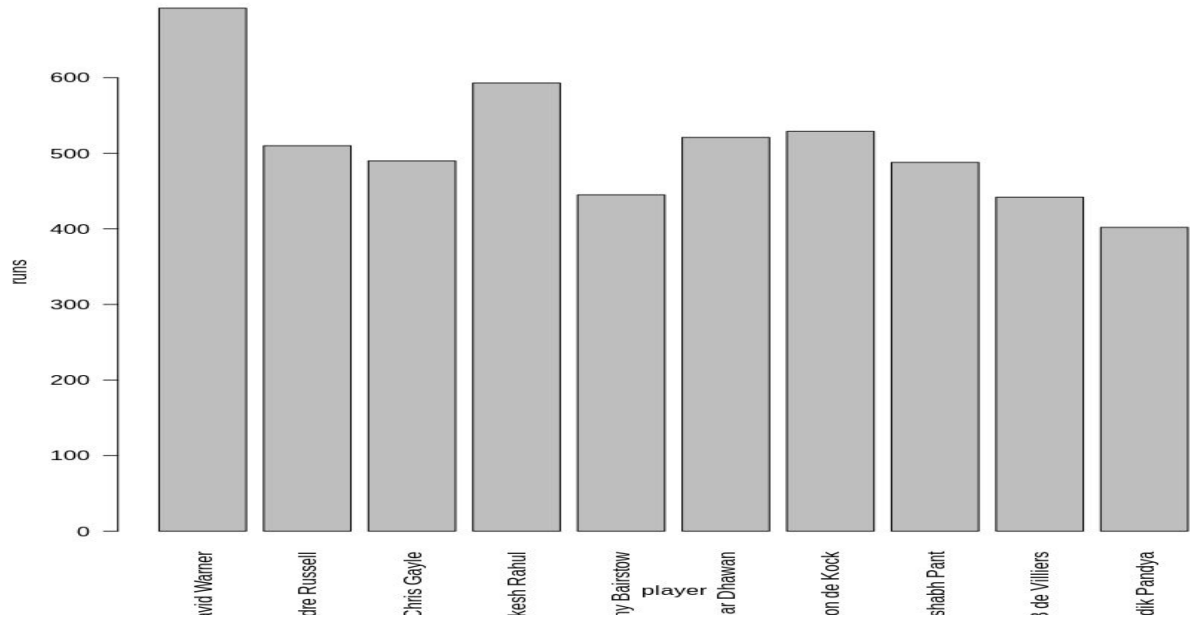
DA LAB ASSIGNMENT : 2

Name : SOMYADEEP SHRIVASTAVA
Roll no. : 17BCS028
TOPIC : IPL DATA ANALYSIS 2019

```
library(tabulizer)
library(dplyr)
library(ggplot2)
library(reshape2)
library(magrittr)
library(tidyr)
# RANKING BALLERS AND BATSMAN DEFINING BY WEIGHTS
library(ggplot2)
> #RANKING BAllERS
> b = read.csv("/home/samroadie/Desktop/DA_Lab/LAB2/baller.csv")
> b['WKTS/MATCHES'] = b['WKTS']/b['MATCHES']
> b['ECONOMY'] = b['RUNS']/(b['BALLS']/6)
> w1 = 5
> w2 = -3
> b['Score'] = w1*b['WKTS/MATCHES'] + w2*b['ECONOMY']
> b['Score'] = b['Score'] + min(b$Score)*(-1)
> b= b[order(b$Score,decreasing = TRUE),]
> b[1:10,]
   POS         PLAYER MATCHES BALLS RUNS WKTS X4.FERS X5.FERS WKTS/MATCHES
ECONOMY    Score
1    1     Imran Tahir      17   386  431   26       2       -     1.529412 6.699482 14.98530
2    2   Kagiso Rabada      12   282  368   25       2       -     2.083333 7.829787 14.36399
9    9     Rashid Khan      15   360  377   17       -       -     1.133333 6.283333 14.25335
5    5  Jasprit Bumrah      16   370  409   19       -       -     1.187500 6.632432 13.47689
10  10 Harbhajan Singh      11   264  312   16       -       -     1.454545 7.090909 13.43669
6    6 K Khaleel Ahmed       9   209  287   19       -       -     2.111111 8.239234 13.27454
12  12 Ravindra Jadeja      16   324  343   15       -       -     0.937500 6.351852 13.06863
4    4   Shreyas Gopal      14   288  347   20       -       -     1.428571 7.229167 12.89205
15  15    Rahul Chahar      13   282  308   13       -       -     1.000000 6.553191 12.77711
21  21     Amit Mishra      11   240  270   11       -       -     1.000000 6.750000 12.18669
> bt = read.csv("/home/samroadie/Desktop/DA_Lab/LAB2/batsman.csv")
> bt = bt[,c("PLAYER","INN","RUNS","AVG","SR","X4S", "X6S")]
```

```
> bt['RUN/INN'] = bt['RUNS']/bt['INN']
> w1 = 7
> w2 = 4
> w3 = 5
> w4 = 1
> bt['Score'] = w1*bt$`RUN/INN` + w2*bt$X4S +w3*bt$X6S + w4*bt$SR
> bt= bt[order(bt$Score,decreasing = TRUE),]
> cd = data.frame(bt[1:10,])
> cd
          PLAYER INN RUNS   AVG     SR X4S X6S  RUN.INN    Score
1     David Warner  12  692 69.20 143.87  57  21 57.66667 880.5367
5    Andre Russell  13  510 56.67 204.82  31  52 39.23077 863.4354
6      Chris Gayle  13  490 40.83 153.61  45  34 37.69231 767.4562
2     Lokesh Rahul  14  593 53.91 135.39  49  25 42.35714 752.8900
10  Jonny Bairstow  10  445 55.62 157.24  48  18 44.50000 750.7400
4   Shikhar Dhawan  16  521 34.73 135.68  64  11 32.56250 674.6175
3  Quinton de Kock  16  529 35.27 132.91  45  25 33.06250 669.3475
7      Rishabh Pant  16  488 37.54 162.67  37  27 30.50000 659.1700
11  AB de Villiers  13  442 44.20 154.01  31  26 34.00000 646.0100
16   Hardik Pandya  15  402 44.67 191.43  28  29 26.80000 636.0300

# plotting top 10 batsman runs

barplot(cd$RUNS,names.arg=cd$PLAYER,xlab='player',ylab='runs',las=2)
```



