Package 'AdvancedBasketballStats'

April 6, 2021

Type Package

Date 2021-04-06

Title Advanced Basketball Statistics

Version 1.0.1

Author Francisco Javier Cantero [aut, cre],
Juan Jose Cuadrado [aut],
Universidad de Alcala de Henares [aut]

Maintainer Francisco Javier Cantero <fco.cantero@edu.uah.es>

Description Provides different functionalities and calculations used in the world of basketball to analyze the statistics of the players, the statistics of the teams, the statistics of the quintets and the statistics of the plays. For more details of the calculations included in the package can be found in the book Basketball on Paper written by Dean Oliver.

License GPL (>= 2)

Encoding UTF-8

LazyData false

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2021-04-06 12:10:07 UTC

R topics documented:

`individuals_advance_stats` *Individual advanced statistics*

Description

This function allows the calculation of advanced individual statistics.

Usage

```
individuals_advance_stats(df1, df2, df3)
```

Arguments

`df1`	Should be a Data Frame that represents the individual statistics or individual defensive statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.
`df2`	Should be a Data Frame that represents the team's statistics. The parameter has to be in the format provided by the team_stats() function.
`df3`	Should be a Data Frame that represents the rival's statistics. The parameter has to be in the format provided by the team_stats() function.

Value

Data frame with the following advanced statistics calculated:

- Player Efficiency Rating (PER)
- Efficiency Field Goals percentage (eFG%)
- True shooting percentage (TS%)
- Three rating (3Par)
- Free Throw rating (FTr)
- Offensive rebounds percentage (ORB%)
- Defensive rebounds percentage (DRB%)
- Total rebounds percentage (TRB%)
- Assists percentage (AST%)
- Steal percentage (STL%)
- Block percentage (BLK%)
- Turnover percentage (TOV%)
- Usage percentage (USG%)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"),"G" = c(67,0),
"GS" = c(62,0),"MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0),"3P" = c(148,0),"3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),"2P" = c(495,0),"2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0),"FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),"AST" = c(684,0),"STL" = c(78,0),"BLK" = c(36,0),
"TOV" = c(261,0),"PF" = c(118,0),"PTS" = c(1698,0),"+/-" = c(0,0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48),"3P" = c(782),"3PA" = c(2242),
"Percentage 3P" = c(0.349),"2P" = c(2224), "2PA" = c(4027),
"Percentage 2P" = c(0.552),"FT" = c(1260),"FTA FG" = c(1728),
"Percentage FT" = c(0.729), "ORB" = c(757),  "DRB" = c(2490),
"TRB" = c(3247), "AST" = c(1803),  "STL" = c(612),"BLK" = c(468),
"TOV" = c(1077),"PF" = c(1471),  "PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
```

```
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))

individuals_advance_stats(df1,df2,df3)
```

`individuals_data_adjustment` *Individual stat adjuster*

Description

The function transform the statistics entered for later use in the rest of the functions that apply to individuals statistics.

Usage

```
individuals_data_adjustment(df1)
```

Arguments

`df1`	Should be a Data Frame that represents the individual statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.

Details

- The data.frame must have the same columns and these represent the same as in the example.
- The input data.frame must have the last row that represents the team's statistics.
- The function allows the transformation of the individual's statistics to which the shooting percentages and the number of total rebounds are added.
- The function allows the transformation of the defensive statistics to which the force missed shot and the forced turnovers.

Value

Data.frame with the transformed statistics for use in the rest of the functions.

The data frame obtained for the individual's statistics will have the following format:

- Name of the player (Name)
- Games played (G)
- Games Started (GS)
- Minutes Played (MP)

- Field Goals Made (FG)
- Field Goals Attempted (FGA)
- Field Goals Percentage (FG%)
- Three Points Made (3P)
- Three Points Attempted (3PA)
- Three Points Percentage (3P
- Two Points Made (2P)
- Two Points Attempted (2PA)
- Two Points Percentage (2P%)
- Free Throw Made (FT)
- Free Throw Attempted (FTA)
- Free Throw Percentage (FT%)
- Offensive Rebounds (ORB)
- Defensive Rebounds (DRB)
- Total Rebounds (TRB)
- Assists (AST)
- Steals (STL)
- Blocks (BLK)
- Turnover (TOV)
- Personal Fouls (PF)
- Points (PTS)
- Plus Minus (+/-)

The data frame obtained for the defensive individual's statistics will have the following format:

- Name of the player (Name)
- Minutes Played (MP)
- Defensive Rebounds (DRB)
- FGA by opposing team (FM)
- Blocks (BLK)
- (TOTAL FM)
- Forced turnover(FTO)
- Steals (STL)
- Total forced turnover (TOTAL FTO)
- FTA by opposing team (FFTA)
- FG made by opposing team (DFGM)
- FT made by opposing team (DFTM)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("James","Team"), "G" = c(67,0), "GS" = c(62,0),
"MP" = c(2316,1), "FG" = c(643,0), "FGA" = c(1303,0),"3P  " = c(148,0),
"3PA" = c(425,0),"2P" = c(495,0), "2PA" = c(878,0),  "FT" = c(264,0),
"FTA" = c(381,0),"ORB" = c(66,0),  "DRB" = c(459,0), "AST" = c(684,0),
"STL" = c(78,0),  "BLK" = c(36,0),"TOV" = c(261,0),  "PF" = c(118,0),
"PTS" = c(1698,0),  "+/-" = c(0,0))

individuals_data_adjustment(df1)

df2 <- data.frame("Name" = c("Witherspoon ","Team"), "MP" = c(14,200),
"DREB" = c(1,0),"FM" = c(4,0), "BLK" = c(0,0),"FTO" = c(0,0),
"STL" = c(1,1), "FFTA" = c(0,0),  "DFGM" = c(1,0), "DFTM" = c(0,0))

individuals_data_adjustment(df2)
```

---

individuals_defensive_actual_floor_stats

Individual's defensive actual statistics

Description

The function allows the calculation of individual defensive actual statistics on court

Usage

```
individuals_defensive_actual_floor_stats(df1, df2, df3)
```

Arguments

df1	Should be a Data Frame that represents the individual defensive statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.
df2	Should be a Data Frame that represents the team's statistics. The parameter has to be in the format provided by the team_stats() function.
df3	Should be a Data Frame that represents the rival's statistics. The parameter has to be in the format provided by the team_stats() function.

