Package ‘nfl4th’

August 21, 2023

Title  Functions to Calculate Optimal Fourth Down Decisions in the National Football League

Version  1.0.4

Description  A set of functions to estimate outcomes of fourth down plays in the National Football League and obtain fourth down plays from <https://www.nfl.com/> and <https://www.espn.com/>.

License  MIT + file LICENSE


BugReports  https://github.com/nflverse/nfl4th/issues

Depends  R (>= 3.6)

Imports  backports (>= 1.1.6), curl, dplyr, glue, httr, janitor, jsonlite, magrittr, mgcv, nflfastR (>= 4.0.0), nflreadr, purrr, rlang, stringr, tibble, tidyrl, tidyselect, xgboost

Suggests  data.table, future, gt, nflplotR, rmarkdown, tictoc, testthat (>= 2.0.0), withr

Encoding  UTF-8

RoxygenNote  7.2.3

Config/testthat/edition  2

NeedsCompilation  no

Author  Ben Baldwin [aut, cre, cph], Sebastian Carl [ctb]

Maintainer  Ben Baldwin <bbaldwin206@gmail.com>

Repository  CRAN

Date/Publication  2023-08-21 09:32:33 UTC


**R topics documented:**

- `add_2pt_probs` ............................... 2
- `add_4th_probs` ............................... 3
- `get_4th_plays` ............................... 4
- `load_4th_pbp` ............................... 5
- `make_2pt_table_data` ....................... 6
- `make_table_data` ......................... 7
- `nfl4th_clear_cache` ......................... 8

**Index** 9

---

**add_2pt_probs**

*Get 2pt decision probabilities*

**Description**

Get various probabilities associated with each option on PATs (go for it, kick PAT).

**Usage**

```
add_2pt_probs(df)
```

**Arguments**

- **df**
  A data frame of decisions to be computed for.

**Value**

Original data frame Data frame plus the following columns added:

- `first_down_prob`, `wp_fail`, `wp_succeed`, `go_wp`, `fg_make_prob`, `miss_fg_wp`, `make_fg_wp`, `fg_wp`, `punt_wp`
- `wp_0` Win probability when scoring 0 points on PAT.
- `wp_1` Win probability when scoring 1 point on PAT.
- `wp_2` Win probability when scoring 2 points on PAT.
- `conv_1pt` Probability of making PAT kick.
- `conv_2pt` Probability of converting 2-pt attempt.
- `wp_go1` Win probability associated with going for 1.
- `wp_go2` Win probability associated with going for 2.
Examples

```r
play <-
  tibble::tibble(
    # things to help find the right game (use "reg" or "post")
    home_team = "GB",
    away_team = "TB",
    posteam = "GB",
    type = "post",
    season = 2020,

    # information about the situation
    qtr = 4,
    quarter_seconds_remaining = 123,
    score_differential = -2,

    home_opening_kickoff = 0,
    posteam_timeouts_remaining = 3,
    defteam_timeouts_remaining = 3
  )

probs <- nfl4th::add_2pt_probs(play)
dplyr::glimpse(probs)
```

---

`add_4th_probs`  
*Get 4th down decision probabilities*

**Description**

Get various probabilities associated with each option on 4th downs (go for it, kick field goal, punt).

**Usage**

`add_4th_probs(df)`

**Arguments**

- `df`  
  A data frame of decisions to be computed for.

**Value**

Original data frame Data frame plus the following columns added:

- `go_boost`  
  Gain (or loss) in win prob associated with choosing to go for it (percentage points).

- `first_down_prob`  
  Probability of earning a first down if going for it on 4th down.

- `wp_fail`  
  Win probability in the event of a failed 4th down attempt.
**wp_succeed**  Win probability in the event of a successful 4th down attempt.

**go_wp**  Average win probability when going for it on 4th down.

**fg_make_prob**  Probability of making field goal.

**miss_fg_wp**  Win probability in the event of a missed field goal.

**make_fg_wp**  Win probability in the event of a made field goal.

**fg_wp**  Average win probability when attempting field goal.

**punt_wp**  Average win probability when punting.

### Examples

```r
play <-
tibble::tibble(
  # things to help find the right game (use "reg" or "post")
  home_team = "GB",
  away_team = "TB",
  posteam = "GB",
  type = "post",
  season = 2020,

  # information about the situation
  qtr = 4,
  quarter_seconds_remaining = 129,
  ydstogo = 8,
  yardline_100 = 8,
  score_differential = -8,

  home_opening_kickoff = 0,
  posteam_timeouts_remaining = 3,
  defteam_timeouts_remaining = 3
)

probs <- nfl4th::add_4th_probs(play)

dplyr::glimpse(probs)
```

---

**get_4th_plays**  

Get 4th down plays from a game

### Description

Get 4th down plays from a game.

