IPL_Data_Analysis

May 24, 2023

1 Problem Statements: Indian Premier League(IPL) Data Analysis

1.1 Description:

- **The Indian Premier League (IPL) is a professional Twenty20 cricket league in India contested by eight teams. The league was founded by the Board of Control for Cricket in India (BCCI) in 2007. The IPL is the most-popular cricket league in the world; in 2014, it was ranked sixth by average attendance among all sports leagues. In 2010, the IPL became the first sporting event to be broadcast live on YouTube. The brand value of the IPL in 2022 was 90,038 crore (US$11 billion). According to BCCI, the 2015 IPL season contributed 1,150 crore (US$140 million) to the GDP of the economy of India. In December 2022, the IPL became a decacorn valued at US$10.9 billion, registering a 75% growth in dollar terms since 2020 when it was valued at $6.2 billion, according to a report by consulting firm D & P Advisory.**
- **The IPL is a very popular event in India, with millions of people tuning in to watch the matches. The matches are also broadcast live in other countries around the world. The IPL has helped to popularize cricket in India and around the world, and it has also helped to generate a lot of revenue for the BCCI.**

2 1. Importing Libraries

[ ]:

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.simplefilter(action = "ignore", category = FutureWarning)
plt.style.use('dark_background')
```

3 2. Datasets

3.1 2.1 Deliveries Data

[ ]:

```
deliveries=pd.read_csv("deliveries.csv")
deliveries.head()
```

[ ]:

```
   match_id  inning           batting_team                bowling_team  over  \
0         1       1  Kolkata Knight Riders  Royal Challengers Bangalore     1
1         1       1  Kolkata Knight Riders  Royal Challengers Bangalore     1
2         1       1  Kolkata Knight Riders  Royal Challengers Bangalore     1
3         1       1  Kolkata Knight Riders  Royal Challengers Bangalore     1
4         1       1  Kolkata Knight Riders  Royal Challengers Bangalore     1

   ball      batsman  non_striker   bowler  is_super_over  …  bye_runs  \
0     1   SC Ganguly  BB McCullum  P Kumar              0  …         0
1     2  BB McCullum   SC Ganguly  P Kumar              0  …         0
2     3  BB McCullum   SC Ganguly  P Kumar              0  …         0
3     4  BB McCullum   SC Ganguly  P Kumar              0  …         0
4     5  BB McCullum   SC Ganguly  P Kumar              0  …         0

   legbye_runs  noball_runs  penalty_runs  batsman_runs  extra_runs  \
0            1            0             0             0           1
1            0            0             0             0           0
2            0            0             0             0           1
3            0            0             0             0           0
4            0            0             0             0           0

   total_runs  player_dismissed dismissal_kind fielder
0           1               NaN            NaN     NaN
1           0               NaN            NaN     NaN
2           1               NaN            NaN     NaN
3           0               NaN            NaN     NaN
4           0               NaN            NaN     NaN

[5 rows x 21 columns]
```

3.2 2.2 Matches Datasets

[ ]:

```
matches=pd.read_csv("matches.csv")
matches.head()
```

[ ]:

```
   id  season       city        date                  team1  \
0   1    2008  Bangalore  2008-04-18  Kolkata Knight Riders
1   2    2008 Chandigarh  2008-04-19    Chennai Super Kings
2   3    2008      Delhi  2008-04-19       Rajasthan Royals
3   4    2008     Mumbai  2008-04-20         Mumbai Indians
4   5    2008    Kolkata  2008-04-20        Deccan Chargers
```

```
                         team2                  toss_winner toss_decision  \
0  Royal Challengers Bangalore  Royal Challengers Bangalore         field
1              Kings XI Punjab          Chennai Super Kings           bat
2             Delhi Daredevils             Rajasthan Royals           bat
3  Royal Challengers Bangalore               Mumbai Indians           bat
4        Kolkata Knight Riders              Deccan Chargers           bat

   result  dl_applied                       winner  win_by_runs  \
0  normal           0        Kolkata Knight Riders          140
1  normal           0          Chennai Super Kings           33
2  normal           0             Delhi Daredevils            0
3  normal           0  Royal Challengers Bangalore            0
4  normal           0        Kolkata Knight Riders            0

   win_by_wickets player_of_match                                       venue  \
0               0     BB McCullum                       M Chinnaswamy Stadium
1               0      MEK Hussey  Punjab Cricket Association Stadium, Mohali
2               9     MF Maharoof                            Feroz Shah Kotla
3               5      MV Boucher                            Wankhede Stadium
4               5       DJ Hussey                                Eden Gardens

     umpire1         umpire2  umpire3
0  Asad Rauf     RE Koertzen      NaN
1  MR Benson      SL Shastri      NaN
2  Aleem Dar  GA Pratapkumar      NaN
3   SJ Davis       DJ Harper      NaN
4  BF Bowden     K Hariharan      NaN
```

3.2.1 Add team score and team extra columns for each match, each inning.

[ ]:

```
team_score = deliveries.groupby(['match_id', 'inning'])['total_runs'].sum().
 ↪unstack().reset_index()
team_score.columns = ['match_id', 'Team1_score', 'Team2_score',␣
 ↪'Team1_superover_score', 'Team2_superover_score']
matches_agg = pd.merge(matches, team_score, left_on = 'id', right_on =␣
 ↪'match_id', how = 'outer')

team_extras = deliveries.groupby(['match_id', 'inning'])['extra_runs'].sum().
 ↪unstack().reset_index()
team_extras.columns = ['match_id', 'Team1_extras', 'Team2_extras',␣
 ↪'Team1_superover_extras', 'Team2_superover_extras']
matches_agg = pd.merge(matches_agg, team_extras, on = 'match_id', how = 'outer')

#Reorder the columns to make the data more readable
```

```
cols = ['match_id', 'season','city','date','team1','team2', 'toss_winner',␣
 ↪'toss_decision', 'result', 'dl_applied', 'winner',␣
 ↪'Team1_score','Team2_score', 'win_by_runs', 'win_by_wickets',␣
 ↪'Team1_extras', 'Team2_extras', 'Team1_superover_score',␣
 ↪'Team2_superover_score', 'Team1_superover_extras', 'Team2_superover_extras',␣
 ↪'player_of_match', 'type', 'venue', 'umpire1', 'umpire2', 'umpire3']
matches_agg = matches_agg[cols]
matches_agg.head(2)
```

[ ]:

```
   match_id  season        city        date                  team1  \
0         1    2008   Bangalore  2008-04-18  Kolkata Knight Riders
1         2    2008  Chandigarh  2008-04-19    Chennai Super Kings

                         team2                  toss_winner toss_decision  \
0  Royal Challengers Bangalore  Royal Challengers Bangalore         field
1              Kings XI Punjab          Chennai Super Kings           bat

   result  dl_applied  … Team1_superover_score  Team2_superover_score  \
0  normal           0  …                   NaN                    NaN
1  normal           0  …                   NaN                    NaN

   Team1_superover_extras  Team2_superover_extras  player_of_match  \
0                     NaN                     NaN      BB McCullum
1                     NaN                     NaN       MEK Hussey

            type                                       venue    umpire1  \
0  pre-qualifier                       M Chinnaswamy Stadium  Asad Rauf
1  pre-qualifier  Punjab Cricket Association Stadium, Mohali  MR Benson

       umpire2  umpire3
0  RE Koertzen      NaN
1   SL Shastri      NaN

[2 rows x 27 columns]
```

3.2.2 Batsmen aggregates (Runs, Balls, 4s, 6s, SR)

[ ]:

