

Package ‘nflfastR’

June 16, 2020

Type Package

Title Functions to Efficiently Scrape NFL Play by Play Data

Version 2.0.3

Description nflfastR is a set of functions to efficiently
scrape NFL play-by-play data.

License MIT + file LICENSE

Depends R (>= 3.5.0)

Imports dplyr,

glue,

httr,

janitor,

jsonlite,

lubridate,

magrittr,

mgcv,

progressr,

purrr,

stats,

stringr,

tibble,

tidyr,

tidyselect,

xgboost (>= 1.1)

Suggests furrr,

future

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

R topics documented:

add_qb_epa	2
calculate_expected_points	2
clean_pbp	3
fast_scraper	4
fast_scraper_schedules	14
teams_colors_logos	15

Index**17**

add_qb_epa	<i>Compute QB epa</i>
------------	-----------------------

Description

Compute QB epa

Usage

```
add_qb_epa(d)
```

Arguments

`d` is a Data frame of play-by-play data scraped using [fast_scraper](#).

Details

Add the variable 'qb_epa', which gives QB credit for EPA for up to the point where a receiver lost a fumble after a completed catch and makes EPA work more like passing yards on plays with fumbles

calculate_expected_points	<i>Compute expected points</i>
---------------------------	--------------------------------

Description

for provided plays. Returns the data with probabilities of each scoring event and EP added. The following columns must be present: season, week, home_team, posteam, roof (coded as 'open', 'closed', or 'retractable'), half_seconds_remaining, yardline_100, ydstogo, posteam_timeouts_remaining, defteam_timeouts_remaining

Usage

```
calculate_expected_points(pbp_data)
```

Arguments

`pbp_data` Play-by-play dataset to estimate expected points for.

Details

Computes expected points for provided plays. Returns the data with probabilities of each scoring event and EP added. The following columns must be present: season, week, home_team, posteam, roof (coded as 'open', 'closed', or 'retractable'), half_seconds_remaining, yardline_100, ydstogo, posteam_timeouts_remaining, defteam_timeouts_remaining

Value

The original pbp_data with the following columns appended to it:

- ep - expected points.
- no_score_prob - probability of no more scoring this half.
- opp_fg_prob - probability next score opponent field goal this half.
- opp_safety_prob - probability next score opponent safety this half.
- opp_td_prob - probability of next score opponent touchdown this half.
- fg_prob - probability next score field goal this half.
- safety_prob - probability next score safety this half.
- td_prob - probability text score touchdown this half.

clean_pbp	<i>Clean Play by Play Data</i>
-----------	--------------------------------

Description

Clean Play by Play Data

Usage

```
clean_pbp(pbp)
```

Arguments

pbp is a Data frame of play-by-play data scraped using [fast_scraper](#).

Details

Build columns that capture what happens on all plays, including penalties, using string extraction from play description. Loosely based on Ben's nflscrapR guide (<https://gist.github.com/guga31bb/5634562c5a2a7b1e996>) but updated to work with the RS data, which has a different player format in the play description; e.g. 24-M.Lynch instead of M.Lynch. The function also standardizes team abbreviations so that, for example, the Chargers are always represented by 'LAC' regardless of which year it was.

Value

The input Data Frame of the paramter 'pbp' with the following columns added:

- success - Binary indicator wheter epa > 0 in the given play.
- passer - Name of the dropback player (scrambles included) including plays with penalties.
- rusher - Name of the rusher (no scrambles) including plays with penalties.
- receiver - Name of the receiver including plays with penalties.
- pass - Binary indicator if the play was a pass play (sacks and scrambles included).
- rush - Binary indicator if the play was a rushing play.
- special - Binary indicator if the play was a special teams play.
- first_down - Binary indicator if the play ended in a first down.

