IPL Auction Analysis

Considered Factors For IPL Auction Analysis

ALL ROUNDER PLAYERS

TOP Wicketkeepers

BOWLERS WITH BEST STRIKE RATE

BOWLERS WITH GOOD ECONOMY

BATSMAN WITH HIGH STRIKE RATE

BATSMAN WITH GOOD AVERAGE

HARD HITTING PLAYERS

DATASETS USED : *IPL_Ball ,IPL_matches*

TABLES CREATED: *IPL_Ball , IPL_match*

Query Used to create tables :

```
create table IPL_Ball(
id int ,inning int ,
over int ,ball int ,batsman varchar, non_striker varchar,
bowler varchar,batsman_runs int,
extra_runs  int ,total_runs  int,
is_wicket int ,dismissal_kind varchar,
player_dismissed  varchar,
fielder varchar, extras_type varchar ,
batting_team varchar, bowling_team varchar
);

copy IPL_Ball from 'C:\Program
Files\PostgreSQL\15\data\dataset\IPL_Ball.csv' delimiter ','
csv header;

select * from IPL_Ball;
```

Query Used to create tables :

```
create table IPL_match(
id int,
city varchar,
date date,
player_of_match varchar,          venue varchar,
neutral_venue  int,
team1 varchar,  team2  varchar, toss_winner  varchar,
toss_decision  varchar, winner varchar, result  varchar,
result_margin  int ,eliminator varchar, method varchar, umpire1 varchar,
umpire2 varchar
);

copy IPL_match from 'C:\Program
Files\PostgreSQL\15\data\dataset\IPL_matches.csv' delimiter ',' csv
header;

select * from IPL_match;
```

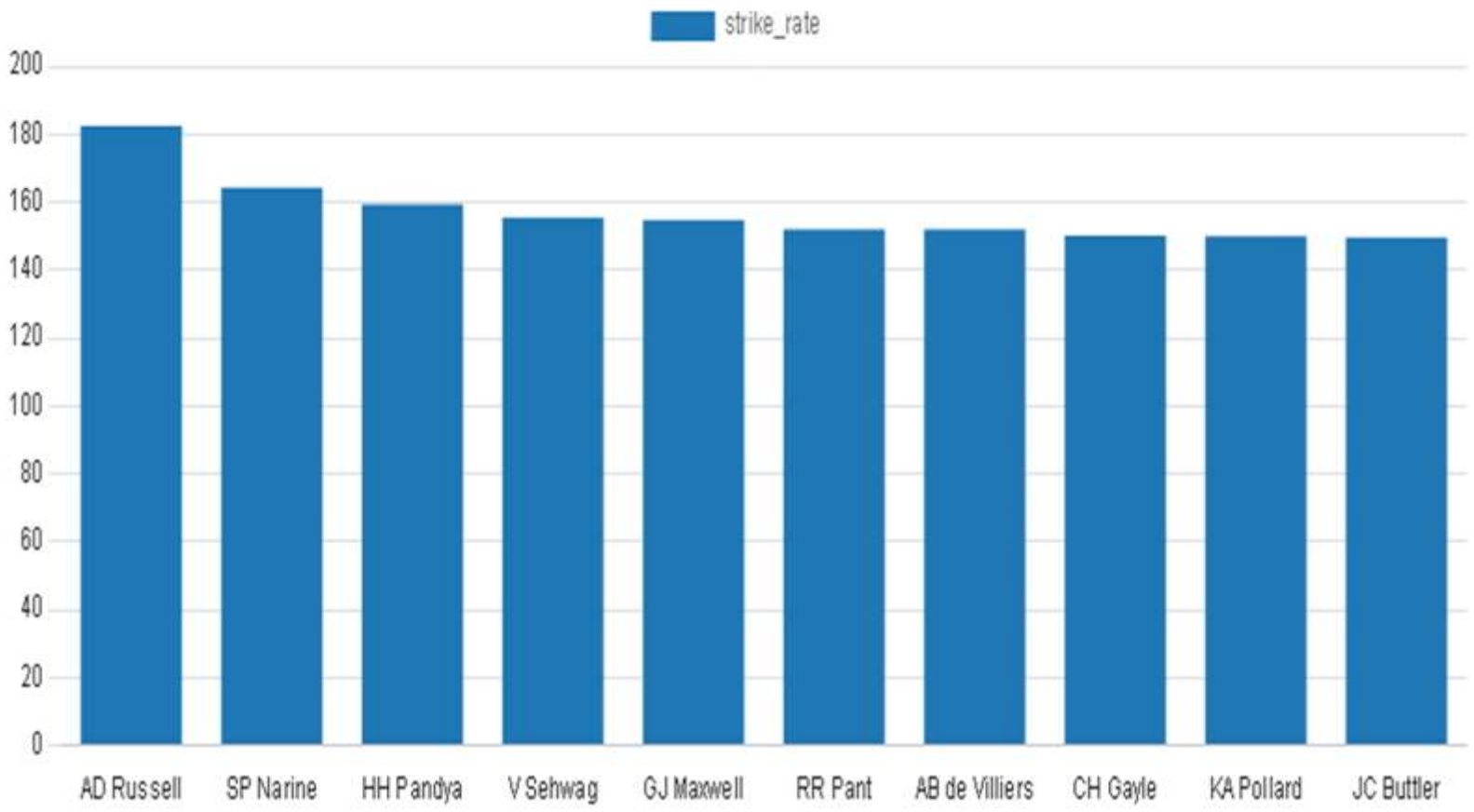
BATSMAN WITH HIGH STRIKE RATE

Q1. To get 2-3 players with high S.R who have faced at least 500 balls

Query used :

```
select batsman,
((sum(batsman_runs))/
(((count(batsman)) - (count(case when
extras_type = 'wides' then  1   end))*1.0)))
*100
as strike_rate
from ipl_ball
group by batsman
having count(batsman) >= 500
order by strike_rate desc limit 10;
```