### Usage

```r
get_4th_plays(gid)
```
Arguments
gid  A game to get 4th down decisions of.

Details
Obtains a data frame that can be used with add_4th_probs(). The following columns must be present:

- game_id : game ID in nflfastR format (eg '2020_20_TB_GB')

Value
Original data frame Data frame plus the following columns added:

- desc Play description from ESPN.
- type_text Play type text from ESPN.
- index Index number of play from a given game. Useful for tracking plays (e.g. for 4th down bot).
- The rest All the columns needed for add_4th_probs().

Examples

plays <- nfl4th::get_4th_plays('2020_20_TB_GB')
dplyr::glimpse(plays)

load_4th_pbp

Safe calculated 4th down probabilities from nflfastR data

Description
Load calculated 4th down probabilities from nflfastR data.

Usage

load_4th_pbp(seasons, fast = FALSE)

Arguments

- seasons  Seasons to load. Must be 2014 and later.
- fast  Defaults to FALSE. If TRUE, loads pre-computed decisions from repository

Value
nflfastR data on 4th downs with the add_4th_probs() columns added and also the following:

- go 100 if a team went for it on 4th down, 0 otherwise. It’s 100 and 0 as a convenience for obtaining percent of times going for it.
Examples

```r
try({
  probs <- load_4th_pbp(2019:2020)
  dplyr::glimpse(probs)
})
```

---

**make_2pt_table_data**  
Get 2pt decision probabilities

---

**Description**

Get a table with the probabilities associated with a 2-pt decision.

**Usage**

```r
make_2pt_table_data(probs)
```

**Arguments**

- `probs`  
  A data frame consisting of one play that has had `add_2pt_probs()` already run on it.

**Value**

A table showing the probabilities associated with each possible choice.

**Examples**

```r
play <-
tibble::tibble(
  # things to help find the right game (use "reg" or "post")
  home_team = "GB",
  away_team = "TB",
  posteam = "GB",
  type = "post",
  season = 2020,

  # information about the situation
  qtr = 4,
  quarter_seconds_remaining = 123,
  score_differential = -2,

  home_opening_kickoff = 0,
  posteam_timeouts_remaining = 3,
  defteam_timeouts_remaining = 3
)
```
make_table_data

probs <- nfl4th::add_2pt_probs(play)
nfl4th::make_2pt_table_data(probs)

make_table_data

Get 4th down decision probabilities

Description

Get a table with the probabilities on 4th down.

Usage

make_table_data(probs)

Arguments

probs

A data frame consisting of one play that has had add_4th_probs() already run on it.

Value

A table showing the probabilities associated with each possible choice.

Examples

play <-
  tibble::tibble(
    home_team = "GB",
    away_team = "TB",
    posteam = "GB",
    type = "post",
    season = 2020,
    qtr = 4,
    quarter_seconds_remaining = 129,
    ydstogo = 8,
    yardline_100 = 8,
    score_differential = -8,
    home_opening_kickoff = 0,
    posteam_timeouts_remaining = 3,
    defteam_timeouts_remaining = 3
  )
probs <- nfl4th::add_4th_probs(play)
nfl4th::make_table_data(probs)

---

### nfl4th_clear_cache

**Reset nfl4th Package Cache**

**Description**

Reset nfl4th Package Cache

**Usage**

```r
nfl4th_clear_cache(type = c("games", "fd_model", "wp_model", "all"))
```

**Arguments**

- `type` One of "games" (the default), "fd_model", or "all". "games" will remove an internally used games file. "fd_model" will remove the nfl4th 4th down model (only necessary in the unlikely case of a model update). "wp_model" will remove the nfl4th win probability model (only necessary in the unlikely case of a model update). "all" will remove all of the above.

**Value**

Returns TRUE invisibly if cache has been cleared.

**Examples**

```r
nfl4th_clear_cache()
```
Index

add_2pt_probs, 2
add_4th_probs, 3

get_4th_plays, 4

load_4th_pbp, 5

make_2pt_table_data, 6
make_table_data, 7

nfl4th_clear_cache, 8