

Package: futebolplotR (via r-universe)

June 3, 2026

Title Brazilian Soccer Badges Plots in 'ggplot2'

Version 0.0.1

Description A set of functions to visualize Brazilian Soccer analysis in 'ggplot2'.

License MIT + file LICENSE

URL <https://github.com/brunomioto/futebolplotR>,
<https://brunomioto.github.io/futebolplotR/>

BugReports <https://github.com/brunomioto/futebolplotR/issues>

Depends R (>= 3.6)

Imports brasileiro, cli (>= 3.0.0), ggpath (>= 1.0.0), ggplot2 (>= 3.3.0), glue, grid, lifecycle, rlang (>= 0.4.11), scales (>= 1.1.0), withr

Suggests dplyr (>= 1.0.0), rsvg (>= 2.0), testthat (>= 3.0.0), vdiff

Remotes williamorim/brasileirao

Config/Needs/website rmarkdown

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Config/pak/sysreqs libmagick+-dev gsfontr libicu-dev libssl-dev

Repository <https://brunomioto.r-universe.dev>

Date/Publication 2024-01-19 12:37:59 UTC

RemoteUrl <https://github.com/brunomioto/futebolplotR>

RemoteRef HEAD

RemoteSha 6af1e039f8559224601c36590c3f4d8bf788104d

Contents

clean_team_abbrs	2
element	2
geom_futebol_badges	4
scale_futebol	7
valid_team_names	9

Index	11
--------------	-----------

clean_team_abbrs	<i>Standardize Brazilian Soccer Team Abbreviations</i>
------------------	--------------------------------------------------------

Description

This function standardizes Brazilian Soccer team abbreviations to brasileiro defaults.

Usage

```
clean_team_abbrs(abbr, keep_non_matches = TRUE)
```

Arguments

abbr	a character vector of abbreviations
keep_non_matches	If TRUE (the default) an element of abbr that can't be matched to any of the internal mapping vectors will be kept as is. Otherwise it will be replaced with NA.

Value

A character vector with the length of abbr and cleaned team abbreviations if they are included in [team_abbr_mapping](#). Non matches may be replaced with NA (depending on the value of keep_non_matches).

element	<i>Theme Elements for Image Grobs</i>
---------	---------------------------------------

Description

In conjunction with the [ggplot2::theme](#) system, the following element_ functions enable images in non-data components of the plot, e.g. axis text.

- `element_futebol_badge()`: draws Brazilian Soccer team badges instead of their abbreviations.

Usage

```
element_futebol_badge(
  alpha = NULL,
  colour = NA,
  hjust = NULL,
  vjust = NULL,
  color = NULL,
  size = 0.5
)
```

Arguments

alpha	The alpha channel, i.e. transparency level, as a numerical value between 0 and 1.
colour, color	The image will be colorized with this color. Use the special character "b/w" to set it to black and white. For more information on valid color names in ggplot2 see https://ggplot2.tidyverse.org/articles/ggplot2-specs.html?q=colour#colour-and-fill .
hjust, vjust	The horizontal and vertical adjustment respectively. Must be a numerical value between 0 and 1.
size	The output grob size in cm (!).

Details

The elements translate Brazilian Soccer team abbreviations into badge images.

Value

An S3 object of class element.

See Also

[geom_futebol_badges\(\)](#), and [ggpath::element_path\(\)](#) for more information on valid team abbreviations, and other parameters.

Examples

```
library(futebolplotR)
library(ggplot2)

team_abbr <- valid_team_names()

df <- data.frame(
  random_value = runif(length(team_abbr), 0, 1),
  team = team_abbr
)

# use badges for x-axis
ggplot(df, aes(x = team, y = random_value)) +
```

```

geom_col(aes(color = team, fill = team), width = 0.5) +
scale_color_futebol(type = "secondary") +
scale_fill_futebol(alpha = 0.4) +
theme_minimal() +
theme(axis.text.x = element_futebol_badge())

# use badges for y-axis
ggplot(df, aes(y = team, x = random_value)) +
geom_col(aes(color = team, fill = team), width = 0.5) +
scale_color_futebol(type = "secondary") +
scale_fill_futebol(alpha = 0.4) +
theme_minimal() +
theme(axis.text.y = element_futebol_badge())

```

geom_futebol_badges *ggplot2 Layer for Visualizing Brazilian Soccer Team badges*

Description

This geom is used to plot Brazilian soccer team badges instead of points in a ggplot. It requires x, y aesthetics as well as a valid Brazilian soccer team abbreviation. The latter can be checked with [valid_team_names\(\)](#).

Usage

```

geom_futebol_badges(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  ...,
  na.rm = FALSE,
  show.legend = FALSE,
  inherit.aes = TRUE
)

```

Arguments

mapping	Set of aesthetic mappings created by aes() . If specified and inherit.aes = TRUE (the default), it is combined with the default mapping at the top level of the plot. You must supply mapping if there is no plot mapping.
data	The data to be displayed in this layer. There are three options: If NULL, the default, the data is inherited from the plot data as specified in the call to ggplot() .

