Package 'soccermatics'

June 18, 2018

Version 0.9.0

Title Visualise tracking and event data from soccer matches

Description Provides tools to visualise x,y-coordinates of soccer players and event data (passes, shots). Uses ggplot to draw soccer pitch and overplot expected goal maps, pass maps, average player positions, player heatmaps, individual player paths, player flow fields, and more.

Depends R (>= 3.4.1)

Imports dplyr, magrittr, ggplot2, ggforce, zoo

License GPL (>=3.0) Note: Use of the name 'soccermatics' was kindly permitted by David Sumpter and is protected from commercial use under EU copyright law.

Encoding UTF-8

LazyData true

Collate 'soccerFlipHoriz.R'
'soccerPitchFG.R'
'soccerHeatmap.R'
'soccerPitchBG.R'
'soccerSpokes.R'
'soccerFlow.R'
'soccerPassmap.R'
'soccerPath.R'
'soccerPositionMap.R'
'soccerResample.R'
'soccerShotmap.R'
'soccerTransformSB.R'
'soccerTransform.R'
'soccerVelocity.R'
'tromso.R'
'tromso_extra.R'

RoxygenNote 6.0.1

R topics documented:

`soccerFlipHoriz` *Flips x,y-coordinates horizontally in one half to account for changing sides at half-time*

Description

Normalises direction of attack in both halves of both teams by flipping x,y-coordinates horizontally in either the first or second half; i.e. teams attack in the same direction all game despite changing sides at half-time.

Usage

```
soccerFlipHoriz(dat, periodVar = "period", periodToFlip = 1,
  pitchLength = 105, pitchWidth = 68)
```

Arguments

`dat`	= dataframe containing unnormalised x,y-coordinates named 'x' and 'y'
`periodVar`	= name of variable containing period labels
`periodToFlip`	= which period to flip
`pitchLength, pitchWidth`	= length, width of pitch in metres

Value

a dataframe

Examples

```
# to flip coordinates in 2nd half of a dataframe with 1st/2nd half identity
# labelled by variable named `period`
soccerFlipHoriz(df, "period", 2, 105, 68)
```

soccerFlow *Draw a flow field on a soccer pitch.*

Description

Draws a flow field showing the mean direction of movement made in each sector of the pitch and adds pitch outlines. Note: This function is prototypical and intended to eventually visualise pass and shot event data, but there are no open-source samples of such data available as yet.

Usage

```
soccerFlow(df, lengthPitch = 105, widthPitch = 68, xBins = 5,
  yBins = NULL, fillPitch = "white", colPitch = "grey60", grass = FALSE,
  lwd = 0.5, border = c(4, 4, 4, 4), plot = NULL)
```

Arguments

df	dataframe containing x,y-coordinates of player position in columns named 'x' and 'y' and angular information (in radians, ranging between -pi and pi) in a column 'direction'.
lengthPitch, widthPitch	numeric, length and width of pitch in metres.
xBins, yBins	integer, the number of horizontal (length-wise) and vertical (width-wise) bins the soccer pitch is to be divided up into. If no value for yBins is provided, it will take the value of xBins.
grass	if TRUE, draws pitch background in green and lines in white. If FALSE, draws pitch background in white and lines in black.
lwd	thickness of arrow lines.
plot	optional, adds wagon wheels to an existing ggplot object if provided
line_col	colour of pitch lines.

Value

a ggplot object of a heatmap on a soccer pitch.

See Also

soccerHeatmap for drawing a heatmap of player position, or soccerSpokes for drawing spokes to show all directions in each area of the pitch.

Examples

```
data(tromso_extra)
# draw flow field showing mean direction of player #8's movement
soccerFlow(subset(tromso_extra, id == 8), xBins = 5, grass = TRUE)
# draw flow field over player heatmap
p <- soccerHeatmap(subset(tromso_extra, id == 8), xBins = 5)
soccerFlow(subset(tromso_extra, id == 8), xBins = 5, plot = p)
```

`soccerHeatmap` *Draw a heatmap on a soccer pitch.*

Description

Draws a heatmap showing player position frequency in each area of the pitch and adds soccer pitch outlines.