Value

Data frame with the following individual defensive actual statistics

- Defensive Stops (DStops)
- Defensive Scores Possesions (DscPoss)
- Defensive Possesions (DPoss)
- Stops percentage (STOPS%)
- (TMDPossS%)
- Defensive Rating (DRtg)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("Witherspoon ","Team"), "MP" = c(14,200),
"DREB" = c(1,0), "FM" = c(4,0), "BLK" = c(0,0),"TOTAL FM" = c(4,0),
"FTO" = c(0,0),"STL" = c(1,1), "TOTAL FTO " = c(1,0), "FFTA" = c(0,0),
"DFGM" = c(1,0), "DFTM" = c(0,0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48),"3P" = c(782),"3PA" = c(2242),
"Percentage 3P" = c(0.349),"2P" = c(2224), "2PA" = c(4027),
"Percentage 2P" = c(0.552),"FT" = c(1260),"FTA FG" = c(1728),
"Percentage FT" = c(0.729), "ORB" = c(757),  "DRB" = c(2490),
"TRB" = c(3247), "AST" = c(1803),  "STL" = c(612),"BLK" = c(468),
"TOV" = c(1077),"PF" = c(1471),  "PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))

individuals_defensive_actual_floor_stats(df1,df2,df3)
```

`individuals_defensive_estimated_floor_stats` *individual's defensive estimated statistics*

Description

The function allows the calculation of individual defensive estimated statistics on court

Usage

```
individuals_defensive_estimated_floor_stats(df1, df2, df3)
```

Arguments

`df1`	Should be a Data Frame. The parameter has to be in the format provided by the data_adjustment() function.
`df2`	Should be a Data Frame. The parameter has to be in the format provided by the team_stats() function.
`df3`	Should be a Data Frame. The parameter has to be in the format provided by the team_stats() function.

Value

Data frame with the following individual defensive estimated statistics

- Defensive Stops (DStops)
- Stops percentage (STOPS%)
- floor percentage (Floor%)
- Defensive Rating (DRtg)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"),"G" = c(67,0),
"GS" = c(62,0),"MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0),"3P" = c(148,0),"3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),"2P" = c(495,0),"2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0),"FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),"AST" = c(684,0),"STL" = c(78,0),"BLK" = c(36,0),
```

```
"TOV" = c(261,0),"PF" = c(118,0),"PTS" = c(1698,0),"+/-" = c(0,0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48),"3P" = c(782),"3PA" = c(2242),
"Percentage 3P" = c(0.349),"2P" = c(2224), "2PA" = c(4027),
"Percentage 2P" = c(0.552),"FT" = c(1260),"FTA FG" = c(1728),
"Percentage FT" = c(0.729), "ORB" = c(757),  "DRB" = c(2490),
"TRB" = c(3247), "AST" = c(1803),  "STL" = c(612),"BLK" = c(468),
"TOV" = c(1077),"PF" = c(1471),  "PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))

individuals_defensive_estimated_floor_stats(df1,df2,df3)
```

`individuals_games_adder` *individual's games adder*

Description

The function allows to perform the sums of two data.frames with the same format adopted after being transformed by individuals_data_adjustment() function,

Usage

```
individuals_games_adder(df1, df2)
```

Arguments

`df1`	Should be a Data Frame that represents the first set of individual statistics or defensive individual statistics of the players. It has to be in the format provided by the individuals_data_adjustment() function
`df2`	Should be a Data Frame that represents the second set of individual statistics or defensive individual statistics of the players. It has to be in the format provided by the individuals_data_adjustment() function

Details

The function will work correctly when the name of the players is the same, in case it is different it will take the players as different.

Value

Data frame with the sum of the statistics of the other two entered data.frame.

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"), "G" = c(67,0),
"GS" = c(62,0), "MP" = c(2316,0), "FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0), "3P" = c(148,0),  "3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),  "2P" = c(495,0), "2PA" = c(878,0),
"Percentage 2P" = c(0.564,0), "FT" = c(264,0), "FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),  "DRB" = c(459,0),
"TRB" = c(525,0),  "AST" = c(684,0),  "STL" = c(78,0),  "BLK" = c(36,0),
"TOV" = c(261,0),  "PF" = c(118,0),  "PTS" = c(1698,0),  "+/-" = c(0,0))

df2 <- data.frame("name" = c("LeBron James","Team"), "G" = c(67,0),
"GS" = c(62,0), "MP" = c(2316,0), "FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0), "3P" = c(148,0),  "3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),  "2P" = c(495,0), "2PA" = c(878,0),
"Percentage 2P" = c(0.564,0), "FT" = c(264,0), "FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),  "DRB" = c(459,0),
"TRB" = c(525,0),  "AST" = c(684,0),  "STL" = c(78,0),  "BLK" = c(36,0),
"TOV" = c(261,0),  "PF" = c(118,0),  "PTS" = c(1698,0),  "+/-" = c(0,0))

individuals_games_adder(df1,df2)
```

`individuals_ofensive_floor_stats` *individual's offensive floor stats*

Description

The function allows the calculation of individual's offensive statistics on court

Usage

```
individuals_ofensive_floor_stats(df1, df2, df3)
```

Arguments

`df1`	Should be a Data Frame that represents the individual statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.
`df2`	Should be a Data Frame that represents the team's statistics. The parameter has to be in the format provided by the team_stats() function.
`df3`	Should be a Data Frame that represents the rival's statistics. The parameter has to be in the format provided by the team_stats() function.

Value

Data frame with the following individual's offensive statistics

- Score possessions (Sc. Poss)
- Possessions attacked (Poss)
- floor percentage (Floor%)
- Offensive Rating (ORtg)
- Point produced per game (Pts Prod/G)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"),"G" = c(67,0),
"GS" = c(62,0),"MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0),"3P" = c(148,0),"3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),"2P" = c(495,0),"2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0),"FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),"AST" = c(684,0),"STL" = c(78,0),"BLK" = c(36,0),
"TOV" = c(261,0),"PF" = c(118,0),"PTS" = c(1698,0),"+/-" = c(0,0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48),"3P" = c(782),"3PA" = c(2242),
"Percentage 3P" = c(0.349),"2P" = c(2224), "2PA" = c(4027),
"Percentage 2P" = c(0.552),"FT" = c(1260),"FTA FG" = c(1728),
"Percentage FT" = c(0.729), "ORB" = c(757),  "DRB" = c(2490),
"TRB" = c(3247), "AST" = c(1803),  "STL" = c(612),"BLK" = c(468),
"TOV" = c(1077),"PF" = c(1471),  "PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))


individuals_ofensive_floor_stats(df1,df2,df3)
```

`individuals_stats_per_game` *individual's statistics per game*

Description

The function allows the calculation of individual statistics per game.