A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. See `fortify()` for which variables will be created.

A function will be called with a single argument, the plot data. The return value must be a data.frame, and will be used as the layer data. A function can be created from a formula (e.g. `~ head(.x, 10)`).

<code>stat</code>	The statistical transformation to use on the data for this layer, either as a ggproto Geom subclass or as a string naming the stat stripped of the <code>stat_</code> prefix (e.g. "count" rather than "stat_count")
<code>position</code>	Position adjustment, either as a string naming the adjustment (e.g. "jitter" to use <code>position_jitter</code>), or the result of a call to a position adjustment function. Use the latter if you need to change the settings of the adjustment.
<code>...</code>	Other arguments passed on to <code>ggplot2::layer()</code> . These are often aesthetics, used to set an aesthetic to a fixed value. See the below section "Aesthetics" for a full list of possible arguments.
<code>na.rm</code>	If FALSE, the default, missing values are removed with a warning. If TRUE, missing values are silently removed.
<code>show.legend</code>	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display.
<code>inherit.aes</code>	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. <code>borders()</code> .

Value

A ggplot2 layer (`ggplot2::layer()`) that can be added to a plot created with `ggplot2::ggplot()`.

Aesthetics

`geom_futebol_badges()` understands the following aesthetics (required aesthetics are in bold):

- **x** - The x-coordinate.
- **y** - The y-coordinate.
- **team_abbrev** - The team abbreviation. Should be one of `valid_team_names()`. The function tries to clean team names internally by calling `clean_team_abbrevs()`.
- `alpha = NULL` - The alpha channel, i.e. transparency level, as a numerical value between 0 and 1.
- `colour = NULL` - The image will be colored with this colour. Use the special character "b/w" to set it to black and white. For more information on valid colour names in ggplot2 see <https://ggplot2.tidyverse.org/articles/ggplot2-specs.html?q=colour#colour-and-fill>
- `angle = 0` - The angle of the image as a numerical value between 0° and 360°.
- `hjust = 0.5` - The horizontal adjustment relative to the given x coordinate. Must be a numerical value between 0 and 1.

- `vjust = 0.5` - The vertical adjustment relative to the given y coordinate. Must be a numerical value between 0 and 1.
- `width = 1.0` - The desired width of the image in npc (Normalised Parent Coordinates). The default value is set to 1.0 which is *big* but it is necessary because all used values are computed relative to the default. A typical size is `width = 0.075` (see below examples).
- `height = 1.0` - The desired height of the image in npc (Normalised Parent Coordinates). The default value is set to 1.0 which is *big* but it is necessary because all used values are computed relative to the default. A typical size is `height = 0.1` (see below examples).

Examples

```
library(futebolplotR)
library(ggplot2)

team_abbr <- futebolplotR::valid_team_names()

df <- data.frame(
  a = rep(1:5, 2),
  b = sort(rep(1:2, 5), decreasing = TRUE),
  teams = team_abbr[1:10]
)

# keep alpha == 1 for all teams including an "A"
matches <- grepl("A", team_abbr[1:10])
df$alpha <- ifelse(matches, 1, 0.2)
# also set a custom fill colour for the non "A" teams
df$colour <- ifelse(matches, NA, "gray")

# scatterplot of all badges
ggplot(df, aes(x = a, y = b)) +
  geom_futebol_badges(aes(team_abbr = teams), width = 0.075) +
  geom_label(aes(label = teams), nudge_y = -0.35, alpha = 0.5) +
  theme_void()

# apply alpha via an aesthetic from inside the dataset `df`
# please note that you have to add scale_alpha_identity() to use the alpha
# values in your dataset!
ggplot(df, aes(x = a, y = b)) +
  geom_futebol_badges(aes(team_abbr = teams, alpha = alpha), width = 0.075) +
  geom_label(aes(label = teams), nudge_y = -0.35, alpha = 0.5) +
  scale_alpha_identity() +
  theme_void()

# apply alpha and colour via an aesthetic from inside the dataset `df`
# please note that you have to add scale_alpha_identity() as well as
# scale_color_identity() to use the alpha and colour values in your dataset!
ggplot(df, aes(x = a, y = b)) +
  geom_futebol_badges(aes(team_abbr = teams, alpha = alpha, colour = colour), width = 0.075) +
  geom_label(aes(label = teams), nudge_y = -0.35, alpha = 0.5) +
  scale_alpha_identity() +
  scale_color_identity() +
  theme_void()
```

```
# apply alpha as constant for all badges
ggplot(df, aes(x = a, y = b)) +
  geom_futebol_badges(aes(team_abbr = teams), width = 0.075, alpha = 0.6) +
  geom_label(aes(label = teams), nudge_y = -0.35, alpha = 0.5) +
  theme_void()
```

scale_futebol

Scales for Brazilian Soccer Team Colors

Description

These functions map soccer team names to their team colors in color and fill aesthetics