RESULT



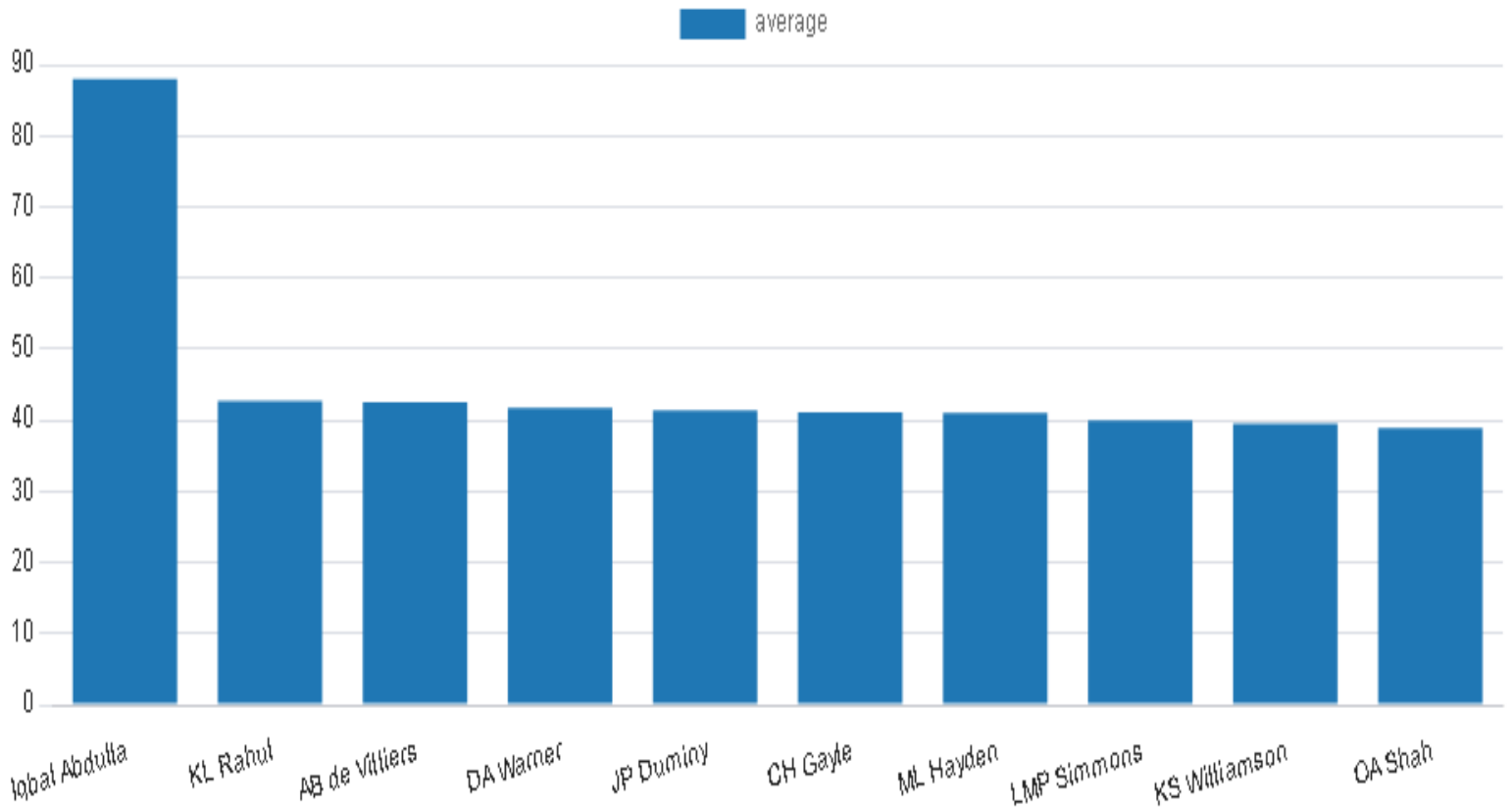
BATSMAN WITH GOOD AVERAGE

Q2. To get 2-3 players with good average who have played more than 2 IPL seasons

Query used :

```
select batsman, ((sum(batsman_runs) *
1.0 )/ (count(case when is_wicket = 1
then 1   end))) as average,
count(distinct extract (year
from(b.date))) as
no_of_seasons_played
from ipl_ball
inner join
ipl_match b
on ipl_ball.id=b.id
group by batsman
having (count(distinct extract (year
from(b.date)))) > 2 and
(sum(is_wicket))>=1
order by average desc
limit 10;
```

RESULT



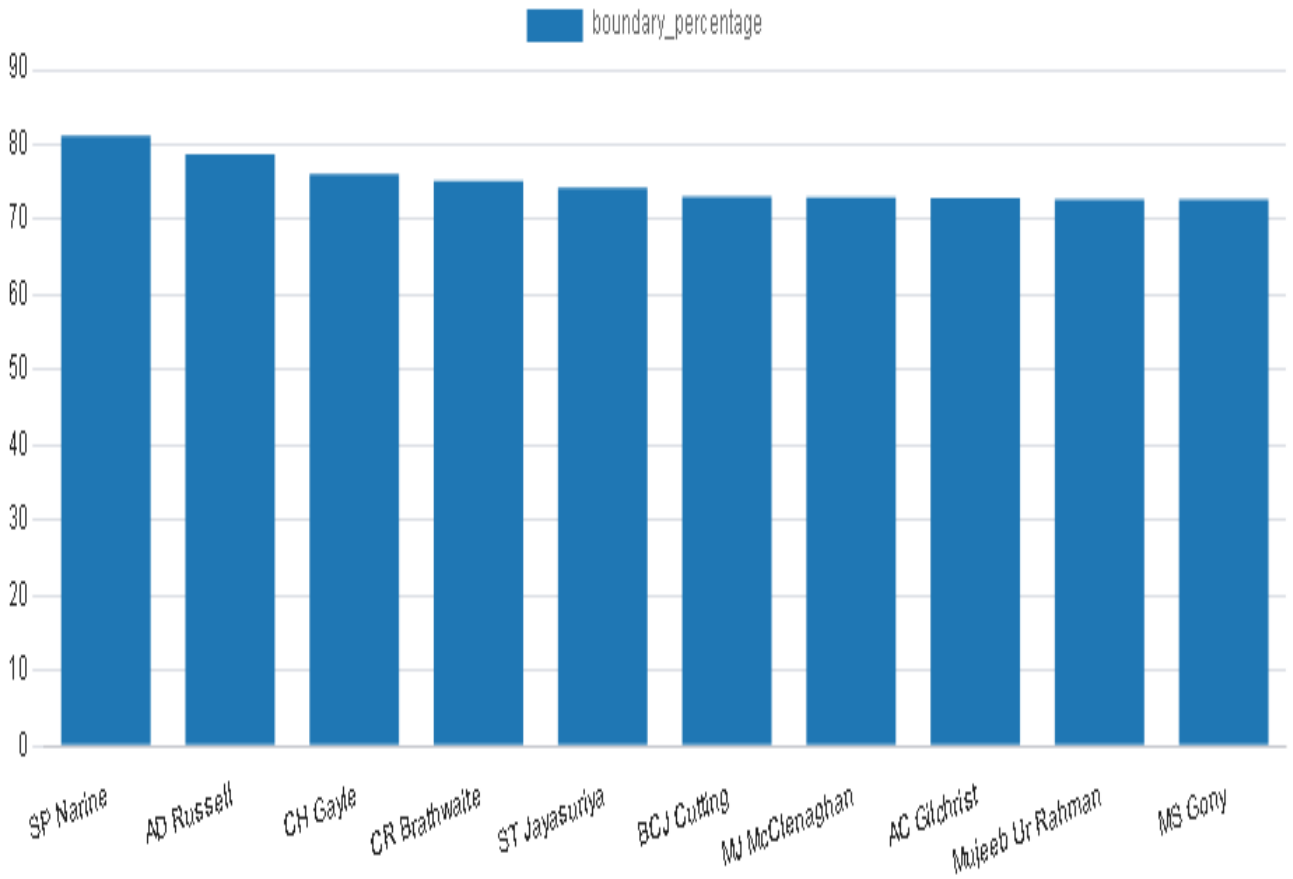
HARD-HITTING BATSMAN

Q3. To get 2-3 Hard-hitting players who have scored most runs in boundaries and have played more the 2 ipl season.