- play - Binary indicator: 1 if the play was a 'normal' play (including penalties), 0 otherwise.
- passer_id - ID of the player in the 'passer' column (NOTE: ids vary pre and post 2011)
- rusher_id - ID of the player in the 'rusher' column (NOTE: ids vary pre and post 2011)
- receiver_id - ID of the player in the 'receiver' column (NOTE: ids vary pre and post 2011)
- name - Name of the 'passer' if it is not 'NA', or name of the 'rusher' otherwise.
- id - ID of the player in the 'name' column.

fast_scraper

Get NFL Play by Play Data

Description

Get NFL Play by Play Data

Usage

```
fast_scraper(game_ids, source = "nfl", pp = FALSE)
```

Arguments

game_ids	Vector of character ids (see details for further information)
source	Character - must now be nfl or unspecified (see details for further information)
pp	Logical - either TRUE or FALSE (see details for further information)

Details

To load valid game_ids please use the package function [fast_scraper_schedules](#).

The source parameter controls from which source the data is being scraped. The old parameters rs as well as gc are not valid anymore. Please use nfl or leave unspecified. The pp parameter controls if the scraper should use parallel processing. Please note that the initiating process takes a few seconds which means it may be better to set pp = FALSE if you are scraping just a few games.

Value

Data frame where each individual row represents a single play for all passed game_ids containing the following detailed information (description partly extracted from nflscrapR):

- play_id - Numeric play id that when used with game_id and drive provides the unique identifier for a single play.
- game_id - Ten digit identifier for NFL game.
- home_team - String abbreviation for the home team.
- away_team - String abbreviation for the away team.
- season_type - 'REG' or 'POST' indicating if the game belongs to regular or post season.
- week - Season week.
- posteam - String abbreviation for the team with possession.
- posteam_type - String indicating whether the posteam team is home or away.
- defteam - String abbreviation for the team on defense.

- side_of_field - String abbreviation for which team's side of the field the team with possession is currently on.
- yardline_100 - Numeric distance in the number of yards from the opponent's endzone for the posteam.
- game_date - Date of the game.
- quarter_seconds_remaining - Numeric seconds remaining in the quarter.
- half_seconds_remaining - Numeric seconds remaining in the half.
- game_seconds_remaining - Numeric seconds remaining in the game.
- game_half - String indicating which half the play is in, either Half1, Half2, or Overtime.
- quarter_end - Binary indicator for whether or not the row of the data is marking the end of a quarter.
- drive - Numeric drive number in the game.
- sp - Binary indicator for whether or not a score occurred on the play.
- qtr - Quarter of the game (5 is overtime).
- down - The down for the given play.
- goal_to_go - Binary indicator for whether or not the posteam is in a goal down situation.
- time - Time at start of play provided in string format as minutes:seconds remaining in the quarter.
- yrdln - String indicating the current field position for a given play.
- ydstogo - Numeric yards in distance from either the first down marker or the endzone in goal down situations.
- ydsnet - Numeric value for total yards gained on the given drive.
- desc - Detailed string description for the given play.
- play_type - String indicating the type of play: pass (includes sacks), run (includes scrambles), punt, field_goal, kickoff, extra_point, qb_kneel, qb_spike, no_play (timeouts and penalties), and missing for rows indicating end of play.
- yards_gained - Numeric yards gained (or lost) for the given play.
- shotgun - Binary indicator for whether or not the play was in shotgun formation.
- no_huddle - Binary indicator for whether or not the play was in no_huddle formation.
- qb_dropback - Binary indicator for whether or not the QB dropped back on the play (pass attempt, sack, or scrambled).
- qb_kneel - Binary indicator for whether or not the QB took a knee.
- qb_spike - Binary indicator for whether or not the QB spiked the ball.
- qb_scramble - Binary indicator for whether or not the QB scrambled.
- pass_length - String indicator for pass length: short or deep.
- pass_location - String indicator for pass location: left, middle, or right.
- air_yards - Numeric value for distance in yards perpendicular to the line of scrimmage at where the targeted receiver either caught or didn't catch the ball.
- yards_after_catch - Numeric value for distance in yards perpendicular to the yard line where the receiver made the reception to where the play ended.
- run_location - String indicator for location of run: left, middle, or right.
- run_gap - String indicator for line gap of run: end, guard, or tackle
- field_goal_result - String indicator for result of field goal attempt: made, missed, or blocked.