Usage

```
individuals_stats_per_game(df1)
```

Arguments

`df1`	Should be a Data Frame that represents the individual statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.

Details

The calculation is made with the number of games played by the player.

Value

Data frame with individual statistics per game

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"),"G" = c(67,0),
"GS" = c(62,0),"MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0),"3P" = c(148,0),"3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),"2P" = c(495,0),"2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0),"FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),"AST" = c(684,0),"STL" = c(78,0),"BLK" = c(36,0),
"TOV" = c(261,0),"PF" = c(118,0),"PTS" = c(1698,0),"+/-" = c(0,0))

individuals_stats_per_game(df1)
```

`individuals_stats_per_minutes` *individual statistics calculator per minutes*

Description

The function allows the calculation of the statistics per game projected to M minutes.

Usage

```
individuals_stats_per_minutes(df1, m)
```

Arguments

`df1`	Should be a Data Frame that represents the individual statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.
`m`	Should be a number. This parameter has to be the number of minutes to which you want to project the statistics.

Details

The statistical projection is made from the relationship between the number of minutes entered and the number of minutes played by the player.

Value

Data frame with statistics by game projected to the minutes entered.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"),"G" = c(67,0),
"GS" = c(62,0),"MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0),"3P" = c(148,0),"3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),"2P" = c(495,0),"2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0),"FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),"AST" = c(684,0),"STL" = c(78,0),"BLK" = c(36,0),
"TOV" = c(261,0),"PF" = c(118,0),"PTS" = c(1698,0),"+/-" = c(0,0))
```

```
m <- 48

individuals_stats_per_minutes(df1,m)
```

`individuals_stats_per_possesion` *individual statistics calculator per possessions*

Description

The function allows the calculation of the statistics per game projected to P possesions.

Usage

```
individuals_stats_per_possesion(df1, df2, df3, p, m)
```

Arguments

`df1`	Should be a Data Frame that represents the individual statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.
`df2`	Should be a Data Frame that represents the team's statistics. The parameter has to be in the format provided by the team_stats() function.
`df3`	Should be a Data Frame that represents the rival's statistics. The parameter has to be in the format provided by the team_stats() function.
`p`	Should be a number. This parameter has to be the number of possessions to which you want to project the statistics.
`m`	should be a number. This parameter has to be the duration of a single game.

Details

The statistical projection is made from the estimation of the possessions that the team plays when the player is on the court.

Value

Data frame with statistics by game projected to the possessions entered

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"),"G" = c(67,0),
"GS" = c(62,0),"MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0),"3P" = c(148,0),"3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),"2P" = c(495,0),"2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0),"FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),"AST" = c(684,0),"STL" = c(78,0),"BLK" = c(36,0),
"TOV" = c(261,0),"PF" = c(118,0),"PTS" = c(1698,0),"+/-" = c(0,0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48),"3P" = c(782),"3PA" = c(2242),
"Percentage 3P" = c(0.349),"2P" = c(2224), "2PA" = c(4027),
"Percentage 2P" = c(0.552),"FT" = c(1260),"FTA FG" = c(1728),
"Percentage FT" = c(0.729), "ORB" = c(757),  "DRB" = c(2490),
"TRB" = c(3247), "AST" = c(1803),  "STL" = c(612),"BLK" = c(468),
"TOV" = c(1077),"PF" = c(1471),  "PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))


p <- 100

m <- 48

individuals_stats_per_possesion(df1,df2,df3,p,m)
```

`lineups_advance_stats` *Lineups advanced statistics*

Description

This function allows the calculation of advanced player statistics.

Usage

```
lineups_advance_stats(df1, m)
```

Arguments

df1	Should be a Data Frame. This parameter has to be in the format provided by the lineups_advance_stats() function.
m	should be a number. This parameter has to be the duration of a single game.

Details

The function only works with the extended statistics of the lineups.

Value

Data frame with the following advanced statistics calculated

- Offensive Rating (ORtg)
- Defensive Rating (DRtg)
- Net Rating (NetRtg)
- Pace (Pace)
- Three rating (3Par)
- True shooting percentage (TS%)
- Efficiency Field Goals percentage (eFG%)
- Assists percentage (AST%)
- Offensive rebounds percentage (ORB%)
- Defensive rebounds percentage (DRB%)
- Total rebounds percentage (TRB%)
- Turnover percentage (TOV%)

Examples

```
df1 <-  data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P" = c(1,0),"X3PA" = c(4,0),"Opp3PA" = c(3,0),
"X2P" = c(4,0),"Opp2P " = c(5,0), "X2PA " = c(6,0),"Opp2PA " = c(8,0) ,
"FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),"OppFTA " = c(1,0),
"OppRB " = c(2,0),"OppOppRB " = c(1,0), "DRB" = c(4,0),"OppDRB" = c(1,0),
"TRB" = c(6,0),"OppTRB" = c(2,0), "AST " = c(5,0),"OppAST " = c(4,0),
"STL " = c(1,0),"OppSTL " = c(3,0), "BLK " = c(0,0),  "OppBLK " = c(1,0),
"TOppV " = c(5,2), "OppTOppV " = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),
"PLUS" = c(15,0),"MINUS" = c(14,3),"P/M" = c(1,-3))

m <- 48

lineups_advance_stats(df1,m)
```

lineups_backcourt *Statistics searcher of backcourt players*

Description

The function allows find the statisticts of backcourt players

Usage