Usage

```
soccerHeatmap(df, lengthPitch = 105, widthPitch = 68, xBins = 10,
  yBins = NULL, colPitch = "black", colLow = "white", colHigh = "red")
```

Arguments

`df`	dataframe containing x,y-coordinates of player position in columns named `'x'` and `'y'`.
`lengthPitch, widthPitch`	numeric, length and width of pitch in metres.
`xBins, yBins`	integer, the number of horizontal (length-wise) and vertical (width-wise) bins the soccer pitch is to be divided up into. If no value for `yBins` is provided, it will take the value of `xBins`.
`colLow, colHigh`	character, colours for the low and high ends of the heatmap gradient.

Details

uses `ggplot2::geom_bin2d` to map 2D bin counts

Value

a ggplot object of a heatmap on a soccer pitch.

See Also

`soccerPitchBG` for a background soccer pitch for the purpose of drawing position maps, player trajectories, etc..

Examples

```
data(tromso)
# simple heatmap of player #9's position
soccerHeatmap(subset(tromso, id == 8), xBins = 10)

# draw heatmap with approximately 5m x 5m bins (pitchLength / 5 = 21, pitchWidth / 5 = 13.6)
soccerHeatmap(subset(tromso, id == 8), xBins = 21, yBins = 14)
```

soccerPassmap *Draw a map of all passes from StatsBomb data*

Description

Draw a map of all passes from StatsBomb data. Compatability with non-StatsBomb data will be added soon.

Usage

```
soccerPassmap(dat, lengthPitch = 105, widthPitch = 68,
  colComplete = "blue", colFail = "red", alpha = 0.8, legend = FALSE,
  fillPitch = "white", colPitch = "grey60", grass = FALSE, lwd = 0.5,
  border = c(4, 4, 4, 4), SB = FALSE)
```

Arguments

lengthPitch, widthPitch	length and width of pitch in metres
alpha	transparency of points
legend	boolean, include legend or not
fillPitch	pitch fill colour
colPitch	pitch line colour
grass	if TRUE, draws a more realistic looking pitch
lwd	pitch line width
border	size of border drawn around pitch perimeter (t,r,b,l)
colGoal, colMiss	colour of circles for scored and missed shots

Value

a ggplot object

See Also

soccerPitchBG for drawing a soccer pitch as foreground over an existing ggplot object

Examples

```
library(StatsBombR)

# get data
Matches <- FreeMatches(37)
d <- allinfo(Matches[1])

# transform all x,y-coordinates of StatsBomb data
d <- soccerTransformSB(d)

# shotmap of Manchester City WFC
d %>%
```

```
  filter(type.name == "Pass",
  team.name == "Manchester City WFC") %>%
  soccerPassmap(SB = TRUE)
```

---

soccerPath *Draw a path of player trajectory on a soccer pitch.*

Description

Draws a path connecting consecutive x,y-coordinates of a player on a soccer pitch.

Usage

```
soccerPath(dat, lengthPitch = 105, widthPitch = 68, id_var = NULL,
  col = "black", fillPitch = "white", colPitch = "grey60",
  grass = FALSE, lwd = 1, legend = TRUE, plot = NULL)
```

Arguments

dat	dataframe containing x,y-coordinates of player position in columns named 'x' and 'y'
lengthPitch, widthPitch	length and width of pitch in metres
id_var	character, the name of the column containing player identity. Only required if 'dat' contains multiple players
col	colour of path if no 'id_var' is provided. If an 'id_var' is present, colours from ColorBrewer's 'Paired' palette are used
grass	if TRUE, draws a more realistic looking pitch
lwd	player path thickness

Value

a ggplot object

Examples

```
data(tromso)
# draw path of player #8 over first 1200 frames
subset(tromso, id == 8)[1:1200,] %>%
  soccerPath(col = "red", grass = TRUE)
# draw path of all players over first 1200 frames
tromso %>%
  dplyr::group_by(id) %>%
  dplyr::slice(1:1200) %>%
  soccerPath("id")
```

soccerPitchBG *Draw a soccer pitch.*

Description

Draws a soccer pitch as a ggplot object for the purpose of adding layers such as player positions, player trajectories, etc..