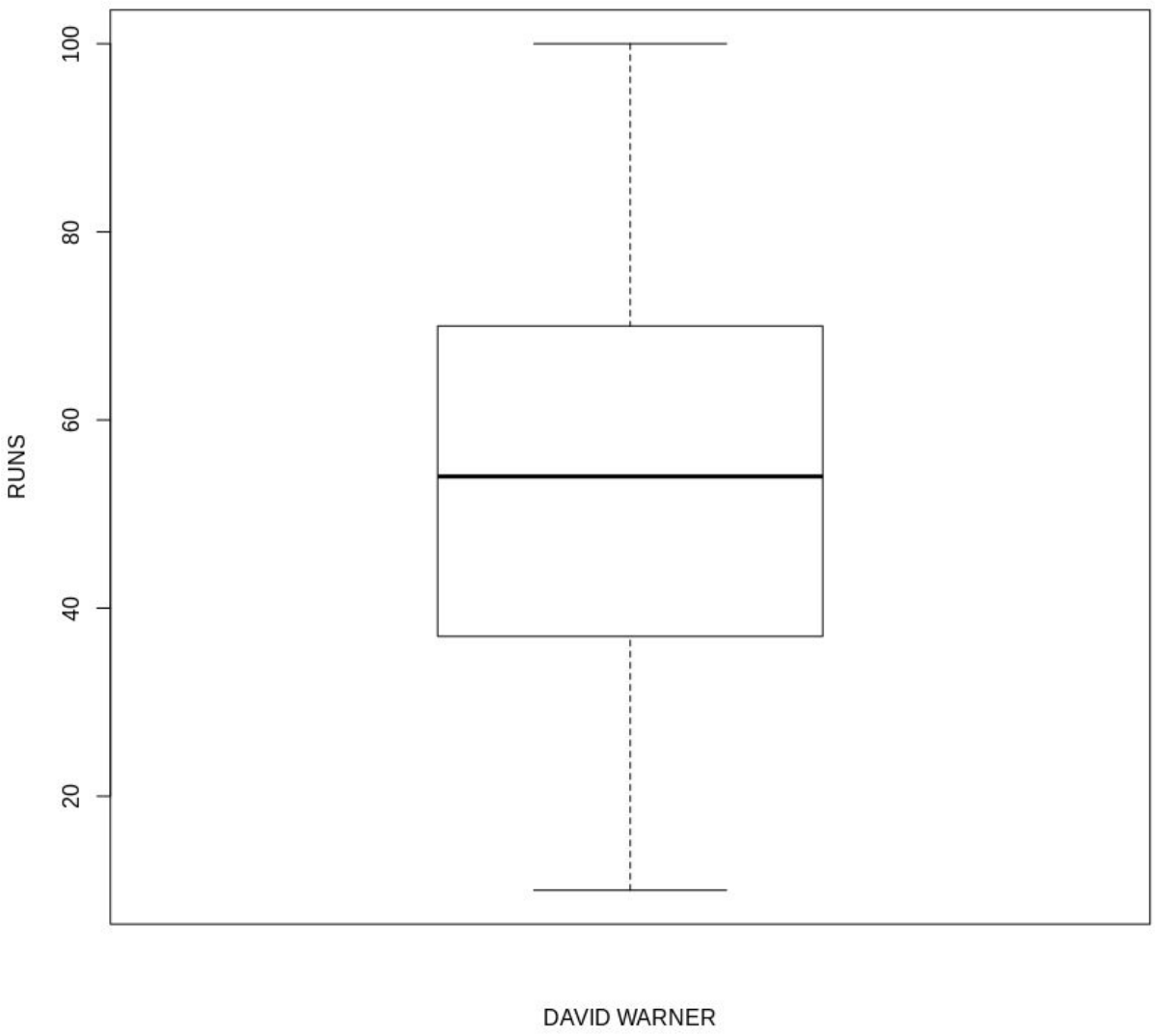
CODE TO MAKE DATASET FOR INDIVIDUAL PLAYER WITH CORRESPONDING RUNS IN EACH MATCH

```
d = read.csv("/home/samroadie/Desktop/DA_Lab/LAB2/deliveries.csv")
d
players_runs = d[,c('match_id','batsman','batsman_runs')]
players_runs

players = unique(players_runs$batsman)
match_id = unique(d$match_id)

batman = c('DA Warner','KL Rahul',  'S Dhawan',    'J Bairstow',    'SS Iyer',         'AD Russell',
'Q de Kock',    'HH Pandya',   'AB de Villiers','RR Pant')

View(d)
s = c()
rs = c()
for(match in match_id){
df = d[which(d$match_id == match),c('batsman','batsman_runs')]
player_name = c()
runs = c()
for(p in unique(df$batsman)){

  if(p %in% batman){
  player_name = c(player_name,p)
  runs =c(runs , sum(df[which(df$batsman==p),'batsman_runs']))}
  print(match)
}
  s = c(s,player_name )
  rs = c(rs,runs)
}
dfo = data.frame(s,rs)
dfo
total_run = c()
j = 1
for (pr in batman)
{

  run  = c()
for(i in rownames(dfo)){
  if(dfo[i,1]== pr){
```

```
    run=c(run,dfo[i,2])
  }

  }
  total_run[[j]] = run
  j = j+1
}
batman
total_run
dfd = data.frame(matrix(ncol = 10,nrow=10))
dfd
colnames(dfd) <- batman
dfd
for(i in rownames(dfd)){
  i= as.numeric(i)
  for(j in 1:10){
  dfd[i,j] = total_run[[j]][i]
  }}
dfd
check = dfd
check
write.csv(check, file = "/home/samroadie/Desktop/DA_Lab/LAB2/topplayer.csv", row.names =
c('match1','match2','match3','match4','match5','match6','match7','match8','match9','match10'))
```