```
lineups_backcourt(df1)
```

Arguments

df1	Should be a Data Frame. The parameter has to be in the format provided by the lineups_data_adjustment() function.

Details

The function works with the basic statistics of the lineups and the extended statistics of the lineups.

Value

Data frame with the statistics of the backcourt players

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

lineups_backcourt(df1)

df1 <-  data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
```

```
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P" = c(1,0),"X3PA" = c(4,0),"Opp3PA" = c(3,0),
"X2P" = c(4,0),"Opp2P " = c(5,0), "X2PA " = c(6,0),"Opp2PA " = c(8,0) ,
"FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),"OppFTA " = c(1,0),
"OppRB " = c(2,0),"OppOppRB " = c(1,0), "DRB" = c(4,0),"OppDRB" = c(1,0),
"TRB" = c(6,0),"OppTRB" = c(2,0), "AST " = c(5,0),"OppAST " = c(4,0),
"STL " = c(1,0),"OppSTL " = c(3,0), "BLK " = c(0,0),  "OppBLK " = c(1,0),
"TOppV " = c(5,2), "OppTOppV " = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),
"PLUS" = c(15,0),"MINUS" = c(14,3),"P/M" = c(1,-3))

lineups_backcourt(df1)
```

`lineups_comparator_stats` *Lineups statistics comparator*

Description

The function allows the comparison of a lineup when it is in the court with the statistics of the rival

Usage

```
lineups_comparator_stats(df1, m)
```

Arguments

`df1`	Should be a Data Frame. The parameter has to be in the format provided by the lineups_data_adjustment() function.
`m`	should be a number. This parameter has to be the duration of a single game.

Details

The function only works with the extended statistics of the lineups.

Value

Data frame with the comparison of statistics and the following values:

- Lineup usage percentage (Team%)
- Pace (Pace)
- Three rating (3Par)
- True shooting percentage (TS%)
- Efficiency Field Goals percentage (eFG%)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <-  data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P" = c(1,0),"X3PA" = c(4,0),"Opp3PA" = c(3,0),
"X2P" = c(4,0),"Opp2P " = c(5,0), "X2PA " = c(6,0),"Opp2PA " = c(8,0) ,
"FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),"OppFTA " = c(1,0),
"OppRB " = c(2,0),"OppOppRB " = c(1,0), "DRB" = c(4,0),"OppDRB" = c(1,0),
"TRB" = c(6,0),"OppTRB" = c(2,0), "AST " = c(5,0),"OppAST " = c(4,0),
"STL " = c(1,0),"OppSTL " = c(3,0), "BLK " = c(0,0),  "OppBLK " = c(1,0),
"TOppV " = c(5,2), "OppTOppV " = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),
"PLUS" = c(15,0),"MINUS" = c(14,3),"P/M" = c(1,-3))

m <- 48

lineups_comparator_stats(df1,m)
```

---

lineups_data_adjustment

Lineups data adjustment

Description

The function transform the statistics entered for later use in the rest of the functions that apply to lineup statistics.

Usage

```
lineups_data_adjustment(df1)
```

Arguments

df1	Should be a Data Frame

Details

- The data.frame must have the same columns and these represent the same as in the example.
- The function allows the transformation of the basic statistics of the lineups to which the shooting percentages, the total rebounds and the plus minus.
- The function allows the transformation of the extended statistics of the lineups to which the total rebounds and the plus minus.

Value

The data frame obtained for the basic statistics of lineups will have the following format:

- Point Guard (PG)
- Shooting Guard (SG)
- Small Forward (SF)
- Paint Forward (PF)
- Center (C)
- Games played (G)
- Games Started (GS)
- Minutes Played (MP)
- Field Goals Made (FG)
- Field Goals Attempted (FGA)
- Field Goals Percentage (FG
- Three Points Made (3P)
- Three Points Attempted (3PA)
- Three Points Percentage (3P%)
- Two Points Made (2P)
- Two Points Attempted (2PA)
- Free Throw Made (FT)
- Offensive Rebounds (ORB)
- Defensive Rebounds (DRB)
- Total Rebounds (TRB)
- Assists (AST)
- Steals (STL)
- Blocks (BLK)
- Turnover (TOV)
- Personal Fouls (PF)
- Points (PTS)
- Plus (+)
- Minus (-)
- Plus Minus (+/-)

For the extended statistics of the lineups it will have the same format as the basic statistics of the lineups but adding the statistics of the opponent against that lineups.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("PG"= c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA"  = c(7,0), "X3P" = c(0,0),"X3PA" = c(2,0),"X2P" = c(4,0),
"X2PA" = c(5,0),  "FT" = c(1,0), "FTA" = c(3,0),"ORB" = c(2,0),
"DRB" = c(5,0), "AST " = c(2,0), "STL " = c(1,0), "BLK " = c(0,0),
"TOV " = c(7,2), "PF" = c(1,0),  "PLUS" = c(9,0),"MINUS" = c(17,3))

lineups_data_adjustment(df1)

df1 <-  data.frame("PG" = c("James","Rondo"),"SG"= c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P  " = c(1,0),"X3PA " = c(4,0),
"Opp3PA  Ã§" = c(3,0),"X2P" = c(4,0),"Opp2P" = c(5,0),"X2PA " = c(6,0),
"Opp2PA" = c(8,0) , "FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),
"OppFTA" = c(1,0),"OppRB" = c(2,0),"OppOppRB" = c(1,0),"DRB" = c(4,0),
"OppDRB" = c(1,0),"AST " = c(5,0),"OppAST " = c(4,0),"STL" = c(1,0),
"OppSTL" = c(3,0),"BLK" = c(0,0),"OppBLK" = c(1,0),"TOppV" = c(5,2),
"OppTOppV" = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),"PLUS" = c(15,0),
"MINUS" = c(14,3))


lineups_data_adjustment(df1)
```

---

lineups_games_adder *Lineups games adder*

Description

The function allows to perform the sums of two data.frames with the same format adopted after being transformed by lineups_data_adjustment() function,

Usage

```
lineups_games_adder(df1, df2)
```

Arguments

df1	Should be a Data Frame that represents the first set of basic or extener lineups statistics. It has to be in the format provided by the lineups_data_adjustment() function.
df2	Should be a Data Frame that represents the second set of basic or extener lineups statistics. It has to be in the format provided by the lineups_data_adjustment() function.