Usage

```
soccerPitchBG(lengthPitch = 105, widthPitch = 68, fillPitch = "white",
  colPitch = "grey60", grass = FALSE, lwd = 0.5, border = c(4, 4, 4, 4),
  direction = c("none", "r", "l"), SB = FALSE)
```

Arguments

lengthPitch, widthPitch	length and width of pitch in metres
fillPitch	pitch fill colour
colPitch	pitch line colour
grass	if TRUE, draws a more realistic looking pitch
lwd	pitch line width
border	size of border drawn around pitch perimeter (t,r,b,l)

Value

a ggplot object

See Also

soccerPitchFG for drawing a soccer pitch as foreground over an existing ggplot object

Examples

```
# get x,y-coords of player #8 during first 10 minutes
data(tromso)
dd <- subset(tromso, id == 9)[1:1200,]
# draw player path on pitch
soccerPitchBG(lengthPitch = 105, widthPitch = 68, grass = TRUE) +
  geom_path(data = dd, aes(x, y))
```

soccerPitchFG *Add soccer pitch outlines to an existing ggplot*

Description

Draws soccer pitch outlines (with transparent fill) over an existing ggplot object to provide context for heatmaps, passing maps, etc..

Usage

```
soccerPitchFG(plot, lengthPitch = 105, widthPitch = 68,
  colPitch = "black", lwd = 0.5, SB = FALSE)
```

Arguments

plot	an existing ggplot object to add layers to.
lengthPitch, widthPitch	numeric, length and width of pitch in metres.

Value

a ggplot object

See Also

soccerPitchBG for a background soccer pitch for the purpose of drawing position maps, player trajectories, etc..

Examples

```
data(tromso)
# draw heatmap of player #9's position
p <- soccerHeatmap(subset(tromso, id == 8), bins = 15, lengthPitch = 105, widthPitch = 68)
# add pitch lines to plot
soccerPitchFG(p, lengthPitch = 105, widthPitch = 68)
```

soccerPositionMap *Plot average player position*

Description

Draws the average x,y-positions of each player from one or both teams on a soccer pitch.

Usage

```
soccerPositionMap(df, lengthPitch = 105, widthPitch = 68, id_var = "id",
  group_var = NULL, x_var = "x", y_var = "y", fill1 = "red",
  col1 = "white", fill2 = "blue", col2 = "white", node_size = 6,
  label_size = 3, label = TRUE, fillPitch = "white",
  colPitch = "grey60", lwd = 0.5, grass = FALSE)
```

Arguments

df	dataframe containing x,y-coordinates of player position in columns named 'x' and 'y'
lengthPitch, widthPitch	numeric, length and width of pitch in metres
id_var	character, the name of the column containing player identity. Defaults to 'id'
group_var	character, the name of the column containing team identity. Optional, defaults to 'NULL'
fill1, fill2	character, fill colour of position points for team one (and team two if 'group_var' provided)
col1, col2	character, border colour of position points for team one (and team two if 'group_var' provided)
node_size	numeric, size of position points
label_size	numeric, size of label names
label	boolean, draw labels or not
fillPitch	pitch fill colour
colPitch	pitch line colour
lwd	pitch line width
grass	if TRUE, draws a more realistic looking pitch
border	size of border drawn around pitch perimeter (t,r,b,l)

See Also

soccerPitchBG for a background soccer pitch for the purpose of drawing position maps, player trajectories, etc...

Examples

```
data(tromso)
# draw average player position of players
p <- soccerPositionMap(tromso, lengthPitch = 105, widthPitch = 68, grass = TRUE)
# draw arrow showing direction of play
soccerDirection(p, "right", lengthPitch = 105, widthPitch = 68, grass = TRUE)
```

---

soccerResample — *Resample the frequency of x,y,t- time series with linear interpolation of x,y-coordinates.*

Description

Downsample or upsample dataframe containing x,y-coordinates and a time variable 't' with linear interpolation of x,y-coordinates and constant interpolation of all other variables.

Usage

```
soccerResample(dat, r = 10)
```

Arguments

dat	= dataframe containing unnormalised x,y-coordinates named ‘x‘ and ‘y‘ and a time variable named ‘t‘
r	resampling rate in frames per second

Value

a dataframe

Examples

```
# resample tromso dataset from ~21 fps to 10 fps
soccerResample(tromso)
```

---

soccerShotmap *Draw a shotmap on a half pitch from StatsBomb data*

Description

Draw a shotmap on a half pitch from StatsBomb data. Compatability with non-StatsBomb data will be added soon.

