

Package: puntr (via r-universe)

September 19, 2026

Type Package
Title Analysis of Punting
License MIT
Version 1.4
Description A package for importing, manipulating, analyzing, and visualizing data related to football punting. Uses play-by-play dataframes in the format used by nflscrapR and nflfastR.
URL <https://puntalytics.github.io/puntr>,
<https://github.com/Puntalytics/puntr>
Encoding UTF-8
LazyData true
Imports dplyr, forcats, ggimage, ggplot2, ggrepel, glue, magrittr, nflfastR, purrr, readr, stringr, tibble, tictoc
Remotes sportsdataverse/cfbfastR
RoxygenNote 7.3.2
Suggests testthat (>= 3.0.0)
Config/testthat/edition 3
Config/pak/sysreqs libcairo2-dev cmake libfontconfig1-dev libfreetype6-dev make libmagick++-dev gsfonts libicu-dev libpng-dev libuv1-dev libssl-dev libx11-dev
Repository <https://sportsdataverse.r-universe.dev>
Date/Publication 2024-09-05 15:05:11 UTC
RemoteUrl <https://github.com/Puntalytics/puntr>
RemoteRef HEAD
RemoteSha cf7b78652d6fd91bf4c3787c43b3d9b3209288b8

Contents

by_punter_games	2
by_punter_seasons	3

by_punters	3
calculate_all	4
college_to_pro	5
create_mini	5
create_miniG	6
create_miniY	7
import_college_punts	7
import_punts	8
import_seasons	9
punt_trim	9
trust_the_process	10
Index	11

by_punter_games	<i>Summarize data for player-games</i>
-----------------	--

Description

This function is essentially a convenient wrapper for `dplyr::summarise` which includes all of the relevant columns from a puntr-style data frame. This function differs from `puntr::create_mini` and `puntr::create_miniY` in that it groups by `punter_player_name`, `season`, and `week` (and adds a convenient `weekid` column to uniquely identify each row). It is unfortunately not customizable beyond the setting of a minimum number of punts to be included; if you have additional parameters you'd like to be summarised, the easiest thing would be to call `summarise` yourself.

Usage

```
by_punter_games(punts, ..., threshold = 1)
```

Arguments

- `punts` The play-by-play punting data to be summarized
- `threshold` The minimum number of punts for a week to be included, defaults to 1

Value

A tibble `miniG` where each row is a punter-week and each column is a stat

Examples

```
## Not run:  
create_miniG(punts)  
  
## End(Not run)
```

by_punter_seasons	<i>Summarize data for player-seasons</i>
-------------------	--

Description

This function is essentially a convenient wrapper for `dplyr::summarise` which includes all of the relevant columns from a puntr-style data frame. This function differs from `puntr::create_mini` in that it groups by both `punter_player_name` and `season` (and adds a convenient `seasonid` column to uniquely identify each row). It is unfortunately not customizable beyond the setting of a minimum number of punts to be included; if you have additional parameters you'd like to be summarised, the easiest thing would be to call `summarise` yourself.

Usage

```
by_punter_seasons(punts, ..., threshold = 32)
```

Arguments

<code>punts</code>	The play-by-play punting data to be summarized
<code>threshold</code>	The minimum number of punts for a season to be included, defaults to 32

Value

A tibble `miniY` where each row is a punter-season and each column is a stat

Examples

```
## Not run:
create_miniY(punts)

## End(Not run)
```

by_punters	<i>Summarize data for player careers</i>
------------	--

Description

This function is essentially a convenient wrapper for `dplyr::summarise` which provides a dataframe with the following columns: `NumPunts`, `pEPA`, `Gross`, `Net`, `RERUN`, `SHARP_RERUN_OF`, `SHARP_RERUN_PD`, `first_year`, `last_year`, `team_logo_espn`, `team_color`, `team_color2`. Additional columns may be added via valid calls to `dplyr::summarise` as additional arguments. For comparison of punter seasons, see `puntr::by_punter_seasons`; for comparison of punter games, see `puntr::by_punter_games`.