- `kick_distance` - Numeric distance in yards for kickoffs, field goals, and punts.
- `extra_point_result` - String indicator for the result of the extra point attempt: good, failed, blocked, safety (touchback in defensive endzone is 1 point apparently), or aborted.
- `two_point_conv_result` - String indicator for result of two point conversion attempt: success, failure, safety (touchback in defensive endzone is 1 point apparently), or return.
- `home_timeouts_remaining` - Numeric timeouts remaining in the half for the home team.
- `away_timeouts_remaining` - Numeric timeouts remaining in the half for the away team.
- `timeout` - Binary indicator for whether or not a timeout was called.
- `timeout_team` - String abbreviation for which team called the timeout.
- `td_team` - String abbreviation for which team scored the touchdown.
- `posteam_timeouts_remaining` - Number of timeouts remaining for the possession team.
- `defteam_timeouts_remaining` - Number of timeouts remaining for the team on defense.
- `total_home_score` - Score for the home team at the start of the play.
- `total_away_score` - Score for the away team at the start of the play.
- `posteam_score` - Score the posteam at the start of the play.
- `defteam_score` - Score the defteam at the start of the play.
- `score_differential` - Score differential between the posteam and defteam at the start of the play.
- `posteam_score_post` - Score for the posteam at the end of the play.
- `defteam_score_post` - Score for the defteam at the end of the play.
- `score_differential_post` - Score differential between the posteam and defteam at the end of the play.
- `no_score_prob` - Predicted probability of no score occurring for the rest of the half based on the expected points model.
- `opp_fg_prob` - Predicted probability of the defteam scoring a FG next.
- `opp_safety_prob` - Predicted probability of the defteam scoring a safety next.
- `opp_td_prob` - Predicted probability of the defteam scoring a TD next.
- `fg_prob` - Predicted probability of the posteam scoring a FG next.
- `safety_prob` - Predicted probability of the posteam scoring a safety next.
- `td_prob` - Predicted probability of the posteam scoring a TD next.
- `extra_point_prob` - Predicted probability of the posteam scoring an extra point.
- `two_point_conversion_prob` - Predicted probability of the posteam scoring the two point conversion.
- `ep` - Using the scoring event probabilities, the estimated expected points with respect to the possession team for the given play.
- `epa` - Expected points added (EPA) by the posteam for the given play.
- `total_home_epa` - Cumulative total EPA for the home team in the game so far.
- `total_away_epa` - Cumulative total EPA for the away team in the game so far.
- `total_home_rush_epa` - Cumulative total rushing EPA for the home team in the game so far.
- `total_away_rush_epa` - Cumulative total rushing EPA for the away team in the game so far.
- `total_home_pass_epa` - Cumulative total passing EPA for the home team in the game so far.
- `total_away_pass_epa` - Cumulative total passing EPA for the away team in the game so far.

- `air_epa` - EPA from the air yards alone. For completions this represents the actual value provided through the air. For incompletions this represents the hypothetical value that could've been added through the air if the pass was completed.
- `yac_epa` - EPA from the yards after catch alone. For completions this represents the actual value provided after the catch. For incompletions this represents the difference between the hypothetical `air_epa` and the play's raw observed EPA (how much the incomplete pass cost the posteam).
- `comp_air_epa` - EPA from the air yards alone only for completions.
- `comp_yac_epa` - EPA from the yards after catch alone only for completions.
- `total_home_comp_air_epa` - Cumulative total completions air EPA for the home team in the game so far.
- `total_away_comp_air_epa` - Cumulative total completions air EPA for the away team in the game so far.
- `total_home_comp_yac_epa` - Cumulative total completions yac EPA for the home team in the game so far.
- `total_away_comp_yac_epa` - Cumulative total completions yac EPA for the away team in the game so far.
- `total_home_raw_air_epa` - Cumulative total raw air EPA for the home team in the game so far.
- `total_away_raw_air_epa` - Cumulative total raw air EPA for the away team in the game so far.
- `total_home_raw_yac_epa` - Cumulative total raw yac EPA for the home team in the game so far.
- `total_away_raw_yac_epa` - Cumulative total raw yac EPA for the away team in the game so far.
- `wp` - Estimated win probability for the posteam given the current situation at the start of the given play.
- `def_wp` - Estimated win probability for the defteam.
- `home_wp` - Estimated win probability for the home team.
- `away_wp` - Estimated win probability for the away team.
- `wpa` - Win probability added (WPA) for the posteam.
- `home_wp_post` - Estimated win probability for the home team at the start of the play.
- `away_wp_post` - Estimated win probability for the away team at the start of the play.
- `vegas_wp` - Estimated win probability for the posteam given the current situation at the start of the given play, incorporating pre-game Vegas line.
- `vegas_home_wp` - Estimated win probability for the home team incorporating pre-game Vegas line.
- `total_home_rush_wpa` - Cumulative total rushing WPA for the home team in the game so far.
- `total_away_rush_wpa` - Cumulative total rushing WPA for the away team in the game so far.
- `total_home_pass_wpa` - Cumulative total passing WPA for the home team in the game so far.
- `total_away_pass_wpa` - Cumulative total passing WPA for the away team in the game so far.
- `air_wpa` - WPA through the air (same logic as `air_epa`).
- `yac_wpa` - WPA from yards after the catch (same logic as `yac_epa`).
- `comp_air_wpa` - The `air_wpa` for completions only.
- `comp_yac_wpa` - The `yac_wpa` for completions only.