Query used :

```
select b.batsman as player_name,
((sum(case when b.batsman_runs = 4 then 4
when b.batsman_runs = 6 then 6 else 0
end)*1.0 )/
sum(b.batsman_runs)) * 100 AS
boundary_percentage,
count(distinct extract(year from m.date)) as
seasons_played
from ipl_ball b
join ipl_match m on b.id = m.id
group by b.batsman
having count(distinct extract(year from m.date))
> 2
order by boundary_percentage desc
limit 10;
```

RESULT



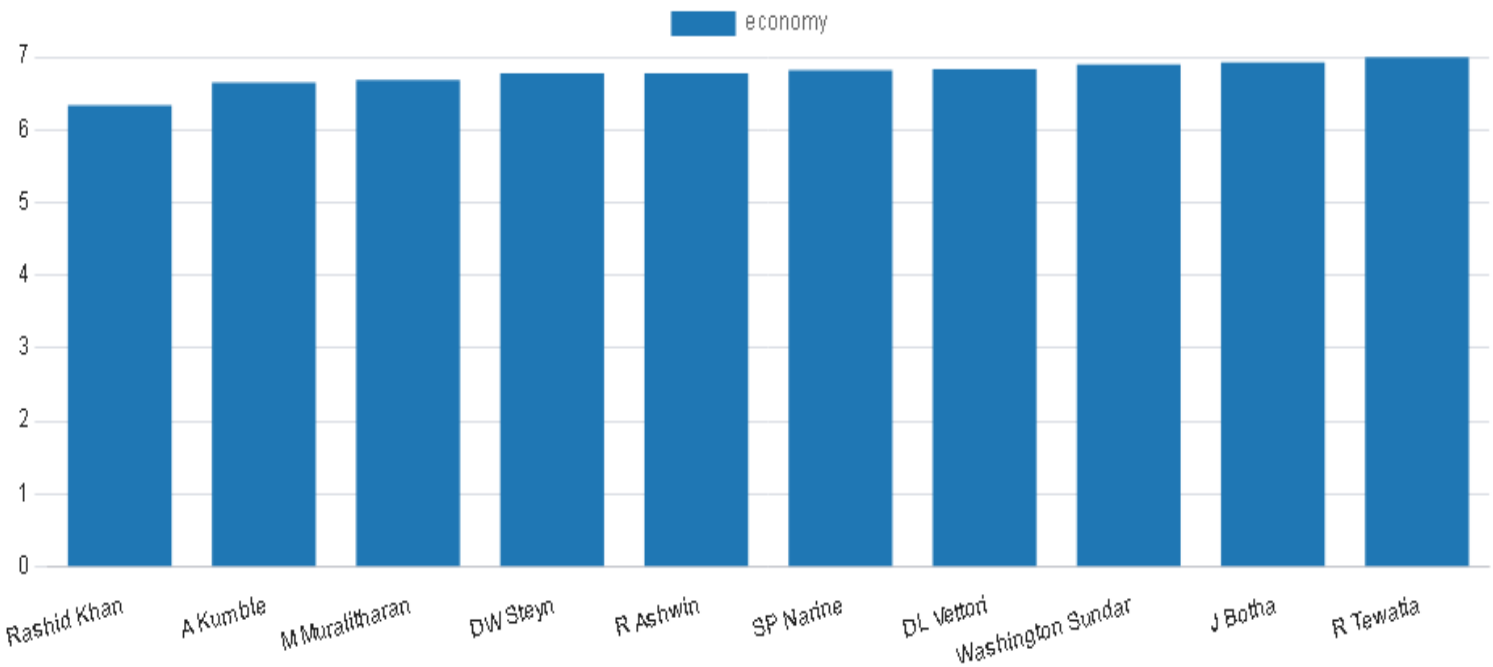
BOWLERS WITH GOOD ECONOMY

Q4. To get 2-3 bowlers with good economy who have bowled at least 500 balls in IPL so far.

Query used :

```
select bowler as bowler_name,
(((sum(total_runs) )/ ((count(
bowler)/ (6.0))))) as economy
from ipl_ball
group by bowler
having count(bowler) >= 500
order by economy asc
limit 10;
```

