TASK 5: Exploratory Data Analysis - Sports

Yogesh S

2023-06-18

OBJECTIVE 1: Perform Exploratory Data Analysis on dataset “Indian Premier League”.

OBJECTIVE 2: Find out the most Successful teams.

OBJECTIVE 3: Identify players contributing to the team’s success.

OBJECTIVE 4: Suggest Teams and players for companies to endorse their products.

Let us perform Exploratory Data Analysis on the Dataset (Objective 1)

Step 1: Load the new Datasets.

```
setwd("C:/R/R Sample Datasets")
library(readr)
library(ggplot2)
myData<- read_csv("deliveries.csv")
```

```
## Rows: 179078 Columns: 21
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (8): batting_team, bowling_team, batsman, non_striker, bowler, player_d...
## dbl (13): match_id, inning, over, ball, is_super_over, wide_runs, bye_runs, ...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
myData2<- read_csv("matches.csv")
```

```
## Rows: 756 Columns: 18
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (13): city, date, team1, team2, toss_winner, toss_decision, result, winn...
## dbl  (5): id, season, dl_applied, win_by_runs, win_by_wickets
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Step 2: View Datasets.

```
View(myData)
head(myData)
```

match_id <dbl>	inning <dbl>	batting_team <chr>	bowling_team <chr>	o… <dbl>	ball <dbl>	batsman <chr>	non_striker <chr>
1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan
1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan
1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan
1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan
1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan
1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	6	S Dhawan	DA Warner

6 rows | 1-8 of 21 columns

```
View(myData2)
head(myData2)
```

id <dbl>	sea… <dbl>	city <chr>	date <chr>	team1 <chr>	team2 <chr>
1	2017	Hyderabad	2017-04-05	Sunrisers Hyderabad	Royal Challengers Bangalore
2	2017	Pune	2017-04-06	Mumbai Indians	Rising Pune Supergiant
3	2017	Rajkot	2017-04-07	Gujarat Lions	Kolkata Knight Riders
4	2017	Indore	2017-04-08	Rising Pune Supergiant	Kings XI Punjab
5	2017	Bangalore	2017-04-08	Royal Challengers Bangalore	Delhi Daredevils
6	2017	Hyderabad	2017-04-09	Gujarat Lions	Sunrisers Hyderabad

6 rows | 1-6 of 18 columns

Step 3: Check for missing values in the datasets using colSums().

Let us Check for missing values in “MATCHES” dataset.

```
colSums(is.na(myData))
```

```
##         match_id           inning     batting_team     bowling_team
##                0                0                0                0
##             over             ball          batsman      non_striker
##                0                0                0                0
##           bowler    is_super_over        wide_runs         bye_runs
##                0                0                0                0
##      legbye_runs      noball_runs     penalty_runs     batsman_runs
##                0                0                0                0
##       extra_runs      total_runs player_dismissed   dismissal_kind
##                0                0           170244           170244
##          fielder
##           172630
```

The last 3 columns have missing values>90%

METHOD1: Removal of largely inconsistent data

Let us remove these inconsistent data columns using the following code:

```
subs1<-subset(myData, select = -c(player_dismissed,dismissal_kind,fielder))
colSums(is.na(subs1))
```

```
##      match_id        inning  batting_team  bowling_team          over
##             0             0             0             0             0
##          ball       batsman   non_striker        bowler is_super_over
##             0             0             0             0             0
##     wide_runs      bye_runs   legbye_runs   noball_runs  penalty_runs
##             0             0             0             0             0
##  batsman_runs    extra_runs    total_runs
##             0             0             0
```

The dataset is now clear from any missing data.

Now let us Check for missing values in “MATCHES” dataset.

```
colSums(is.na(myData2))
```

```
##              id          season            city            date           team1
##               0               0               7               0               0
##           team2     toss_winner   toss_decision          result      dl_applied
##               0               0               0               0               0
##          winner     win_by_runs  win_by_wickets player_of_match           venue
##               4               0               0               4               0
##         umpire1         umpire2         umpire3
##               2               2             637
```

Remove the columns that have (missing values at random.)

```
subs2<-subset(myData2, select = -c(umpire3))
colSums(is.na(subs2))
```

```
##              id          season            city            date           team1
##               0               0               7               0               0
##           team2     toss_winner   toss_decision          result      dl_applied
##               0               0               0               0               0
##          winner     win_by_runs  win_by_wickets player_of_match           venue
##               4               0               0               4               0
##         umpire1         umpire2
##               2               2
```

METHOD2: Replace value with other column value if NA values are found.

Select subset containing NA values in the city column and select a replacement column.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
test <- subs2 %>% filter(is.na(city),)
select(test,city,venue)
```

city <chr>	venue <chr>
NA	Dubai International Cricket Stadium
NA	Dubai International Cricket Stadium
NA	Dubai International Cricket Stadium
NA	Dubai International Cricket Stadium
NA	Dubai International Cricket Stadium
NA	Dubai International Cricket Stadium
NA	Dubai International Cricket Stadium

7 rows

```
subs2$city <- ifelse(is.na(subs2$city), subs2$venue, subs2$city)
colSums(is.na(subs2))
```

```
##              id          season            city            date           team1
##               0               0               0               0               0
##           team2     toss_winner   toss_decision          result      dl_applied
##               0               0               0               0               0
##          winner     win_by_runs  win_by_wickets player_of_match           venue
##               4               0               0               4               0
##         umpire1         umpire2
##               2               2
```

```
subs2<-na.omit(subs2)
colSums(is.na(subs2))
```

```
##             id         season           city           date          team1
##              0              0              0              0              0
##          team2    toss_winner  toss_decision         result     dl_applied
##              0              0              0              0              0
##         winner    win_by_runs win_by_wickets player_of_match          venue
##              0              0              0              0              0
##        umpire1        umpire2
##              0              0
```

Step 4: Summarize and understand the dataset better before analysis and visualization.

```
str(subs1)
```

```
## tibble [179,078 × 18] (S3: tbl_df/tbl/data.frame)
##  $ match_id     : num [1:179078] 1 1 1 1 1 1 1 1 1 1 ...
##  $ inning       : num [1:179078] 1 1 1 1 1 1 1 1 1 1 ...
##  $ batting_team : chr [1:179078] "Sunrisers Hyderabad" "Sunrisers Hyderabad" "Sunrisers Hyderab
ad" "Sunrisers Hyderabad" ...
##  $ bowling_team : chr [1:179078] "Royal Challengers Bangalore" "Royal Challengers Bangalore" "R
oyal Challengers Bangalore" "Royal Challengers Bangalore" ...
##  $ over         : num [1:179078] 1 1 1 1 1 1 1 2 2 2 ...
##  $ ball         : num [1:179078] 1 2 3 4 5 6 7 1 2 3 ...
##  $ batsman      : chr [1:179078] "DA Warner" "DA Warner" "DA Warner" "DA Warner" ...
##  $ non_striker  : chr [1:179078] "S Dhawan" "S Dhawan" "S Dhawan" "S Dhawan" ...
##  $ bowler       : chr [1:179078] "TS Mills" "TS Mills" "TS Mills" "TS Mills" ...
##  $ is_super_over: num [1:179078] 0 0 0 0 0 0 0 0 0 0 ...
##  $ wide_runs    : num [1:179078] 0 0 0 0 2 0 0 0 0 0 ...
##  $ bye_runs     : num [1:179078] 0 0 0 0 0 0 0 0 0 0 ...
##  $ legbye_runs  : num [1:179078] 0 0 0 0 0 0 1 0 0 0 ...
##  $ noball_runs  : num [1:179078] 0 0 0 0 0 0 0 0 0 1 ...
##  $ penalty_runs : num [1:179078] 0 0 0 0 0 0 0 0 0 0 ...
##  $ batsman_runs : num [1:179078] 0 0 4 0 0 0 0 1 4 0 ...
##  $ extra_runs   : num [1:179078] 0 0 0 0 2 0 1 0 0 1 ...
##  $ total_runs   : num [1:179078] 0 0 4 0 2 0 1 1 4 1 ...
```

```
str(subs2)
```

```
## tibble [750 × 17] (S3: tbl_df/tbl/data.frame)
##  $ id              : num [1:750] 1 2 3 4 6 7 8 9 10 11 ...
##  $ season          : num [1:750] 2017 2017 2017 2017 2017 ...
##  $ city            : chr [1:750] "Hyderabad" "Pune" "Rajkot" "Indore" ...
##  $ date            : chr [1:750] "2017-04-05" "2017-04-06" "2017-04-07" "2017-04-08" ...
##  $ team1           : chr [1:750] "Sunrisers Hyderabad" "Mumbai Indians" "Gujarat Lions" "Rising
Pune Supergiant" ...
##  $ team2           : chr [1:750] "Royal Challengers Bangalore" "Rising Pune Supergiant" "Kolkata
Knight Riders" "Kings XI Punjab" ...
##  $ toss_winner     : chr [1:750] "Royal Challengers Bangalore" "Rising Pune Supergiant" "Kolkata
Knight Riders" "Kings XI Punjab" ...
##  $ toss_decision   : chr [1:750] "field" "field" "field" "field" ...
##  $ result          : chr [1:750] "normal" "normal" "normal" "normal" ...
##  $ dl_applied      : num [1:750] 0 0 0 0 0 0 0 0 0 0 ...
##  $ winner          : chr [1:750] "Sunrisers Hyderabad" "Rising Pune Supergiant" "Kolkata Knight
Riders" "Kings XI Punjab" ...
##  $ win_by_runs     : num [1:750] 35 0 0 0 0 0 0 97 0 0 ...
##  $ win_by_wickets : num [1:750] 0 7 10 6 9 4 8 0 4 8 ...
##  $ player_of_match: chr [1:750] "Yuvraj Singh" "SPD Smith" "CA Lynn" "GJ Maxwell" ...
##  $ venue           : chr [1:750] "Rajiv Gandhi International Stadium, Uppal" "Maharashtra Cricke
t Association Stadium" "Saurashtra Cricket Association Stadium" "Holkar Cricket Stadium" ...
##  $ umpire1         : chr [1:750] "AY Dandekar" "A Nand Kishore" "Nitin Menon" "AK Chaudhary" ...
##  $ umpire2         : chr [1:750] "NJ Llong" "S Ravi" "CK Nandan" "C Shamshuddin" ...
##  - attr(*, "na.action")= 'omit' Named int [1:6] 5 301 546 571 745 754
##   ..- attr(*, "names")= chr [1:6] "5" "301" "546" "571" ...
```

```
summary(subs1)
```

```
##     match_id         inning      batting_team       bowling_team
##  Min.   :    1   Min.   :1.000   Length:179078      Length:179078
##  1st Qu.:  190   1st Qu.:1.000   Class :character   Class :character
##  Median :  379   Median :1.000   Mode  :character   Mode  :character
##  Mean   : 1802   Mean   :1.483
##  3rd Qu.:  567   3rd Qu.:2.000
##  Max.   :11415   Max.   :5.000
##       over            ball         batsman          non_striker
##  Min.   : 1.00   Min.   :1.000   Length:179078      Length:179078
##  1st Qu.: 5.00   1st Qu.:2.000   Class :character   Class :character
##  Median :10.00   Median :4.000   Mode  :character   Mode  :character
##  Mean   :10.16   Mean   :3.616
##  3rd Qu.:15.00   3rd Qu.:5.000
##  Max.   :20.00   Max.   :9.000
##     bowler          is_super_over         wide_runs          bye_runs
##  Length:179078      Min.   :0.0000000   Min.   :0.00000   Min.   :0.000000
##  Class :character   1st Qu.:0.0000000   1st Qu.:0.00000   1st Qu.:0.000000
##  Mode  :character   Median :0.0000000   Median :0.00000   Median :0.000000
##                     Mean   :0.0004523   Mean   :0.03672   Mean   :0.004936
##                     3rd Qu.:0.0000000   3rd Qu.:0.00000   3rd Qu.:0.000000
##                     Max.   :1.0000000   Max.   :5.00000   Max.   :4.000000
##   legbye_runs       noball_runs        penalty_runs      batsman_runs
##  Min.   :0.00000   Min.   :0.000000   Min.   :0.0e+00   Min.   :0.000
##  1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:0.0e+00   1st Qu.:0.000
##  Median :0.00000   Median :0.000000   Median :0.0e+00   Median :1.000
##  Mean   :0.02114   Mean   :0.004183   Mean   :5.6e-05   Mean   :1.247
##  3rd Qu.:0.00000   3rd Qu.:0.000000   3rd Qu.:0.0e+00   3rd Qu.:1.000
##  Max.   :5.00000   Max.   :5.000000   Max.   :5.0e+00   Max.   :7.000
##    extra_runs        total_runs
##  Min.   :0.00000   Min.   : 0.000
##  1st Qu.:0.00000   1st Qu.: 0.000
##  Median :0.00000   Median : 1.000
##  Mean   :0.06703   Mean   : 1.314
##  3rd Qu.:0.00000   3rd Qu.: 1.000
##  Max.   :7.00000   Max.   :10.000
```

```
summary(subs2)
```

```
##        id              season         city              date          
##  Min.   :    1.0   Min.   :2008   Length:750         Length:750        
##  1st Qu.:  189.2   1st Qu.:2011   Class :character   Class :character  
##  Median :  377.5   Median :2013   Mode  :character   Mode  :character  
##  Mean   : 1774.3   Mean   :2013                                        
##  3rd Qu.:  565.8   3rd Qu.:2016                                        
##  Max.   :11415.0   Max.   :2019                                        
##     team1              team2           toss_winner        toss_decision     
##  Length:750         Length:750         Length:750         Length:750        
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##     result            dl_applied         winner           win_by_runs     
##  Length:750         Min.   :0.00000   Length:750         Min.   :  0.00  
##  Class :character   1st Qu.:0.00000   Class :character   1st Qu.:  0.00  
##  Mode  :character   Median :0.00000   Mode  :character   Median :  0.00  
##                     Mean   :0.02533                      Mean   : 13.37  
##                     3rd Qu.:0.00000                      3rd Qu.: 19.00  
##                     Max.   :1.00000                      Max.   :146.00  
##  win_by_wickets   player_of_match       venue             umpire1         
##  Min.   : 0.000   Length:750         Length:750         Length:750        
##  1st Qu.: 0.000   Class :character   Class :character   Class :character  
##  Median : 4.000   Mode  :character   Mode  :character   Mode  :character  
##  Mean   : 3.375                                                           
##  3rd Qu.: 6.000                                                           
##  Max.   :10.000                                                           
##    umpire2         
##  Length:750        
##  Class :character  
##  Mode  :character  
##                    
##                    
##                    
```