 # Finding descriptive statistics and coefficient of variance of top 10 players

```
> ########  loading data of top 10 players and their individual 10 match score ######
> ind_ply <- read.csv("/home/samroadie/Desktop/DA_Lab/LAB2/topplayer.csv")
> summary(ind_ply)
    match     DA.Warner        KL.Rahul       S.Dhawan      J.Bairstow       SS.Iyer       AD.Rusell
Q.de.Kock
 match1 :1   Min.   : 10.00   Min.   :  1.0   Min.   : 0.0   Min.   :  0.00   Min.   : 3.0   Min.   :10.00
Min.   : 4.00
 match10:1   1st Qu.: 40.25   1st Qu.:  9.0   1st Qu.:13.0   1st Qu.: 21.75   1st Qu.:16.5   1st
Qu.:31.75   1st Qu.:23.25
 match2 :1   Median : 54.00   Median : 33.5   Median :32.5   Median : 43.00   Median :35.5
Median :51.00   Median :31.00
 match3 :1   Mean   : 53.80   Mean   : 38.9   Mean   :34.7   Mean   : 44.50   Mean   :32.7   Mean
:48.30   Mean   :37.80
 match4 :1   3rd Qu.: 69.25   3rd Qu.: 67.0   3rd Qu.:49.0   3rd Qu.: 57.75   3rd Qu.:44.5   3rd
Qu.:65.50   3rd Qu.:55.00
 match5 :1   Max.   :100.00   Max.   :100.0   Max.   :97.0   Max.   :114.00   Max.   :67.0   Max.
:86.00   Max.   :81.00
 (Other):4
   HH.Pandya     AB.de.Villiers     RR.Pant
 Min.   : 0.00   Min.   : 1.00   Min.   : 5.00
 1st Qu.:20.00   1st Qu.:14.25   1st Qu.:18.00
 Median :26.50   Median :42.50   Median :26.00
 Mean   :24.10   Mean   :42.60   Mean   :35.00
 3rd Qu.:31.75   3rd Qu.:71.00   3rd Qu.:46.75
 Max.   :37.00   Max.   :86.00   Max.   :80.00

> #BOX PLOT OF TOP MOST PLAYER DAVID WARNER

> boxplot(ind_ply$DA.Warner, xlab = 'DAVID WARNER',ylab ='RUNS')
```




```
> #Coefficient of variance of various players
> sd(ind_ply$DA.Warner)/mean(ind_ply$DA.Warner)*100
[1] 52.021
> sd(ind_ply$KL.Rahul)/mean(ind_ply$KL.Rahul)*100
[1] 90.39726
> sd(ind_ply$S.Dhawan)/mean(ind_ply$S.Dhawan)*100
[1] 83.34738
> sd(ind_ply$J.Bairstow)/mean(ind_ply$J.Bairstow)*100
[1] 78.92764
> sd(ind_ply$SS.Iyer)/mean(ind_ply$SS.Iyer)*100
[1] 66.783
> sd(ind_ply$AD.Rusell)/mean(ind_ply$AD.Rusell)*100
```

```
[1] 51.14472
> sd(ind_ply$Q.de.Kock)/mean(ind_ply$Q.de.Kock)*100
[1] 63.18267
> sd(ind_ply$HH.Pandya)/mean(ind_ply$HH.Pandya)*100
[1] 45.13528
> sd(ind_ply$AB.de.Villiers)/mean(ind_ply$AB.de.Villiers)*100
[1] 75.50111
> sd(ind_ply$RR.Pant)/mean(ind_ply$RR.Pant)*100
[1] 77.52404
```

Those who have highest covariance will have least consistency, here kl rahul has least consistency.
Those who have lowest covariance will have highest consistency, here AD Russell have highest consistency.

```
> cd
          PLAYER INN RUNS   AVG     SR X4S X6S  RUN.INN    Score
1     David Warner  12  692 69.20 143.87  57  21 57.66667 880.5367
5    Andre Russell  13  510 56.67 204.82  31  52 39.23077 863.4354
6      Chris Gayle  13  490 40.83 153.61  45  34 37.69231 767.4562
2     Lokesh Rahul  14  593 53.91 135.39  49  25 42.35714 752.8900
10  Jonny Bairstow  10  445 55.62 157.24  48  18 44.50000 750.7400
4   Shikhar Dhawan  16  521 34.73 135.68  64  11 32.56250 674.6175
3  Quinton de Kock  16  529 35.27 132.91  45  25 33.06250 669.3475
7      Rishabh Pant  16  488 37.54 162.67  37  27 30.50000 659.1700
11  AB de Villiers  13  442 44.20 154.01  31  26 34.00000 646.0100
16   Hardik Pandya  15  402 44.67 191.43  28  29 26.80000 636.0300
> ######## correlation between RUNS and AVG of various players  #############
> cor.test(cd$AVG, cd$RUNS,  method = "pearson")

        Pearson's product-moment correlation

data:  cd$AVG and cd$RUNS
t = 1.7796, df = 8, p-value = 0.113
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.1460633  0.8703395
sample estimates:
      cor
0.5325381

> cor.test(ind_ply$DA.Warner, ind_ply$S.Dhawan ,method = "spear")
```

```
        Spearman's rank correlation rho

data:  ind_ply$DA.Warner and ind_ply$S.Dhawan
S = 84, p-value = 0.1544
alternative hypothesis: true rho is not equal to 0
sample estimates: rho  0.4909091
```