```
batsman_grp = deliveries.groupby(["match_id", "inning", "batting_team",␣
 ↪"batsman"])
batsmen = batsman_grp["batsman_runs"].sum().reset_index()

# Ignore the wide balls.
balls_faced = deliveries[deliveries["wide_runs"] == 0]
balls_faced = balls_faced.groupby(["match_id", "inning",␣
 ↪"batsman"])["batsman_runs"].count().reset_index()
balls_faced.columns = ["match_id", "inning", "batsman", "balls_faced"]
batsmen = batsmen.merge(balls_faced, left_on=["match_id", "inning", "batsman"],
                        right_on=["match_id", "inning", "batsman"], how="left")

fours = deliveries[ deliveries["batsman_runs"] == 4]
sixes = deliveries[ deliveries["batsman_runs"] == 6]

fours_per_batsman = fours.groupby(["match_id", "inning",␣
 ↪"batsman"])["batsman_runs"].count().reset_index()
sixes_per_batsman = sixes.groupby(["match_id", "inning",␣
 ↪"batsman"])["batsman_runs"].count().reset_index()

fours_per_batsman.columns = ["match_id", "inning", "batsman", "4s"]
sixes_per_batsman.columns = ["match_id", "inning", "batsman", "6s"]

batsmen = batsmen.merge(fours_per_batsman, left_on=["match_id", "inning",␣
 ↪"batsman"],
                        right_on=["match_id", "inning", "batsman"], how="left")
batsmen = batsmen.merge(sixes_per_batsman, left_on=["match_id", "inning",␣
 ↪"batsman"],
                        right_on=["match_id", "inning", "batsman"], how="left")
batsmen['SR'] = np.round(batsmen['batsman_runs'] / batsmen['balls_faced'] *␣
 ↪100, 2)

for col in ["batsman_runs", "4s", "6s", "balls_faced", "SR"]:
    batsmen[col] = batsmen[col].fillna(0)

dismissals = deliveries[ pd.notnull(deliveries["player_dismissed"])]
dismissals = dismissals[["match_id", "inning", "player_dismissed",␣
 ↪"dismissal_kind", "fielder"]]
dismissals.rename(columns={"player_dismissed": "batsman"}, inplace=True)
batsmen = batsmen.merge(dismissals, left_on=["match_id", "inning", "batsman"],
                        right_on=["match_id", "inning", "batsman"], how="left")

batsmen = matches[['id','season']].merge(batsmen, left_on = 'id', right_on =␣
 ↪'match_id', how = 'left').drop('id', axis = 1)
batsmen.head(2)
```

```
[ ]:    season  match_id  inning           batting_team      batsman  batsman_runs  \
     0    2008         1       1  Kolkata Knight Riders  BB McCullum           158
     1    2008         1       1  Kolkata Knight Riders    DJ Hussey            12

        balls_faced    4s    6s      SR dismissal_kind   fielder
     0         73.0  10.0  13.0  216.44            NaN       NaN
     1         12.0   1.0   0.0  100.00         caught  CL White
```

3.2.3 Bowler Aggregates

[ ]:

```
bowler_grp = deliveries.groupby(["match_id", "inning", "bowling_team",␣
 ↪"bowler", "over"])
bowlers = bowler_grp["total_runs", "wide_runs", "bye_runs", "legbye_runs",␣
 ↪"noball_runs"].sum().reset_index()

bowlers["runs"] = bowlers["total_runs"] - (bowlers["bye_runs"] +␣
 ↪bowlers["legbye_runs"])
bowlers["extras"] = bowlers["wide_runs"] + bowlers["noball_runs"]

del( bowlers["bye_runs"])
del( bowlers["legbye_runs"])
del( bowlers["total_runs"])

dismissal_kinds_for_bowler = ["bowled", "caught", "lbw", "stumped", "caught and␣
 ↪bowled", "hit wicket"]
dismissals = deliveries[deliveries["dismissal_kind"].
 ↪isin(dismissal_kinds_for_bowler)]
dismissals = dismissals.groupby(["match_id", "inning", "bowling_team",␣
 ↪"bowler", "over"])["dismissal_kind"].count().reset_index()
dismissals.rename(columns={"dismissal_kind": "wickets"}, inplace=True)

bowlers = bowlers.merge(dismissals, left_on=["match_id", "inning",␣
 ↪"bowling_team", "bowler", "over"],
                        right_on=["match_id", "inning", "bowling_team",␣
 ↪"bowler", "over"], how="left")
bowlers["wickets"] = bowlers["wickets"].fillna(0)

bowlers_over = bowlers.groupby(['match_id', 'inning', 'bowling_team',␣
 ↪'bowler'])['over'].count().reset_index()
bowlers = bowlers.groupby(['match_id', 'inning', 'bowling_team', 'bowler']).
 ↪sum().reset_index().drop('over', 1)
bowlers = bowlers_over.merge(bowlers, on=["match_id", "inning", "bowling_team",␣
 ↪"bowler"], how = 'left')
bowlers['Econ'] = np.round(bowlers['runs'] / bowlers['over'] , 2)
bowlers = matches[['id','season']].merge(bowlers, left_on = 'id', right_on =␣
 ↪'match_id', how = 'left').drop('id', axis = 1)