RESULT



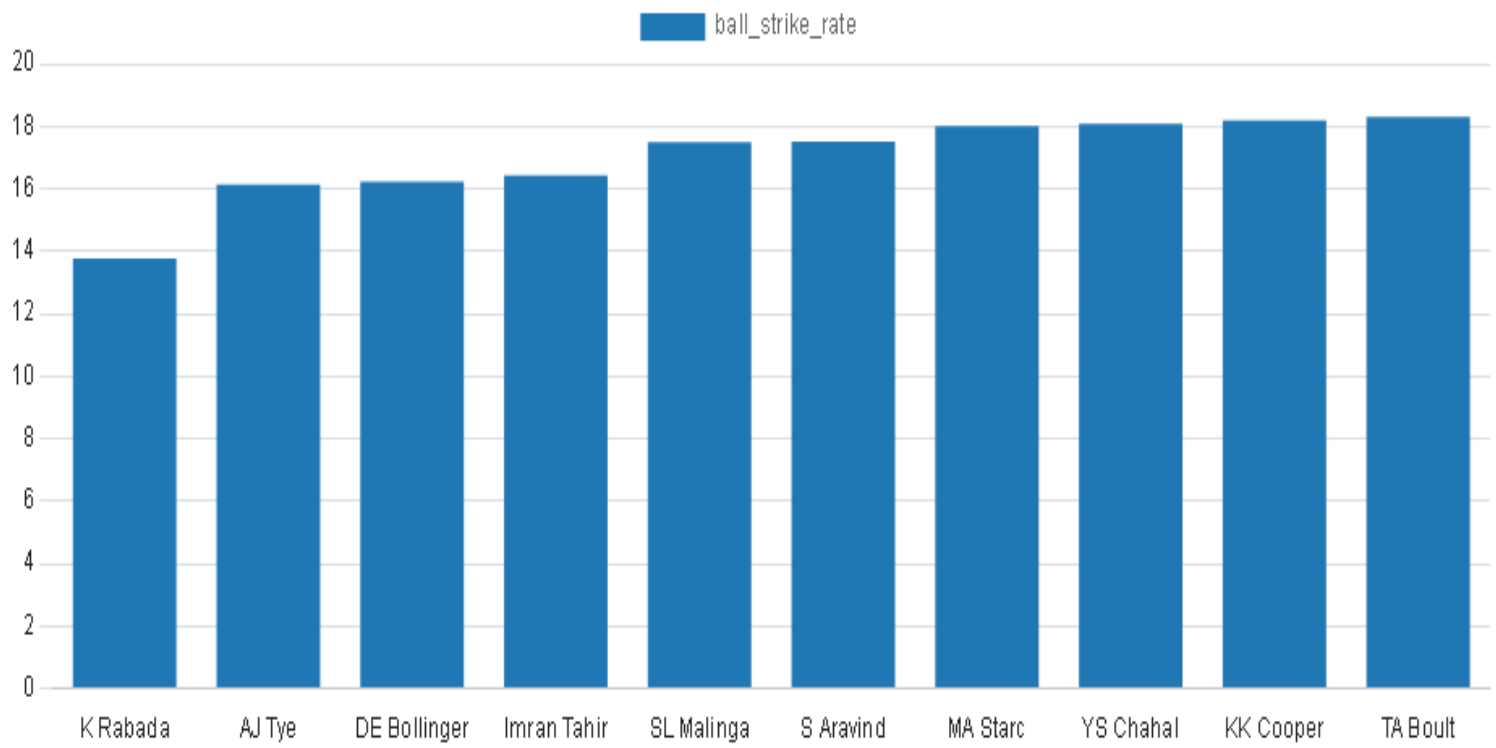
BOWLERS WITH BEST STRIKE RATE

Q5. To get 2-3 bowlers with the best strike rate and who have bowled at least 500 balls in IPL so far.

Query used :

```
select bowler as
bowler_name,count(bowler) as
number_of_balls_bowled,
((count(bowler)*1.0 ) / (sum(case when
dismissal_kind = 'bowled' then 1
when dismissal_kind = 'caught' then 1
when dismissal_kind = 'caught and bowled'
then 1
when dismissal_kind = 'hit wicket' then 1
when dismissal_kind = 'lbw' then 1
when dismissal_kind = 'stumped' then 1
else 0 end))) as ball_strike_rate
from ipl_ball group by bowler
having count(bowler) >= 500
order by ball_strike_rate
LIMIT 10;
```

RESULT



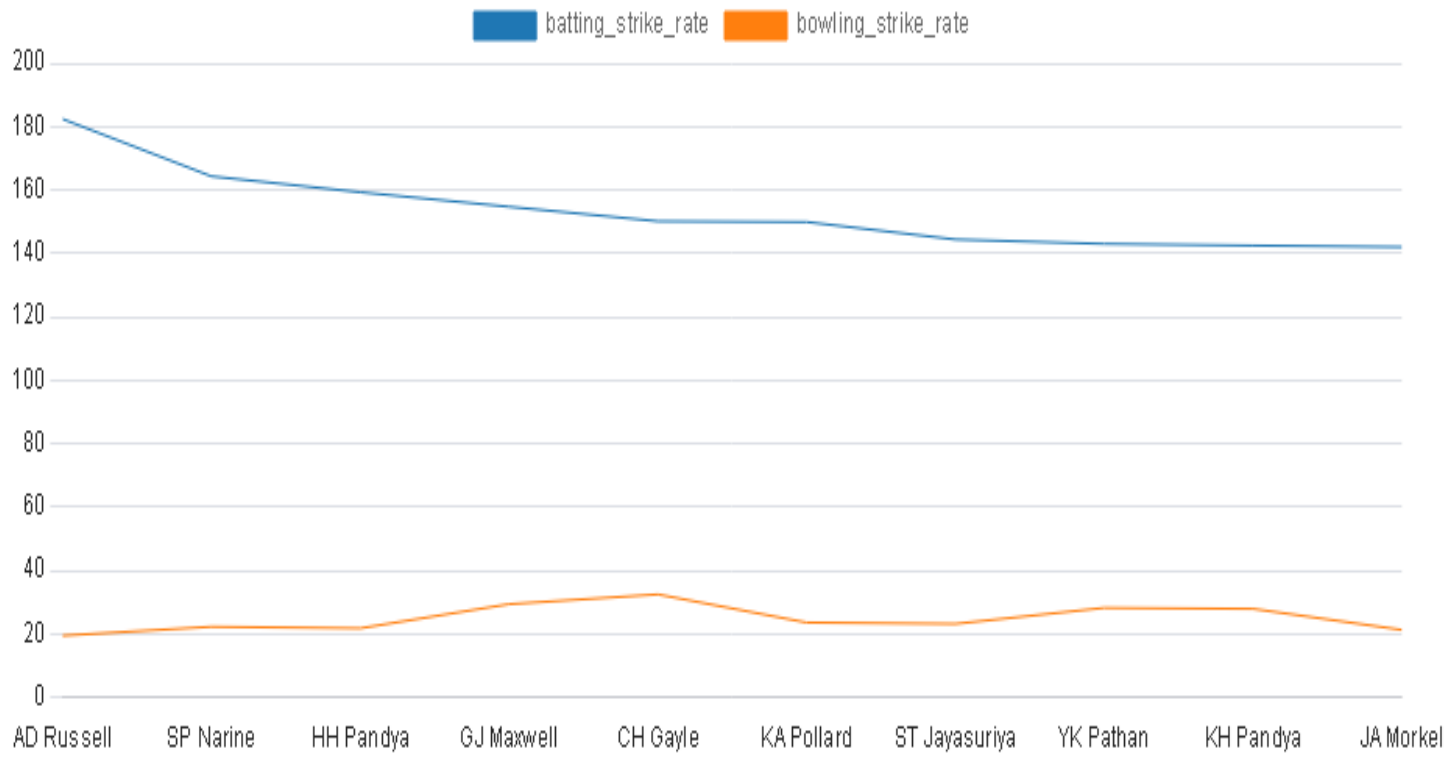
ALL ROUNDER PLAYERS

Q6. To get 2-3 All_rounders with the best batting as well as bowling strike rate and who have faced at least 500 balls in IPL so far and have bowled minimum 300 balls

Query used :

```
select
    batting.batsman as player_name,
    batting.batting_strike_rate,
    bowling.bowler_name,
    bowling.number_of_balls_bowled,
    bowling.bowling_strike_rate
from
    ( select batsman,((sum(batsman_runs)) /
((count(batsman) - count(case when extras_type = 'wides' then 1 end)) * 1.0)) *
100 as batting_strike_rate
from ipl_ball group by batsman  having count(batsman) >= 500 ) as batting
inner join
    (select bowler as bowler_name,count(bowler) as number_of_balls_bowled,
((count(bowler) * 1.0) / (sum(case when dismissal_kind = 'bowled' then 1
                                   when dismissal_kind = 'caught' then 1
                                   when dismissal_kind = 'caught and
bowled' then 1
                                   when dismissal_kind = 'hit wicket' then 1
                                   when dismissal_kind = 'lbw' then 1
                                   when dismissal_kind = 'stumped' then 1
ELSE 0 END)))
as bowling_strike_rate from ipl_ball group by bowler
having count(bowler) >= 300)as bowling
ON batting.batsman = bowling.bowler_name
ORDER BY
    batting.batting_strike_rate DESC,
    bowling.bowling_strike_rate
LIMIT 10;
```

RESULT



BEST WICKETKEEPER

Q7. To get the criteria for choosing the two best wicket keepers from a list of wicketkeeper names

Criteria used:

Let's consider we have a list of wicketkeepers: W1, W2, W3, W4, W5, and so on. Now we have to evaluate each wicketkeeper based on the given criteria and assign a score to each criteria. The wicketkeepers with the highest cumulative scores will be considered as best wicketkeeper.