Details

- The function will work correctly when the name of the players is the same, in case it is different it will take the lineups as different.
- The function sums data sets that have an identical size,

Value

Data frame with the sum of the statistics of the other two entered data.frame.

Examples

```
df1 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

df2 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

lineups_games_adder(df1,df2)
```

lineups_paint *Statistics searcher of paint players*

Description

The function allows find the statisticts of paint players

Usage

```
lineups_paint(df1)
```

Arguments

df1	Should be a Data Frame. The parameter has to be in the format provided by the lineups_data_adjustment() function.

Details

The function works with the basic statistics of the lineups and the extended statistics of the lineups.

Value

Data frame with the statistics of the paint players

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

lineups_paint(df1)

df1 <-  data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
```

```
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P" = c(1,0),"X3PA" = c(4,0),"Opp3PA" = c(3,0),
"X2P" = c(4,0),"Opp2P " = c(5,0), "X2PA " = c(6,0),"Opp2PA " = c(8,0) ,
"FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),"OppFTA " = c(1,0),
"OppRB " = c(2,0),"OppOppRB " = c(1,0), "DRB" = c(4,0),"OppDRB" = c(1,0),
"TRB" = c(6,0),"OppTRB" = c(2,0), "AST " = c(5,0),"OppAST " = c(4,0),
"STL " = c(1,0),"OppSTL " = c(3,0), "BLK " = c(0,0),  "OppBLK " = c(1,0),
"TOppV " = c(5,2), "OppTOppV " = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),
"PLUS" = c(15,0),"MINUS" = c(14,3),"P/M" = c(1,-3))

lineups_paint(df1)
```

`lineups_players` *Statistics by position*

Description

The function allows you to search for statistics by position within the lineup.

Usage

```
lineups_players(df1, n)
```

Arguments

`df1`	Should be a Data Frame. The parameter has to be in the format provided by the lineups_data_adjustment() function.
`n`	Should be a number. It represents the position on which you want to perform the search.

Details

The function allows you to search for paint players both in basic statistics and in extended statistics. The supported values for n are as follows:

- If the value entered for n is 1, it will return the statistics of the grouped Point Guards.
- If the value entered for n is 2, it will return the statistics of the grouped Small Guards.
- If the value entered for n is 3, it will return the statistics of the grouped Small Forwards.
- If the value entered for n is 4, it will return the statistics of the grouped Paint Forwards.
- If the value entered for n is 5, it will return the statistics of the grouped Centers.

Value

Data frame with the statistics by position.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

 n <- 1

lineups_players(df1,n)

df1 <-  data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P" = c(1,0),"X3PA" = c(4,0),"Opp3PA" = c(3,0),
"X2P" = c(4,0),"Opp2P " = c(5,0), "X2PA " = c(6,0),"Opp2PA " = c(8,0) ,
"FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),"OppFTA " = c(1,0),
"OppRB " = c(2,0),"OppOppRB " = c(1,0), "DRB" = c(4,0),"OppDRB" = c(1,0),
"TRB" = c(6,0),"OppTRB" = c(2,0), "AST " = c(5,0),"OppAST " = c(4,0),
"STL " = c(1,0),"OppSTL " = c(3,0), "BLK " = c(0,0),  "OppBLK " = c(1,0),
"TOppV " = c(5,2), "OppTOppV " = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),
"PLUS" = c(15,0),"MINUS" = c(14,3),"P/M" = c(1,-3))

n <- 5

lineups_players(df1,n)
```

---

lineups_searcher *Statistics searcher*

Description

The function allows the statistical search of the lineups where the entered players appear.

Usage

```
lineups_searcher(df1, n, p1, p2, p3, p4)
```

Arguments

df1	Should be a Data Frame. This parameter has to be in the format provided by the lineups_advance_stats() function.
n	Should be a numer. It represents the number of player to be found.
p1	Should be a String. Represents the name of the first player to be found.
p2	Should be a String. Represents the name of the second player to be found.
p3	Should be a String. Represents the name of a player to be found.
p4	Should be a String. Represents the name of a player to be found.

Details

- The function allows you to search for paint players both in basic statistics and in extended statistics.
- The values allowed by n are 1, 2, 3 and 4.The number entered in N must be equal to the number of players searched.
- The name entered in the function must be the same as the one inside the data frame.

Value

Data frame with the statistics of the lineups where the entered players appear

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

n <- 2
```

```
p1 <- "James"

p2 <- "Davis"

p3 <- ""

p4 <- ""


lineups_searcher(df1,n,p1,p2,p3,p4)
```

`lineups_separator` *Statistics separator*

Description

The function allows to separate the extended statistics of the lineups. Therefore, you can obtain the statistics of the lineups or the rival with respect to the lineups for later analysis

Usage

```
lineups_separator(df1, n)
```

Arguments

`df1`	Should be a Data Frame. The parameter has to be in the format provided by the lineups_data_adjustment() function.
`n`	Should be a number. it Represents the statistics we want to obtain.

Details

The function only works with the extended statistics of the lineups. The supported values for n are as follows:

- If n takes the value of 1, the function will return the statistics of the lineup.
- If n takes the value of 2, the function will return the statistics of the rival with respect to the lineup.

Value

Data frame with the statistics separated in the format of the basic statistics of the lineups.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henare

Examples

```
df1 <-  data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(6,0),
"OppFG " = c(6,0), "FGA " = c(10,0),"OppFGA " = c(9,0),
"X3P  " = c(2,0),"Opp3P" = c(1,0),"X3PA" = c(4,0),"Opp3PA" = c(3,0),
"X2P" = c(4,0),"Opp2P " = c(5,0), "X2PA " = c(6,0),"Opp2PA " = c(8,0) ,
"FT " = c(0,0),"OppFT " = c(1,0), "FTA " = c(0,0),"OppFTA " = c(1,0),
"OppRB " = c(2,0),"OppOppRB " = c(1,0), "DRB" = c(4,0),"OppDRB" = c(1,0),
"TRB" = c(6,0),"OppTRB" = c(2,0), "AST " = c(5,0),"OppAST " = c(4,0),
"STL " = c(1,0),"OppSTL " = c(3,0), "BLK " = c(0,0),  "OppBLK " = c(1,0),
"TOppV " = c(5,2), "OppTOppV " = c(3,2),"PF" = c(1,0),"OppPF" = c(3,0),
"PLUS" = c(15,0),"MINUS" = c(14,3),"P/M" = c(1,-3))

n <- 1

lineups_separator(df1,n)

n <- 2

lineups_separator(df1,n)
```

`lineups_stats_per_possesion` *Lineups stats per possesion*

Description

The function do the calculation of statistics per p possesion for the differents lineups

Usage