Usage

```
scale_color_futebol(
  type = c("primary", "secondary"),
  values = NULL,
  ...,
  aesthetics = "colour",
  breaks = ggplot2::waiver(),
  na.value = "grey50",
  guide = NULL,
  alpha = NA
)
```

```
scale_colour_futebol(
  type = c("primary", "secondary"),
  values = NULL,
  ...,
  aesthetics = "colour",
  breaks = ggplot2::waiver(),
  na.value = "grey50",
  guide = NULL,
  alpha = NA
)
```

```
scale_fill_futebol(
  type = c("primary", "secondary"),
  values = NULL,
  ...,
  aesthetics = "fill",
  breaks = ggplot2::waiver(),
  na.value = "grey50",
  guide = NULL,
)
```

```

    alpha = NA
  )

```

Arguments

type	One of "primary" or "secondary" to decide which color type to use.
values	If NULL (the default) use the internal team color vectors. Otherwise a set of aesthetic values to map data values to. The values will be matched in order (usually alphabetical) with the limits of the scale, or with breaks if provided. If this is a named vector, then the values will be matched based on the names instead. Data values that don't match will be given <code>na.value</code> .
...	Arguments passed on to discrete_scale
palette	A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., <code>scales::hue_pal()</code>).
limits	One of: <ul style="list-style-type: none"> • NULL to use the default scale values • A character vector that defines possible values of the scale and their order • A function that accepts the existing (automatic) values and returns new ones. Also accepts rlang lambda function notation.
drop	Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.
na.translate	Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify <code>na.translate = FALSE</code> .
scale_name	The name of the scale that should be used for error messages associated with this scale.
name	The name of the scale. Used as the axis or legend title. If <code>waiver()</code> , the default, the name of the scale is taken from the first mapping used for that aesthetic. If NULL, the legend title will be omitted.
labels	One of: <ul style="list-style-type: none"> • NULL for no labels • <code>waiver()</code> for the default labels computed by the transformation object • A character vector giving labels (must be same length as breaks) • An expression vector (must be the same length as breaks). See <code>?plot-math</code> for details. • A function that takes the breaks as input and returns labels as output. Also accepts rlang lambda function notation.
guide	A function used to create a guide or its name. See guides() for more information.
super	The super class to use for the constructed scale
aesthetics	Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the <code>colour</code> and <code>fill</code> aesthetics at the same time, via <code>aesthetics = c("colour", "fill")</code> .

breaks	One of: <ul style="list-style-type: none"> • NULL for no breaks • <code>waiver()</code> for the default breaks (the scale limits) • A character vector of breaks • A function that takes the limits as input and returns breaks as output
na.value	The aesthetic value to use for missing (NA) values
guide	A function used to create a guide or its name. If NULL (the default) no guide will be plotted for this scale. See ggplot2::guides() for more information.
alpha	Factor to modify color transparency via a call to scales::alpha() . If NA (the default) no transparency will be applied. Can also be a vector of alphas. All alpha levels must be in range $[0, 1]$.

See Also

The theme elements [element_futebol_badge\(\)](#) to replace axis text labels with badges.

Examples

```
library(futebolplotR)
library(ggplot2)

team_abbr <- valid_team_names()

df <- data.frame(
  random_value = runif(length(team_abbr), 0, 1),
  teams = team_abbr
)
ggplot(df, aes(x = teams, y = random_value)) +
  geom_col(aes(color = teams, fill = teams), width = 0.5) +
  scale_color_futebol(type = "secondary") +
  scale_fill_futebol(alpha = 0.4) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

 valid_team_names

Output Valid Brazilian Soccer Team Abbreviations

Description

Output Valid Brazilian Soccer Team Abbreviations

Usage

```
valid_team_names()
```

10

valid_team_names

Value

A vector of type "character".

Examples

```
# List valid Brazilian soccer team abbreviations  
valid_team_names()
```

Index

`aes()`, 4

`borders()`, 5

`clean_team_abbrs`, 2
`clean_team_abbrs()`, 5

`discrete_scale`, 8

`element`, 2
`element_futebol_badge (element)`, 2
`element_futebol_badge()`, 9

`fortify()`, 5

`geom_futebol_badges`, 4
`geom_futebol_badges()`, 3
`ggpath::element_path()`, 3
`ggplot()`, 4
`ggplot2::ggplot()`, 5
`ggplot2::guides()`, 9
`ggplot2::layer()`, 5
`ggplot2::theme`, 2
`guides()`, 8

`lambda`, 8

`scale_color_futebol (scale_futebol)`, 7
`scale_colour_futebol (scale_futebol)`, 7
`scale_fill_futebol (scale_futebol)`, 7
`scale_futebol`, 7
`scales::alpha()`, 9
`scales::hue_pal()`, 8

`team_abbr_mapping`, 2

`valid_team_names`, 9
`valid_team_names()`, 4, 5