bowlers.head(2)
```

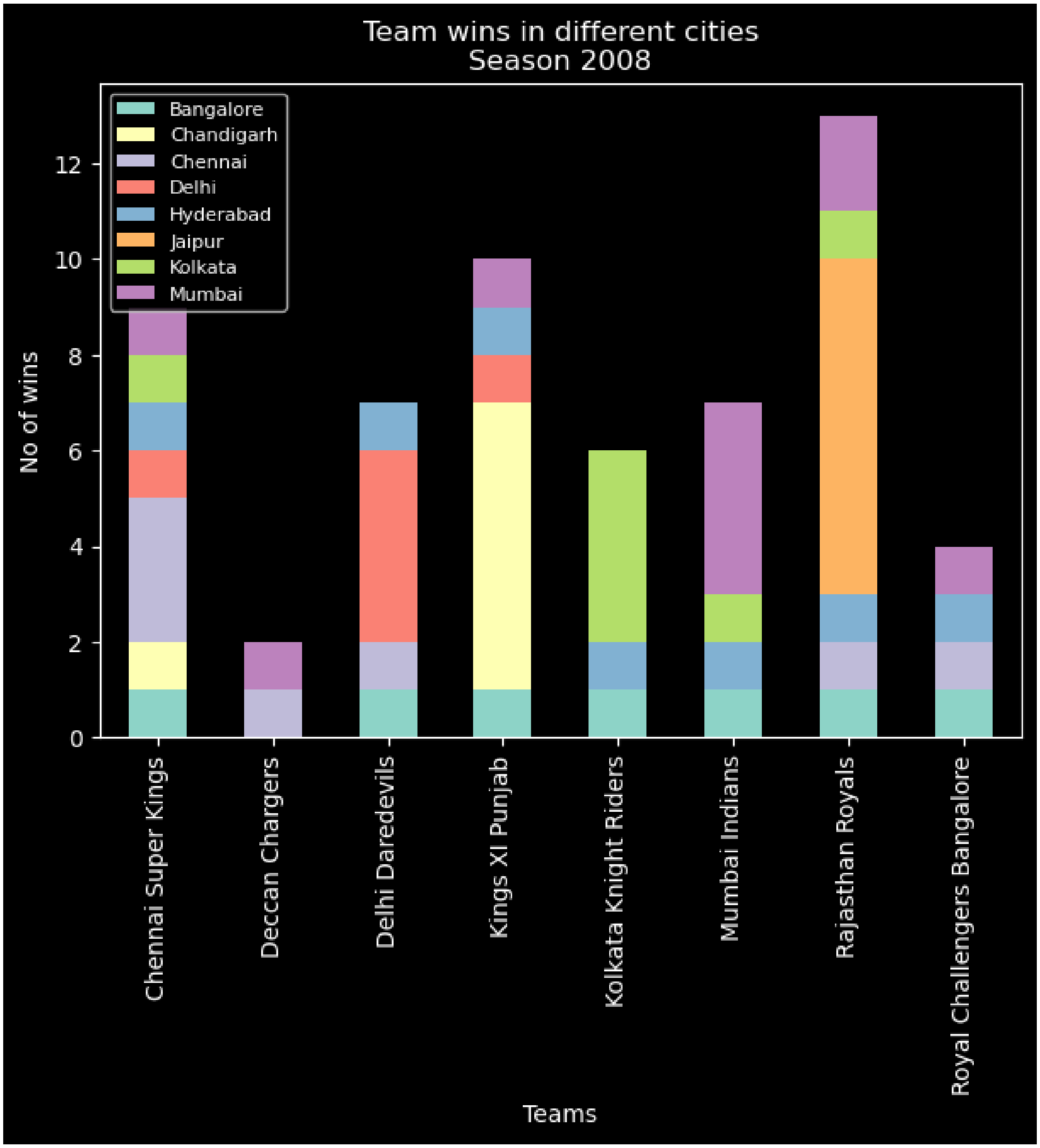
[ ]:

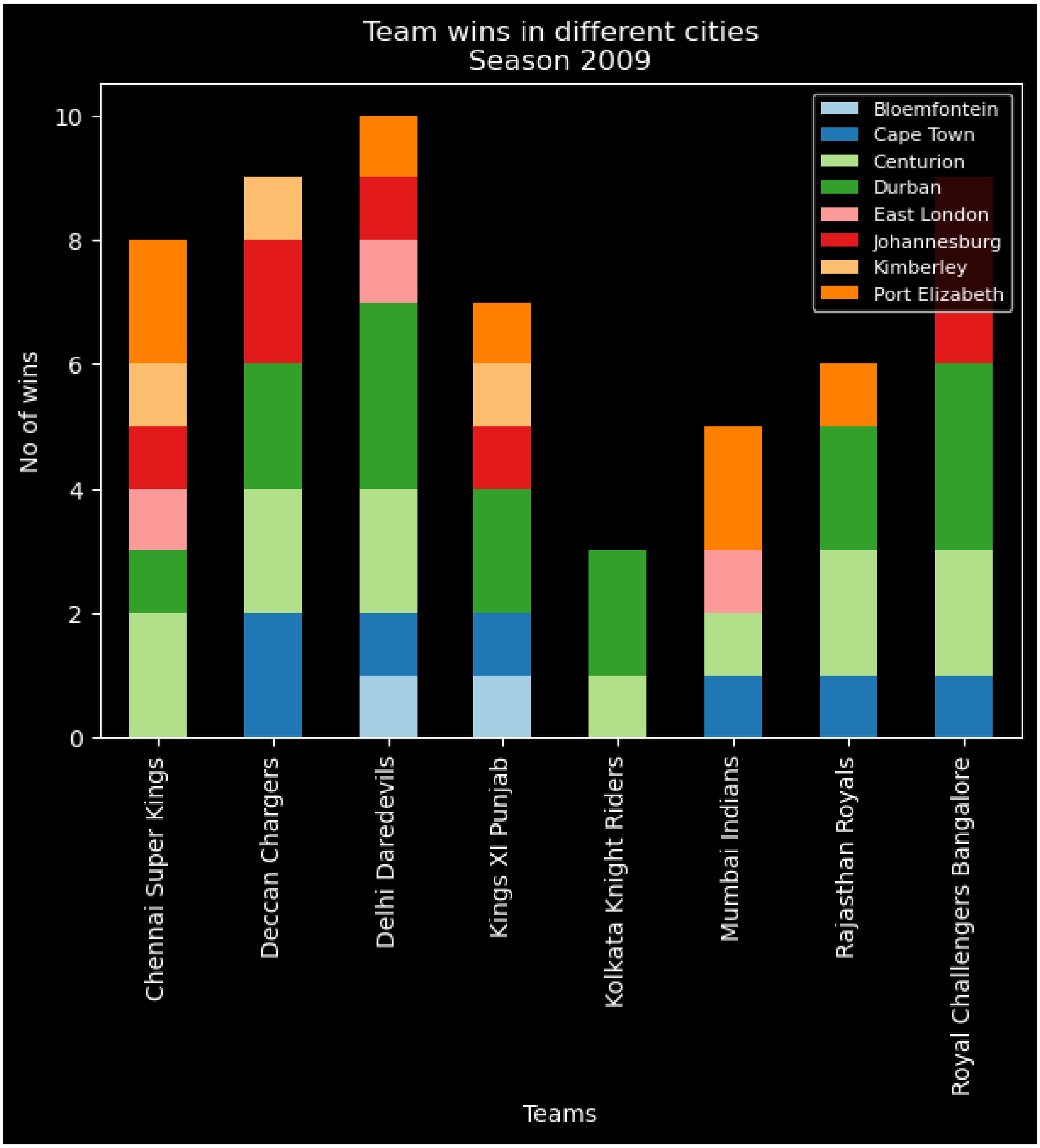
```
   season  match_id  inning                 bowling_team     bowler  over  \
0    2008         1       1  Royal Challengers Bangalore  AA Noffke     4
1    2008         1       1  Royal Challengers Bangalore   CL White     1