```
lineups_stats_per_possesion(df1, df2, df3, p, m)
```

Arguments

df1	Should be a Data Frame. This parameter has to be in the format provided by the lineups_advance_stats() function.
df2	Should be a Data Frame that represents the team's statistics. The parameter has to be in the format provided by the team_stats() function.
df3	Should be a Data Frame that represents the rival's statistics. The parameter has to be in the format provided by the team_stats() function.
p	Should be a number. This parameter has to be the number of possessions to which you want to project the statistics.
m	should be a number. This parameter has to be the duration of a single game.

Details

- The function only works with the basic statistics of the lineups.
- The statistical projection is made from the estimation of the possessions that the team plays when the lineups is on the court.

Value

Data frame whit statistics per p possesion

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("PG" = c("James","Rondo"),"SG" = c("Green","Caruso"),
"SF" = c("Caldwell","Kuzma"), "PF" = c("Davis","Davis"),
"C" = c("Howard ","Howard"),"MP" = c(7,1), "FG " = c(4,0),
"FGA " = c(7,0),"Percentage FG" = c(0.571,0),
"X3P  " = c(0,0),"X3PA  " = c(2,0),"Percentage 3P" = c(0,0),
"X2P " = c(4,0), "X2PA " = c(5,0), "Percentage 2P" = c(0.8,0),
"FT " = c(1,0), "FTA " = c(3,0), "Percentage FT" = c(0.333,0),
"ORB " = c(2,0), "DRB " = c(5,0),"TRB " = c(7,0), "AST " = c(2,0),
"STL " = c(1,0), "BLK " = c(0,0),"TOV " = c(7,2), "PF" = c(1,0),
"PLUS" = c(9,0),"MINUS" = c(17,3),"P/M" = c(-8,-3))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48),"3P" = c(782),"3PA" = c(2242),
"Percentage 3P" = c(0.349),"2P" = c(2224), "2PA" = c(4027),
"Percentage 2P" = c(0.552),"FT" = c(1260),"FTA FG" = c(1728),
"Percentage FT" = c(0.729), "ORB" = c(757),  "DRB" = c(2490),
"TRB" = c(3247), "AST" = c(1803),  "STL" = c(612),"BLK" = c(468),
"TOV" = c(1077),"PF" = c(1471),  "PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))


p <- 100

m <- 48
```

```
lineups_stats_per_possesion(df1,df2,df3,p,m)
```

`play_advance_stats` *Play advanced statistics*

Description

This function allows the calculation of advanced play statistics.

Usage

```
play_advance_stats(df1)
```

Arguments

`df1`	Should be a Data Frame that represents the play's statistics. The parameter has to be in the format provided by the play_data_adjustment() function.

Value

Data frame with the following advanced statistics calculated:

- Points Per Possession (PPP)
- Possessions (POSS)
- Frequency (Freq)
- Efficiency Field Goals percentage (eFG%)
- Free Throw Percentage (FT%)
- Assists percentage (AST%)
- Turnover percentage (TOV%)
- And One percentage (AndOne%)
- Score percentage (Score%)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"FGA Percentage" = c(0.552,1),"3P" = c(6,1),"3PA" = c(18,1),
"3P Percentage" = c(0.333,1),"2P" = c(149,0),"2PA" = c(263,0),
"2P Percentage" = c(0.567,0),"FT" = c(39,1),  "FTA" = c(53,1),
"FT Percentage" = c(0.736,1),  "ANDONE" = c(12,1), "AST" = c(0,1),
 "TOV" = c(27,1))

play_advance_stats(df1)
```

---

play_data_adjustment *Play data adjustment*

Description

The function transform the statistics entered for later use in the rest of the functions that apply to play statistics.

Usage

```
play_data_adjustment(df1)
```

Arguments

df1	Should be a Data Frame that represents the play's statistics. The parameter has to be in the format provided by the play_data_adjustment() function.

Details

- The data.frame must have the same columns and these represent the same as in the example.
- The input data.frame must have the last row that represents the team's statistics.
- The function allows the transformation of the play statistics to which the shooting percentages.

Value

The data frame obtained for the play statistics will have the following format:

- Name of the player (Name)
- Games Started (GS)
- Points (PTS)
- Field Goals Made (FG)
- Field Goals Attempted (FGA)

- Field Goals Percentage (FG%)
- Three Points Made (3P)
- Three Points Attempted (3PA)
- Three Points Percentage (3P%)
- Two Points Made (2P)
- Two Points Attempted (2PA)
- Two Points Percentage (2P%)
- Free Throw Made (FT)
- Free Throw Attempted (FTA)
- Free Throw Percentage (FT%)
- And One Times (ANDONE)
- Assists (AST)
- Turnover (TOV)

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"3P" = c(6,1),"3PA" = c(18,1), "FT" = c(39,1),  "FTA" = c(53,1),
"ANDONE" = c(12,1), "AST" = c(0,1), "TOV" = c(27,1))

play_data_adjustment(df1)
```

---

play_games_adder *Play games adder*

Description

The function allows to perform the sums of two data.frames with the same format adopted after being transformed by play_data_adjustment() function,

Usage

```
play_games_adder(df1, df2)
```

Arguments

df1	Should be a Data Frame that represents the first set of play's statistics. It has to be in the format provided by the play_data_adjustment() function.
df2	Should be a Data Frame that represents the second set of play's statistics. It has to be in the format provided by the play_data_adjustment() function.

Details

The function will work correctly when the name of the players is the same, in case it is different it will take the players as different.

Value

Data frame with the sum of the statistics of the other two entered data frame.

