs in the women's game has a more profound impact on winning than in the men's game. While the detailed location data may no longer be available, `fastRhockey` allows for further exploration of research questions like the above.

In addition to team box scores, `fastRhockey` returns player box score-level data, for both skaters and goalies.

```
phf_player_box(game_id = 514869)$skaters
```

```
## # A tibble: 36 x 23
##    player_jersey player_name       position goals assists points penalty_minutes
##            <int> <chr>             <chr>    <int>   <int>  <int>           <int>
##  1             8 Alyssa Wohlfeiler F            0       1      1               2
##  2            12 Allie Munroe      D            0       1      1               0
##  3            17 Taylor Girard     F            1       0      1               0
##  4            22 Kennedy Marchment F            0       1      1               2
##  5            24 Janine Weber      F            0       1      1               0
##  6            88 Amanda Conway     F            1       0      1               0
##  7             5 Tori Howran       D            0       0      0               0
##  8             6 Shannon Turner    D            0       0      0               0
##  9             9 Kaycie Anderson   F            0       0      0               0
## 10            11 Emily Fluke       F            0       0      0               0
## # ... with 26 more rows, and 16 more variables: plus_minus <int>,
## #   shots_on_goal <int>, blocks <int>, giveaways <int>, takeaways <int>,
## #   faceoffs_won_lost <chr>, faceoffs_win_pct <dbl>, powerplay_goals <int>,
## #   shorthanded_goals <int>, shots <int>, shots_blocked <int>,
## #   faceoffs_won <int>, faceoffs_lost <int>, skaters_href <chr>,
## #   player_id <chr>, game_id <dbl>
```

```
phf_player_box(game_id = 514869)$goalies
```

```
## # A tibble: 4 x 13
##   player_jersey player_name       shots_against goals_against saves save_percent
##           <int> <chr>                     <int>         <int> <int>        <dbl>
## 1            35 Abbie Ives                   29             3    26        0.897
## 2            55 Mariah Fujimagari             0             0     0        0
## 3            35 Lovisa Selander               0             0     0        0
## 4            88 Katie Burt                   34             2    32        0.941
## # ... with 7 more variables: minutes_played <chr>, penalty_minutes <int>,
## #   goals <int>, assists <int>, goalies_href <chr>, player_id <chr>,
## #   game_id <dbl>
```

One potential public usage of this data is creating the opportunity to play fantasy PHF hockey using these stats.

However, the most notable return of `fastRhockey` is the play-by-play and event-level data that can be extracted from the `phf_pbp()` function. The function returns about 78 columns of data, anywhere from metadata about an event to the player involved, or the players on the ice in the event of a goal.

```
phf_pbp(game_id = 514869)
```

```
## # A tibble: 168 x 78
##    play_type team       time  play_description scoring_team_ab~ scoring_team_on~
##    <chr>     <chr>      <chr> <chr>            <chr>            <chr>
##  1 Goalie    Boston Pr~ 00:00 Starting Goalie~ <NA>             <NA>
##  2 Goalie    Connectic~ 00:00 Starting Goalie~ <NA>             <NA>
##  3 Faceoff   Boston Pr~ 00:00 #18 Taylor Wenc~ <NA>             <NA>
##  4 Shot      Connectic~ 00:17 #22 Kennedy Mar~ <NA>             <NA>
##  5 Faceoff   Boston Pr~ 00:17 #29 Kayla Fries~ <NA>             <NA>
```

```
##  6 Giveaway  Connectic~ 00:35 #22 Kennedy Mar~ <NA>              <NA>
##  7 Giveaway  Boston Pr~ 01:25 #14 Jillian Dem~ <NA>              <NA>
##  8 Faceoff   Boston Pr~ 01:27 #11 Evelina Ras~ <NA>              <NA>
##  9 Faceoff   Connectic~ 01:55 #22 Kennedy Mar~ <NA>              <NA>
## 10 Shot      Connectic~ 02:04 #5 Tori Howran ~ <NA>              <NA>
## # ... with 158 more rows, and 72 more variables: offensive_player_name_1 <chr>,
## #   offensive_player_name_2 <chr>, offensive_player_name_3 <chr>,
## #   offensive_player_name_4 <chr>, offensive_player_name_5 <chr>,
## #   offensive_player_name_6 <chr>, defending_team_abbrev <chr>,
## #   defending_team_on_ice <chr>, defensive_player_name_1 <chr>,
## #   defensive_player_name_2 <chr>, defensive_player_name_3 <chr>,
## #   defensive_player_name_4 <chr>, defensive_player_name_5 <chr>, ...
```

The combination of this play-by-play data and the player box scores has facilitated further women's hockey faceoff research, specifically regarding the usage of ELO Ratings to evaluate players in faceoff situations.

