

# Package ‘spartyR’

September 24, 2022

**Title** Plot Scaled 'ggplot' Representations of Sports Playing Surfaces

**Version** 2.0.1

**Maintainer** Ross Drucker <ross.a.drucker@gmail.com>

**Description** Create scaled 'ggplot' representations of playing surfaces.  
Playing surfaces are drawn pursuant to rule-book specifications.  
This package should be used as a baseline plot for displaying any type of tracking data.

**License** GPL (>= 3)

**Encoding** UTF-8

**RoxygenNote** 7.2.1

**Imports** ggfittext, ggplot2, glue, grid

**Depends** R (>= 3.3)

**Suggests** data.table, gganimate, testthat (>= 3.0.0), roxygen2, knitr,  
rmarkdown, curl, gifski

**Config/testthat/edition** 3

**URL** <https://github.com/sportsdataverse/spartyR>

**BugReports** <https://github.com/sportsdataverse/spartyR/issues>

**VignetteBuilder** knitr

**SystemRequirements** pandoc (>= 1.12.3), pandoc-citeproc

**NeedsCompilation** no

**Author** Ross Drucker [aut, cre] (<<https://orcid.org/0000-0002-0688-0235>>)

**Repository** CRAN

**Date/Publication** 2022-09-24 18:50:02 UTC

## R topics documented:

cani_color_league_features . . . . .	2
cani_plot_league . . . . .	3
cani_plot_sport . . . . .	3

convert_units . . . . .	4
geom_baseball . . . . .	4
geom_basketball . . . . .	6
geom_curling . . . . .	8
geom_football . . . . .	10
geom_hockey . . . . .	12
geom_soccer . . . . .	14
geom_tennis . . . . .	15

<b>Index</b>	<b>18</b>
--------------	-----------

---

cani\_color\_league\_features

*Check to see what features of a surface can be colored*

---

## Description

Check to see what features of a surface can be colored

## Usage

```
cani_color_league_features(league_code, sport_name = NULL)
```

## Arguments

league_code	The case-insensitive league code to be plotted
sport_name	The name of a sport to use in the event that the league_code supplied has more than one sport associated with it. Default: NULL

## Value

Nothing, but a message is sent to the console

## Examples

```
cani_color_league_features("NCAA", "basketball")
```

---

cani_plot_league	<i>Check to see if a league can be plotted, and alert as to which function(s) that league will work for</i>
------------------	---

---

**Description**

Check to see if a league can be plotted, and alert as to which function(s) that league will work for

**Usage**

```
cani_plot_league(league_code)
```

**Arguments**

league\_code      The case-insensitive league code to be plotted

**Value**

Nothing, but a message is sent to the console

**Examples**

```
cani_plot_league("MLB")
```

---

cani_plot_sport	<i>Check to see if a sport can be plotted, and alert as to which league(s) are plottable for the sport</i>
-----------------	--

---

**Description**

Check to see if a sport can be plotted, and alert as to which league(s) are plottable for the sport

**Usage**

```
cani_plot_sport(sport_code)
```

**Arguments**

sport\_code      The case-insensitive sport name

**Value**

Nothing, but a message is sent to the console

**Examples**

```
cani_plot_sport("basketball")
```

---

convert_units	<i>Convert all units, regardless of starting and ending units</i>
---------------	---

---

**Description**

Convert all units, regardless of starting and ending units

**Usage**

```
convert_units(meas, from_unit, to_unit, conversion_columns = NULL)
```

**Arguments**

meas	A measurement in any unit of length
from_unit	A string containing the original unit of measure to be converted
to_unit	A string containing the ending unit of measure
conversion_columns	A vector containing the columns to convert if meas is of type data.frame

**Value**

The measurement in converted units

**Examples**

```
convert_units(1, "in", "cm")
convert_units(100, "cm", "m")
```

---

geom_baseball	<i>Generate a ggplot2 instance containing a baseball field for a specified league</i>
---------------	---