#Descriptive and inferential statistics of IPL 2019 and plots

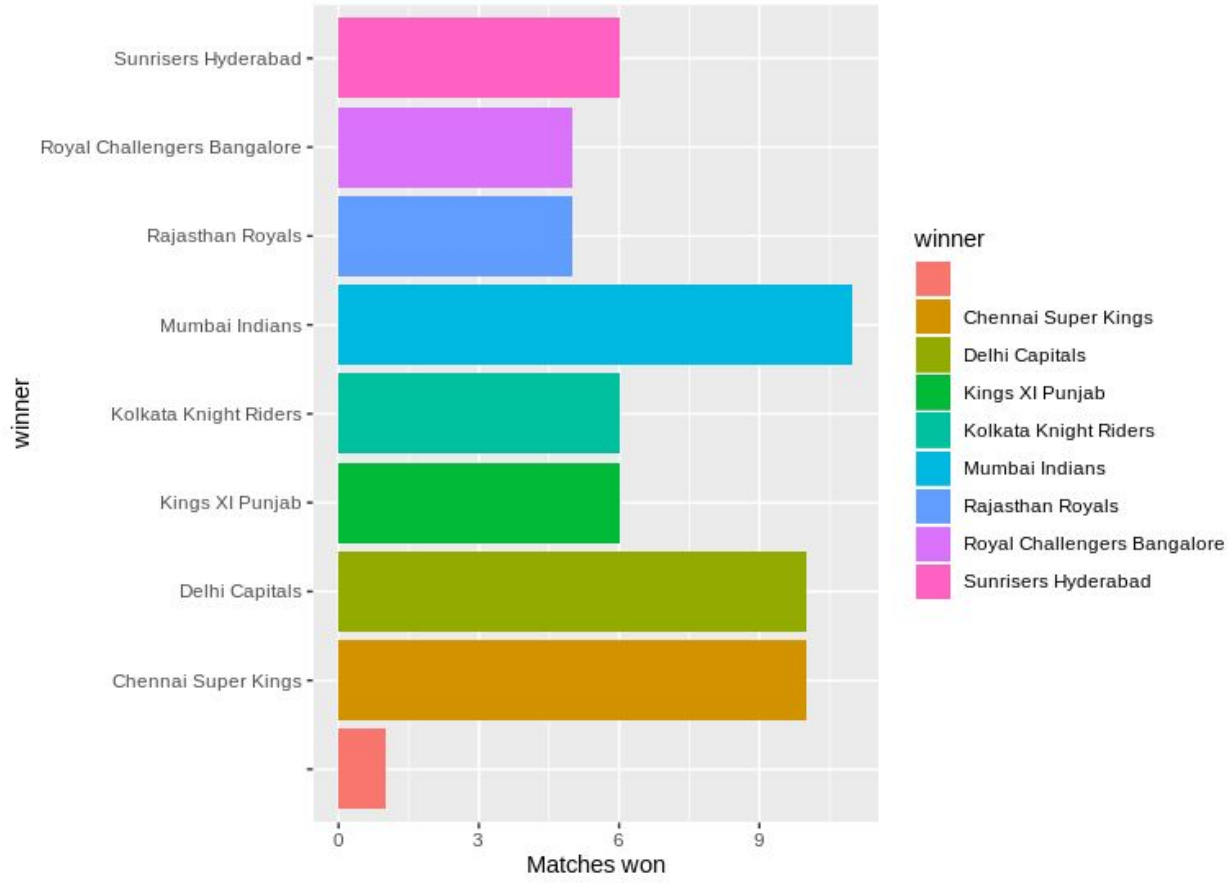
```
matches <- read.csv("/home/samroadie/Desktop/Link to Clg/Sem 6/Data Analytics/DA LAB/Lab 2/match.csv", stringsAsFactors = FALSE)

data <- read.csv("/home/samroadie/Desktop/Link to Clg/Sem 6/Data Analytics/DA LAB/Lab 2/data.csv", stringsAsFactors = FALSE)

matches <- matches[,-18]
data$wickets <- as.numeric(ifelse(data$player_dismissed =="" ,"",1))

> summarize(matches,no_of_matches = n())
  no_of_matches
1            60

> max_run <- matches[which.max(matches$win_by_runs),]
> select(max_run, winner, win_by_runs)
               winner win_by_runs
11 Sunrisers Hyderabad         118
> # Sunrisers Hyderabad by 118 runs

> max_run <- matches[which.max(matches$win_by_wickets),]
> select(max_run, winner, win_by_wickets)
               winner win_by_wickets
38 Sunrisers Hyderabad             9
> # Sunrisers Hyderabad by 9 wicket
```

##

Teams and matches won

```
matches%>%
  group_by(winner)%>%
  summarize(most_win = n())%>%
  ggplot(aes(x = winner,y = most_win,fill = winner))+
  geom_bar(stat = "identity")+
```

```
coord_flip()+
scale_y_continuous("Matches won")
```