Usage

```
soccerShotmap(dat, lengthPitch = 105, widthPitch = 68,
  colGoal = "skyblue", colMiss = "grey60", alpha = 0.8, legend = FALSE,
  fillPitch = "white", colPitch = "grey60", grass = FALSE, lwd = 0.5,
  border = c(4, 4, 4, 4), SB = FALSE)
```

Arguments

lengthPitch, widthPitch	length and width of pitch in metres
colGoal, colMiss	colour of points representing scored and missed shots
alpha	transparency of points
legend	boolean, include legend or not
fillPitch	pitch fill colour
colPitch	pitch line colour
grass	if TRUE, draws a more realistic looking pitch
lwd	pitch line width
border	size of border drawn around pitch perimeter (t,r,b,l)

Value

a ggplot object

See Also

soccerPitchBG for drawing a soccer pitch as foreground over an existing ggplot object

Examples

```
library(StatsBombR)

# get data
Matches <- FreeMatches(37)
d <- allinfo(Matches[1])

# transform all x,y-coordinates of StatsBomb data
d <- soccerTransformSB(d)

# pass map of Manchester City WFC
d %>%
  filter(type.name == "Shot",
  team.name == "Manchester City WFC") %>%
  soccerShotmap(fillPitch = "#1C1F26", colPitch = "white", SB = TRUE)
```

---

soccerSpokes *Visualise movement direction on a soccer pitch.*

Description

Draws spokes showing the direction of x,y-movements made in each sector of the pitch.

Usage

```
soccerSpokes(plot, df, lengthPitch = 105, widthPitch = 68, xBins,
  yBins = NULL, angleBins = 16, lwd = 0.5, minLength = 0.6,
  minAlpha = 0.4, legend = TRUE)
```

Arguments

plot	plot of soccer pitch returned by soccerPitchBG to add spokes to
df	dataframe containing x,y-coordinates of player position in columns named x and y and angular information (in radians, ranging between -pi and pi) in a column direction.
lengthPitch, widthPitch	length and width of pitch in metres.
xBins, yBins	integer, the number of horizontal (length-wise) and vertical (width-wise) bins the soccer pitch is to be divided up into. If no value for yBins is provided, it will take the value of xBins.
angleBins	integer, the number of angle bins movement directions are divided up into. For example, a value of 4 clusters directions in each bin into north, east, south and west.
lwd	thickness of arrow lines
minLength	numeric, ratio between size of shortest arrow and longest arrow depending on number of events.
legend	if TRUE, adds legend showing relationship between arrow transparency and number of events

Value

a ggplot object

See Also

soccerPitchBG for drawing a heatmap of player position, soccerHeatmap for drawing a heatmap of player position

Examples

```
data(tromso_extra)
# resample movement dataset to plot 100 movement directions
# (in absence of pass / shot event data as yet)
id8 <- tromso_extra %>%
  dplyr::filter(id == 8) %>%
  dplyr::sample_n(100)
# 5x5 x,y-bins, 16 angle-bins, blank pitch
soccerPitchBG(pitchLength, pitchWidth) %>%
  soccerSpokes(id8, xBins = 5, angleBins = 16, minLength = 0.4)
# 10x10 x,y-bins, 8 angle-bins, grass pitch
soccerPitchBG(pitchLength, pitchWidth, grass = T) %>%
  soccerSpokes(id8, xBins = 10, angleBins = 8, minLength = 0.2, lwd = 1)
# draw spokes over player heatmap w/ 5x5 x,y-bins, 8 angle-bins
soccerHeatmap(id8, xBins = 5) %>%
  soccerSpokes(id8, xBins = 5, angleBins = 8, lwd = 1)
```

---

soccerTransform *Normalises x,y-coordinates to metres units for use with soccermatics functions*

Description

Normalises x,y-coordinates from between any arbitrary bounds to metre units bounded by [0 < x < pitchLength, 0 < y < pitchWidth]

Usage

```
soccerTransform(dat, xMin, xMax, yMin, yMax, pitchLength, pitchWidth)
```

Arguments

dat	dataframe containing unnormalised x,y-coordinates named ‘x‘ and ‘y‘
xMin, xMax, yMin, yMax	range of x,y-coordinates possible in the raw dataset
pitchLength, pitchWidth	length, width of pitch in metres