Step 5: Analyze the data using Descriptive Statistics.

```
subset2<-filter(subs1,batsman=='MS Dhoni')
subset4<-filter(subs2,winner=='Chennai Super Kings')
mean(subset2$total_runs)
```

```
## [1] 1.413803
```

```
mean(subset4$win_by_runs)
```

```
## [1] 17.78
```

```
var(subset4$win_by_wickets)
```

```
## [1] 10.86657
```

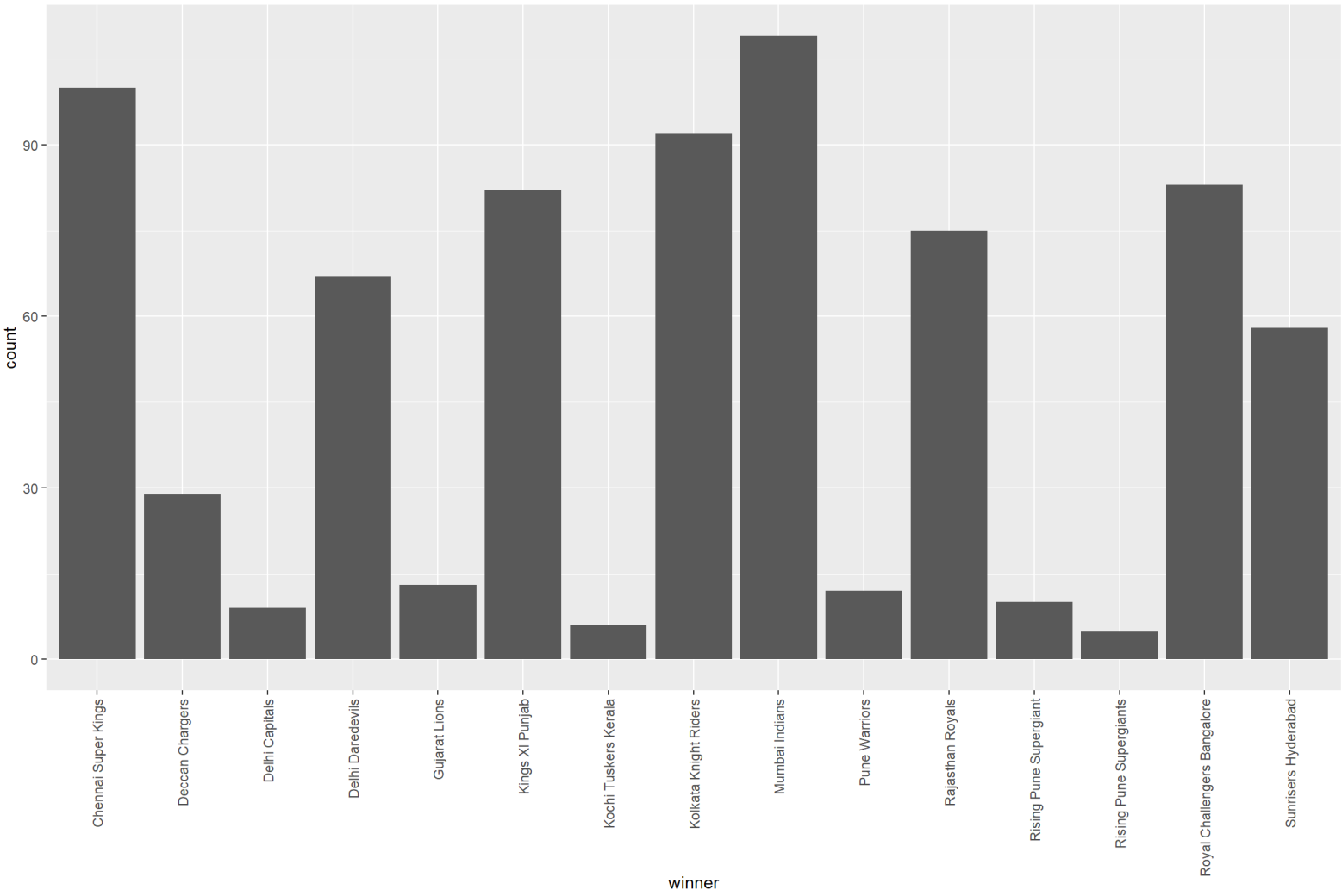
```
median(subset2$total_runs)
```

```
## [1] 1
```

Step 6: VISUALIZATION: We shall start Visualizing the Data when we solve the objectives of the task.

OBJECTIVE 2: Find the most successful teams

```
q<-ggplot(subs2, aes(x=winner)) + geom_bar()
q + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



According to the visualization, the most successful teams in IPL rank in the order:

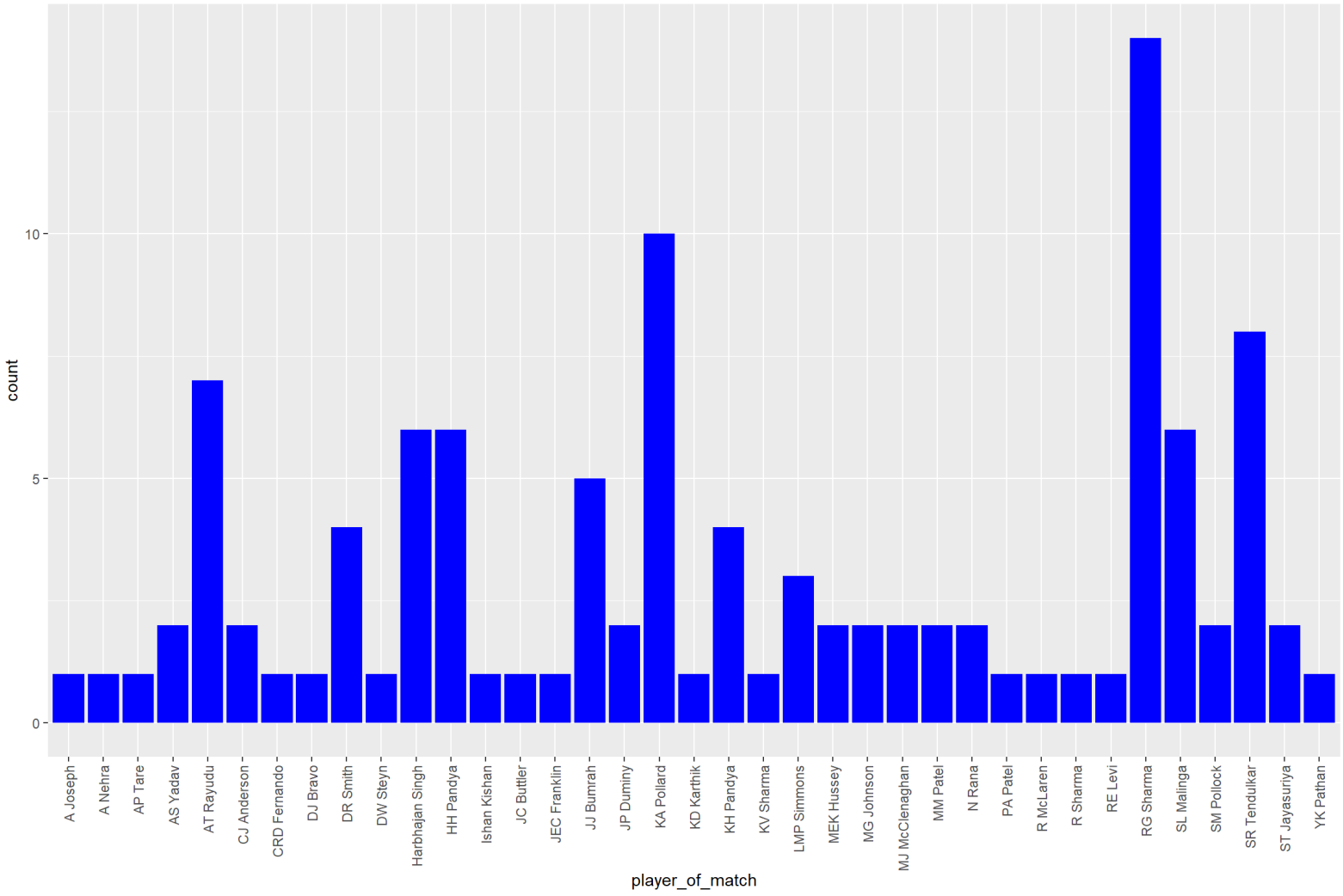
1. Mumbai Indians
2. Chennai Super Kings
3. Kolkata Knight Riders
4. Royal Challengers Bangalore

OBJECTIVE 3: Find the players contributing to the success of each team

Each team is colour coded according to their 'JERSEY'

Mumbai Indians

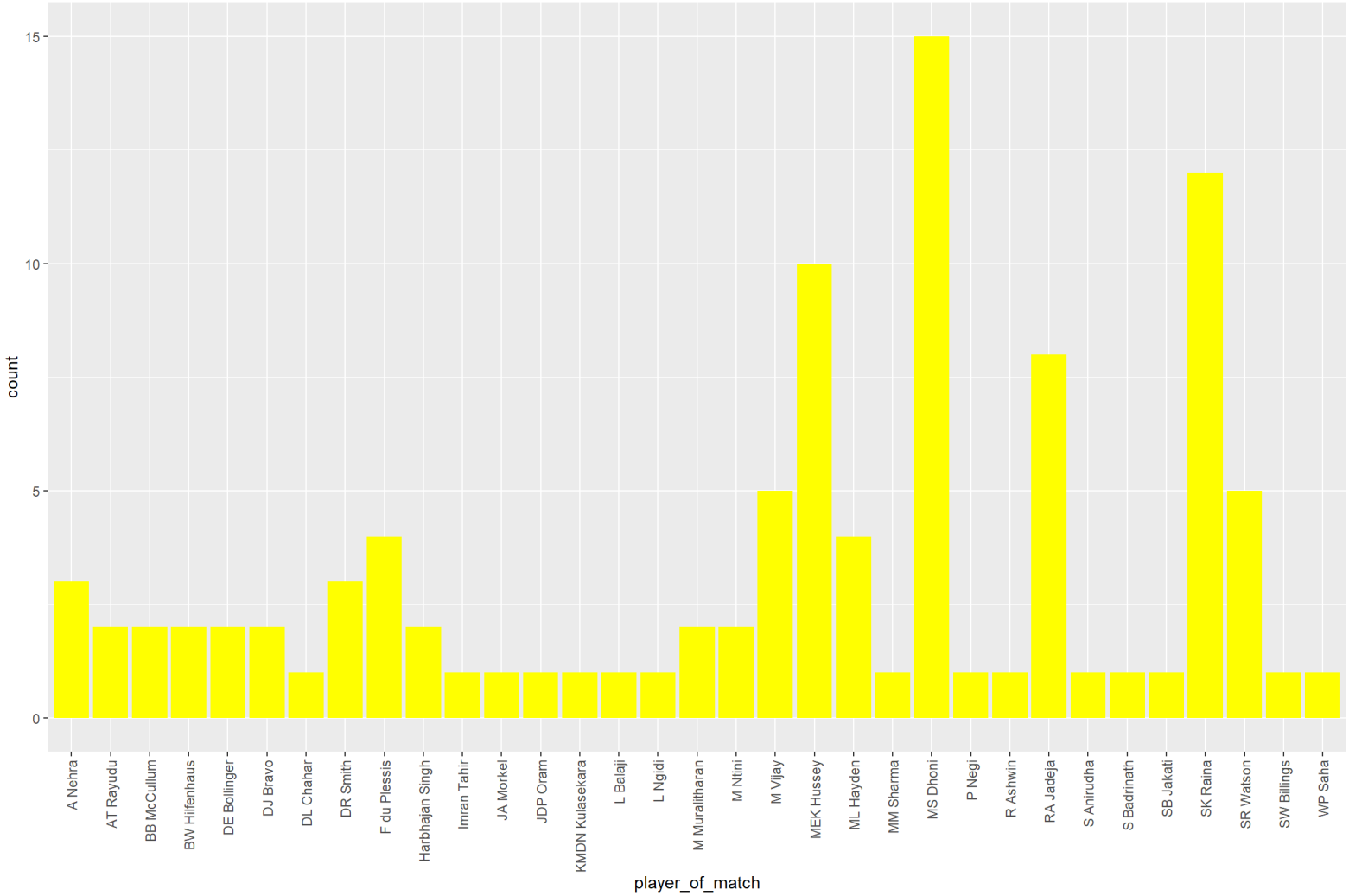
```
MI_POM<-filter(subs2,winner=='Mumbai Indians')
MiMoM<-ggplot(MI_POM, aes(x=player_of_match)) + geom_bar(fill="blue")
MiMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: RG Sharma