---

**Description**

Generate a ggplot2 instance containing a baseball field for a specified league

**Usage**

```
geom_baseball(
  league,
  display_range = "full",
  field_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
```

```

    y_trans = 0,
    field_units = NULL,
    xlims = NULL,
    ylims = NULL
  )

```

### Arguments

league	The league for which to draw the surface. This is case-insensitive
display_range	A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function. The possible display ranges are: "full" The full field. This is the default "infield" The infield on the baseball field
field_updates	A list of updates to the field's parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the field's default colors, which are set by <a href="#">baseball_features_set_colors()</a>
rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the back tip of home plate towards the left-handed batter's box (the first base side of the field)
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the back tip of home plate towards straight-away center field
field_units	The units with which to draw the field. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the display_range parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the display_range parameter

### Value

A ggplot2 instance with a full-surface representation of a baseball field

### Examples

```

## Not run:
  geom_baseball(league = "MLB", rotation = 270, display_range = "infield")
  geom_baseball(league = "little league", field_units = "m")

## End(Not run)

```

---

geom_basketball	<i>Generate a ggplot2 instance containing a basketball court for a specified league</i>
-----------------	---

---

### Description

Generate a ggplot2 instance containing a basketball court for a specified league

### Usage

```
geom_basketball(
  league,
  display_range = "full",
  court_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
  y_trans = 0,
  court_units = NULL,
  xlims = NULL,
  ylims = NULL
)
```

### Arguments

league	The league for which to draw the surface. This is case-insensitive
display_range	<p>A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function.</p> <p>The possible display ranges are:</p> <ul style="list-style-type: none"> <li>"full" The full court. This is the default</li> <li>"offense" The TV-right half of the court half-court. This is considered the offensive half of the court</li> <li>"offence" The TV-right half of the court half-court. This is considered the offensive half of the court</li> <li>"offensivehalfcourt" The TV-right half of the court half-court. This is considered the offensive half of the court</li> <li>"offensive_half_court" The TV-right half of the court half-court. This is considered the offensive half of the court</li> <li>"offensive half court" The TV-right half of the court half-court. This is considered the offensive half of the court</li> <li>"defense" The TV-left half of the court half-court. This is considered the defensive half of the court</li> <li>"defence" The TV-left half of the court half-court. This is considered the defensive half of the court</li> </ul>

"defensivehalfcourt" The TV-left half of the court half-court. This is considered the defensive half of the court

"defensive\_half\_court" The TV-left half of the court half-court. This is considered the defensive half of the court

"defensive half court" The TV-left half of the court half-court. This is considered the defensive half of the court

"offensivekey" The TV-right offensive key (three-point line and two-point range)

"offensive\_key" The TV-right offensive key (three-point line and two-point range)

"offensive key" The TV-right offensive key (three-point line and two-point range)

"attackingkey" The TV-right offensive key (three-point line and two-point range)

"attacking\_key" The TV-right offensive key (three-point line and two-point range)

"attacking key" The TV-right offensive key (three-point line and two-point range)

"defensivekey" The TV-left defensive key (three-point line and two-point range)

"defensive\_key" The TV-left defensive key (three-point line and two-point range)

"defensive key" The TV-left defensive key (three-point line and two-point range)

"defendingkey" The TV-left defensive key (three-point line and two-point range)

"defending\_key" The TV-left defensive key (three-point line and two-point range)

"defending key" The TV-left defensive key (three-point line and two-point range)