   wide_runs  noball_runs  runs  extras  wickets  Econ
0          5            0    40       5      1.0  10.0
```

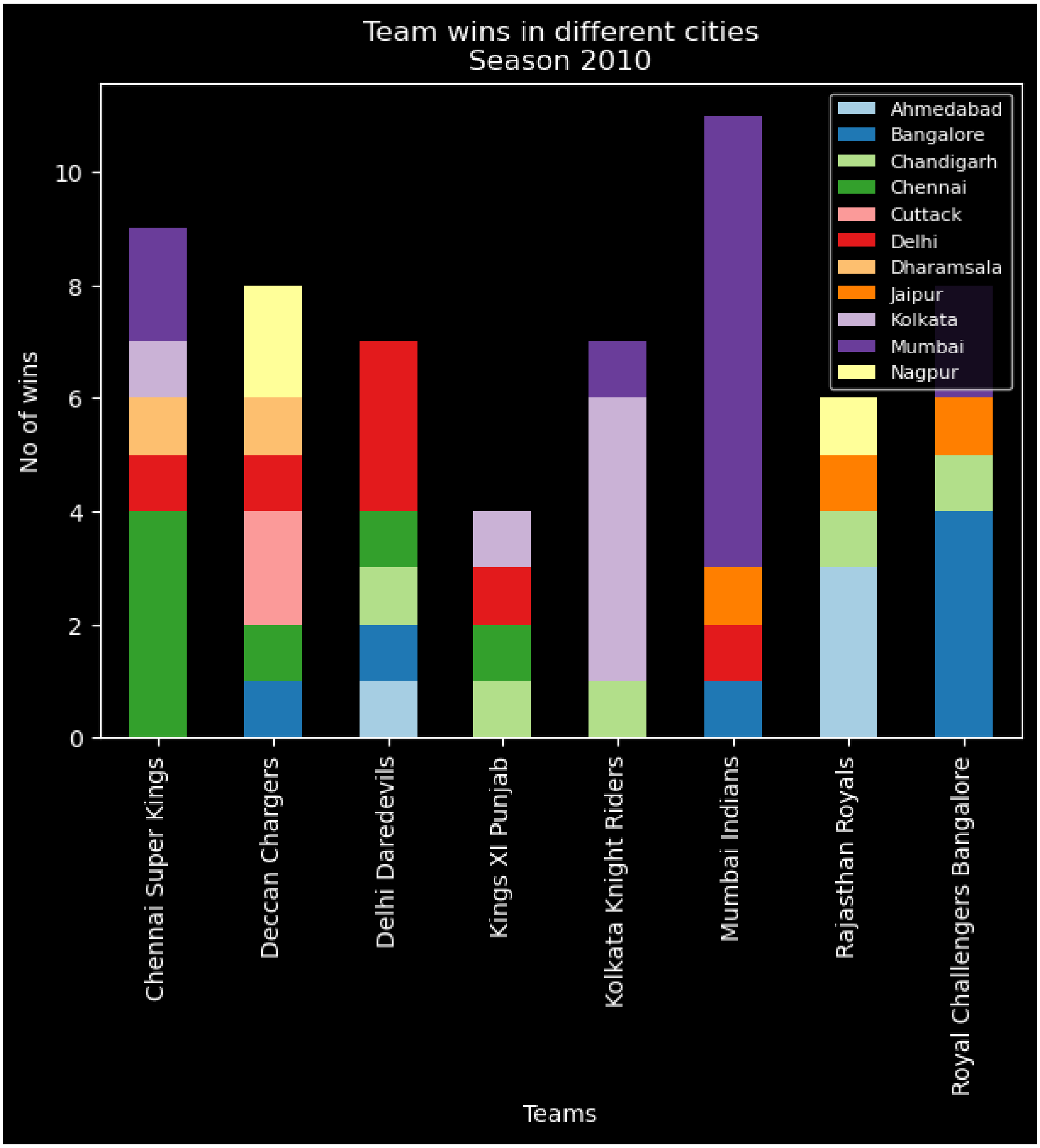
```
1          2            0    24       2      0.0  24.0
```

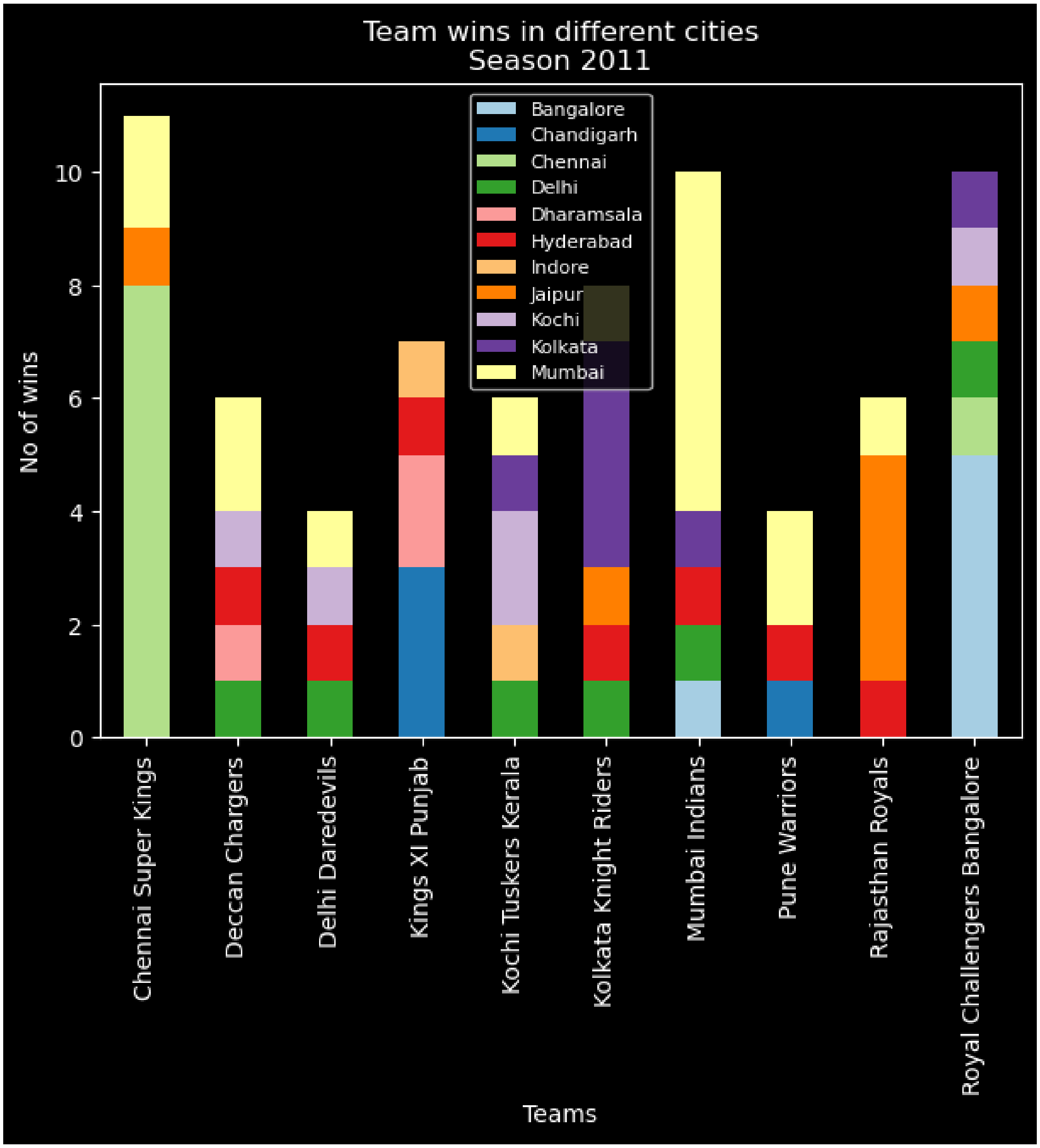
- **Team wins in home city vs other cities Each team plays two matches against the other teams, one in its home city and other in the home city of the opposite team. It would be interesting see if playing in home city increases a teams chances of a win.**

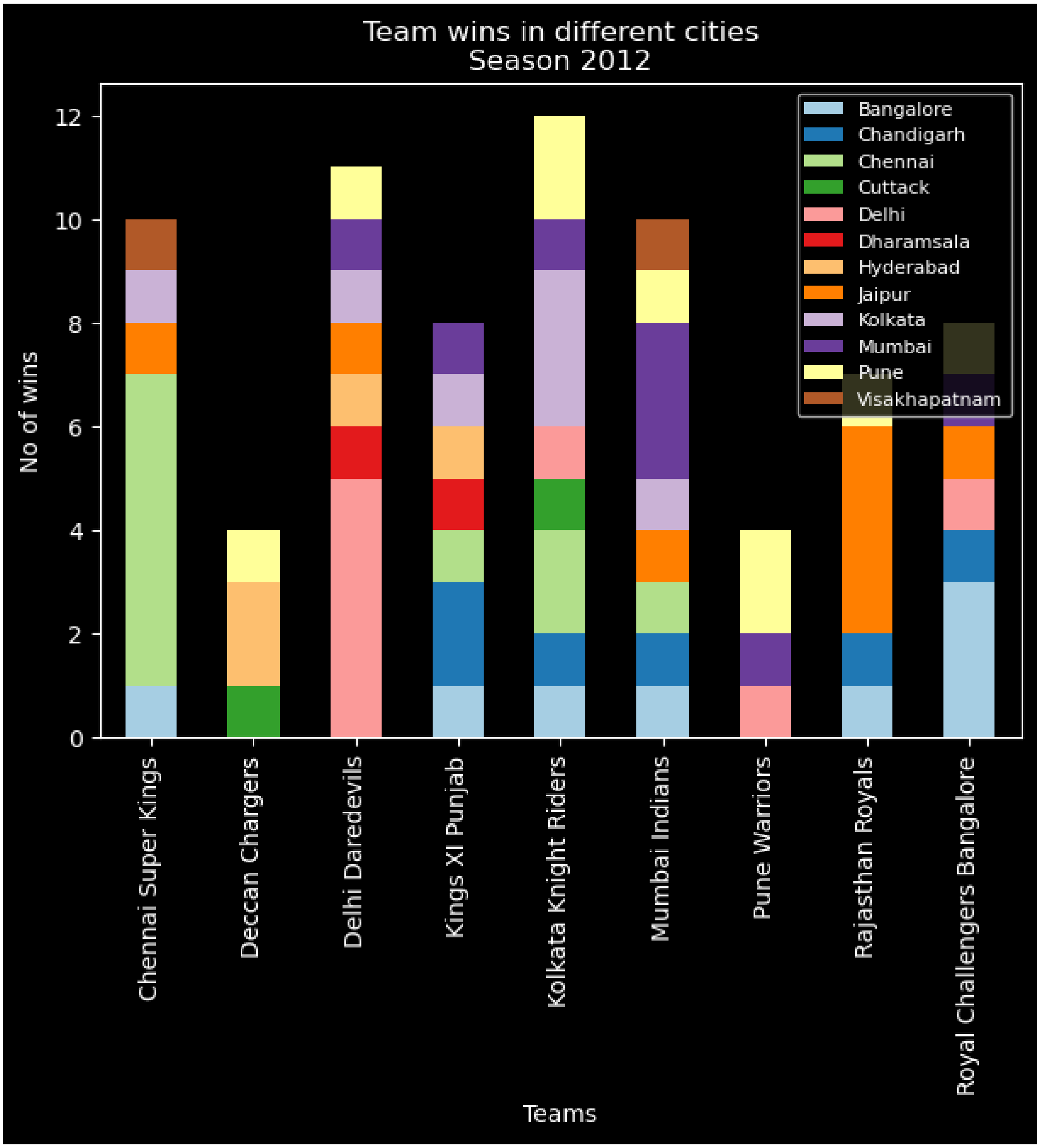
```
[ ]: #No of wins by team and season in each city
     x, y = 2008, 2017
     while x < y:
         wins_percity = matches_agg[matches_agg['season'] == x].groupby(['winner',␣
      ↪'city'])['match_id'].count().unstack()
         plot = wins_percity.plot(kind='bar', stacked=True, title="Team wins in␣
      ↪different cities\nSeason "+str(x), figsize=(7, 5))
         sns.set_palette("Paired", len(matches_agg['city'].unique()))