Chennai Super Kings

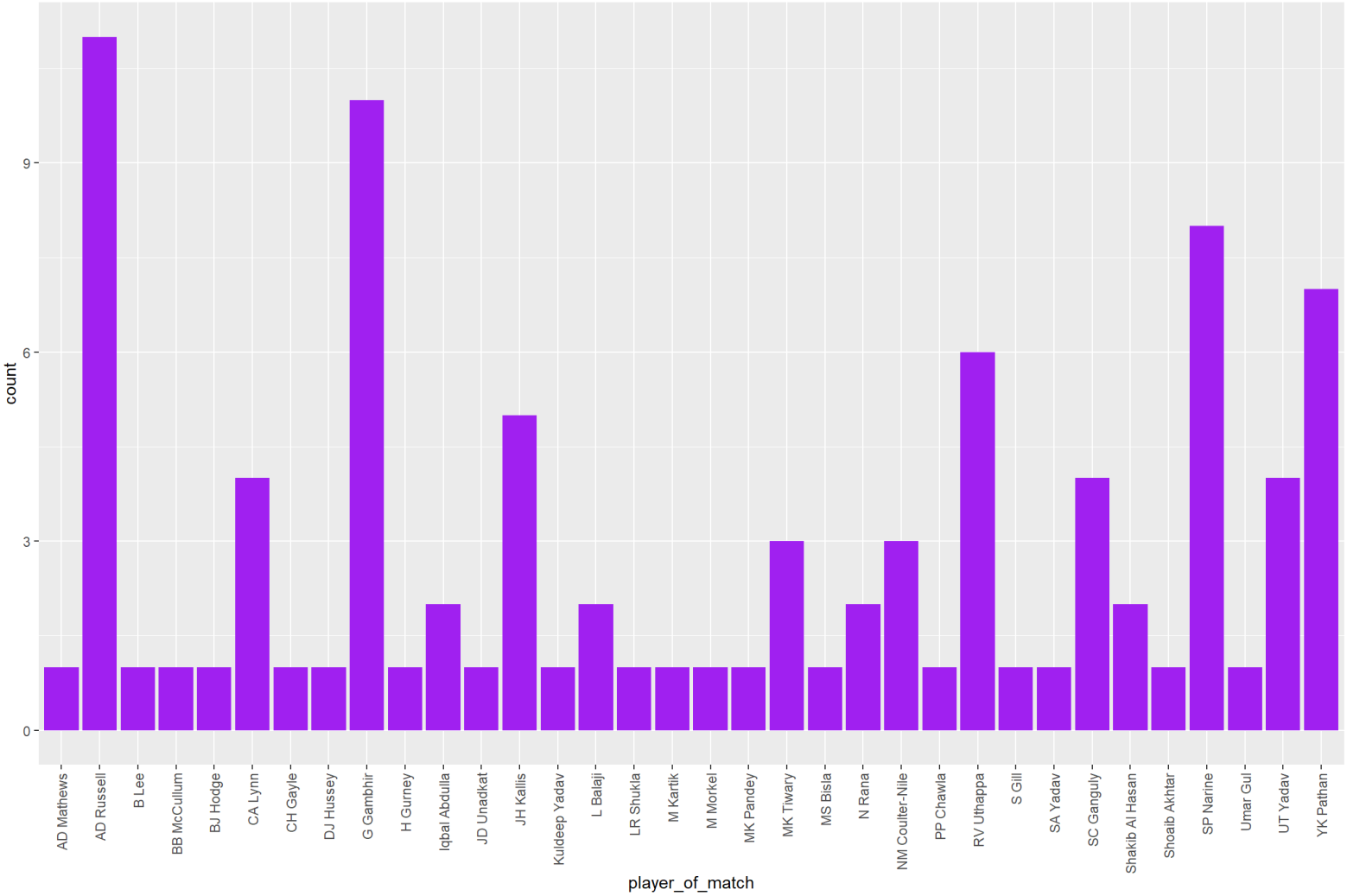
```
CSK_POM<-filter(subs2,winner=='Chennai Super Kings')
CskMoM<-ggplot(CSK_POM, aes(x=player_of_match)) + geom_bar(fill="yellow")
CskMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: MS Dhoni

Kolkata Knight Riders

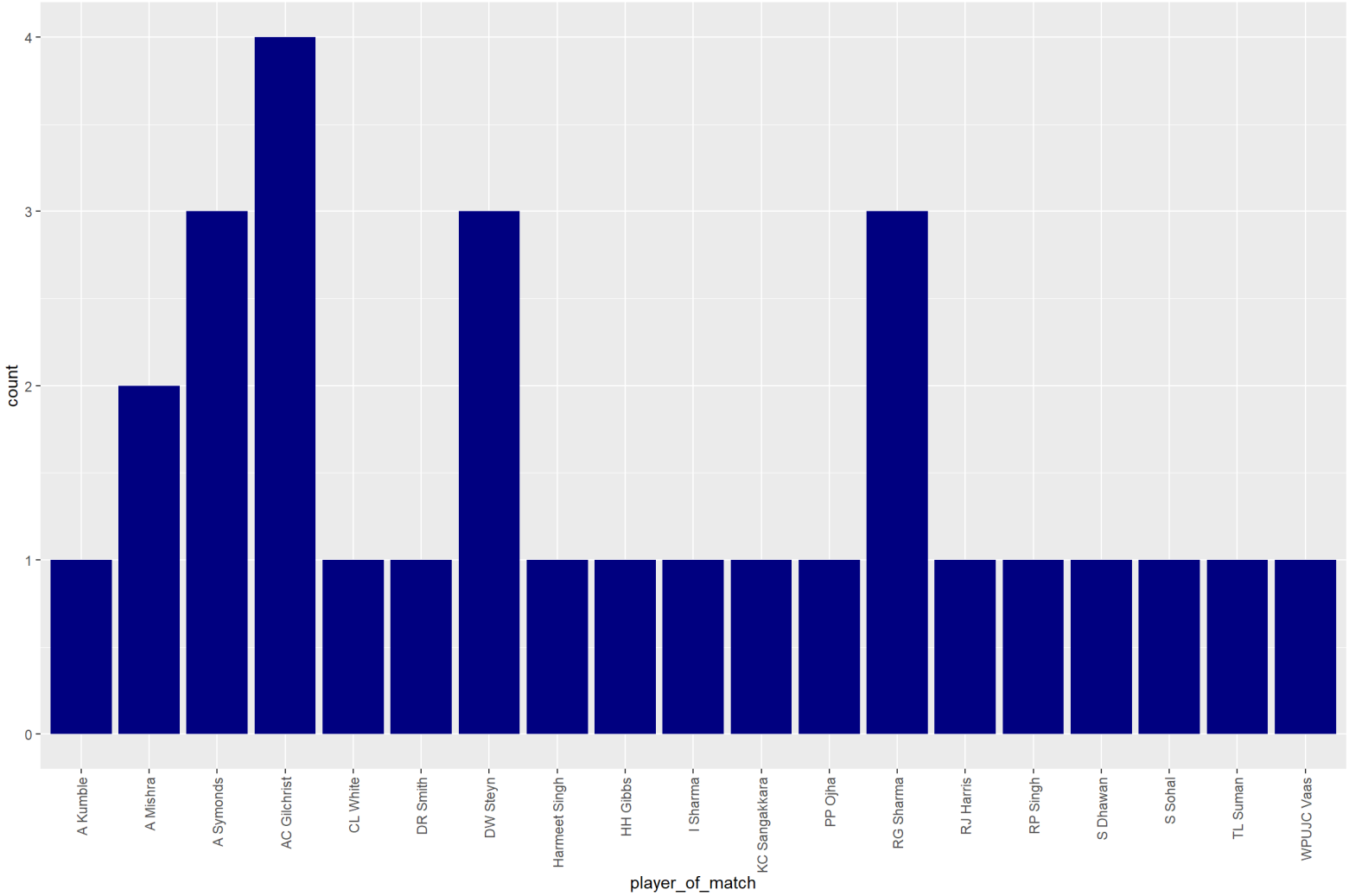
```
KKR_POM<-filter(subs2,winner=='Kolkata Knight Riders')
KkrMoM<-ggplot(KKR_POM, aes(x=player_of_match)) + geom_bar(fill="purple")
KkrMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: AD Russell