Batting Ability : Assign a score to each wicketkeeper based on their batting performance. We can use a scale of 1 to 10, where a higher score indicates better batting ability.
For example:
W1: Batting Ability = 8
W2: Batting Ability = 9
W3: Batting Ability = 7
W4: Batting Ability = 9
W5: Batting Ability = 8

Adaptability : Assign a score to each wicketkeeper based on their ability to adapt to different match situations and batting roles.
For example:
W1: Adaptability = 7
W2: Adaptability = 8
W3: Adaptability = 6
W4: Adaptability = 7
W5: Adaptability = 9

Fielding Skills : Assign a score to each wicketkeeper based on their fielding skills, including catching and stumping abilities.
For example:
W1: Fielding Skills = 9
W2: Fielding Skills = 8
W3: Fielding Skills = 7
W4: Fielding Skills = 6
W5: Fielding Skills = 8

Communication and Leadership : Assign a score to each wicketkeeper based on their communication skills and leadership qualities.
For example:
W1: Communication and Leadership = 6
W2: Communication and Leadership = 9
W3: Communication and Leadership = 8
W4: Communication and Leadership = 7
W5: Communication and Leadership = 7

Versatility : Assign a score to each wicketkeeper based on their ability to contribute with bowling (spin or medium pace).
For example:
W1: Versatility = 5
W2: Versatility = 7
W3: Versatility = 6
W4: Versatility = 9
W5: Versatility = 6

Consistency and Experience: Assign a score to each wicketkeeper based on their consistent performances and experience in IPL .
For example:
W1: Consistency and Experience = 8
W2: Consistency and Experience = 9
W3: Consistency and Experience = 7
W4: Consistency and Experience = 8
W5: Consistency and Experience = 9

Now, based on these scores for each criterion, we can calculate the cumulative score for each wicketkeeper by summing up the scores:

W1: Cumulative Score = 8 + 7 + 9 + 6 + 5 + 8 = 43
W2: Cumulative Score = 9 + 8 + 8 + 9 + 7 + 9 = 50
W3: Cumulative Score = 7 + 6 + 7 + 8 + 6 + 7 = 41
W4: Cumulative Score = 9 + 7 + 6 + 7 + 9 + 8 = 46
W5: Cumulative Score = 8 + 9 + 8 + 7 + 6 + 9 = 47

Based on these cumulative scores, we can conclude that W2, W5 are the top 2 wicketkeeper.

EXPLANATION OF EACH FACTOR APPLIED IN ORDER TO FIND TOP 2 BEST WICKETKEEPER

- **Batting Ability**: In IPL, scoring runs quickly is crucial. A good wicketkeeper should be able to contribute to the team's run total with aggressive batting. Look for wicketkeepers who have a strong strike rate, good shot selection, and the ability to hit boundaries consistently.

- **Adaptability:** IPL tournament often needs quick decision-making and adaptability to different match situations. The wicketkeeper should be able to adjust their batting style according to the team's needs, whether it's playing aggressively at the start of the innings or playing a responsible anchor role in the middle or late stages.

- **Fielding Skills**: Defensively, a wicketkeeper's role is to catch and stump the batsmen. Find for wicketkeepers with excellent catching skills, agility, and reflexes. They should be able to anticipate and react quickly to edges and stumpings, potentially leading to crucial dismissals.

- **Communication and Leadership**: A wicketkeeper is often the closest player to the bowler and has a good view of the field. They should have strong communication skills to coordinate with the bowlers, provide feedback, and make strategic decisions on the field. Leadership qualities are also beneficial as they can effectively guide and motivate the team.

- **Versatility**: While not essential, having a wicketkeeper who can also contribute with a few overs of spin or medium pace can be an added advantage. This provides flexibility in team selection and allows for tactical changes during the match.

- **Consistency and Experience**: Look for wicketkeepers who have a track record of consistent performances in IPL. Experience in high-pressure situations can also be valuable as it demonstrates the ability to handle the demands of a competitive tournament.

TABLE CREATED FOR ANALYSIS ON ADDITIONAL QUESTIONS

TABLE CREATED: deliveries, matches

QUERY USED :

```
create table deliveries(
id int ,inning int ,over int , ball int ,
batsman varchar , non_striker varchar , bowler varchar,
batsman_runs int,
extra_runs int ,total_runs int ,          is_wicket              int ,dismissal_kind varchar,
player_dismissed varchar, fielder varchar , extras_type varchar ,
batting_team  varchar , bowling_team varchar
);

copy deliveries from 'C:\Program Files\PostgreSQL\15\data\dataset\IPL_Ball.csv' delimiter ',' csv header;

select * from deliveries;
```

QUERY USED :

```
create table matches(
id int,
city varchar,
date date,
player_of_match varchar ,  venue varchar,
neutral_venue int,
team1 varchar, team2  varchar ,  toss_winner varchar ,  toss_decision  varchar ,  winner varchar,
result varchar , result_margin int ,
Eliminator varchar ,  method varchar,
umpire1 varchar , umpire2 varchar
);

copy matches from 'C:\Program Files\PostgreSQL\15\data\dataset\IPL_matches.csv' delimiter ',' csv header;

select * from matches;
```

Q1. Get the count of cities that have hosted an IPL match

Query used :

```
select count(distinct city) from matches;
```

OUTPUT:
Count : 33

Q2. Create table deliveries_v02 with all the columns of the table ‘deliveries’ and an additional column ball_result containing values boundary, dot or other depending on the total_run (boundary for >= 4, dot for 0 and other for any other number)

Query used :