         plot.set_xlabel("Teams")
         plot.set_ylabel("No of wins")
         plot.legend(loc='best', prop={'size':8})
         x+=1
```

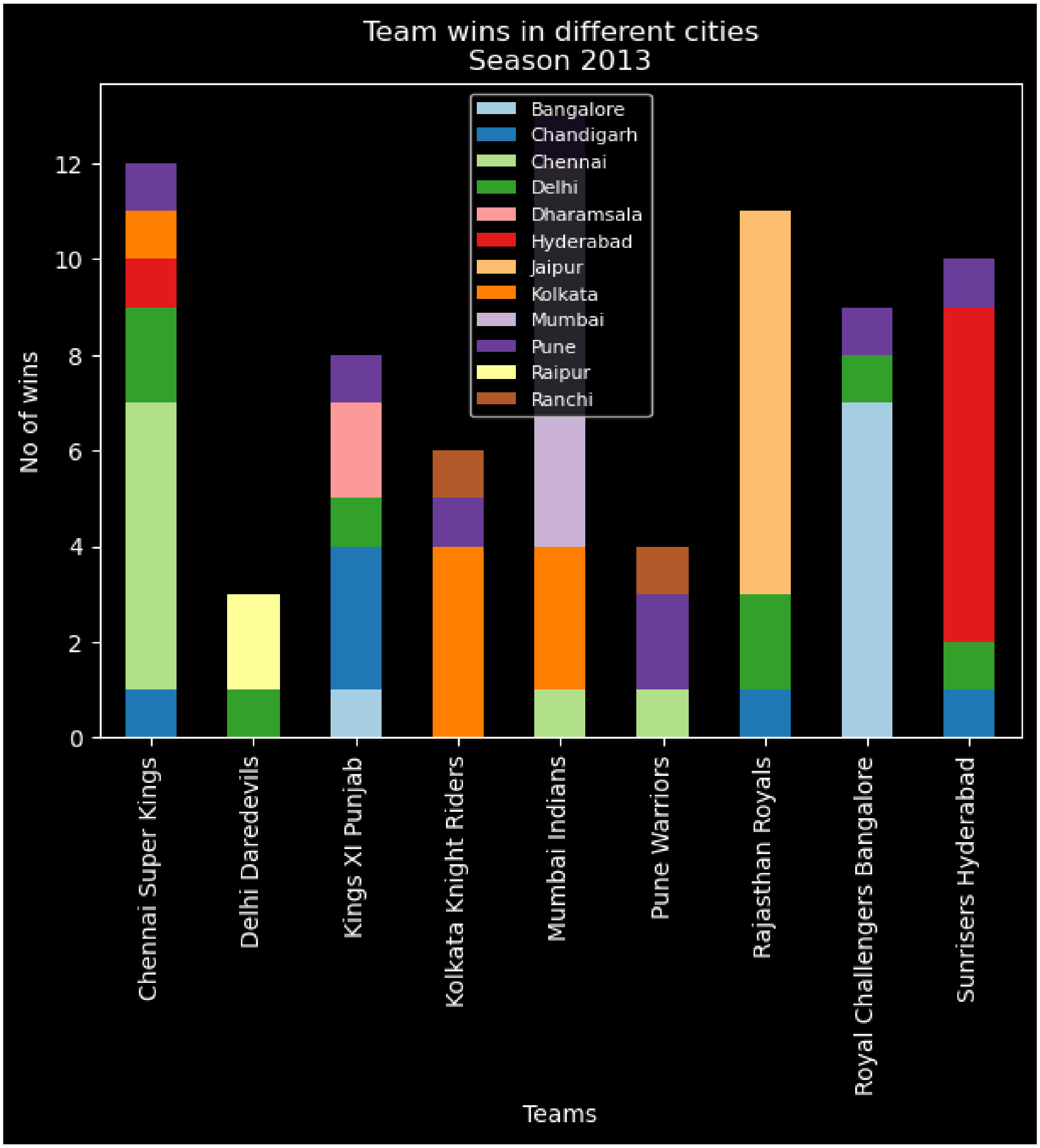

Team wins in different cities
Season 2008
Bangalore
Chandigarh
Chennai
Delhi
Hyderabad
Jaipur
Kolkata
Mumbai
No of wins
0
2
4
6
8
10
12
Chennai Super Kings
Deccan Chargers
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Rajasthan Royals
Royal Challengers Bangalore
Teams

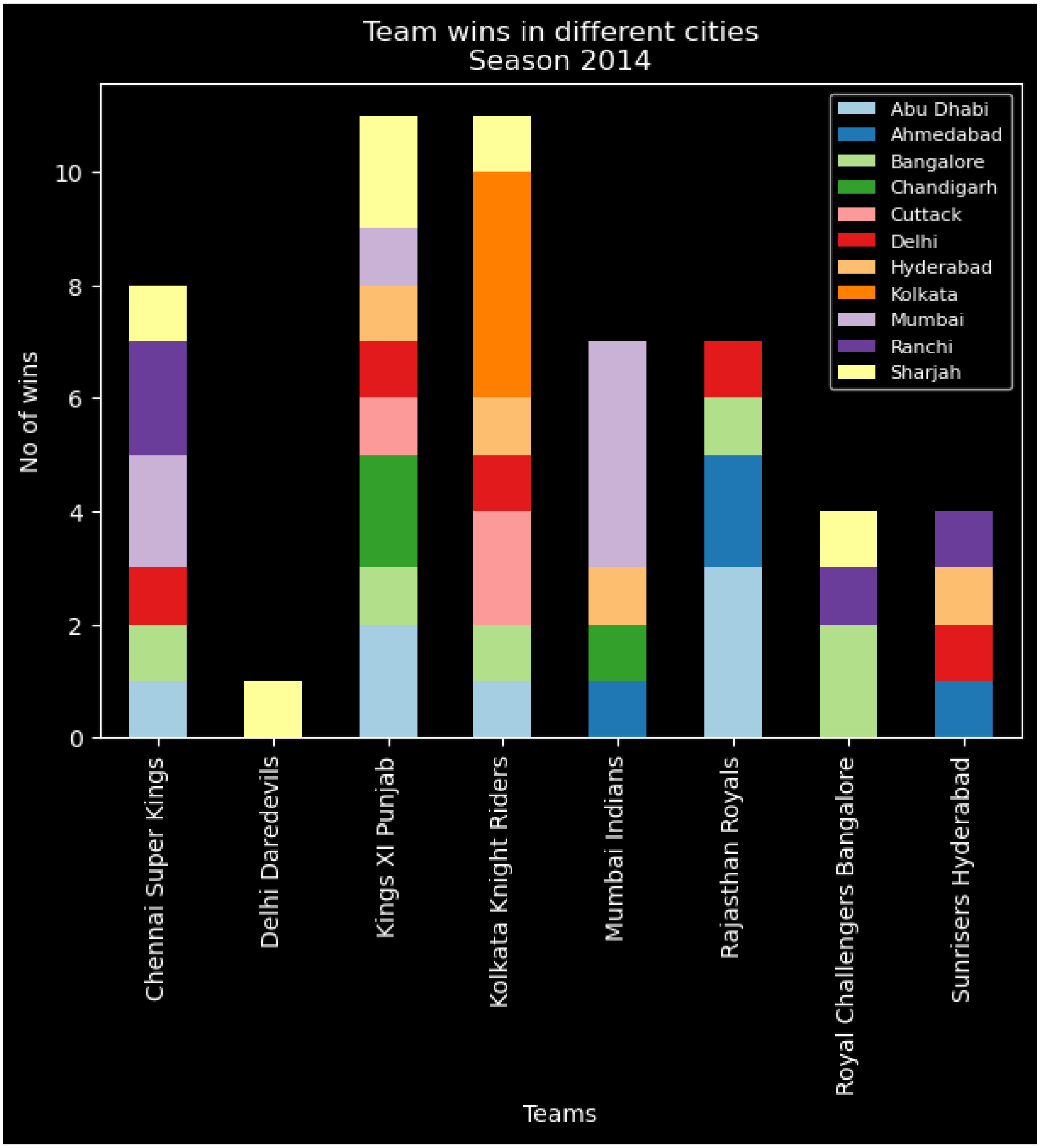

Team wins in different cities
Season 2009
No of wins
0
2
4
6
8
10
Bloemfontein
Cape Town
Centurion
Durban
East London
Johannesburg
Kimberley
Port Elizabeth
Chennai Super Kings
Deccan Chargers
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Rajasthan Royals
Royal Challengers Bangalore
Teams

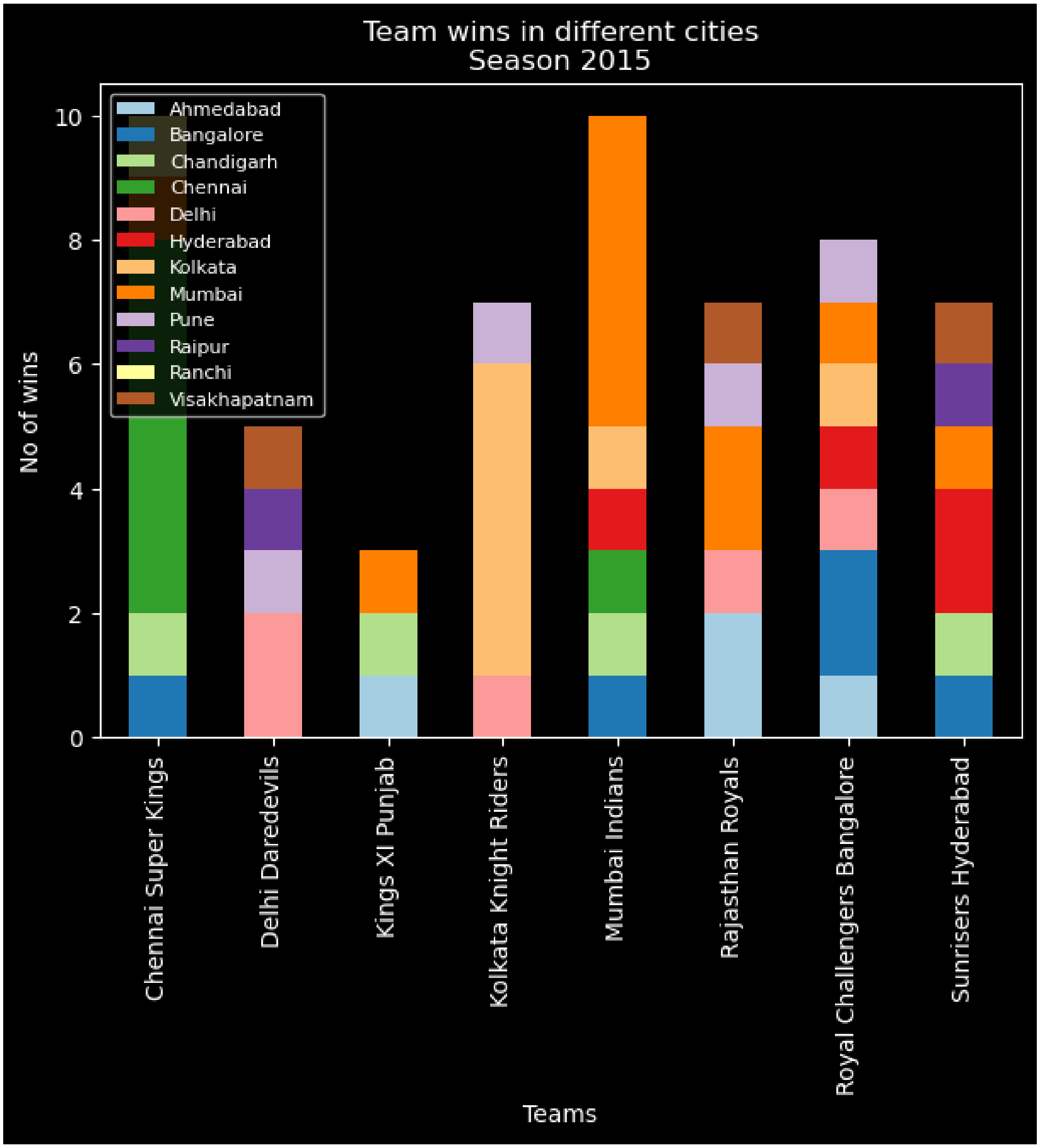

Team wins in different cities
Season 2010
No of wins
0
2
4
6
8
10
Teams
Chennai Super Kings
Deccan Chargers
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Rajasthan Royals