Deccan Chargers

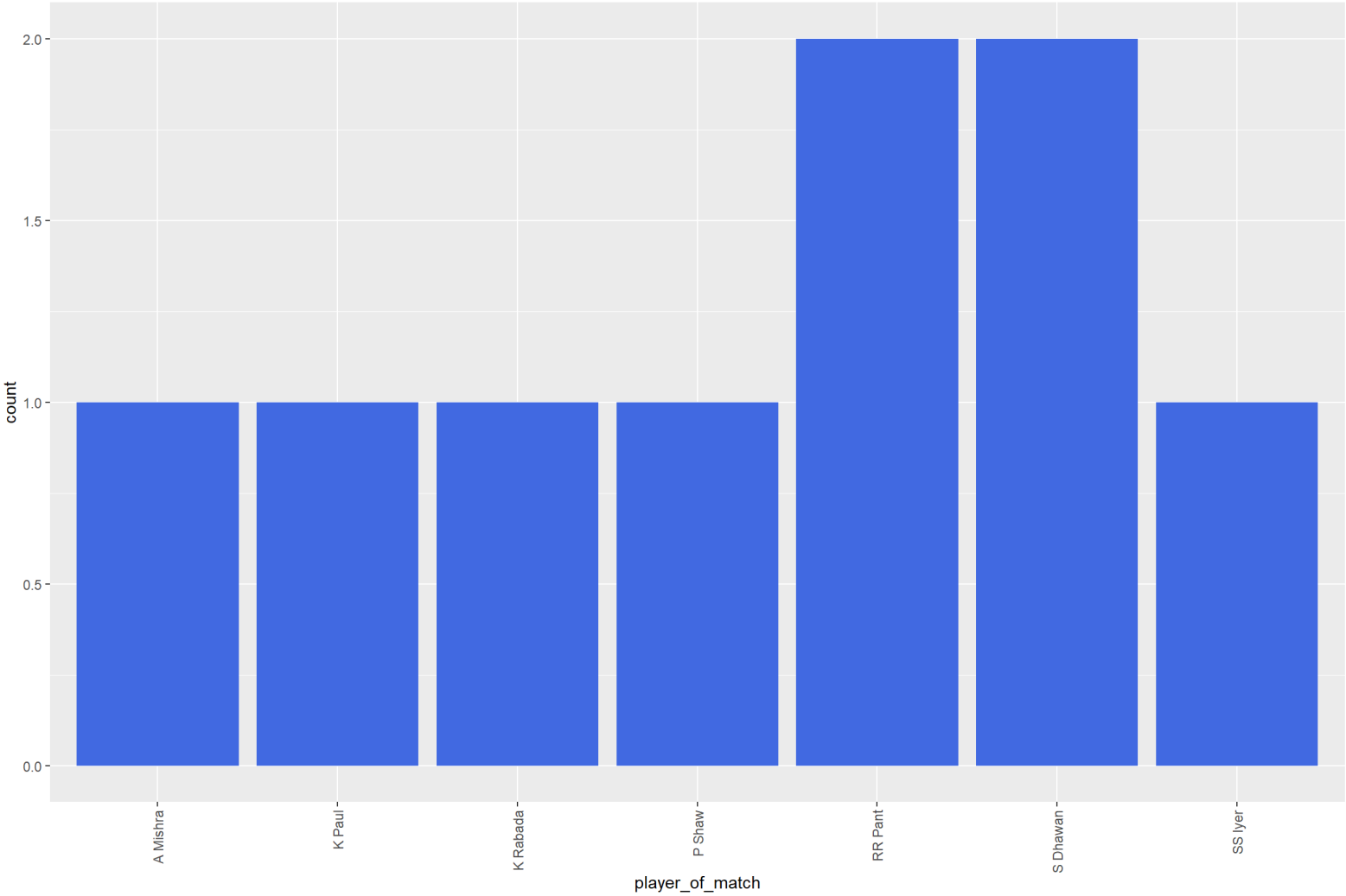
```
DEC_POM<-filter(subs2,winner=='Deccan Chargers')
DecMoM<-ggplot(DEC_POM, aes(x=player_of_match)) + geom_bar(fill="navy")
DecMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team’s success: AC Gilchrist

Delhi Capitals

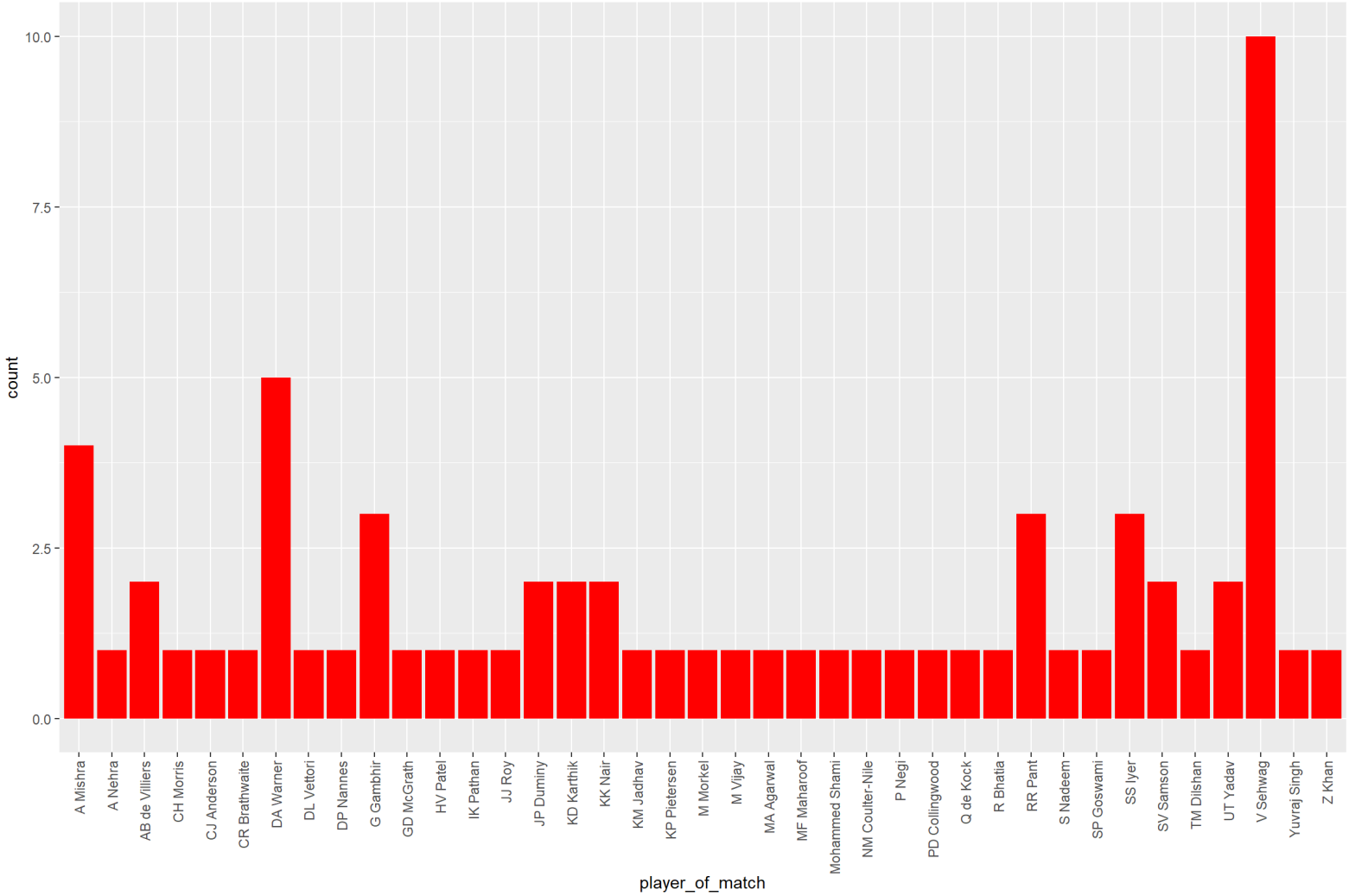
```
DC_POM<-filter(subs2,winner=='Delhi Capitals')
DcMoM<-ggplot(DC_POM, aes(x=player_of_match)) + geom_bar(fill="royalblue")
DcMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team’s success: RR Pant & S Dhawan

Delhi Daredevils

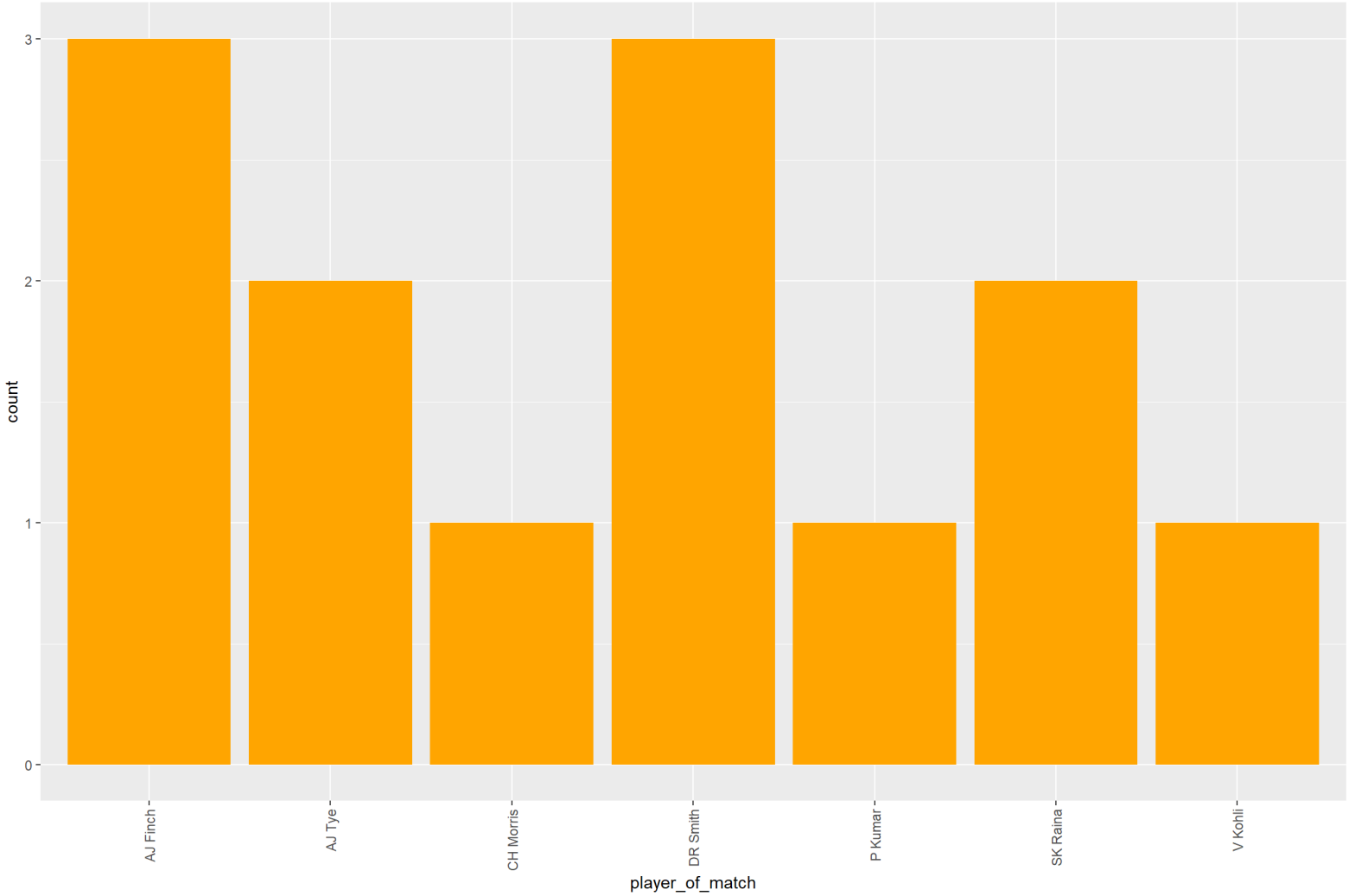
```
DD_POM<-filter(subs2,winner=='Delhi Daredevils')
DdMoM<-ggplot(DD_POM, aes(x=player_of_match)) + geom_bar(fill="red")
DdMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: V Sehwag

Gujarat Lions

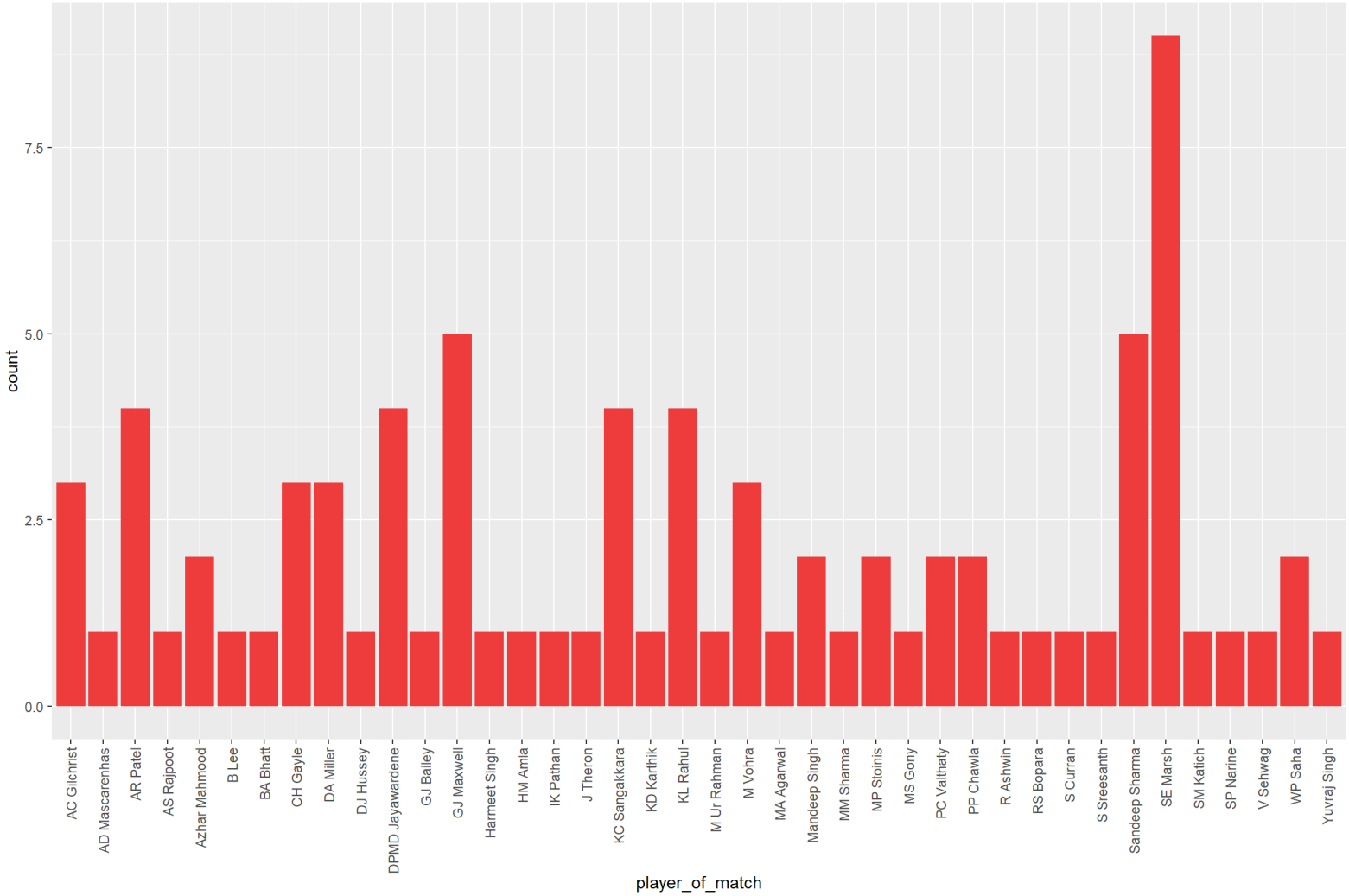
```
GL_POM<-filter(subs2,winner=='Gujarat Lions')
GlMoM<-ggplot(GL_POM, aes(x=player_of_match)) + geom_bar(fill="orange")
GlMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: AJ Finch & DR Smith