Examples

```
df1 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"FGA Percentage" = c(0.552,1),"3P" = c(6,1),"3PA" = c(18,1),
"3P Percentage" = c(0.333,1),"2P" = c(149,0),"2PA" = c(263,0),
"2P Percentage" = c(0.567,0),"FT" = c(39,1),  "FTA" = c(53,1),
"FT Percentage" = c(0.736,1),  "ANDONE" = c(12,1), "AST" = c(0,1),
"TOV" = c(27,1))

df2 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"FGA Percentage" = c(0.552,1),"3P" = c(6,1),"3PA" = c(18,1),
"3P Percentage" = c(0.333,1),"2P" = c(149,0),"2PA" = c(263,0),
"2P Percentage" = c(0.567,0),"FT" = c(39,1),  "FTA" = c(53,1),
"FT Percentage" = c(0.736,1),  "ANDONE" = c(12,1), "AST" = c(0,1),
"TOV" = c(27,1))

play_games_adder(df1,df2)
```

---

play_stats_per_game *Play stats per game*

Description

The function allows the calculation of play statistics per game.

Usage

```
play_stats_per_game(df1)
```

Arguments

df1	Should be a Data Frame that represents the play's statistics. The parameter has to be in the format provided by the play_data_adjustment() function.

Details

The calculation is made with the number of games played by the player.

Value

Data frame with play statistics per game

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"FGA Percentage" = c(0.552,1),"3P" = c(6,1),"3PA" = c(18,1),
"3P Percentage" = c(0.333,1),"2P" = c(149,0),"2PA" = c(263,0),
"2P Percentage" = c(0.567,0),"FT" = c(39,1),  "FTA" = c(53,1),
"FT Percentage" = c(0.736,1),  "ANDONE" = c(12,1), "AST" = c(0,1),
"TOV" = c(27,1))

play_stats_per_game(df1)
```

---

play_stats_per_possesion

Play stats per possesion

Description

The function allows the calculation of the statistics per game projected to P possesions.

Usage

```
play_stats_per_possesion(df1, df2, df3, p, m)
```

Arguments

df1	Should be a Data Frame that represents the play's statistics. The parameter has to be in the format provided by the play_data_adjustment() function.
df2	Should be a Data Frame that represents the team's statistics. The parameter has to be in the format provided by the team_stats() function.
df3	Should be a Data Frame that represents the rival's statistics. The parameter has to be in the format provided by the team_stats() function.
p	Should be a number. This parameter has to be the number of possessions to which you want to project the statistics.
m	should be a number. This parameter has to be the duration of a single game.

Details

The statistical projection is made from the estimation of the possessions that the team plays when the player is on the court.

Value

Data frame with statistics by game projected to the possesions entered

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"FGA Percentage" = c(0.552,1),"3P" = c(6,1),"3PA" = c(18,1),
"3P Percentage" = c(0.333,1),"2P" = c(149,0),"2PA" = c(263,0),
"2P Percentage" = c(0.567,0),"FT" = c(39,1),  "FTA" = c(53,1),
"FT Percentage" = c(0.736,1),  "ANDONE" = c(12,1), "AST" = c(0,1),
"TOV" = c(27,1))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48), "3P" = c(782),
"3PA" = c(2242), "Percentage 3P" = c(0.349),  "2P" = c(2224),
"2PA" = c(4027), "Percentage 2P" = c(0.552), "FT" = c(1260),
"FTA FG" = c(1728),  "Percentage FT" = c(0.729), "ORB" = c(757),
"DRB" = c(2490),"TRB" = c(3247),"AST" = c(1803),"STL" = c(612),
"BLK" = c(468),   "TOV" = c(1077),  "PF" = c(1471),
"PTS" = c(8054),  "+/-" = c(0))

df3 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
```

```
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))

p<- 100

m <- 48

play_stats_per_possesion(df1,df2,df3,p,m)
```

`play_team_stats` *Team play statistics*

Description

The function performs the sum of the statistical sections to obtain a data.frame with the total statistics of the team.

Usage

```
play_team_stats(df1)
```

Arguments

`df1`	Should be a Data Frame that represents the play's statistics. This parameter has to be in the format provided by the play_data_adjustment() function.

Value

Data frame whit the team's plays statistics

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("Name" = c("Sabonis ","Team"), "GP" = c(62,71),
"PTS" = c(387,0), "FG" = c(155,1), "FGA" = c(281,1),
"FGA Percentage" = c(0.552,1),"3P" = c(6,1),"3PA" = c(18,1),
"3P Percentage" = c(0.333,1),"2P" = c(149,0),"2PA" = c(263,0),
"2P Percentage" = c(0.567,0),"FT" = c(39,1),  "FTA" = c(53,1),
"FT Percentage" = c(0.736,1),  "ANDONE" = c(12,1), "AST" = c(0,1),
```

```
"TOV" = c(27,1))

play_team_stats(df1)
```

`team_advanced_stats` *Team advanced statistics*

Description

This function allows the calculation of advanced team's statistics.

Usage

```
team_advanced_stats(df1, df2, m)
```

Arguments

`df1`	Should be a Data Frame that represents the team statistics. This parameter has to be in the format provided by the team_stats() function.
`df2`	Should be a Data Frame that represents the rival statistics. This parameter has to be in the format provided by the team_stats() function.
`m`	should be a number. This parameter has to be the duration of a single game.

Value

Data frame with the following advanced statistics calculated:

- Offensive Rating (ORtg)
- Defensive Rating (DRtg)
- Net Rating (NetRtg)
- Pace (Pace)
- Three rating (3Par)
- True shooting percentage (TS%)
- Efficiency Field Goals percentage (eFG%)
- Assists percentage (AST%)
- Assist to Turnover Ratio (AST/TO)
- Assist Ratio (ASTRATIO)
- Offensive rebounds percentage (ORB%)
- Defensive rebounds percentage (DRB%)
- Total rebounds percentage (TRB%)
- Turnover percentage (TOV%)
- Free Throw rating (FTr)

- Opponent Efficiency Field Goals percentage (Opp eFG%)
- Opponent Turnover percentage (Opp TOV%)
- Opponent Defensive rebounds percentage (Opp DRB%)
- Opponent Free Throw rating (Opp FTr)

Examples