Royal Challengers Bangalore
Ahmedabad
Bangalore
Chandigarh
Chennai
Cuttack
Delhi
Dharamsala
Jaipur
Kolkata
Mumbai
Nagpur

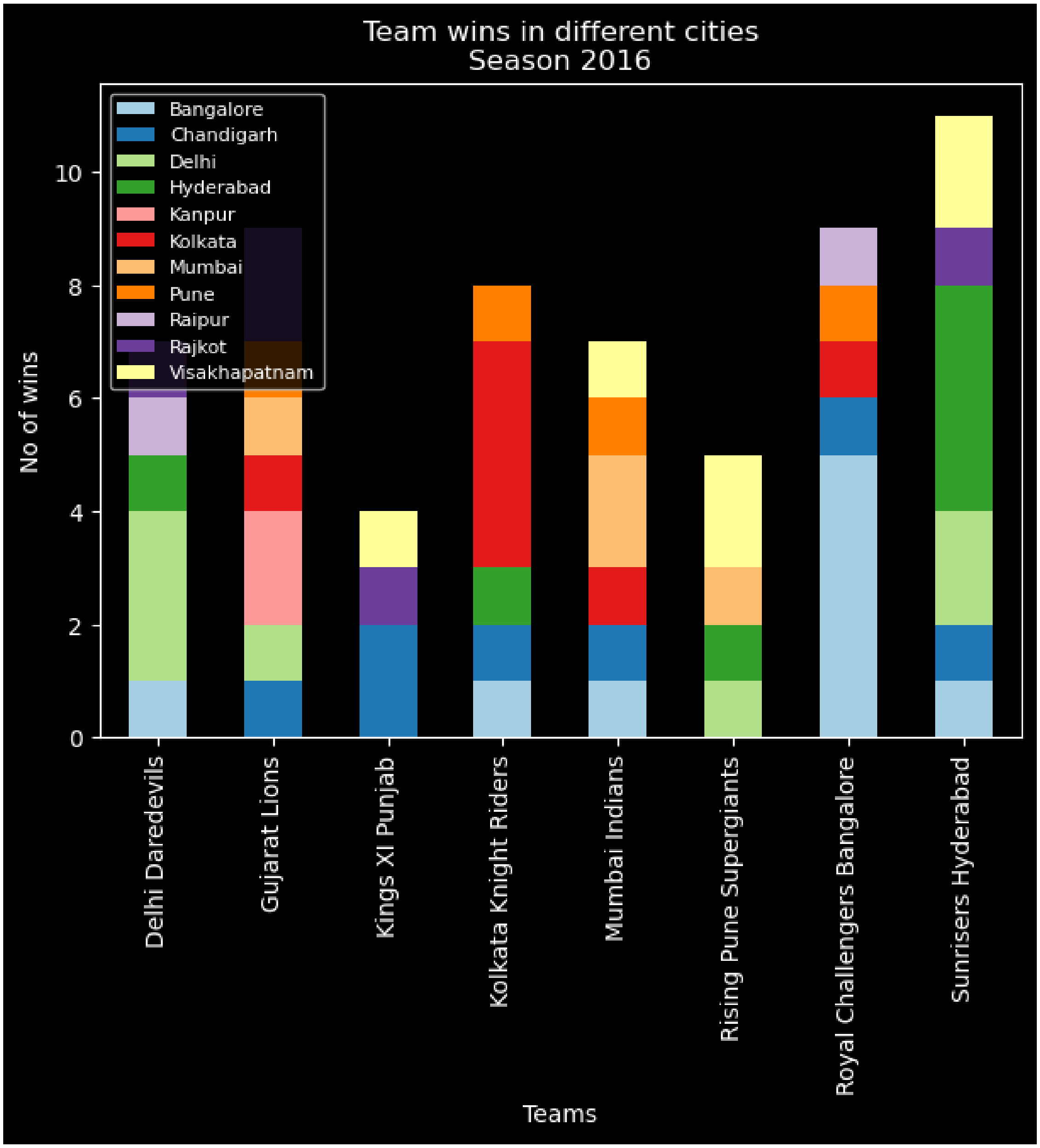

Team wins in different cities
Season 2011
No of wins
0
2
4
6
8
10
Bangalore
Chandigarh
Chennai
Delhi
Dharamsala
Hyderabad
Indore
Jaipur
Kochi
Kolkata
Mumbai
Chennai Super Kings
Deccan Chargers
Delhi Daredevils
Kings XI Punjab
Kochi Tuskers Kerala
Kolkata Knight Riders
Mumbai Indians
Pune Warriors
Rajasthan Royals
Royal Challengers Bangalore
Teams


Team wins in different cities
Season 2012
No of wins
0
2
4
6
8
10
12
Bangalore
Chandigarh
Chennai
Cuttack
Delhi
Dharamsala
Hyderabad
Jaipur
Kolkata
Mumbai
Pune
Visakhapatnam
Chennai Super Kings
Deccan Chargers
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Pune Warriors
Rajasthan Royals
Royal Challengers Bangalore
Teams


Team wins in different cities
Season 2013
Bangalore
Chandigarh
Chennai
Delhi
Dharamsala
Hyderabad
Jaipur
Kolkata
Mumbai
Pune
Raipur
Ranchi
No of wins
0