Kings XI Punjab

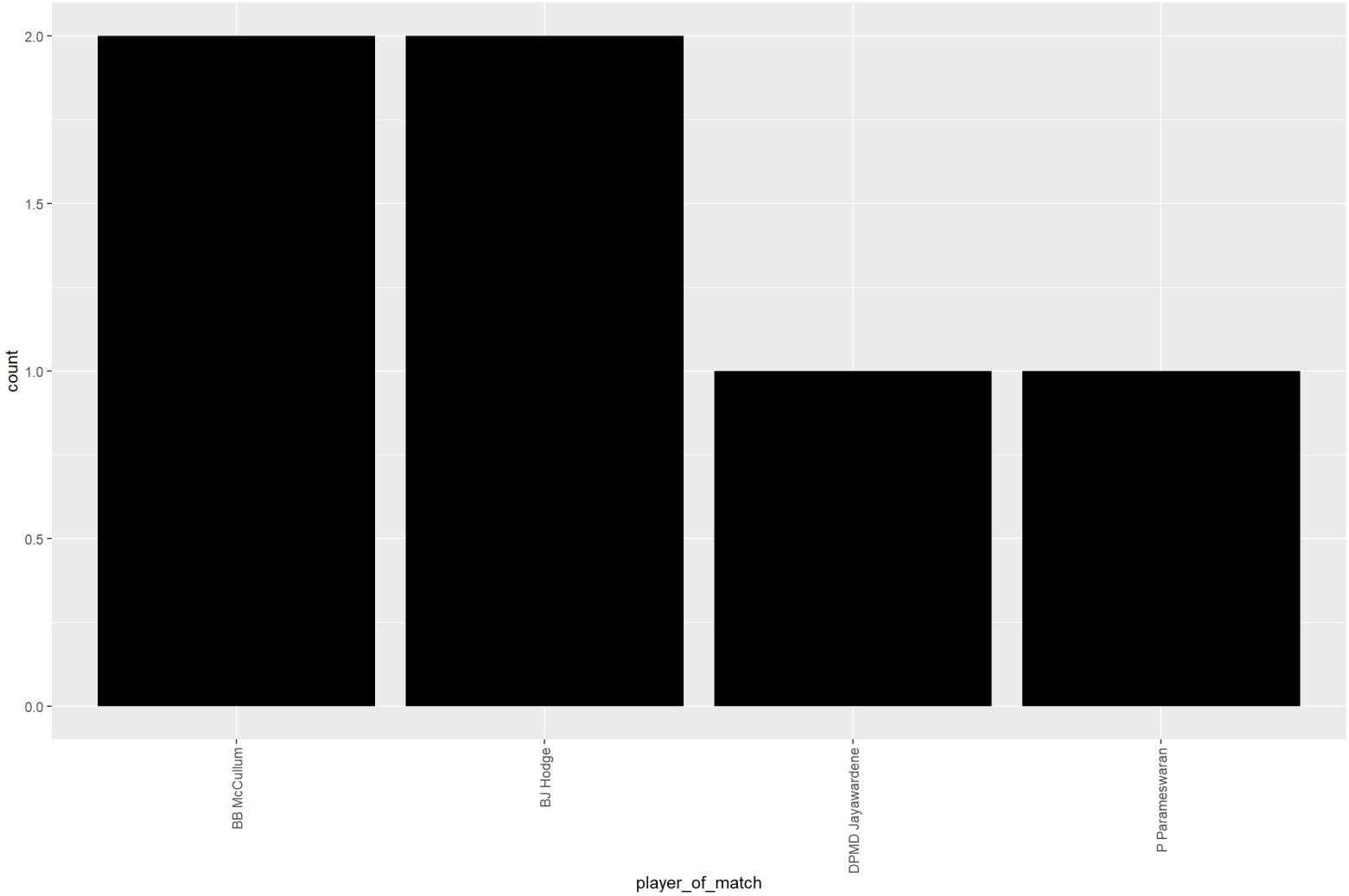
```
KXIP_POM<-filter(subs2,winner=='Kings XI Punjab')
KxipMoM<-ggplot(KXIP_POM, aes(x=player_of_match)) + geom_bar(fill="brown2")
KxipMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: SE Marsh

Kochi Tuskers Kerala

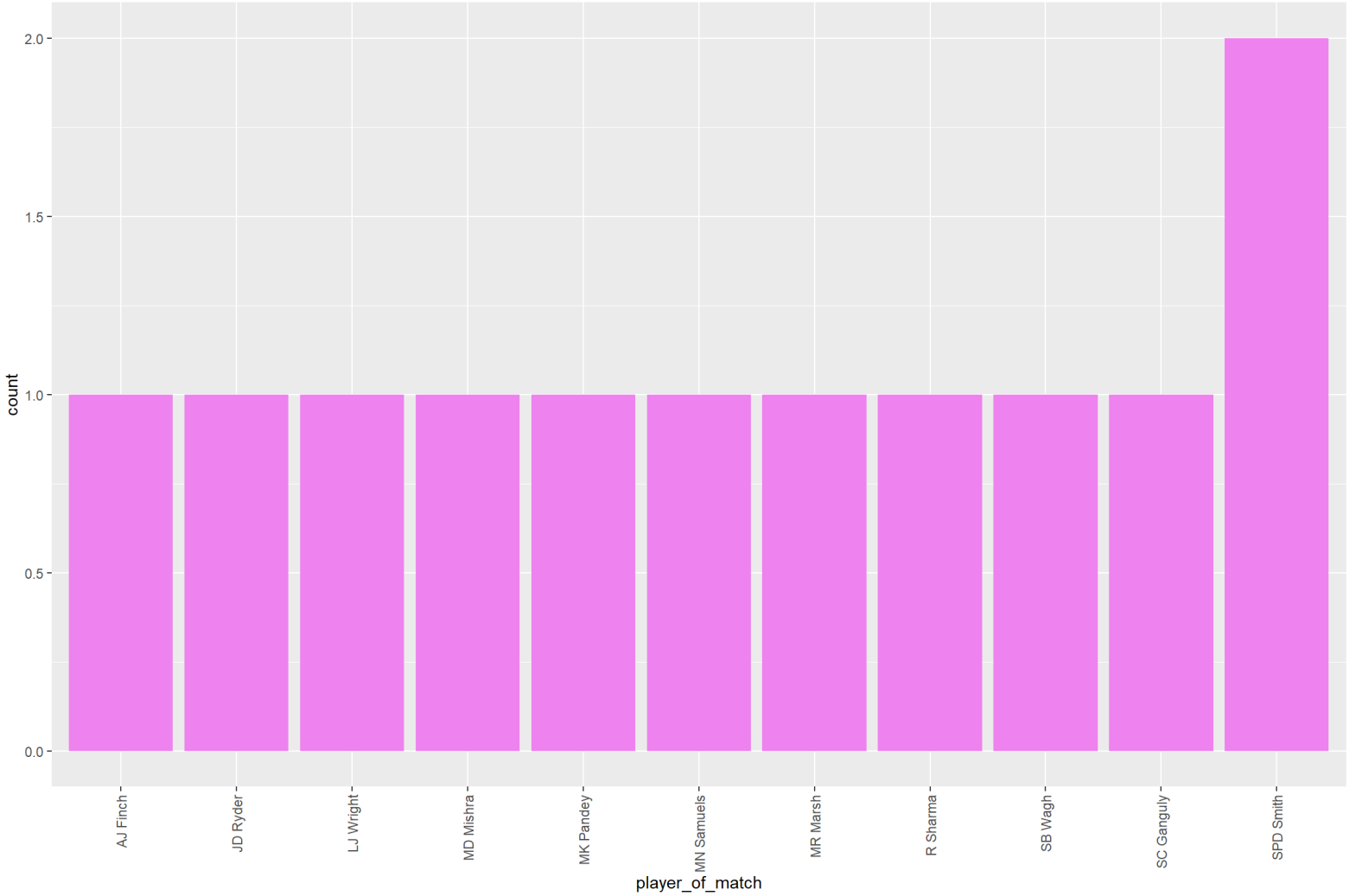
```
KTK_POM<-filter(subs2,winner=='Kochi Tuskers Kerala')
KtkMoM<-ggplot(KTK_POM, aes(x=player_of_match)) + geom_bar(fill="black")
KtkMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: BB McCullum & BJ Hodge