- total_home_comp_air_wpa - Cumulative total completions air WPA for the home team in the game so far.
- total_away_comp_air_wpa - Cumulative total completions air WPA for the away team in the game so far.
- total_home_comp_yac_wpa - Cumulative total completions yac WPA for the home team in the game so far.
- total_away_comp_yac_wpa - Cumulative total completions yac WPA for the away team in the game so far.
- total_home_raw_air_wpa - Cumulative total raw air WPA for the home team in the game so far.
- total_away_raw_air_wpa - Cumulative total raw air WPA for the away team in the game so far.
- total_home_raw_yac_wpa - Cumulative total raw yac WPA for the home team in the game so far.
- total_away_raw_yac_wpa - Cumulative total raw yac WPA for the away team in the game so far.
- punt_blocked - Binary indicator for if the punt was blocked.
- first_down_rush - Binary indicator for if a running play converted the first down.
- first_down_pass - Binary indicator for if a passing play converted the first down.
- first_down_penalty - Binary indicator for if a penalty converted the first down.
- third_down_converted - Binary indicator for if the first down was converted on third down.
- third_down_failed - Binary indicator for if the posteam failed to convert first down on third down.
- fourth_down_converted - Binary indicator for if the first down was converted on fourth down.
- fourth_down_failed - Binary indicator for if the posteam failed to convert first down on fourth down.
- incomplete_pass - Binary indicator for if the pass was incomplete.
- touchback - Binary indicator for if a touchback occurred on the play.
- interception - Binary indicator for if the pass was intercepted.
- punt_inside_twenty - Binary indicator for if the punt ended inside the twenty yard line.
- punt_in_endzone - Binary indicator for if the punt was in the endzone.
- punt_out_of_bounds - Binary indicator for if the punt went out of bounds.
- punt_downed - Binary indicator for if the punt was downed.
- punt_fair_catch - Binary indicator for if the punt was caught with a fair catch.
- kickoff_inside_twenty - Binary indicator for if the kickoff ended inside the twenty yard line.
- kickoff_in_endzone - Binary indicator for if the kickoff was in the endzone.
- kickoff_out_of_bounds - Binary indicator for if the kickoff went out of bounds.
- kickoff_downed - Binary indicator for if the kickoff was downed.
- kickoff_fair_catch - Binary indicator for if the kickoff was caught with a fair catch.
- fumble_forced - Binary indicator for if the fumble was forced.
- fumble_not_forced - Binary indicator for if the fumble was not forced.
- fumble_out_of_bounds - Binary indicator for if the fumble went out of bounds.
- solo_tackle - Binary indicator if the play had a solo tackle (could be multiple due to fumbles).