```
df1 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48), "3P" = c(782),
"3PA" = c(2242), "Percentage 3P" = c(0.349),  "2P" = c(2224),
"2PA" = c(4027), "Percentage 2P" = c(0.552), "FT" = c(1260),
"FTA FG" = c(1728),  "Percentage FT" = c(0.729), "ORB" = c(757),
"DRB" = c(2490),"TRB" = c(3247),"AST" = c(1803),"STL" = c(612),
"BLK" = c(468),   "TOV" = c(1077),  "PF" = c(1471),
"PTS" = c(8054),  "+/-" = c(0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))

m <- 48

team_advanced_stats(df1,df2,m)
```

---

team_stats *Team statistics*

Description

This function allows the team statistics from individual statistics.

Usage

```
team_stats(df1)
```

Arguments

df1	Should be a Data Frame that represents the individual statistics or individual defensive statistics of the players. The parameter has to be in the format provided by the data_adjustment() function.

Details

The function performs the sum of the statistical sections to obtain a data.frame with the total statistics of the team.

Value

Data frame with the sum of the team's statistics.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("name" = c("LeBron James","Team"), "G" = c(67,0),
"GS" = c(62,0), "MP" = c(2316,0),"FG" = c(643,0), "FGA" = c(1303,0),
"Percentage FG" = c(0.493,0), "3P" = c(148,0),  "3PA" = c(425,0),
"Percentage 3P" = c(0.348,0),  "2P" = c(495,0), "2PA" = c(878,0),
"Percentage 2P" = c(0.564,0),"FT" = c(264,0), "FTA FG" = c(381,0),
"Percentage FT" = c(0.693,0), "ORB" = c(66,0),"DRB" = c(459,0),
"TRB" = c(525,0),  "AST" = c(684,0),  "STL" = c(78,0),
"BLK" = c(36,0),"TOV" = c(261,0),  "PF" = c(118,0),
"PTS" = c(1698,0), "+/-" = c(0,0))

team_stats(df1)
```

---

team_stats_per_game *Team stats per game*

Description

The function allows the calculation of team's statistics per game.

Usage

```
team_stats_per_game(df1)
```

Arguments

df1	Should be a Data Frame that represents the team statistics. This parameter has to be in the format provided by the team_stats() function.

Details

The calculation is made with the number of games played by the team.

Value

Data frame whit the team's statistics per game

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48), "3P" = c(782),
"3PA" = c(2242), "Percentage 3P" = c(0.349),  "2P" = c(2224),
"2PA" = c(4027), "Percentage 2P" = c(0.552), "FT" = c(1260),
"FTA FG" = c(1728),  "Percentage FT" = c(0.729), "ORB" = c(757),
"DRB" = c(2490),"TRB" = c(3247),  "AST" = c(1803),  "STL" = c(612),
"BLK" = c(468),   "TOV" = c(1077),  "PF" = c(1471),  "PTS" = c(8054),
"+/-" = c(0))

team_stats_per_game(df1)
```

---

team_stats_per_minutes

Team stats per minutes

Description

The function allows the calculation of the team statistics per game projected to M minutes.

Usage

```
team_stats_per_minutes(df1, m)
```

Arguments

df1	Should be a Data Frame that represents the team statistics. This parameter has to be in the format provided by the team_stats() function.
m	Should be a number. This parameter has to be the number of minutes to which you want to project the statistics.

Details

The statistical projection is made from the relationship between the number of minutes entered and the number of minutes played by the team

Value

Data frame with statistics by game projected to the minutes entered.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48), "3P" = c(782),
"3PA" = c(2242), "Percentage 3P" = c(0.349),  "2P" = c(2224),
"2PA" = c(4027), "Percentage 2P" = c(0.552), "FT" = c(1260),
"FTA FG" = c(1728),  "Percentage FT" = c(0.729), "ORB" = c(757),
"DRB" = c(2490),"TRB" = c(3247),  "AST" = c(1803),  "STL" = c(612),
"BLK" = c(468),   "TOV" = c(1077),  "PF" = c(1471),  "PTS" = c(8054),
"+/-" = c(0))

m <- 48

team_stats_per_minutes(df1,m)
```

---

team_stats_per_possesion

Team stats per possesion

Description

The function allows the calculation of the statistics per game projected to P possesions.

Usage

```
team_stats_per_possesion(df1, df2, p)
```

Arguments

df1	Should be a Data Frame that represents the team statistics. This parameter has to be in the format provided by the team_stats() function.
df2	Should be a Data Frame that represents the rival statistics. This parameter has to be in the format provided by the team_stats() function.
p	Should be a number. This parameter has to be the number of possessions to which you want to project the statistics.

Details

The statistical projection is made from the estimation of the possessions that the team plays.

Value

Data frame with statistics by game projected to the possessions entered.

Author(s)

Fco Javier Cantero <fco.cantero@edu.uah.es>

Juan José Cuadrado <jjcg@uah.es>

Universidad de Alcalá de Henares

Examples

```
df1 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(3006),
"FGA" = c(6269),"Percentage FG" = c(0.48), "3P" = c(782),
"3PA" = c(2242), "Percentage 3P" = c(0.349),  "2P" = c(2224),
"2PA" = c(4027), "Percentage 2P" = c(0.552), "FT" = c(1260),
"FTA FG" = c(1728),  "Percentage FT" = c(0.729), "ORB" = c(757),
"DRB" = c(2490),"TRB" = c(3247),"AST" = c(1803),"STL" = c(612),
"BLK" = c(468),   "TOV" = c(1077),  "PF" = c(1471),
"PTS" = c(8054),  "+/-" = c(0))

df2 <- data.frame("G" = c(71), "MP" = c(17090), "FG" = c(2773),
"FGA" = c(6187),"Percentage FG" = c(0.448), "3P" = c(827),
"3PA" = c(2373), "Percentage 3P" = c(0.349),  "2P" = c(1946),
"2PA" = c(3814), "Percentage 2P" = c(0.510), "FT" = c(1270),
"FTA FG" = c(1626),  "Percentage FT" = c(0.781), "ORB" = c(668),
"DRB" = c(2333),"TRB" = c(3001),  "AST" = c(1662),"STL" = c(585),
"BLK" = c(263),   "TOV" = c(1130),  "PF" = c(1544),
"PTS" = c(7643),  "+/-" = c(0))

p <- 100

team_stats_per_possesion(df1,df2,p)
```

Index