Pune Warriors

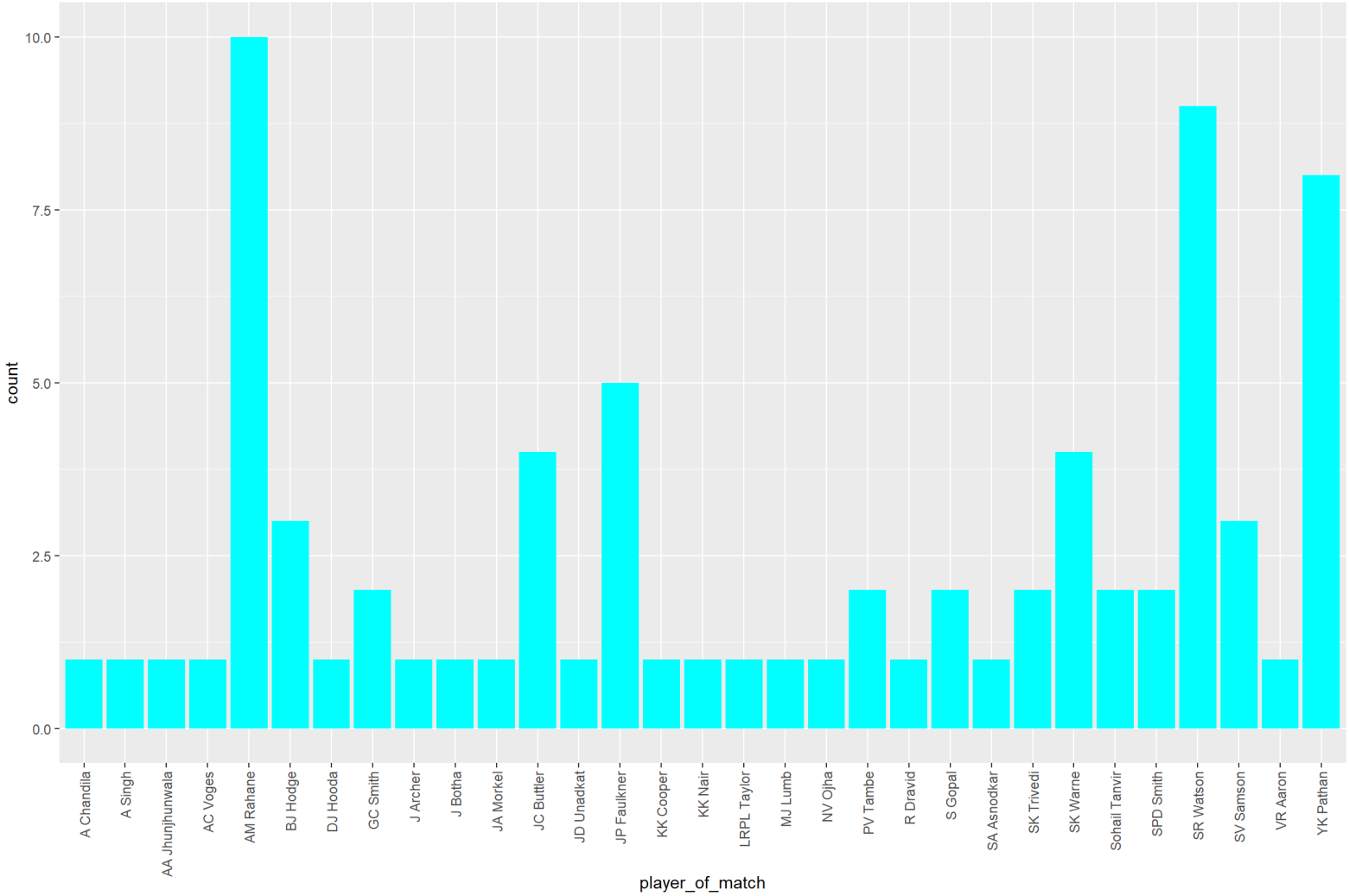
```
PW_POM<-filter(subs2,winner=='Pune Warriors')
PwMoM<-ggplot(PW_POM, aes(x=player_of_match)) + geom_bar(fill="violet")
PwMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: SPD Smith

Rajasthan Royals

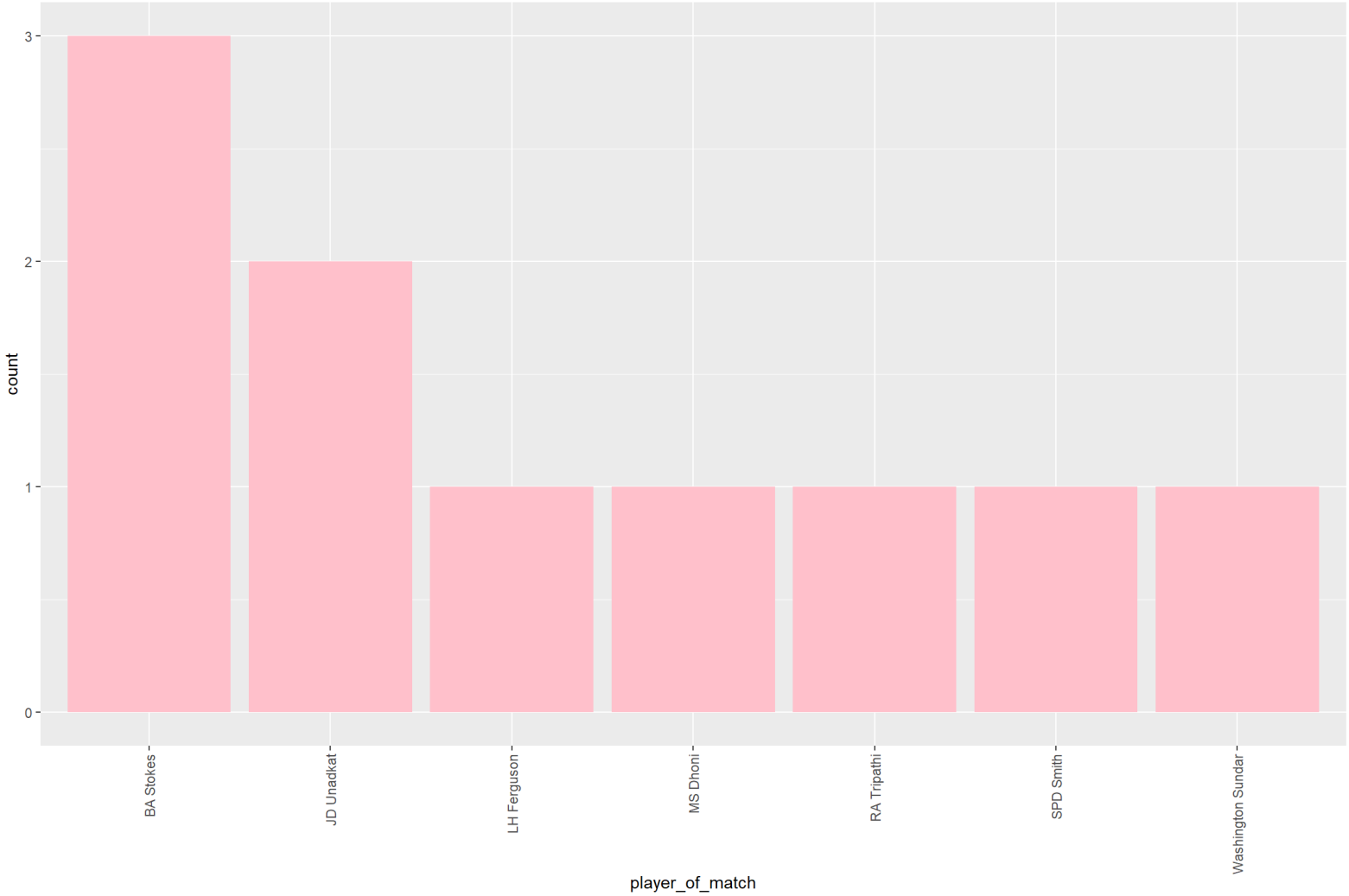
```
RR_POM<-filter(subs2,winner=='Rajasthan Royals')
RrMoM<-ggplot(RR_POM, aes(x=player_of_match)) + geom_bar(fill="cyan")
RrMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: AM Rahane

Rising Pune SupergiantS

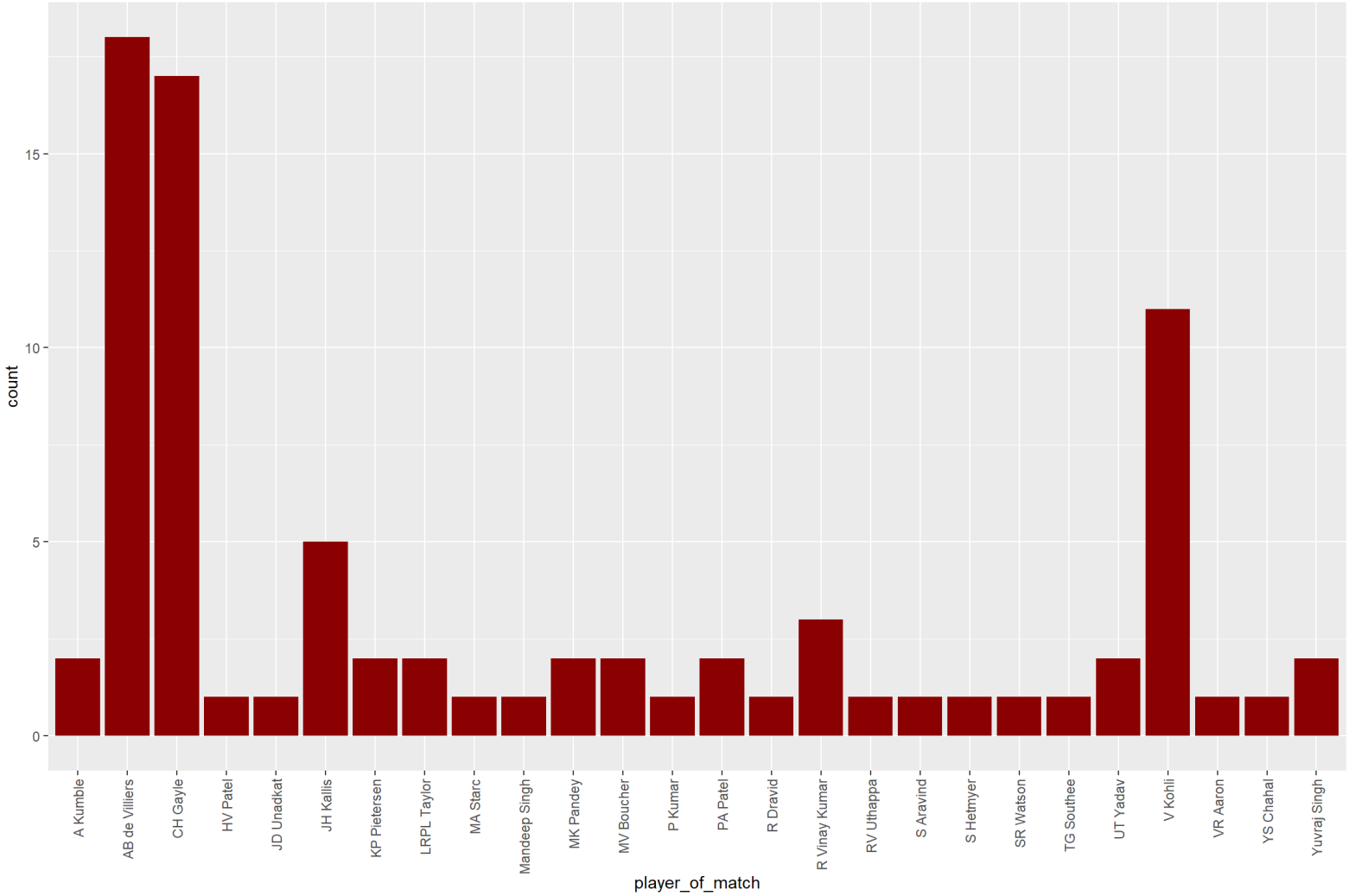
```
RPS_POM<-filter(subs2,winner=='Rising Pune Supergiant')
RpsMoM<-ggplot(RPS_POM, aes(x=player_of_match)) + geom_bar(fill="pink")
RpsMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team’s success: BA Stokes

Royal Challengers Bangalore

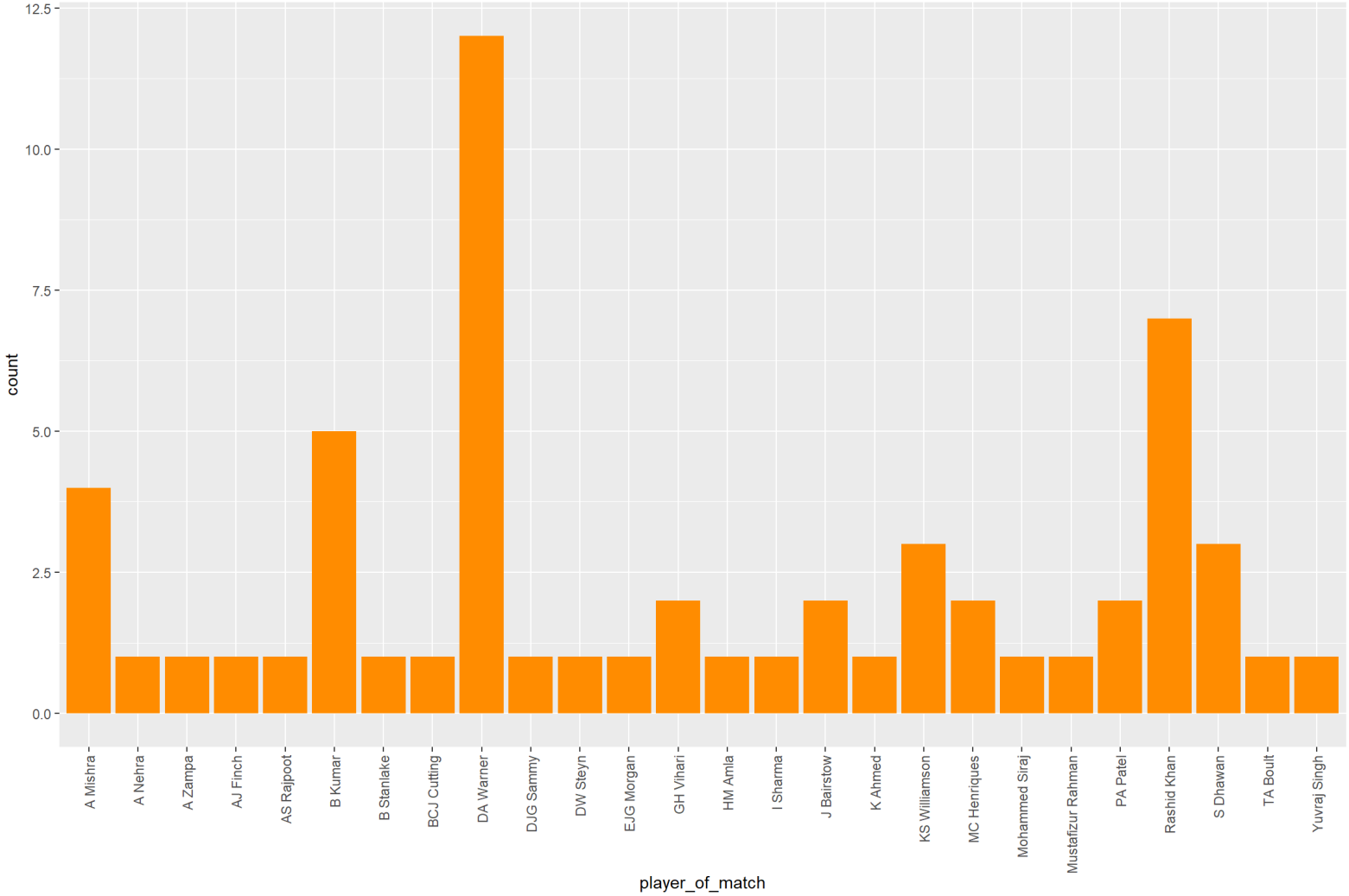
```
RCB_POM<-filter(subs2,winner=='Royal Challengers Bangalore')
RcbMoM<-ggplot(RCB_POM, aes(x=player_of_match)) + geom_bar(fill="darkred")
RcbMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Player contributing most to the team's success: AB de Villiers