- safety - Binary indicator for whether or not a safety occurred.
- penalty - Binary indicator for whether or not a penalty occurred.
- tackled_for_loss - Binary indicator for whether or not a tackle for loss occurred.
- fumble_lost - Binary indicator for if the fumble was lost.
- own_kickoff_recovery - Binary indicator for if the kicking team recovered the kickoff.
- own_kickoff_recovery_td - Binary indicator for if the kicking team recovered the kickoff and scored a TD.
- qb_hit - Binary indicator if the QB was hit on the play.
- rush_attempt - Binary indicator for if the play was a run.
- pass_attempt - Binary indicator for if the play was a pass attempt (includes sacks).
- sack - Binary indicator for if the play ended in a sack.
- touchdown - Binary indicator for if the play resulted in a TD.
- pass_touchdown - Binary indicator for if the play resulted in a passing TD.
- rush_touchdown - Binary indicator for if the play resulted in a rushing TD.
- return_touchdown - Binary indicator for if the play resulted in a return TD.
- extra_point_attempt - Binary indicator for extra point attempt.
- two_point_attempt - Binary indicator for two point conversion attempt.
- field_goal_attempt - Binary indicator for field goal attempt.
- kickoff_attempt - Binary indicator for kickoff.
- punt_attempt - Binary indicator for punts.
- fumble - Binary indicator for if a fumble occurred.
- complete_pass - Binary indicator for if the pass was completed.
- assist_tackle - Binary indicator for if an assist tackle occurred.
- lateral_reception - Binary indicator for if a lateral occurred on the reception.
- lateral_rush - Binary indicator for if a lateral occurred on a run.
- lateral_return - Binary indicator for if a lateral occurred on a return.
- lateral_recovery - Binary indicator for if a lateral occurred on a fumble recovery.
- passer_player_id - Unique identifier for the player that attempted the pass.
- passer_player_name - String name for the player that attempted the pass.
- receiver_player_id - Unique identifier for the receiver that was targeted on the pass.
- receiver_player_name - String name for the targeted receiver.
- rusher_player_id - Unique identifier for the player that attempted the run.
- rusher_player_name - String name for the player that attempted the run.
- lateral_receiver_player_id - Unique identifier for the player that received the lateral on a reception.
- lateral_receiver_player_name - String name for the player that received the lateral on a reception.
- lateral_rusher_player_id - Unique identifier for the player that received the lateral on a run.
- lateral_rusher_player_name - String name for the player that received the lateral on a run.
- lateral_sack_player_id - Unique identifier for the player that received the lateral on a sack.
- lateral_sack_player_name - String name for the player that received the lateral on a sack.

- `interception_player_id` - Unique identifier for the player that intercepted the pass.
- `interception_player_name` - String name for the player that intercepted the pass.
- `lateral_interception_player_id` - Unique identifier for the player that received the lateral on an interception.
- `lateral_interception_player_name` - String name for the player that received the lateral on an interception.
- `punt_returner_player_id` - Unique identifier for the punt returner.
- `punt_returner_player_name` - String name for the punt returner.
- `lateral_punt_returner_player_id` - Unique identifier for the player that received the lateral on a punt return.
- `lateral_punt_returner_player_name` - String name for the player that received the lateral on a punt return.
- `kickoff_returner_player_name` - String name for the kickoff returner.
- `kickoff_returner_player_id` - Unique identifier for the kickoff returner.
- `lateral_kickoff_returner_player_id` - Unique identifier for the player that received the lateral on a kickoff return.
- `lateral_kickoff_returner_player_name` - String name for the player that received the lateral on a kickoff return.
- `punter_player_id` - Unique identifier for the punter.
- `punter_player_name` - String name for the punter.
- `kicker_player_name` - String name for the kicker on FG or kickoff.
- `kicker_player_id` - Unique identifier for the kicker on FG or kickoff.
- `own_kickoff_recovery_player_id` - Unique identifier for the player that recovered their own kickoff.
- `own_kickoff_recovery_player_name` - String name for the player that recovered their own kickoff.
- `blocked_player_id` - Unique identifier for the player that blocked the punt or FG.
- `blocked_player_name` - String name for the player that blocked the punt or FG.
- `tackle_for_loss_1_player_id` - Unique identifier for one of the potential players with the tackle for loss.
- `tackle_for_loss_1_player_name` - String name for one of the potential players with the tackle for loss.
- `tackle_for_loss_2_player_id` - Unique identifier for one of the potential players with the tackle for loss.
- `tackle_for_loss_2_player_name` - String name for one of the potential players with the tackle for loss.
- `qb_hit_1_player_id` - Unique identifier for one of the potential players that hit the QB.
- `qb_hit_1_player_name` - String name for one of the potential players that hit the QB.
- `qb_hit_2_player_id` - Unique identifier for one of the potential players that hit the QB.
- `qb_hit_2_player_name` - String name for one of the potential players that hit the QB.
- `forced_fumble_player_1_team` - Team of one of the players with a forced fumble.
- `forced_fumble_player_1_player_id` - Unique identifier of one of the players with a forced fumble.