"offensivepaint" The TV-right offensive free-throw lane

"offensive\_paint" The TV-right offensive free-throw lane

"offensive paint" The TV-right offensive free-throw lane

"attackingpaint" The TV-right offensive free-throw lane

"attacking\_paint" The TV-right offensive free-throw lane

"attacking paint" The TV-right offensive free-throw lane

"offensivelane" The TV-right offensive free-throw lane

"offensive\_lane" The TV-right offensive free-throw lane

"offensive lane" The TV-right offensive free-throw lane

"attackinglane" The TV-right offensive free-throw lane

"attacking\_lane" The TV-right offensive free-throw lane

"attacking lane" The TV-right offensive free-throw lane

"defensivepaint" The TV-left defensive free-throw lane

"defensive\_paint" The TV-left defensive free-throw lane

"defensive paint" The TV-left defensive free-throw lane

"defendingpaint" The TV-left defensive free-throw lane

	"defending_paint" The TV-left defensive free-throw lane
	"defending paint" The TV-left defensive free-throw lane
	"defensivelane" The TV-left defensive free-throw lane
	"defensive_lane" The TV-left defensive free-throw lane
	"defensive lane" The TV-left defensive free-throw lane
	"defendinglane" The TV-left defensive free-throw lane
	"defending_lane" The TV-left defensive free-throw lane
	"defending lane" The TV-left defensive free-throw lane
court_updates	A list of updates to the courts' parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the courts' default colors, which are set by <code>basketball_features_set_colors()</code>
rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the center of the court towards the right-hand basket when viewing the court in TV View
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the center of the court towards the top of the court when viewing the court in TV view
court_units	The units with which to draw the court. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the <code>display_range</code> parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the <code>display_range</code> parameter

**Value**

A ggplot2 instance with a full-surface representation of a basketball court

**Examples**

```
## Not run:
  geom_basketball(league = "NBA", rotation = 270, display_range = "offense")
  geom_basketball(league = "fiba", court_units = "ft")

## End(Not run)
```

---

geom_curling	<i>Generate a ggplot2 instance containing a curling sheet for a specified league</i>
--------------	--

---

**Description**

Generate a ggplot2 instance containing a curling sheet for a specified league



**Usage**

```
geom_curling(
  league,
  display_range = "full",
  sheet_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
  y_trans = 0,
  sheet_units = NULL,
  xlims = NULL,
  ylims = NULL
)
```

**Arguments**

league	The league for which to draw the surface. This is case-insensitive
display_range	A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function. The possible display ranges are: "full" The full sheet. This is the default "house" A single house, which defaults to the top house in TV view
sheet_updates	A list of updates to the sheet's parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the sheet's default colors, which are set by <a href="#">curling_features_set_colors()</a>
rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the center of the sheet towards the right-hand goal when viewing the sheet in TV View
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the center of the sheet towards the top of the sheet when viewing the sheet in TV view
sheet_units	The units with which to draw the sheet. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the display_range parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the display_range parameter

**Value**

A ggplot2 instance with a full-surface representation of a curling sheet

**Examples**

```
## Not run:
geom_curling(league = "wcf", rotation = 270, display_range = "house")
geom_curling(league = "wcf", sheet_units = "ft")

## End(Not run)
```

---

geom_football	<i>Generate a ggplot2 instance containing a football field for a specified league</i>
---------------	---

---

**Description**

Generate a ggplot2 instance containing a football field for a specified league

**Usage**

```
geom_football(
  league,
  display_range = "full",
  field_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
  y_trans = 0,
  field_units = NULL,
  xlims = NULL,
  ylims = NULL
)
```

**Arguments**

league	The league for which to draw the surface. This is case-insensitive
display_range	A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function. The possible display ranges are: "full" The full field. This is the default "offense" The TV-right half of the field "offence" The TV-right half of the field "offensivehalffield" The TV-right half of the field "offensive_half_field" The TV-right half of the field "offensive half field" The TV-right half of the field "defense" The TV-left half of the field "defence" The TV-left half of the field

	"defensivehalffield" The TV-left half of the field
	"defensive_half_field" The TV-left half of the field
	"defensive half field" The TV-left half of the field
	"redzone" The offensive red zone of the field. This is by definition 20 yards from the goal line
	"red_zone" The offensive red zone of the field. This is by definition 20 yards from the goal line
	"red zone" The offensive red zone of the field. This is by definition 20 yards from the goal line
	"oredzone" The offensive red zone of the field. This is by definition 20 yards from the goal line
	"offensive_red_zone" The offensive red zone of the field. This is by definition 20 yards from the goal line
	"offensive red zone" The offensive red zone of the field. This is by definition 20 yards from the goal line
	"dredzone" The defensive red zone of the field. This is by definition 20 yards from the goal line
	"defensive_red_zone" The defensive red zone of the field. This is by definition 20 yards from the goal line
	"defensive red zone" The defensive red zone of the field. This is by definition 20 yards from the goal line
field_updates	A list of updates to the field's parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the field's default colors, which are set by <a href="#">football_features_set_colors()</a>
rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the center of the field towards the right-hand endzone when viewing the field in TV View
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the center of the field towards the sideline when viewing the field in TV view
field_units	The units with which to draw the field. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the <code>display_range</code> parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the <code>display_range</code> parameter