##

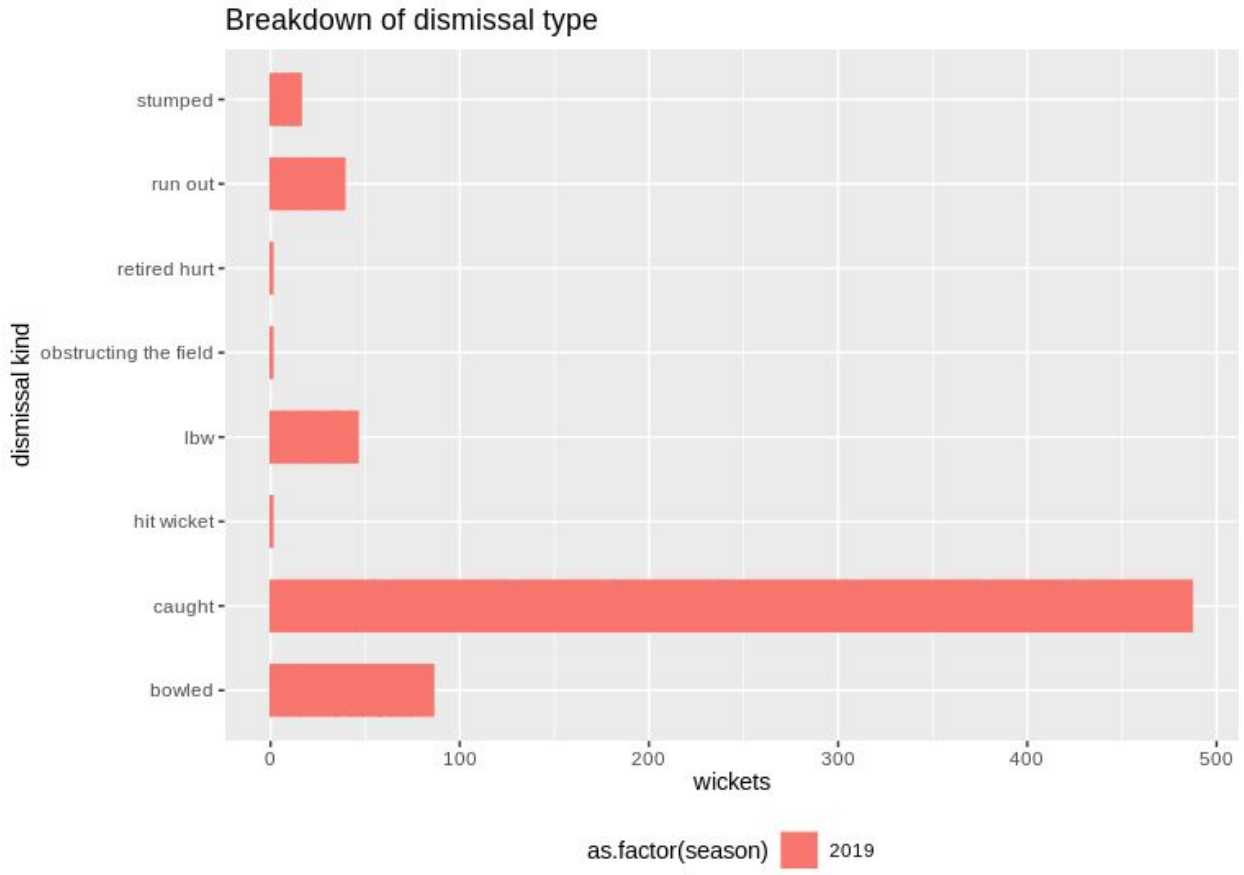
```
teams <- data %>% select(batting_team)%>%
  distinct()
teams <- rename(teams, team = batting_team)
teams
s_team <- c("RCB","CSK","SRH","KKR","DC","MI","KXIP","RR")
teams <- cbind(teams, s_team)
player_of_match <- matches%>% select(id,player_of_match,season) %>%
```

```
  distinct()
player_of_match <- rename(player_of_match, player=player_of_match)

matches$city <- as.character(matches$city)
matches$city[matches$city==""] <- "Dubai"
venue_city <- matches %>%
  select(city)%>%
  distinct()
```

##

Dismissal type and number of dismissal

```
dismissal <- data%>%
  left_join(matches, by=c("match_id"="id"))%>%
  left_join(teams,by=c("batting_team"="team"))%>%
  filter(dismissal_kind!="")%>%
  group_by(season,dismissal_kind,s_team)%>%
  summarize(wickets =n())
ggplot(dismissal,aes(x=dismissal_kind,y=wickets,colour=as.factor(season),
fill=as.factor(season)))+
  geom_bar(position = "stack", show.legend = TRUE, width =.6,stat="identity")+
  theme(legend.position="bottom")+
  coord_flip()+
  theme(legend.direction = "horizontal") +
  scale_y_continuous(name="wickets")+
  scale_x_discrete(name="dismissal kind")+
  ggtitle("Breakdown of dismissal type ")
```