Value

a dataframe

See Also

soccerTransformSB readily transforms data from StatsBomb for use with soccermatics

soccerTransformSB for transforming data from StatsBomb

Examples

```
# Three examples with true pitch dimesions (in metres):
lengthPitch <- 101
widthPitch <- 68

# Example 1. Opta-style ------------------------------------------------------
# limits = [0 < x < 100, 0 < y < 100]
# centre of pitch = [50,50]

df <- data.frame(t = 1:12,
                 x = c(50,55,61,66,62,58,51,44,45,42,41,32),
                 y = c(50,48,47,40,42,45,49,51,59,75,88,100))

df <- soccerTransform(df, 0, 100, 0, 100, lengthPitch, widthPitch)

soccerPath(df, lengthPitch = lengthPitch, widthPitch = widthPitch)


# Example 2. StrataBet-style -------------------------------------------------
# limits = [0 < x < 420, -136 < y < 136]
# centre of pitch = [210,0]

df <- data.frame(t = 1:12,
                 x = c(210,222,201,192,178,170,143,122,104,91,75,60),
                 y = c(0,-5,-20,-12,-8,-2,4,8,13,20,30,45))

df <- soccerTransform(df, 0, 420, -136, 136, lengthPitch, widthPitch)

soccerPath(df, lengthPitch = lengthPitch, widthPitch = widthPitch)


# Example 3. Other -----------------------------------------------------------
# limits = [-5250 < x < 5250, -3400 < y < 3400]
# centre of pitch = [0,0]

xMin <- -5250
xMax <- 5250
yMin <- -3400
yMax <- 3400

df <- data.frame(x = c(0,-452,-982,-1099,-1586,-2088,-2422,-2999,-3200,-3857),
                 y = c(0,150,300,550,820,915,750,620,400,264))

df <- soccerTransform(df, -5250, 5250, -3400, 3400, lengthPitch, widthPitch)

soccerPath(df, lengthPitch = lengthPitch, widthPitch = widthPitch)
```

soccerTransformSB *Normalises all x,y-coordinate data from StatsBomb data*

Description

Normalises x,y-coordinates from StatsBomb data to metre units for use with soccermatics functions. [StatsBomb pitch coordinates](https://github.com/statsbomb/open-data/blob/master/doc/StatsBomb

Usage

```
soccerTransformSB(dat)
```

Arguments

dat	dataframe returned from StatsBombR

Value

a dataframe

See Also

soccerPitchBG readily visualises StatsBomb data with the argument 'SB = TRUE'. soccerTransform transforms any arbitrary x,y-coordinates to metre units

Examples

```
# get data
library(StatsBombR)
Matches <- FreeMatches(37)
d <- allinfo(Matches[1])

# transform
d <- soccerTransformSB(d)
```

soccerVelocity *Compute instantaneous distance, speed and direction from x,y-coordinates*

Description

Compute instantaneous distance moved (in metres), speed (in metres per second), and direction (in radians) between subsequent frames in a dataframe of x,y-coordinates.

Usage

```
soccerVelocity(dat)
```

Arguments

dat	= dataframe containing unnormalised x,y-coordinates ‘x‘ and ‘y‘, time variable ‘t‘, and identifier ‘id‘

Value

a dataframe

Examples

```
# calculate distance, speed, and direction for tromso dataset
soccerVelocity(tromso)
```

---

tromso *x,y-coordinates of 11 soccer players over 12000 frames each*

Description

x,y-coordinates of 11 soccer players over 10 minutes (Tromsø IL vs. Anzhi, 2013-11-07), captured at 20 Hz using the ZXY Sport Tracking system and made available in the publication ZXY Sport Tracking.

Usage

```
data(tromso)
```

Format

A dataframe containing 12000 frames of x,y-coordinates and timestamps from 11 players.

Source

ZXY Sport Tracking

References

Pettersen et al. (2014) Proceedings of the International Conference on Multimedia Systems (MMSys)

Examples

```
data(tromso)
# draw path of player #8 on a soccer pitch
soccerPitchBG(lengthPitch = 105, widthPitch = 68, grass = TRUE) +
  geom_path(data = subset(tromso, id == 8), aes(x, y), lwd = 2)
```

`tromso_extra` *x,y-coordinates and additional positional information on 11 soccer players over 12000 frames each*

Description

x,y-coordinates of 11 soccer players over 10 minutes (Tromsø IL vs. Anzhi, 2013-11-07), plus additional information on player heading, direction, energy, speed, and total distance. Data captured at 20 Hz using the ZXY Sport Tracking system and made available in the publication ZXY Sport Tracking.

Usage

```
data(tromso_extra)
```

Format

A dataframe containing 12000 frames of x,y-coordinates and timestamps from 11 players.

Source

ZXY Sport Tracking

References

Pettersen et al. (2014) Proceedings of the International Conference on Multimedia Systems (MMSys) (pdf)

Examples

```
data(tromso_extra)
# draw flow field showing mean direction of player #8's movement
soccerFlow(subset(tromso_extra, id == 8), bins = 5, grass = TRUE)
```

Index