2
4
6
8
10
12
Chennai Super Kings
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Pune Warriors
Rajasthan Royals
Royal Challengers Bangalore
Sunrisers Hyderabad
Teams


Team wins in different cities
Season 2014
No of wins
0
2
4
6
8
10
Abu Dhabi
Ahmedabad
Bangalore
Chandigarh
Cuttack
Delhi
Hyderabad
Kolkata
Mumbai
Ranchi
Sharjah
Chennai Super Kings
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Rajasthan Royals
Royal Challengers Bangalore
Sunrisers Hyderabad
Teams


Team wins in different cities
Season 2015
No of wins
0
2
4
6
8
10
Ahmedabad
Bangalore
Chandigarh
Chennai
Delhi
Hyderabad
Kolkata
Mumbai
Pune
Raipur
Ranchi
Visakhapatnam
Chennai Super Kings
Delhi Daredevils
Kings XI Punjab
Kolkata Knight Riders
Mumbai Indians
Rajasthan Royals
Royal Challengers Bangalore
Sunrisers Hyderabad
Teams

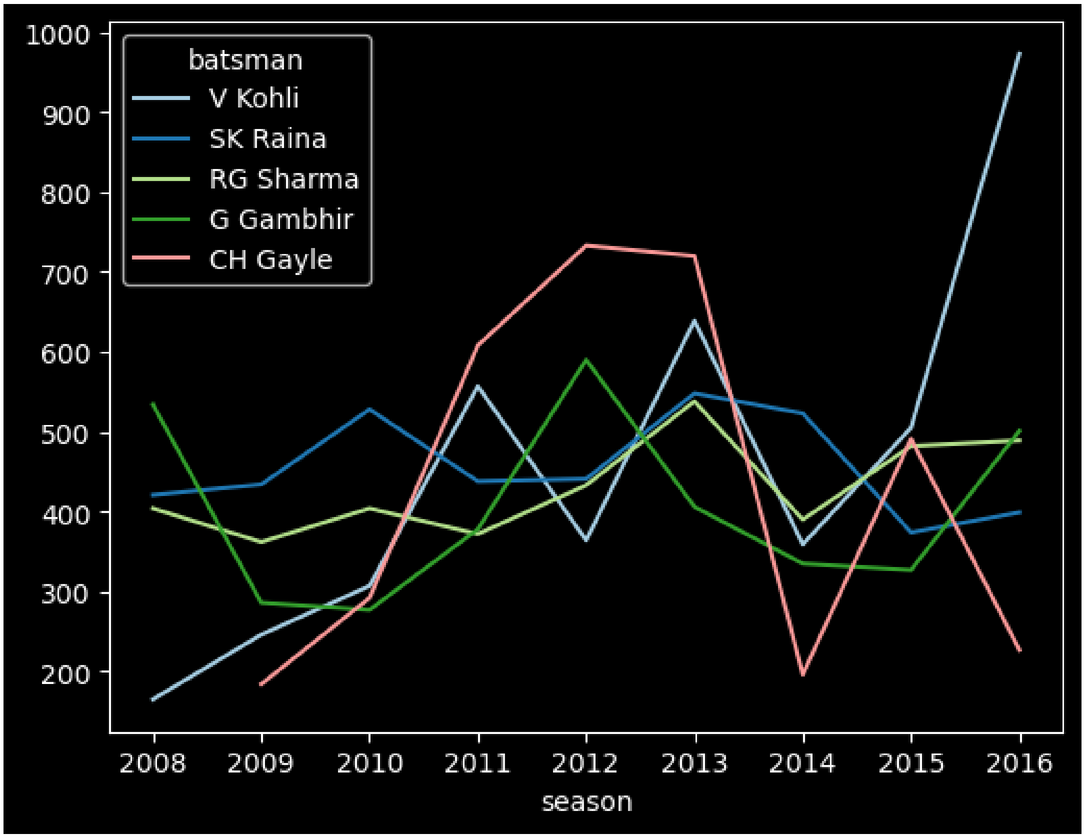


3.2.4 Plot the performance of top 5 batsmen over seasons

Virat Kohli show a steady improvement over season and C. Gayle and SK Raina show a slump