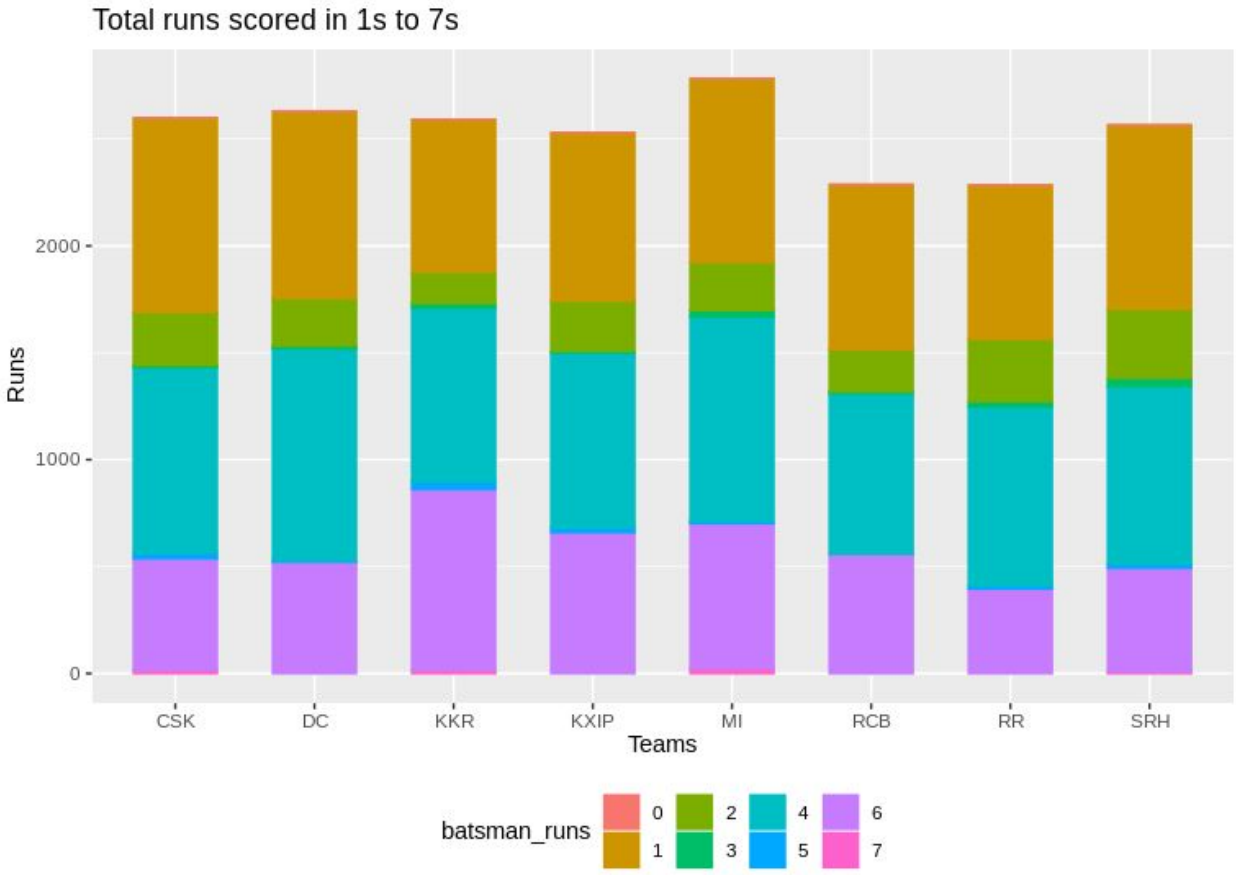


##

Run scored in 1s to 7s

```
runs_cat <- data %>%
  left_join(matches,by=c("match_id"="id"))%>%
  left_join(teams,by=c("batting_team"="team"))%>%
  group_by(s_team,batsman_runs)%>%
  summarize(no=n(),runs=sum(total_runs))

runs_cat$batsman_runs <- as.factor(runs_cat$batsman_runs)

ggplot(runs_cat,aes(x=s_team,y=runs,colour=batsman_runs,fill=batsman_runs))+
  geom_bar(position = "stack", show.legend = TRUE, width =.6,stat="identity")+
  theme(legend.position="bottom")+
```

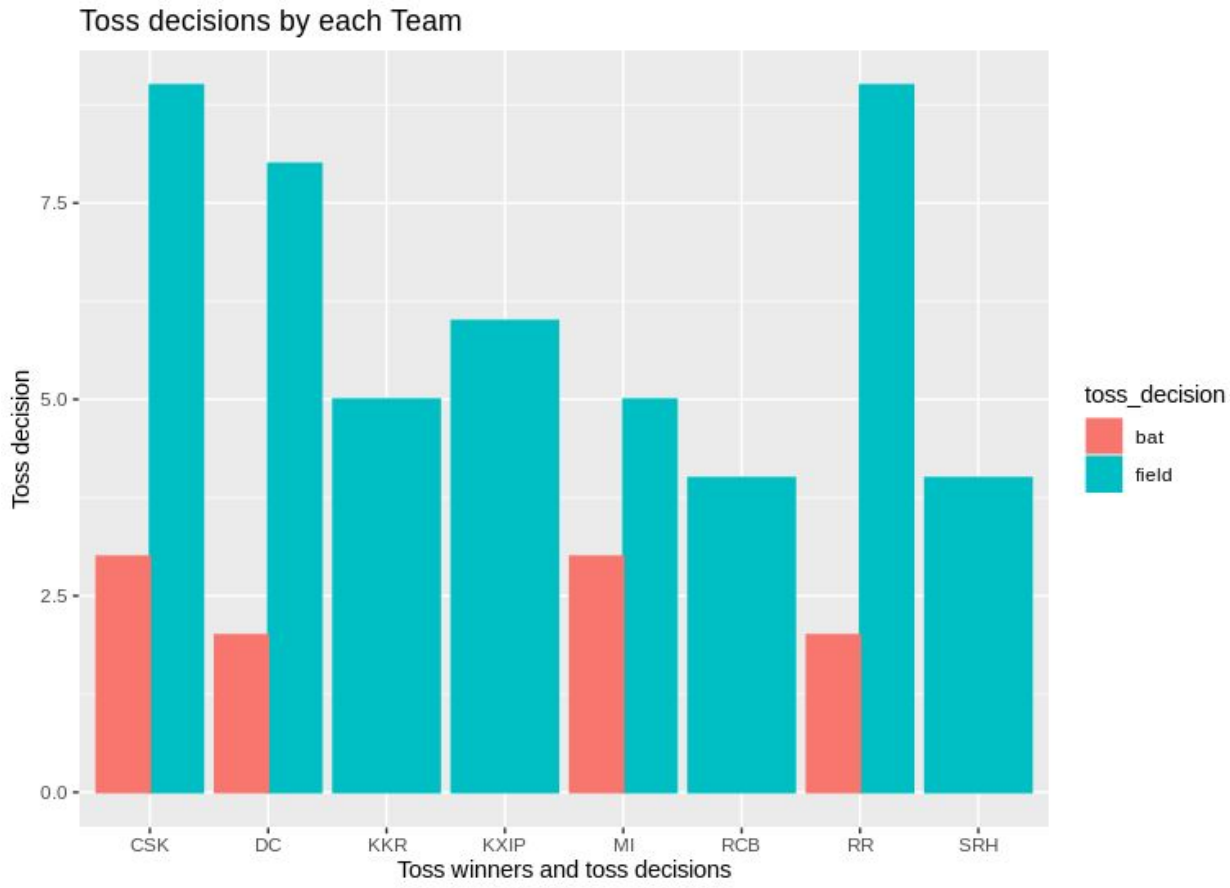
```
theme(legend.direction = "horizontal") +
scale_y_continuous(name="Runs")+
scale_x_discrete(name="Teams")+
ggtitle("Total runs scored in 1s to 7s")
```