**Value**

A `ggplot2` instance with a full-surface representation of a football field

**Examples**

```
## Not run:
geom_football(league = "NFL", rotation = 270, display_range = "red_zone")
geom_football(league = "cfl", field_units = "ft")

## End(Not run)
```

---

geom_hockey	<i>Generate a ggplot2 instance containing an ice rink for a specified league</i>
-------------	--

---

**Description**

Generate a ggplot2 instance containing an ice rink for a specified league

**Usage**

```
geom_hockey(
  league,
  display_range = "full",
  rink_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
  y_trans = 0,
  rink_units = NULL,
  xlims = NULL,
  ylims = NULL
)
```

**Arguments**

league	The league for which to draw the surface. This is case-insensitive
display_range	A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function. The possible display ranges are: "full" The full ice surface. This is the default "offense" The TV-right half of the ice surface "offence" The TV-right half of the ice surface "defense" The TV-left half of the ice surface "defence" The TV-left half of the ice surface "ozone" The TV-right zone of the ice surface "offensive_zone" The TV-right zone of the ice surface "offensive zone" The TV-right zone of the ice surface

	"attacking_zone" The TV-right zone of the ice surface
	"attacking zone" The TV-right zone of the ice surface
	"dzone" The TV-left zone of the ice surface
	"defensive_zone" The TV-left zone of the ice surface
	"defensive zone" The TV-left zone of the ice surface
	"defending_zone" The TV-left zone of the ice surface
	"defending zone" The TV-left zone of the ice surface
	"nzone" The middle zone of the ice surface
	"neutral" The middle zone of the ice surface
	"neutral_zone" The middle zone of the ice surface
	"neutral zone" The middle zone of the ice surface
rink_updates	A list of updates to the rink's parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the courts' default colors, which are set by <code>hockey_features_set_colors()</code>
rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the center of the ice surface towards the right-hand goal when viewing the rink in TV View
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the center of the ice surface towards the top of the rink when viewing the rink in TV view
rink_units	The units with which to draw the rink. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the <code>display_range</code> parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the <code>display_range</code> parameter

**Value**

A ggplot2 instance with a full-surface representation of an ice hockey rink

**Examples**

```
## Not run:
geom_hockey(league = "NHL", rotation = 270, display_range = "ozone")
geom_hockey(league = "iihf", rink_units = "ft")

## End(Not run)
```

---

geom_soccer	<i>Generate a ggplot2 instance containing a soccer pitch for a specified league</i>
-------------	---

---

### Description

Generate a ggplot2 instance containing a soccer pitch for a specified league

### Usage

```
geom_soccer(
  league,
  display_range = "full",
  pitch_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
  y_trans = 0,
  pitch_units = NULL,
  xlims = NULL,
  ylims = NULL
)
```

### Arguments

league	The league for which to draw the surface. This is case-insensitive
display_range	<p>A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function.</p> <p>The possible display ranges are:</p> <ul style="list-style-type: none"> <li>"full" The full pitch. This is the default</li> <li>"offense" The TV-right half of the pitch</li> <li>"offence" The TV-right half of the pitch</li> <li>"offensivehalfpitch" The TV-right half of the pitch</li> <li>"offensive_half_pitch" The TV-right half of the pitch</li> <li>"offensive half pitch" The TV-right half of the pitch</li> <li>"defense" The TV-left half of the pitch</li> <li>"defence" The TV-left half of the pitch</li> <li>"defensivehalfpitch" The TV-left half of the pitch</li> <li>"defensive_half_pitch" The TV-left half of the pitch</li> <li>"defensive half pitch" The TV-left half of the pitch</li> </ul>
pitch_updates	A list of updates to the pitch's parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the pitch's default colors, which are set by <a href="#">soccer_features_set_colors()</a>

rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the center of the pitch towards the right-hand goal when viewing the pitch in TV View
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the center of the pitch towards the top of the pitch when viewing the pitch in TV view
pitch_units	The units with which to draw the pitch. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the display_range parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the display_range parameter