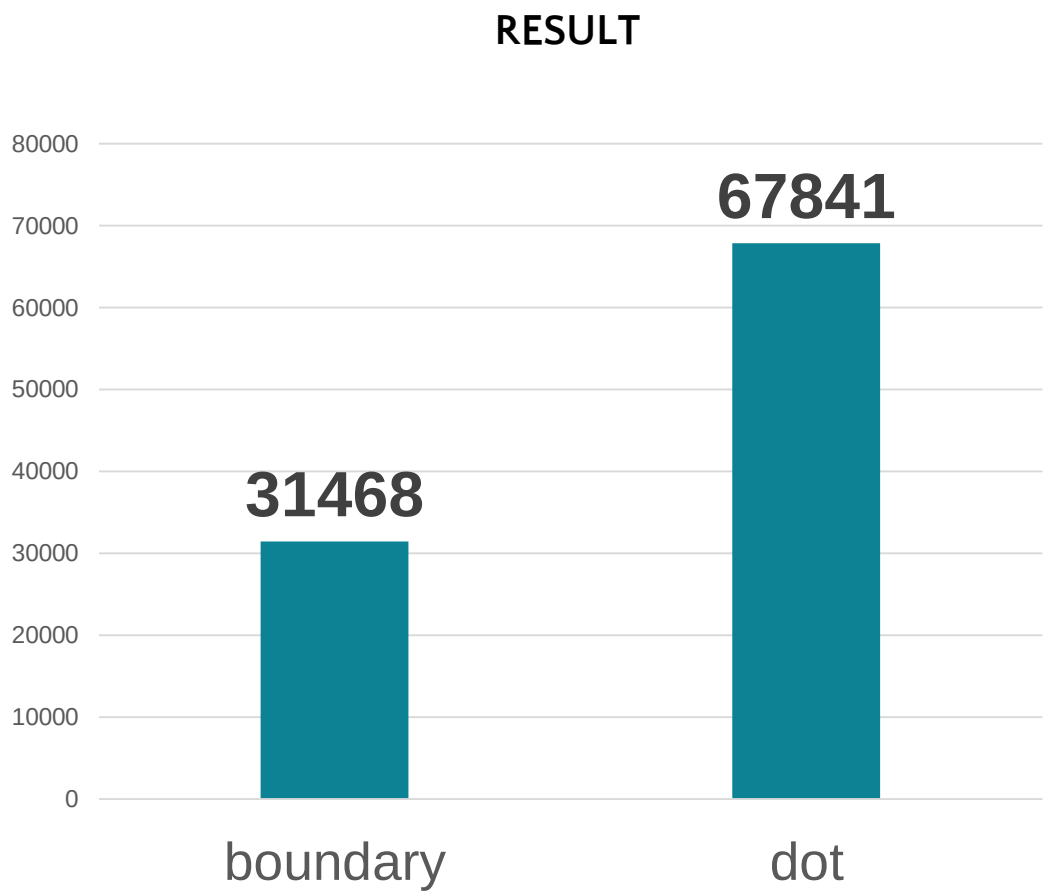
```
create table deliveries_v02 as
select*,
    case
        when total_runs >= 4 then 'boundary'
        when total_runs = 0 then 'dot'
        else 'other'
    end as ball_result
from deliveries;

select * from deliveries_v02;
```

Q3. Write a query to fetch the total number of boundaries and dot balls from the deliveries_v02 table.

Query used :

```
select ball_result , count(*) as count
from deliveries_v02
where ball_result in('boundary',
'dot')
group by ball_result;
```

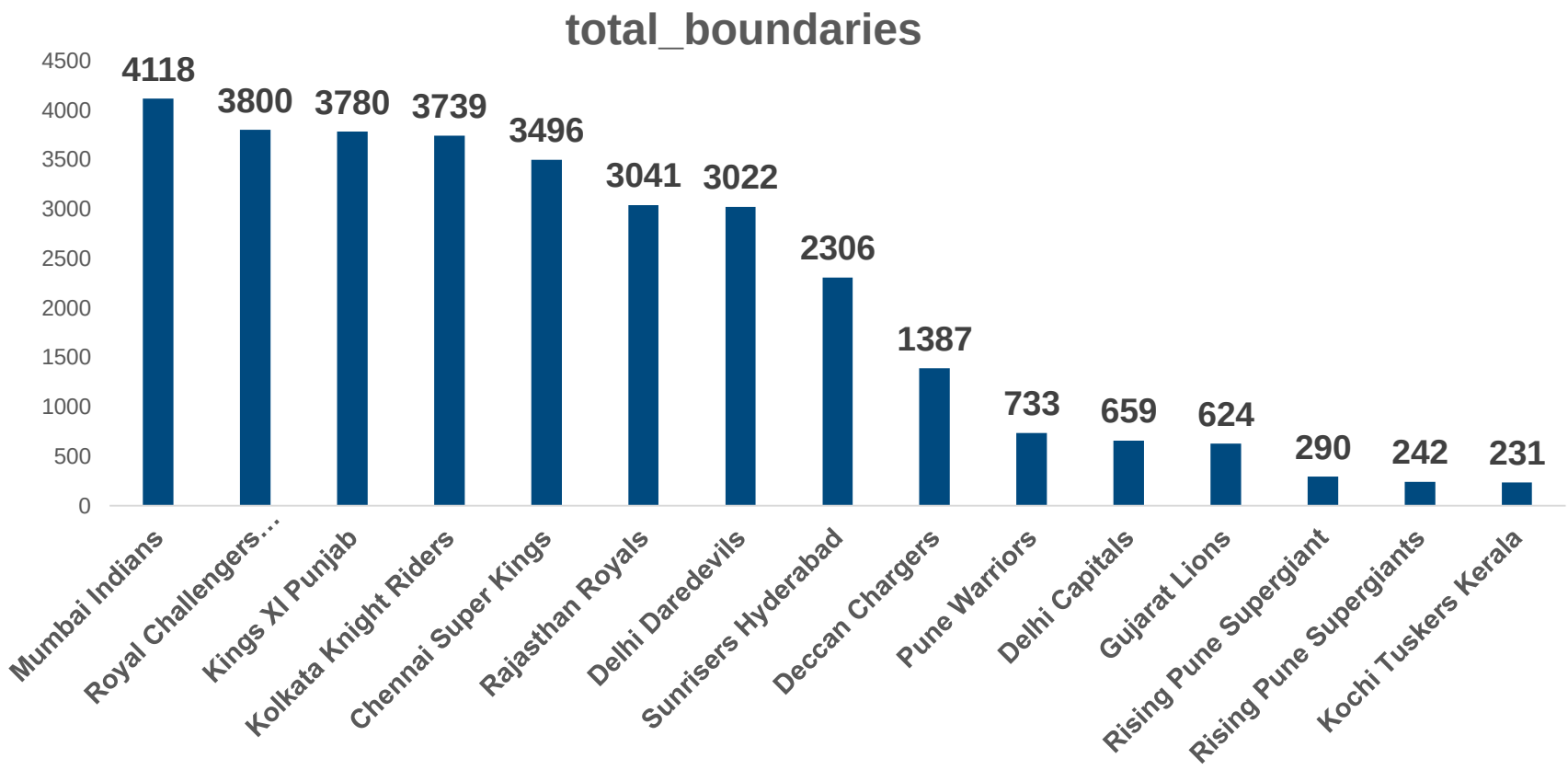


Q4. Write a query to fetch the total number of boundaries scored by each team from the deliveries_v02 table and order it in descending order of the number of boundaries scored.