##

toss decision of toss winner

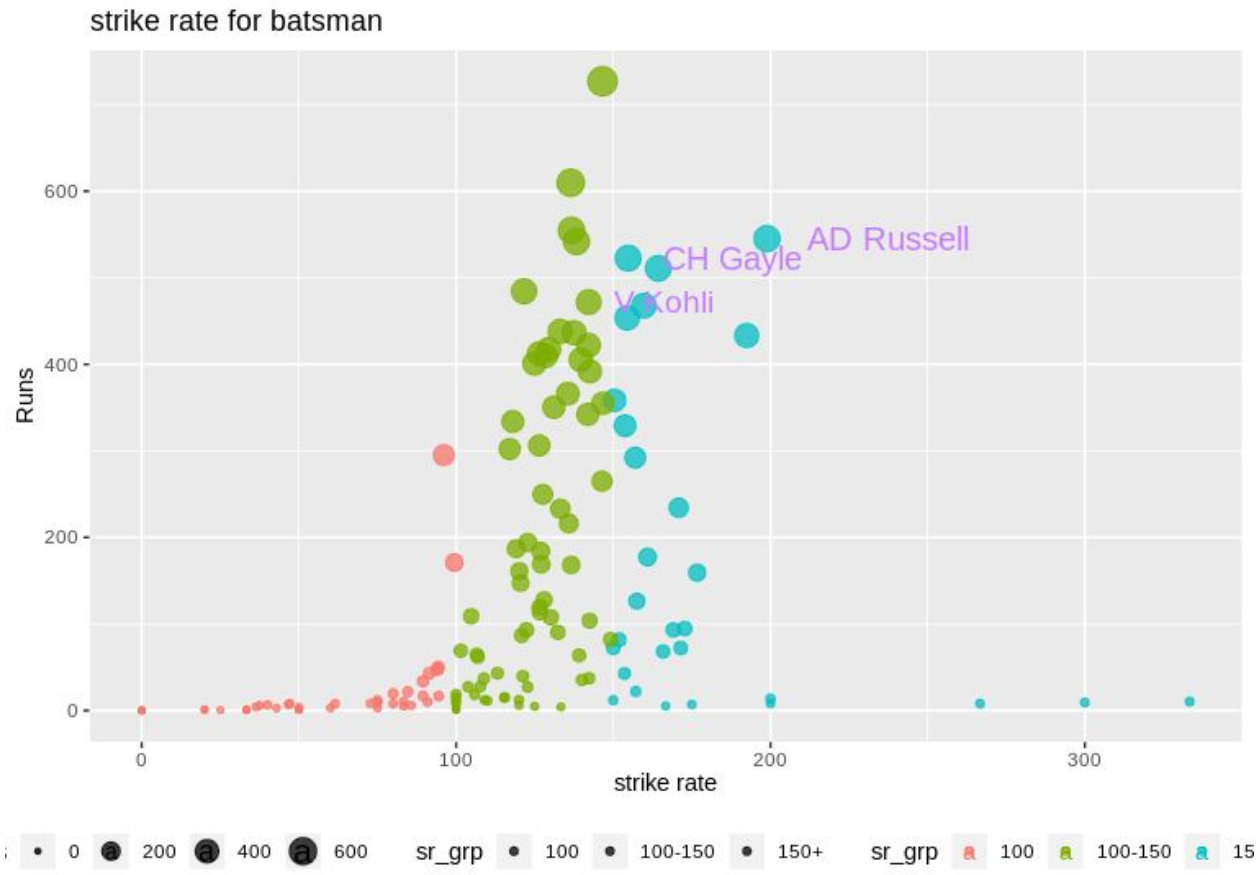
```
wins_1 <- matches%>%
  left_join(teams,by=c("toss_winner"="team") )%>%
  select(s_team,toss_winner,toss_decision)%>%
  group_by(s_team,toss_decision)%>%
  summarize(wins=n())


ggplot(wins_1,aes(x=s_team,y=wins,colour=toss_decision,fill=toss_decision))+
  geom_bar(position = "dodge",stat = "identity")+
  theme(legend.position="right")+
  scale_y_continuous(name="Toss decision")+
  scale_x_discrete(name="Toss winners and toss decisions")+
  ggtitle("Toss decisions by each Team")
```