- forced_fumble_player_1_player_name - String name of one of the players with a forced fumble.
- forced_fumble_player_2_team - Team of one of the players with a forced fumble.
- forced_fumble_player_2_player_id - Unique identifier of one of the players with a forced fumble.
- forced_fumble_player_2_player_name - String name of one of the players with a forced fumble.
- solo_tackle_1_team - Team of one of the players with a solo tackle.
- solo_tackle_2_team - Team of one of the players with a solo tackle.
- solo_tackle_1_player_id - Unique identifier of one of the players with a solo tackle.
- solo_tackle_2_player_id - Unique identifier of one of the players with a solo tackle.
- solo_tackle_1_player_name - String name of one of the players with a solo tackle.
- solo_tackle_2_player_name - String name of one of the players with a solo tackle.
- assist_tackle_1_player_id - Unique identifier of one of the players with a tackle assist.
- assist_tackle_1_player_name - String name of one of the players with a tackle assist.
- assist_tackle_1_team - Team of one of the players with a tackle assist.
- assist_tackle_2_player_id - Unique identifier of one of the players with a tackle assist.
- assist_tackle_2_player_name - String name of one of the players with a tackle assist.
- assist_tackle_2_team - Team of one of the players with a tackle assist.
- assist_tackle_3_player_id - Unique identifier of one of the players with a tackle assist.
- assist_tackle_3_player_name - String name of one of the players with a tackle assist.
- assist_tackle_3_team - Team of one of the players with a tackle assist.
- assist_tackle_4_player_id - Unique identifier of one of the players with a tackle assist.
- assist_tackle_4_player_name - String name of one of the players with a tackle assist.
- assist_tackle_4_team - Team of one of the players with a tackle assist.
- pass_defense_1_player_id - Unique identifier of one of the players with a pass defense.
- pass_defense_1_player_name - String name of one of the players with a pass defense.
- pass_defense_2_player_id - Unique identifier of one of the players with a pass defense.
- pass_defense_2_player_name - String name of one of the players with a pass defense.
- fumbled_1_team - Team of one of the players with a fumble.
- fumbled_1_player_id - Unique identifier of one of the players with a fumble.
- fumbled_1_player_name - String name of one of the players with a fumble.
- fumbled_2_player_id - Unique identifier of one of the players with a fumble.
- fumbled_2_player_name - String name of one of the players with a fumble.
- fumbled_2_team - Team of one of the players with a fumble.
- fumble_recovery_1_team - Team of one of the players with a fumble recovery.
- fumble_recovery_1_yards - Yards gained by one of the players with a fumble recovery.
- fumble_recovery_1_player_id - Unique identifier of one of the players with a fumble recovery.
- fumble_recovery_1_player_name - String name of one of the players with a fumble recovery.
- fumble_recovery_2_team - Team of one of the players with a fumble recovery.
- fumble_recovery_2_yards - Yards gained by one of the players with a fumble recovery.