Usage

```
by_punters(punts, ..., threshold = 64)
```

Arguments

punts	The play-by-play punting data to be summarized
...	Any additional arguments will be passed through to <code>dplyr::summarise</code>
threshold	The minimum number of career punts needed to be included, defaults to 64

Value

A tibble punters where each row is a punter and each column is a stat

Examples

```
## Not run:
create_mini(punts)

## End(Not run)
```

calculate_all	<i>Calculate metrics</i>
---------------	--------------------------

Description

This function adds all Puntalytics metrics to your punting data: RERUN, SHARP, SHARPnet and SHARP_RERUN (plus open-field and pin-deep versions of the 3 SHARPs) as well as average and cumulative EPA (our custom EPA metric.)

Usage

```
calculate_all(punts)
```

Arguments

punts	The already-processed play-by-play punting data to which metrics will be added. Must contain at least three seasons and 1000 punts.
-------	---

Value

A tibble punts with the calculated metrics added

Examples

```
## Not run:
calculate_all(punts)

## End(Not run)
```

college_to_pro	<i>Convert college data to puntr format</i>
----------------	---

Description

Rename columns and process data such that the output can be plugged directly into `puntr::calculate_all`, and the output of that can be plugged directly into `puntr::create_mini` (or `puntr::create_miniY`).

Usage

```
college_to_pro(punts, power_five = TRUE)
```

Arguments

punts	A data frame containing punts in the cfbfastR format
power_five	Logical, defaults to TRUE to include only punters from Power 5 teams

Value

A tibble punts in a format usable for `puntr::calculate_all`

Examples

```
## Not run:
college_to_pro(punts)

## End(Not run)
```

create_mini	<i>Summarize data for players</i>
-------------	-----------------------------------

Description

This function is essentially a convenient wrapper for `dplyr::summarise` which includes all of the relevant columns from a puntr-style data frame. NOTE: `puntr::create_mini`, `puntr::create_miniY`, and `puntr::create_miniG` are being phased out in favor of `puntr::by_punters`, `puntr::by_punter_seasons`, and `puntr::by_punter_games`

Usage

```
create_mini(punts, ..., threshold = 64)
```

Arguments

punts	The play-by-play punting data to be summarized
...	Any additional arguments will be passed through to <code>dplyr::summarise</code>
threshold	The minimum number of career punts needed to be included, defaults to 64

Value

A tibble mini where each row is a punter and each column is a stat

Examples

```
## Not run:
create_mini(punts)

## End(Not run)
```

create_miniG

Summarize data for player-games

Description

This function is essentially a convenient wrapper for `dplyr::summarise` which includes all of the relevant columns from a puntr-style data frame. This function differs from `puntr::create_mini` and `puntr::create_miniY` in that it groups by `punter_player_name`, `season`, and `week` (and adds a convenient `weekid` column to uniquely identify each row). NOTE: `puntr::create_mini`, `puntr::create_miniY`, and `puntr::create_miniG` are being phased out in favor of `puntr::by_punters`, `puntr::by_punter_seasons`, and `puntr::by_punter_games`

Usage

```
create_miniG(punts, ..., threshold = 1)
```

Arguments

<code>punts</code>	The play-by-play punting data to be summarized
<code>...</code>	Any additional arguments will be passed through to <code>dplyr::summarise</code>
<code>threshold</code>	The minimum number of punts for a week to be included, defaults to 1

Value

A tibble miniG where each row is a punter-week and each column is a stat

Examples

```
## Not run:
create_miniG(punts)

## End(Not run)
```

create_miniY	<i>Summarize data for player-seasons</i>
--------------	--

Description

This function is essentially a convenient wrapper for `dplyr::summarise` which includes all of the relevant columns from a puntr-style data frame. This function differs from `puntr::create_mini` in that it groups by both `punter_player_name` and `season` (and adds a convenient `seasonid` column to uniquely identify each row). NOTE: `puntr::create_mini`, `puntr::create_miniY`, and `puntr::create_miniG` are being phased out in favor of `puntr::by_punters`, `puntr::by_punter_seasons`, and `puntr::by_punter_games`

Usage

```
create_miniY(punts, ..., threshold = 32)
```

Arguments

<code>punts</code>	The play-by-play punting data to be summarized
<code>...</code>	Any additional arguments will be passed through to <code>dplyr::summarise</code>
<code>threshold</code>	The minimum number of punts for a season to be included, defaults to 32

Value

A tibble `miniY` where each row is a punter-season and each column is a stat

Examples

```
## Not run:
create_miniY(punts)

## End(Not run)
```

import_college_punts	<i>Import college punting data</i>
----------------------	------------------------------------

Description

Import college punting data for seasons in the scope of `cfbfastR` (back to 2014). This function is a wrapper around `cfbfastR::load_cfb_pbp`.