Query used :

```
select batting_team,
count (*) as
total_boundaries
from  deliveries_v02
where ball_result =
'boundary'
group by batting_team
order by
total_boundaries desc;
```

RESULT

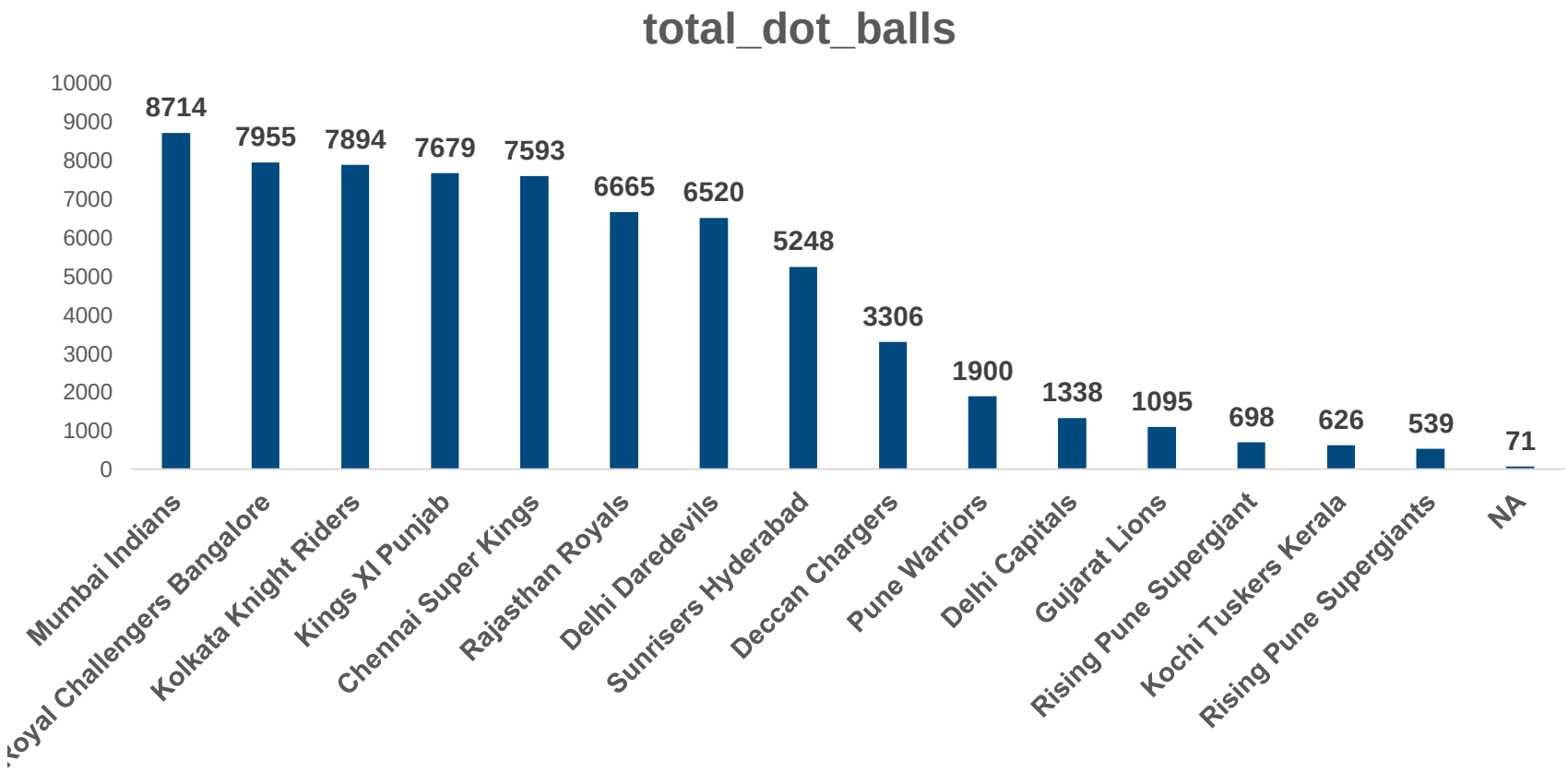


Q5. Write a query to fetch the total number of dot balls bowled by each team and order it in descending order of the total number of dot balls bowled

Query used :

```
select bowling_team,
count(*) as total_dot_balls
from  deliveries_v02
where ball_result = 'dot'
group by bowling_team
order by total_dot_balls
desc;
```

RESULT

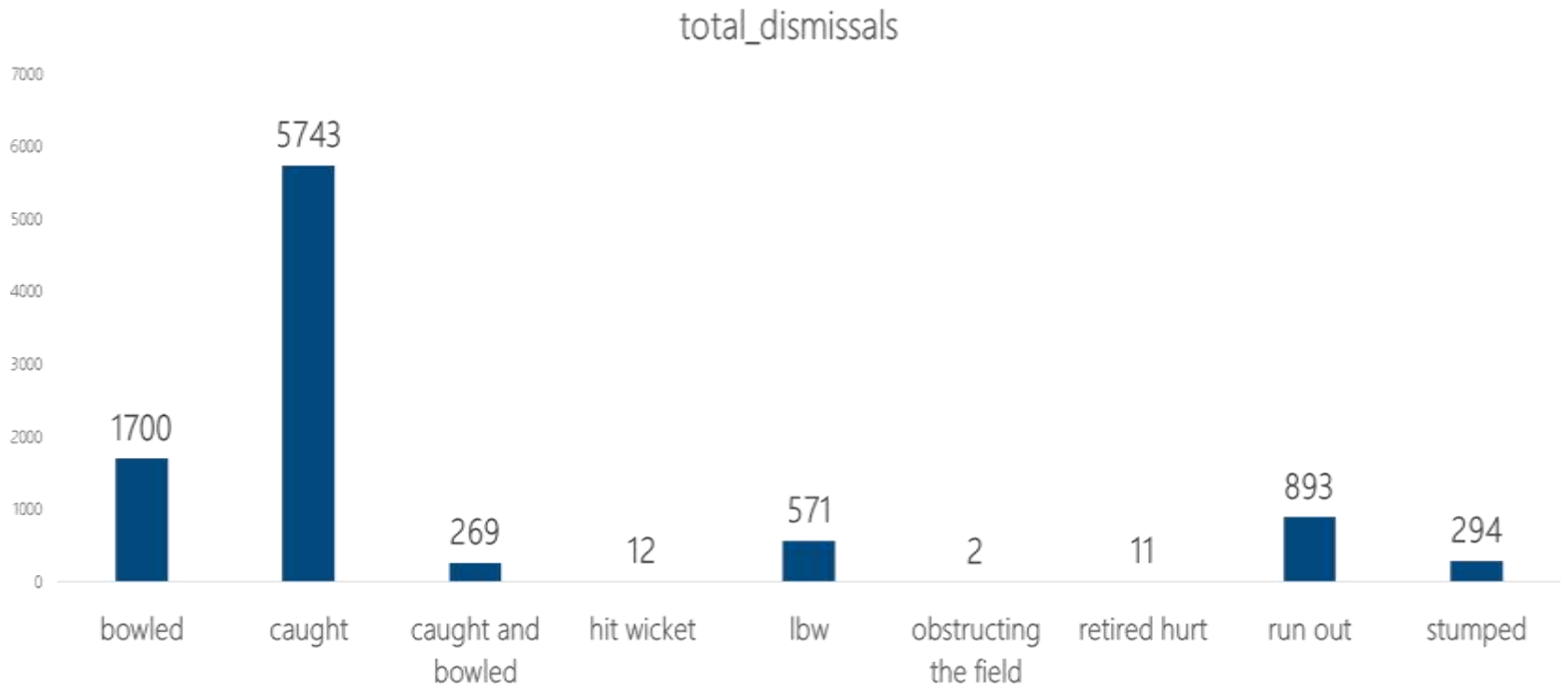


Q6. Write a query to fetch the total number of dismissals by dismissal kinds where dismissal kind is not NA