Sunrisers Hyderabad

```
SH_POM<-filter(subs2,winner=='Sunrisers Hyderabad')
ShMoM<-ggplot(SH_POM, aes(x=player_of_match)) + geom_bar(fill="darkorange")
ShMoM + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

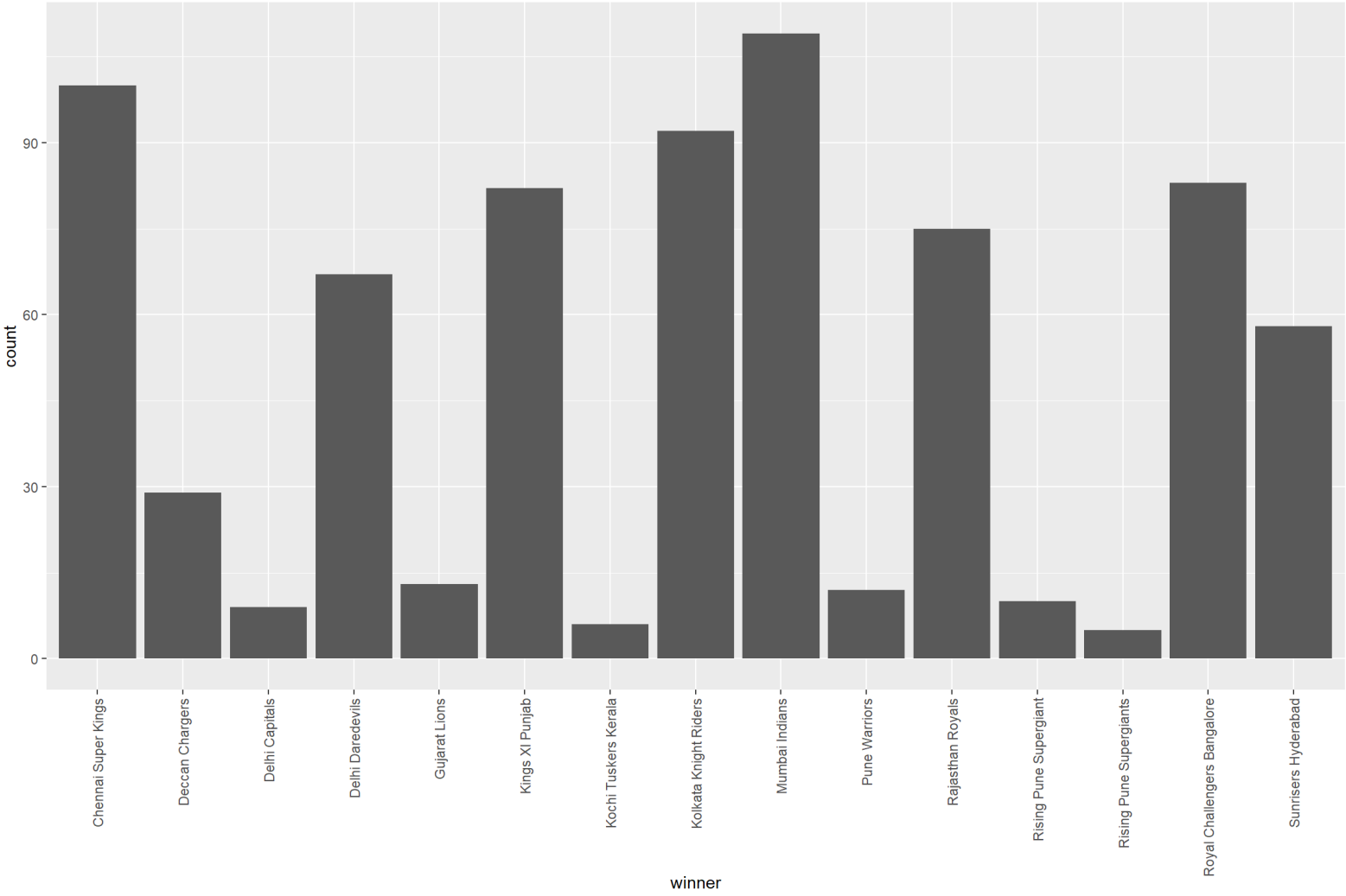


Player contributing most to the team's success: DA Warner

OBJECTIVE 4: Suggest players or teams for a company to endorse for its products

Let us rerun our result from OBJECTIVE 2 to select successful teams to suggest.

```
q + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



The successful teams are:

1.Mumbai Indians

2.Chennai Super Kings

3.Kolkata Knight Riders

4.Royal Challengers Bangalore

Let us analyze the popularity of players in the above teams by measuring the total runs scored by them

Higher Performance of the player = Greater Popularity

```
library(dplyr)
```

Mumbai Indians

```
MI<-filter(subs1,batting_team=='Mumbai Indians')
MI_tbl <- MI %>% group_by(batsman) %>%
summarise(total_count=n(),
.groups = 'drop')
MI_tbl
```

batsman <chr>	total_count <int>
A Dananjaya	5
A Joseph	13
A Nehra	13
A Symonds	143
AC Blizzard	91
AG Murtaza	5
AJ Finch	33
AM Nayar	348
AM Rahane	147
AN Ahmed	21

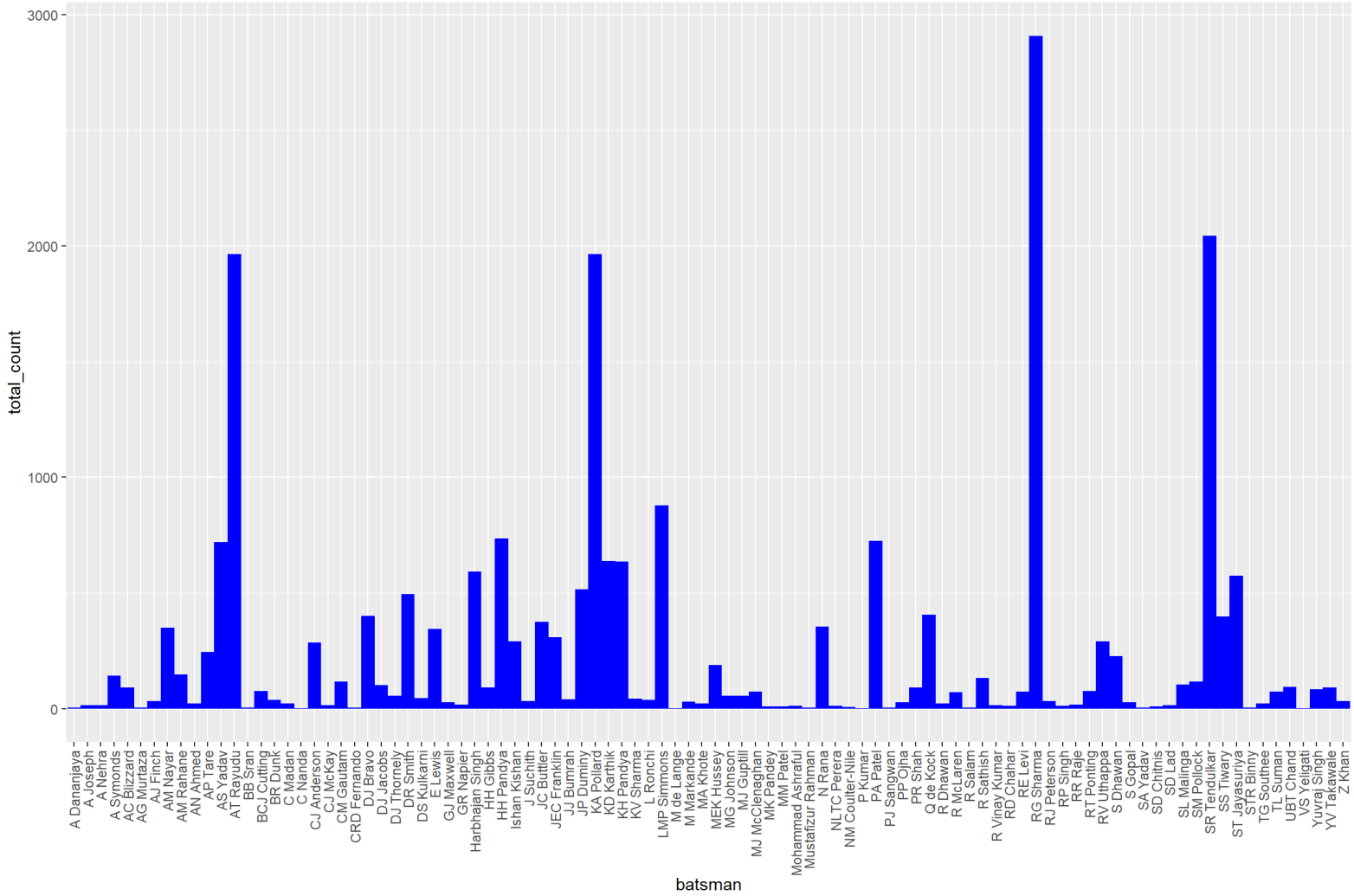
1-10 of 96 rows Previous **1** 2 3 4 5 6 … 10 Next

```
MI2 <- MI_tbl %>% as.data.frame()
MI2
```

batsman <chr>	total_count <int>
A Dananjaya	5
A Joseph	13
A Nehra	13
A Symonds	143
AC Blizzard	91
AG Murtaza	5
AJ Finch	33
AM Nayar	348
AM Rahane	147
AN Ahmed	21

1-10 of 96 rows Previous **1** 2 3 4 5 6 … 10 Next

```
MIplot<- ggplot(MI2, aes(x=batsman, y=total_count))+geom_bar(width = 1, stat = "identity",fill="blue")
MIplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Players with most popularity for ads are:

1. RG Sharma
2. SR Tendulkar
3. KA Pollard

Chennai Super Kings

```
CSK<-filter(subs1,batting_team=='Chennai Super Kings')
CSK_tbl <- CSK %>% group_by(batsman) %>%
  summarise(total_count=n(),
            .groups = 'drop')
CSK_tbl
```

batsman <chr>	total_count <int>
A Flintoff	57
A Mukund	1
A Nehra	7
AS Rajpoot	6

batsman <chr>	total_count <int>
AT Rayudu	717
B Laughlin	9
BB McCullum	634
CH Morris	17
CK Kapugedera	24
D Shorey	20

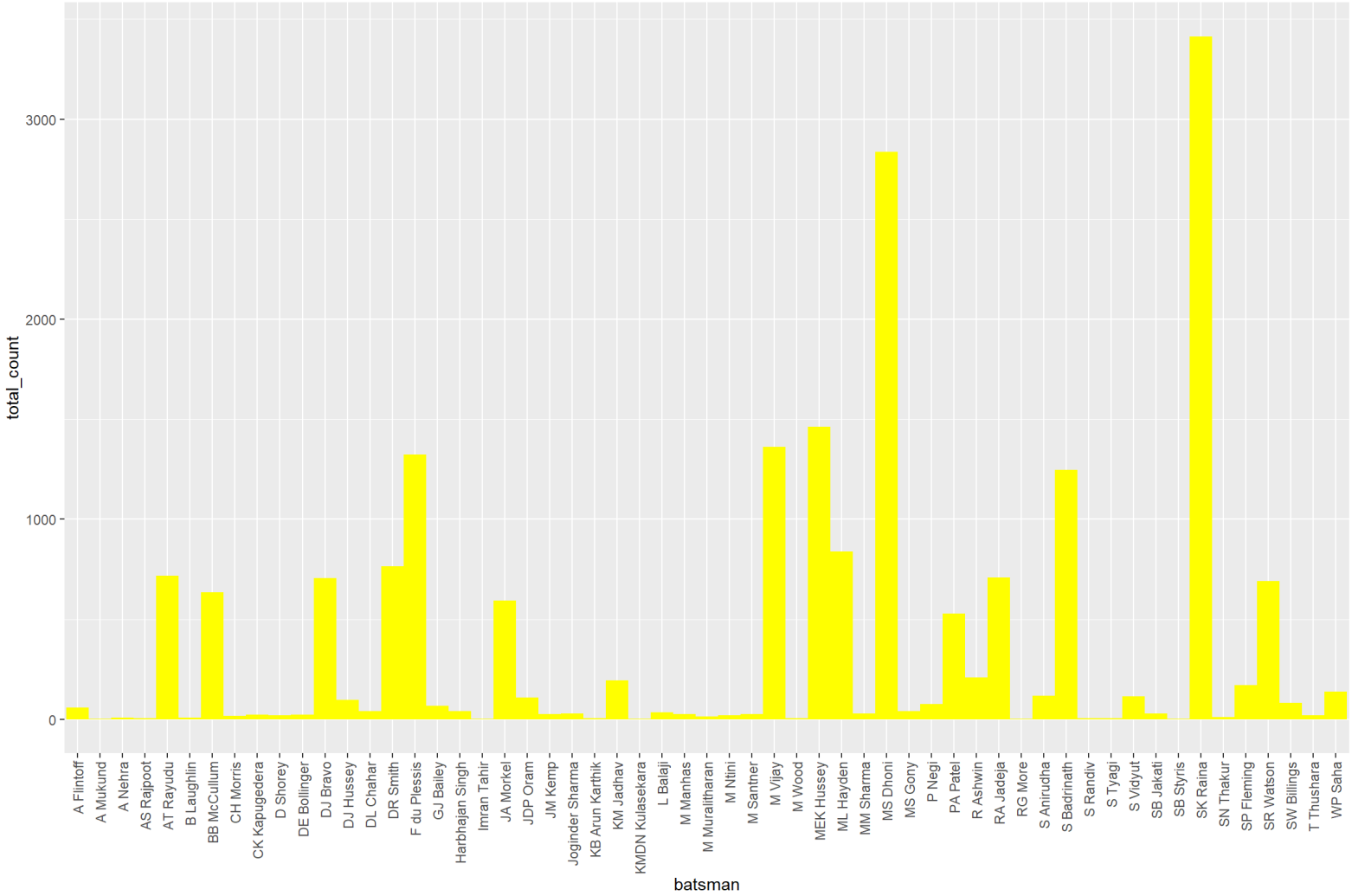
1-10 of 57 rows Previous **1** 2 3 4 5 6 Next