**Value**

A ggplot2 instance with a full-surface representation of a soccer pitch

**Examples**

```
## Not run:
geom_soccer(league = "EPL", rotation = 270, display_range = "offense")
geom_soccer(league = "fifa", pitch_units = "ft")

## End(Not run)
```

---

geom_tennis	<i>Generate a ggplot2 instance containing a tennis court for a specified league</i>
-------------	---

---

**Description**

Generate a ggplot2 instance containing a tennis court for a specified league

**Usage**

```
geom_tennis(
  league,
  display_range = "full",
  court_updates = list(),
  color_updates = list(),
  rotation = 0,
  x_trans = 0,
  y_trans = 0,
  court_units = NULL,
  xlims = NULL,
  ylims = NULL
)
```

**Arguments**

league	The league for which to draw the surface. This is case-insensitive
display_range	A case-insensitive string indicating the display range to use for the plot. The default is "full", which will be returned when either an invalid or no value is passed to the function. The possible display ranges are: "full" The full court. This is the default "serve" The serving half of the court "serving" The serving half of the court "servicehalf" The serving half of the court "service_half" The serving half of the court "service half" The serving half of the court "servinghalf" The serving half of the court "serving_half" The serving half of the court "serving half" The serving half of the court "receive" The receiving half of the court "receiving" The receiving half of the court "receivicehalf" The receiving half of the court "receivice_half" The receiving half of the court "receivice half" The receiving half of the court "receivinghalf" The receiving half of the court "receiving_half" The receiving half of the court "receiving half" The receiving half of the court
court_updates	A list of updates to the courts' parameters. These will overwrite the parameters of the league
color_updates	A list of updates to the courts' default colors, which are set by <a href="#">tennis_features_set_colors()</a>
rotation	An angle, given in degrees, through which the plot should be rotated
x_trans	The amount that the x coordinates are to be shifted. By convention, the +x axis extends from the center of the court towards the right-hand serviceline when viewing the court in TV View
y_trans	The amount that the y coordinates are to be shifted. By convention, the +y axis extends from the center of the court towards the sideline when viewing the court in TV view
court_units	The units with which to draw the court. The default is NULL, which will apply the rule-book specified units
xlims	The limits on the final display in the x direction. The default is NULL, which will utilize the xlims specified by the display_range parameter
ylims	The limits on the final display in the y direction. The default is NULL, which will utilize the ylims specified by the display_range parameter

**Value**

A ggplot2 instance with a full-surface representation of a tennis court



**Examples**

```
## Not run:  
  geom_tennis(league = "USTA", rotation = 270, display_range = "serving")  
  geom_tennis(league = "itf", court_units = "m")  
  
## End(Not run)
```

# Index

baseball\_features\_set\_colors(), [5](#)  
basketball\_features\_set\_colors(), [8](#)  
  
cani\_color\_league\_features, [2](#)  
cani\_plot\_league, [3](#)  
cani\_plot\_sport, [3](#)  
convert\_units, [4](#)  
curling\_features\_set\_colors(), [9](#)  
  
football\_features\_set\_colors(), [11](#)  
  
geom\_baseball, [4](#)  
geom\_basketball, [6](#)  
geom\_curling, [8](#)  
geom\_football, [10](#)  
geom\_hockey, [12](#)  
geom\_soccer, [14](#)  
geom\_tennis, [15](#)  
  
hockey\_features\_set\_colors(), [13](#)  
  
soccer\_features\_set\_colors(), [14](#)  
  
tennis\_features\_set\_colors(), [16](#)