Query used :

```
select dismissal_kind,
count(*) as
total_dismissals
from deliveries
where not
dismissal_kind = 'NA'
group by
dismissal_kind;
```

RESULT

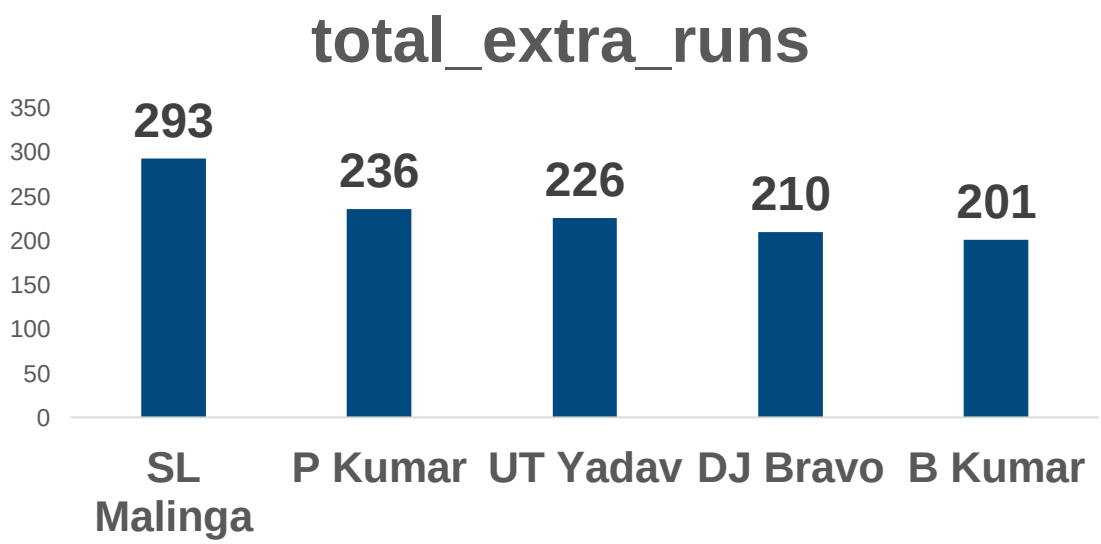


Q7. Write a query to get the top 5 bowlers who conceded maximum extra runs from the deliveries table

Query used :

```
select bowler ,
sum(extra_runs) as
total_extra_runs
from deliveries
group by  bowler
order by total_extra_runs
desc
LIMIT 5;
```

RESULT



Q8. Write a query to create a table named deliveries_v03 with all the columns of deliveries_v02 table and two additional column (named venue and match_date) of venue and date from table matches

Query used :

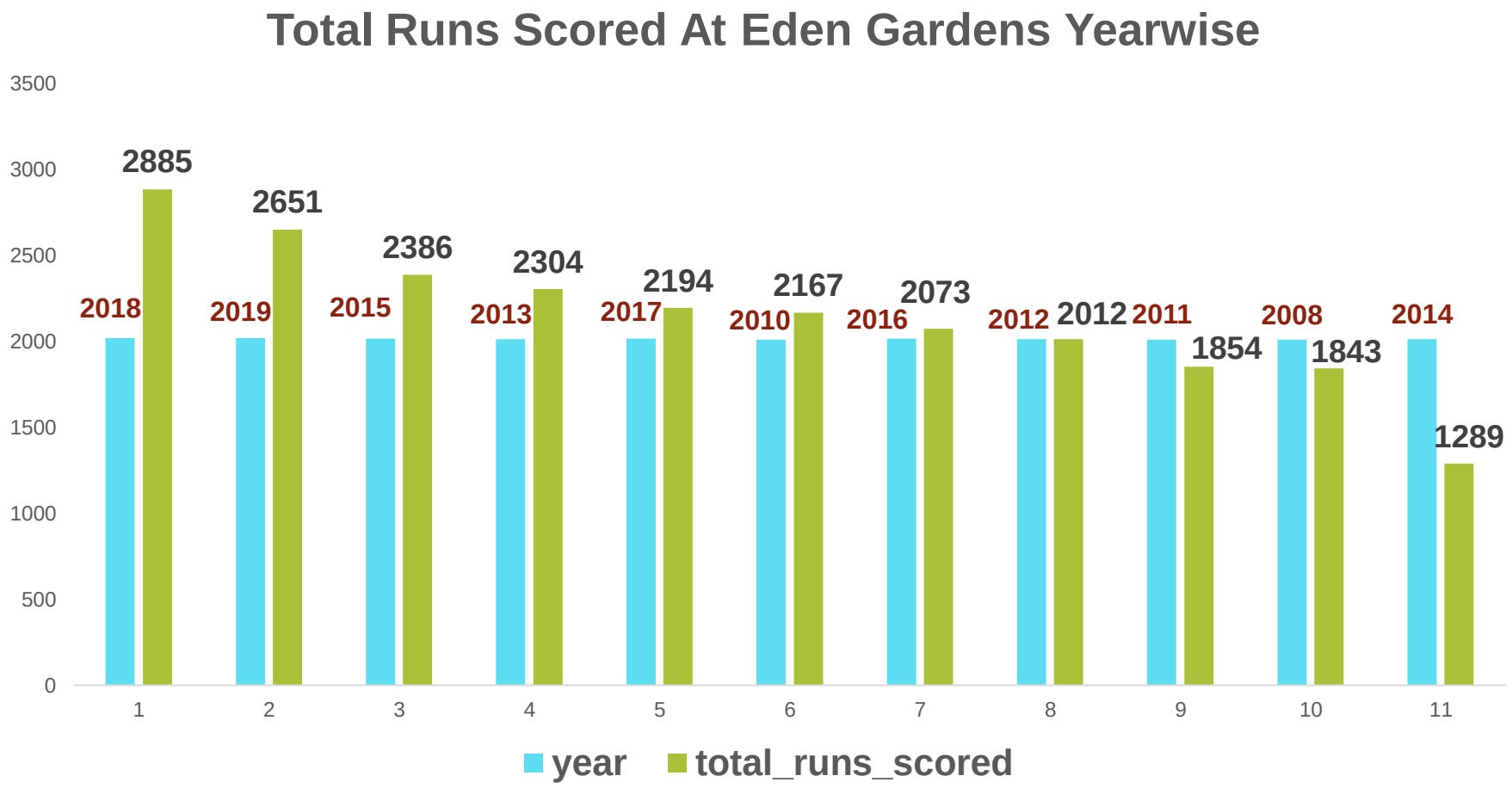
```
create table deliveries_v03 as
select d.*,m.venue,
m.date as match_date
from deliveries_v02  as d
inner join matches as m on d.id = m.id;

select * from  deliveries_v03;
```

Q9. Write a query to fetch the year-wise total runs scored at Eden Gardens and order it in the descending order of total runs scored.

Query used :

```
select extract(year from match_date)
as year,
sum(total_runs) as total_runs_scored
from deliveries_v03
where venue = 'Eden Gardens'
group by year
order by total_runs_scored desc;
```



Thank You