##

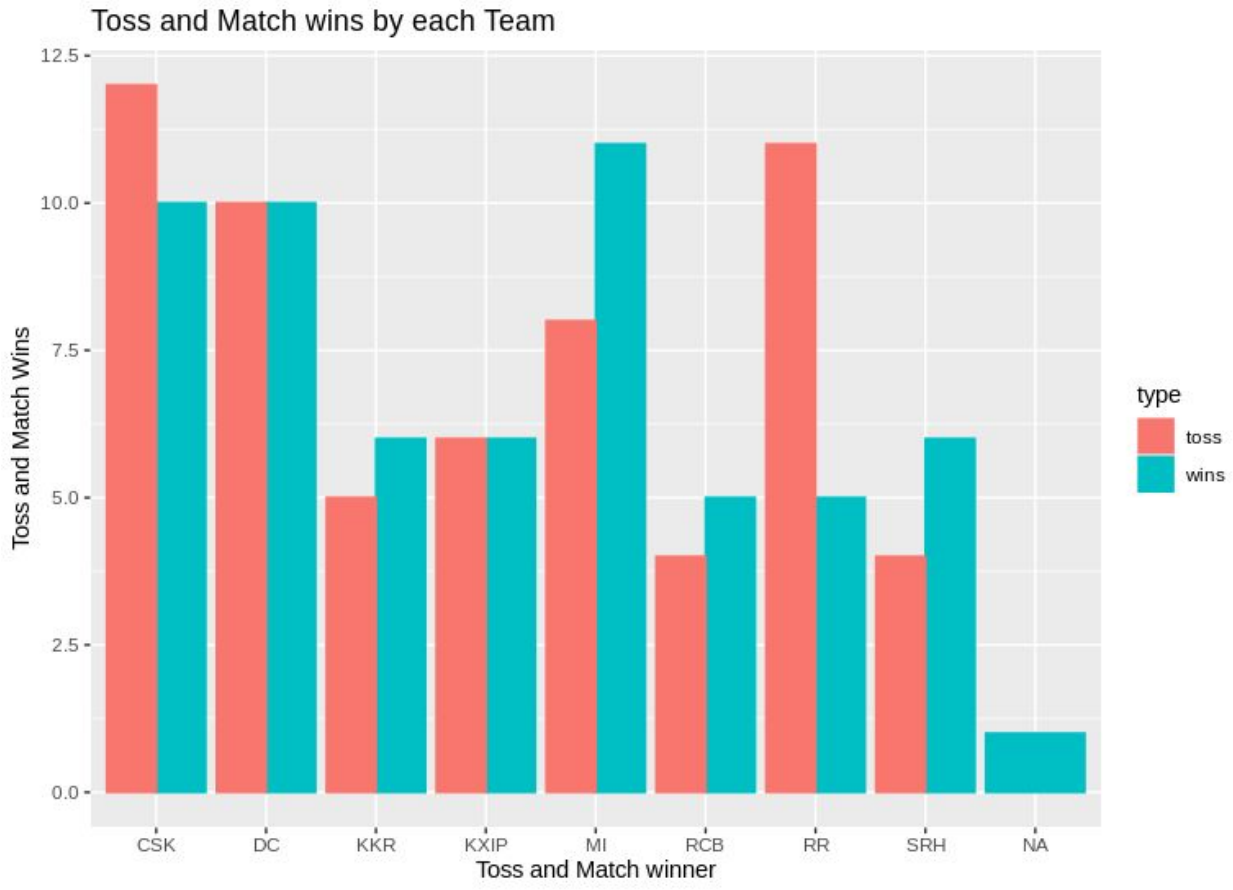
Strike rate of all batsman

```
Bat_sr<- data %>%
  left_join(matches,by=c("match_id"="id"))%>%
  left_join(teams,by=c("batting_team"="team"))%>%
  group_by(batsman)%>%
  summarize(balls=n(),runs=sum(batsman_runs))%>%
  mutate(sr=runs*100/balls)%>%
  arrange(desc(sr))%>%
  mutate(sr_grp=ifelse(sr<100,"100",ifelse(sr<150,"100-150","150+")))%>%
  mutate(player_lab=ifelse(batsman=="AD Russell","AD Russell",ifelse(batsman=="V
Sehwag","V Sehwag",ifelse(batsman=="V Kohli","V Kohli",ifelse(batsman=="CH Gayle","CH
Gayle",""))))))

ggplot(Bat_sr,aes(x=sr,y=runs,colour=sr_grp,fill=sr_grp,size=runs))+
  geom_jitter(show.legend = TRUE,alpha=.75)+
  theme(legend.position="bottom")+
  theme(legend.direction = "horizontal") +
  geom_text(aes(label=player_lab,hjust=-.25, colour="red"))+
  scale_y_continuous(name="Runs")+
  scale_x_continuous(name="strike rate")+
  ggtitle("strike rate for batsman   ")
```



Number of Toss and Match wins by each team

```
toss <- matches%>%
  left_join(teams,by=c("toss_winner"="team") )%>%
  select(s_team,toss_winner)%>%
  group_by(s_team)%>%
  summarize(wins=n())

toss$type <- "toss"

wins <-matches%>%
  left_join(teams,by=c("winner"="team") )%>%
  select(s_team,winner)%>%
  group_by(s_team)%>%
  summarize(wins=n())

wins$type <- "wins"

toss_w <- rbind(toss,wins)

toss_w <- toss_w %>%
  group_by(s_team, type)%>%
  summarize(wins=sum(wins))


ggplot(toss_w,aes(x=s_team,y=wins,colour=type,fill=type))+
  geom_bar(position = "dodge",stat = "identity")+
  theme(legend.position="right")+
  scale_y_continuous(name="Toss and Match Wins")+
  scale_x_discrete(name="Toss and Match winner")+
  ggtitle("Toss and Match wins by each Team")
```



##

Economy rate for all bowlers

```
ball_sr<- data %>%
  left_join(matches,by=c("match_id"="id"))%>%
  left_join(teams,by=c("bowling_team"="team"))%>%
  group_by(bowler)%>%
  summarize(balls=n(),runs=sum(total_runs,na.rm=TRUE))

ball_wk <-data %>%
  left_join(matches,by=c("match_id"="id"))%>%
  left_join(teams,by=c("bowling_team"="team"))%>%
  filter(dismissal_kind!="run out")%>%
  group_by(bowler)%>%
  summarize(wickets=sum(wickets,na.rm=TRUE))

ball_sr <-ball_sr%>%
```

```
  left_join(ball_wk,by=c("bowler"="bowler"))%>%
  mutate(sr=runs/wickets)%>%
  mutate(er=runs/(balls/6))%>%
  arrange(desc(sr))%>%
  mutate(sr_grp=ifelse(sr<10,"10",ifelse(sr<40,"11-40","41+")))%>%
  mutate(er_grp=ifelse(er<6,"6",ifelse(er<10,"6-10","11+")))%>%
  mutate(player_l=ifelse(bowler=="SL Malinga","SL Malinga",ifelse(bowler=="DJ Bravo","DJ
Bravo",ifelse(bowler=="R Ashwin","R Ashwin",ifelse(bowler=="DW Steyn","DW Steyn",""))))))
```

```
ggplot(ball_sr,aes(x=er,y=runs,colour=er_grp,fill=er_grp,size=runs))+
  geom_jitter(show.legend = TRUE,alpha=.75)+
  theme(legend.position="bottom")+
  theme(legend.direction = "horizontal") +
  geom_text(aes(label=player_l,hjust=-.25, colour="red"))+
  scale_y_continuous(name="Runs")+
  scale_x_continuous(name="Economy rate ")+
  ggtitle("Economy rate for bowlers  ")
```