- fumble_recovery_2_player_id - Unique identifier of one of the players with a fumble recovery.
- fumble_recovery_2_player_name - String name of one of the players with a fumble recovery.
- return_team - String abbreviation of the return team.
- return_yards - Yards gained by the return team.
- penalty_team - String abbreviation of the team with the penalty.
- penalty_player_id - Unique identifier for the player with the penalty.
- penalty_player_name - String name for the player with the penalty.
- penalty_yards - Yards gained (or lost) by the posteam from the penalty.
- replay_or_challenge - Binary indicator for whether or not a replay or challenge.
- replay_or_challenge_result - String indicating the result of the replay or challenge.
- penalty_type - String indicating the penalty type.
- defensive_two_point_attempt - Binary indicator whether or not the defense was able to have an attempt on a two point conversion, this results following a turnover.
- defensive_two_point_conv - Binary indicator whether or not the defense successfully scored on the two point conversion.
- defensive_extra_point_attempt - Binary indicator whether or not the defense was able to have an attempt on an extra point attempt, this results following a blocked attempt that the defense recovers the ball.
- defensive_extra_point_conv - Binary indicator whether or not the defense successfully scored on an extra point attempt.
- season - 4 digit number indicating to which season the game belongs to.
- cp - Numeric value indicating the probability for a complete pass based on comparable game situations.
- cpoe - For a single pass play this is 1 - cp when the pass was completed or 0 - cp when the pass was incomplete. Analyzed for a whole game or season an indicator for the passer how much over or under expectation his completion percentage was.
- series - Starts at 1, each new first down increments, numbers shared across both teams NA: kickoffs, extra point/two point conversion attempts, non-plays, no posteam
- series_success - 1: scored touchdown, gained enough yards for first down 0: punt, interception, fumble lost, turnover on downs, FG attempt NA: series is NA, series contains QB spike/kneel
- start_time - Kickoff time in eastern time zone.
- stadium - Game site name.
- weather - String describing the weather including temperature, humidity and wind (direction and speed). Doesn't change during the game!
- nfl_api_id - UUID of the game in the new NFL API.
- play_clock - Time on the playclock when the ball was snapped.
- play_deleted - Binary indicator for deleted plays.
- play_type_nfl - Play type as listed in the NFL source. Slightly different to the regular play_type variable.
- end_clock_time - Game time at the end of a given play.
- end_yard_line - String indicating the yardline at the end of the given play consisting of team half and yard line number.

- `drive_real_start_time` - Local day time when the drive started (currently not used by the NFL and therefore mostly 'NA').
- `drive_play_count` - Numeric value of how many regular plays happened in a given drive.
- `drive_time_of_possession` - Time of possession in a given drive.
- `drive_first_downs` - Number of first downs in a given drive.
- `drive_inside20` - Binary indicator if the offense was able to get inside the opponents 20 yard line.
- `drive_ended_with_score` - Binary indicator the drive ended with a score.
- `drive_quarter_start` - Numeric value indicating in which quarter the given drive has started.
- `drive_quarter_end` - Numeric value indicating in which quarter the given drive has ended.
- `drive_yards_penalized` - Numeric value of how many yards the offense gained or lost through penalties in the given drive.
- `drive_start_transition` - String indicating how the offense got the ball.
- `drive_end_transition` - String indicating how the offense lost the ball.
- `drive_game_clock_start` - Game time at the beginning of a given drive.
- `drive_game_clock_end` - Game time at the end of a given drive.
- `drive_start_yard_line` - String indicating where a given drive started consisting of team half and yard line number.
- `drive_end_yard_line` - String indicating where a given drive ended consisting of team half and yard line number.
- `drive_play_id_started` - `Play_id` of the first play in the given drive.
- `drive_play_id_ended` - `Play_id` of the last play in the given drive.
- `away_score` - Total points scored by the away team.
- `home_score` - Total points scored by the home team.
- `location` - Either 'Home' or 'Neutral' indicating if the home team played at home or at a neutral site.
- `result` - Equals `home_score - away_score` and means the game outcome from the perspective of the home team.
- `total` - Equals `home_score + away_score` and means the total points scored in the given game.
- `spread_line` - The closing spread line for the game. A positive number means the home team was favored by that many points, a negative number means the away team was favored by that many points. (Source: Pro-Football-Reference)
- `total_line` - The closing total line for the game. (Source: Pro-Football-Reference)
- `div_game` - Binary indicator for if the given game was a division game.
- `roof` - One of 'dome', 'outdoors', 'closed', 'open' indicating the roof status of the stadium the game was played in. (Source: Pro-Football-Reference)
- `surface` - What type of ground the game was played on. (Source: Pro-Football-Reference)
- `temp` - The temperature at the stadium only for 'roof' = 'outdoors' or 'open'. (Source: Pro-Football-Reference)
- `wind` - The speed of the wind in miles/hour only for 'roof' = 'outdoors' or 'open'. (Source: Pro-Football-Reference)
- `home_coach` - First and last name of the home team coach. (Source: Pro-Football-Reference)
- `away_coach` - First and last name of the away team coach. (Source: Pro-Football-Reference)
- `stadium_id` - ID of the stadium the game was played in. (Source: Pro-Football-Reference)
- `game_stadium` - Name of the stadium the game was played in. (Source: Pro-Football-Reference)