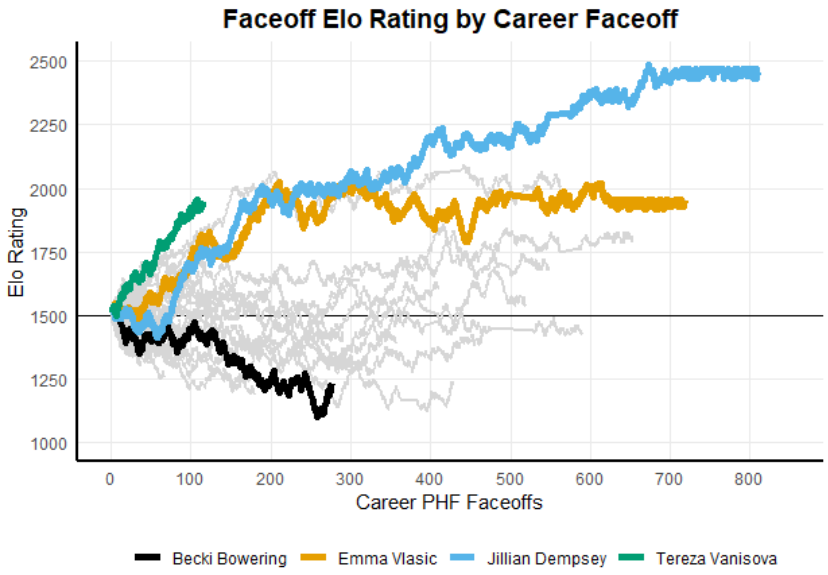

Figure 1: Rolling ELO Faceoff Ratings for the PHF.

The above graph shows rolling ELO ratings by career faceoff across the PHF/NWHL. Using the idea of ELO ratings, this approach compares skater wins against their level of competition and assigns value to each won or lost faceoff.

The play-by-play data returned by `phf_pbp()` allows for these rolling ELO ratings where the ratings are calculated by subsequent faceoffs, ensuring that the faceoff ratings are always up to date when they're calculated, even within an individual game.

4 NHL Data and Capabilities

`fastRhockey` was integrated with `hockeyR` to add NHL capabilities and functions to the `fastRhockey` package as a part of the SportsDataverse. The NHL data available through `fastRhockey` is quite expansive, covering all aspects of the NHL with the following functions and more:

- `nhl_teams`: metadata on the teams in the NHL

- `nhl_game_boxscore`: NHL boxscore data on a game level
- `nhl_game_shifts`: returns NHL shift and time on ice data; there is no PHF equivalent for this data
- `nhl_player_stats`: stats on a player level in the NHL
  - `nhl_player_info`: contains player metadata for the NHL
- `nhl_schedule`: schedule data for the NHL
- `nhl_draft`: data for the NHL draft, specifically picks, prospects, and prospect info
  - `nhl_draft_prospects` and `nhl_draft_prospects_info` for various draft prospect information
- `nhl_team_logos`: returns data on the logos for the NHL teams; equivalent of `phf_team_logos`
- `load_nhl_pbp`: loading full seasons of NHL play-by-play data

The data and functions returned from the NHL capability of `fastRhockey` are impressive and cover a lot of ground, but were integrated from `hockeyR` and NHL capability was not the primary motivation for developing `fastRhockey`. However, it is a huge and important addition to incorporate the major hockey data sources under one package umbrella and turns `fastRhockey` into the go-to source for anyone looking to do any sort of hockey analysis.

5 Conclusion

`fastRhockey` was developed as an answer to the lack of consistent and publicly available women's hockey data for the Premier Hockey Federation (PHF). The PHF, and women's hockey in general, has been rapidly growing in popularity, and with that, comes a growing desire for numbers and stats about the PHF. `fastRhockey` is a reliable and consistent package that is designed to handle the irregularities and inconsistencies that are the norm with women's hockey data. The package contains additional NHL capabilities to consolidate major hockey stats under one package.

`fastRhockey` can be found on GitHub and as well as CRAN, using `install.packages("fastRhockey")`.