```
# Convert tibble to df
CSK2 <- CSK_tbl %>% as.data.frame()
CSK2
```

batsman <chr>	total_count <int>
A Flintoff	57
A Mukund	1
A Nehra	7
AS Rajpoot	6
AT Rayudu	717
B Laughlin	9
BB McCullum	634
CH Morris	17
CK Kapugedera	24
D Shorey	20

1-10 of 57 rows Previous **1** 2 3 4 5 6 Next

```
CSKplot<- ggplot(CSK2, aes(x=batsman, y=total_count))+geom_bar(width = 1, stat = "identity",fill
="yellow")
CSKplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Players with most popularity for ads are:

1. SK Raina
2. MS Dhoni
3. MEK Hussey

Kolkata Knight Riders

```
KKR<-filter(subs1,batting_team=='Kolkata Knight Riders')
KKR_tbl <- KKR %>% group_by(batsman) %>%
  summarise(total_count=n(),
            .groups = 'drop')
KKR_tbl
```

batsman <chr>	total_count <int>
A Chopra	75
AB Agarkar	137
AB Dinda	8
AD Mathews	211

batsman <chr>	total_count <int>
AD Russell	761
AN Ghosh	8
AS Rajpoot	7
Azhar Mahmood	8
B Lee	42
BAW Mendis	2

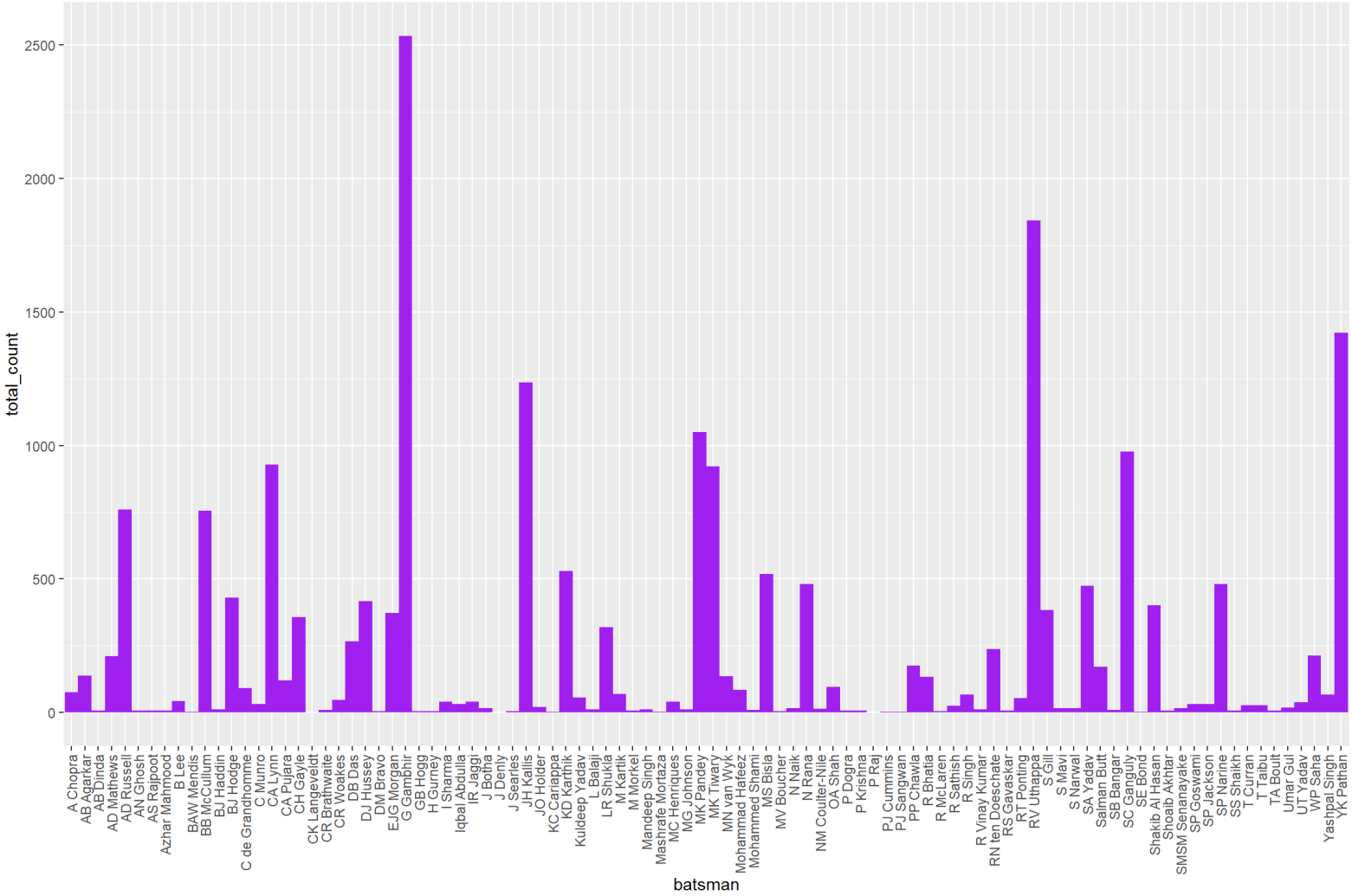
1-10 of 96 rows Previous **1** 2 3 4 5 6 … 10 Next

```
# Convert tibble to df
KKR2 <- KKR_tbl %>% as.data.frame()
KKR2
```

batsman <chr>	total_count <int>
A Chopra	75
AB Agarkar	137
AB Dinda	8
AD Mathews	211
AD Russell	761
AN Ghosh	8
AS Rajpoot	7
Azhar Mahmood	8
B Lee	42
BAW Mendis	2

1-10 of 96 rows Previous **1** 2 3 4 5 6 … 10 Next

```
KKRplot<- ggplot(KKR2, aes(x=batsman, y=total_count))+geom_bar(width = 1, stat = "identity",fill ="purple")
KKRplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Players with most popularity for ads are:

1. G Gambhir
2. RV Uthappa
3. YK Pathan

Royal Challengers Bangalore

```
RCB<-filter(subs1,batting_team=='Royal Challengers Bangalore')
RCB_tbl <- RCB %>% group_by(batsman) %>%
  summarise(total_count=n(),
            .groups = 'drop')
RCB_tbl
```

batsman <chr>	total_count <int>
A Choudhary	20
A Kumble	49
A Mithun	26
A Mukund	22

batsman <chr>	total_count <int>
AA Noffke	12
AB Dinda	13
AB McDonald	38
AB de Villiers	2395
AD Nath	58
AF Milne	12

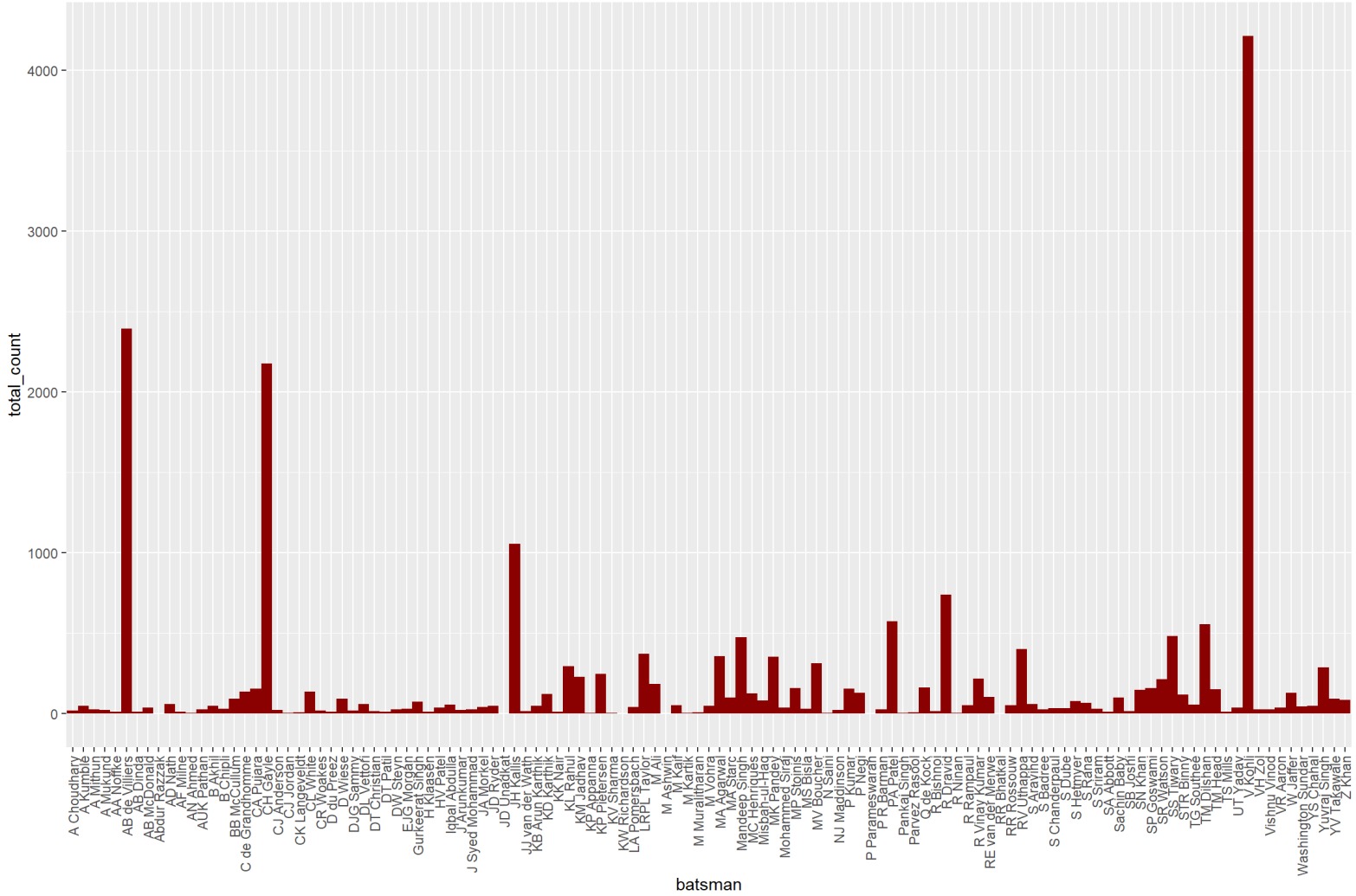
1-10 of 119 rows Previous **1** 2 3 4 5 6 … 12 Next

```
# Convert tibble to df
RCB2 <- RCB_tbl %>% as.data.frame()
RCB2
```

batsman <chr>	total_count <int>
A Choudhary	20
A Kumble	49
A Mithun	26
A Mukund	22
AA Noffke	12
AB Dinda	13
AB McDonald	38
AB de Villiers	2395
AD Nath	58
AF Milne	12

1-10 of 119 rows Previous **1** 2 3 4 5 6 … 12 Next

```
RCBplot<- ggplot(RCB2, aes(x=batsman, y=total_count))+geom_bar(width = 1, stat = "identity",fill
="darkred")
RCBplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Players with most popularity for ads are:

1. V Kohli
2. AB de Villiers
3. CH Gayle

RESULT:

Thus, we have successfully completed an extensive “Exploratory Data Analysis” using the dataset “Indian Premier League”

We have also analyzed successful teams and suggested players and teams that can be used by companies to endorse their products.