Examples

```
## Not run:
# Get pbp data for two games using parallel processing
game_ids <- c("2019_01_GB_CHI", "2013_21_SEA_DEN")
pbp <- fast_scraper(game_ids, pp = TRUE)

## End(Not run)
```

fast_scraper_schedules

Get NFL Season Schedules

Description

Get NFL Season Schedules

Usage

```
fast_scraper_schedules(seasons, pp = FALSE)
```

Arguments

seasons	Vector of numeric or character 4 digit seasons
pp	Logical - either TRUE or FALSE (see details for further information)

Details

This functions now incorporates the games file provided and maintained by Lee Sharpe.

The pp parameter controls if the scraper should use parallel processing. Please note that the initiating process takes a few seconds which means it may be better to set pp = FALSE if you are scraping less than 10 seasons.

Value

Data frame containing the following detailed game information:

- game_id - Character identifier including season, week, away team and home team
- season - 4 digit season year.
- game_type - One of 'REG', 'WC', 'DIV', 'CON', 'SB' indicating if a game was a regular season game or one of the playoff rounds.
- week - Numeric week number.
- gameday - Game date in format yyyy/mm/dd.
- weekday - The day of the week on which the game occurred.
- gametime - The kickoff time of the game. This is represented in 24-hour time and the Eastern time zone, regardless of what time zone the game was being played in.
- away_team - Away team abbreviation.
- home_team - Home team abbreviation.
- away_score - The number of points the away team scored. Is 'NA' for games which haven't yet been played.

- **home_score** - The number of points the home team scored. Is 'NA' for games which haven't yet been played.
- **home_result** - Equals **home_score** - **away_score** and means the game outcome from the perspective of the home team.
- **stadium** - Name of the stadium the game was or will be played in. (Source: Pro-Football-Reference)
- **location** - Either 'Home' or 'Neutral' indicating if the home team played at home or at a neutral site.
- **roof** - One of 'dome', 'outdoors', 'closed', 'open' indicating the roof status of the stadium the game was played in. (Source: Pro-Football-Reference)
- **surface** - What type of ground the game was played on. (Source: Pro-Football-Reference)
- **old_game_id** - Unique game identifier of the old NFL API.

Examples

```
## Not run:
# Get schedules for the whole 2015 - 2018 seasons
seasons <- 2015:2018
schedules <- fast_scraper_schedules(seasons)

## End(Not run)
```

teams_colors_logos	<i>NFL Team names, colors and logo urls.</i>
--------------------	--

Description

NFL Team names, colors and logo urls.

Usage

```
teams_colors_logos
```

Format

A data frame with 36 rows and 10 variables containing NFL team level information, including franchises in multiple cities:

team_abbr Team abbreviation
team_name Complete Team name
team_id Team id used in the roster function
team_nick Nickname
team_color Primary color
team_color2 Secondary color
team_color3 Tertiary color
team_color4 Quaternary color
team_logo_wikipedia Url to Team logo on wikipedia
team_logo_espn Url to higher quality logo on espn

The colors are taken from Lee Sharpe's `teamcolors.csv` who has taken them from the 'teamcolors' package created by Ben Baumer and Gregory Matthews. The Wikipedia logo urls are taken from Lee Sharpe's `logos.csv`

Index

* **datasets**

teams_colors_logos, [15](#)

add_qb_epa, [2](#)

calculate_expected_points, [2](#)

clean_pbp, [3](#)

fast_scraper, [2](#), [3](#), [4](#)

fast_scraper_schedules, [4](#), [14](#)

teams_colors_logos, [15](#)