Usage

```
import_college_punts(years)
```

Arguments

years A year or range of years to be scraped

Value

A tibble punts of punts in the cfbfastR format

Examples

```
## Not run:
import_college_punts(2018:2021)

## End(Not run)
```

import_punts	<i>Import punting data</i>
--------------	----------------------------

Description

Easily import punting data for completed seasons. This data has been pre-scraped and compiled [here](#). The only cleaning performed on this data before storing is `filter(play_type == 'punt')`.

Usage

```
import_punts(years, local = FALSE, path = NULL)
```

Arguments

years A year or range of years between 1999 and 2020, inclusive

local Defaults to 'FALSE'. If 'TRUE', must also specify the local path to the 'puntr-data' folder. Importing a local copy of the data is much faster (~3 seconds, vs ~15 minutes)

path The local path to the 'puntr-data' folder. Required when 'local = TRUE'. Omit the trailing '/', e.g. '/Users/yourname/footballfolder'

Value

A tibble punts containing play-by-play punting data for the specified years

Examples

```
## Not run:
import_punts(1999:2020)
import_punts(1999:2020, local=TRUE, path='/my/local/path')

## End(Not run)
```

import_seasons	<i>Import play-by-play data</i>
----------------	---------------------------------

Description

Grab all play-by-play data, not just punts. This function pulls data directly from the nflfastR-data repo, and is purely a wrapper for ease of use.

Usage

```
import_seasons(years)
```

Arguments

years	A year or range of years between 1999 and 2020, inclusive
-------	---

Value

A tibble pbp containing play-by-play data for the specified years

Examples

```
## Not run:
import_seasons(1999:2019)

## End(Not run)
```

punt_trim	<i>Filter columns relevant to punting</i>
-----------	---

Description

This function is exported in case you would like include custom columns; if you would NOT, you can just use [puntr::trust_the_process]. If you do use puntr::punt_trim yourself, be sure to include the trim=FALSE flag in puntr::trust_the_process.

Usage

```
punt_trim(punts, columns = NULL)
```

Arguments

punts	The play-by-play punting data
columns	Defaults to NULL indicating trimming down to the default columns. If columns is a list of other columns, those will be additionally included

Value

A tibble punts containing only the specified columns

Examples

```
## Not run:
punt_trim(punts)
punt_trim(punts, columns=c("solo_tackle", "assist_tackle"))

## End(Not run)
```

trust_the_process	<i>Clean and process play-by-play punting data</i>
-------------------	--

Description

This function handles all of the standard data cleaning necessary/preferable for puntalytics.

Usage

```
trust_the_process(punts, seastype = "REG", trim = TRUE)
```

Arguments

punts	The play-by-play punting data to be cleaned and processed
seastype	One of "REG" (default) or "POST" to filter data, or NULL to include all data
trim	Specify trim=FALSE if you would like to use puntr::punt_trim to create custom columns (or not use puntr::punt_trim at all and include all columns)

Value

A tibble punts with cleaned and processed play-by-play punting data

Examples

```
## Not run:
trust_the_process(punts)
trust_the_process(punts_custom_trimmed_already, trim=FALSE)

## End(Not run)
```

Index

by_punter_games, [2](#)
by_punter_seasons, [3](#)
by_punters, [3](#)

calculate_all, [4](#)
college_to_pro, [5](#)
create_mini, [5](#)
create_miniG, [6](#)
create_miniY, [7](#)

import_college_punts, [7](#)
import_punts, [8](#)
import_seasons, [9](#)

punt_trim, [9](#)

trust_the_process, [10](#)