```
[ ]: batsman_runsperseason = batsmen.groupby(['season', 'batting_team',␣
       ↪'batsman'])['batsman_runs'].sum().reset_index()
     batsman_runsperseason = batsman_runsperseason.groupby(['season',␣
       ↪'batsman'])['batsman_runs'].sum().unstack().T
     batsman_runsperseason['Total'] = batsman_runsperseason.sum(axis=1) #add total␣
       ↪column to find batsman with the highest runs
```

```
batsman_runsperseason = batsman_runsperseason.sort_values(by = 'Total',␣
↪ascending = False).drop('Total', 1)
ax = batsman_runsperseason[:5].T.plot()
```

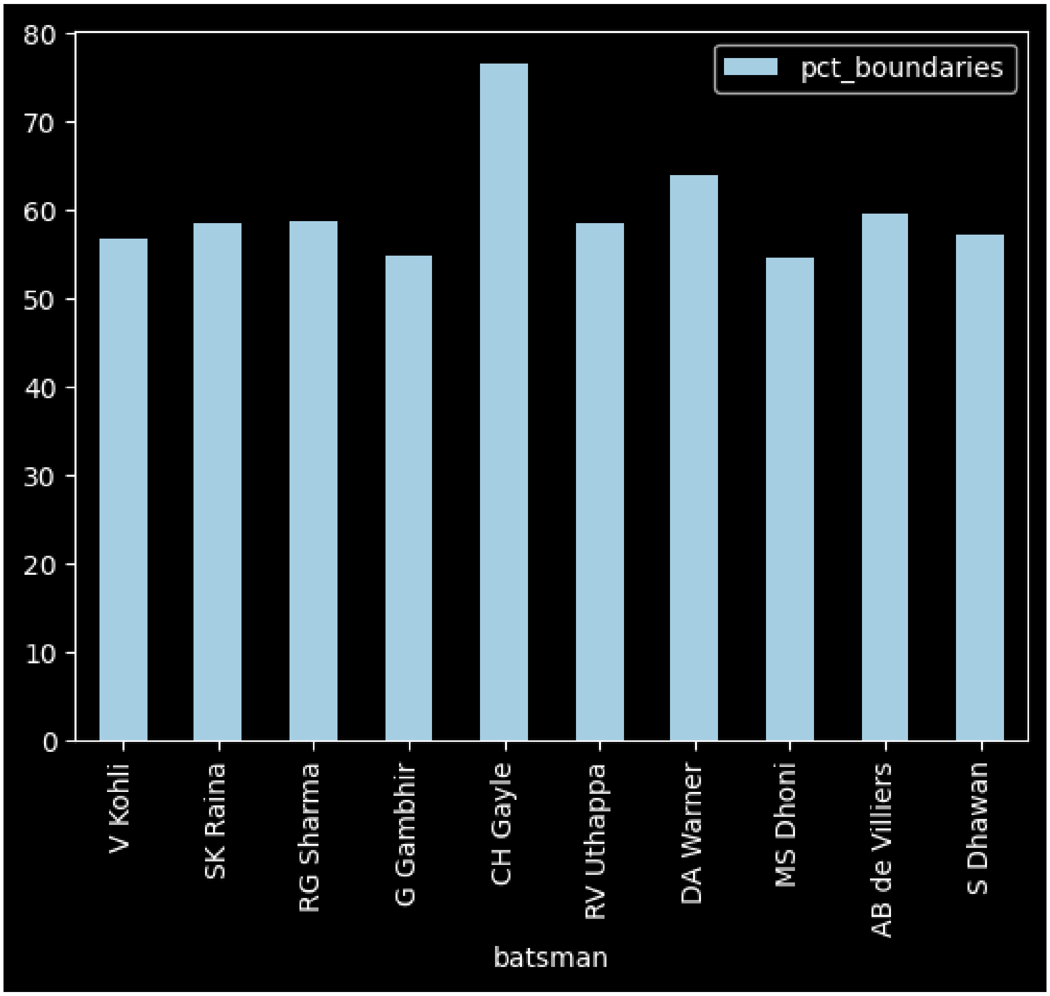


3.2.5 Percentage of total runs scored through boundaries for each batsman

The average for top batsmen is around 58-60% with exception of CH Gayle at 76%. Interestingly, **MS Dhoni** who is known for helicopter shots(6s) gets close to 45% of his runs through singles

```
[ ]: batsman_runs = batsmen.groupby(['batsman'])['batsman_runs', '4s', '6s'].sum().
     ↪reset_index()
     batsman_runs['4s_6s'] = batsman_runs['4s'] * 4 + batsman_runs['6s'] * 6
     batsman_runs['pct_boundaries'] = np.round(batsman_runs['4s_6s'] /␣
     ↪batsman_runs['batsman_runs'] * 100, 2)
     batsman_runs = batsman_runs.sort_values(by = 'batsman_runs', ascending = False)
     batsman_runs[:10].plot(x= 'batsman', y = 'pct_boundaries', kind = 'bar')
```

```
[ ]: <AxesSubplot: xlabel='batsman'>
```

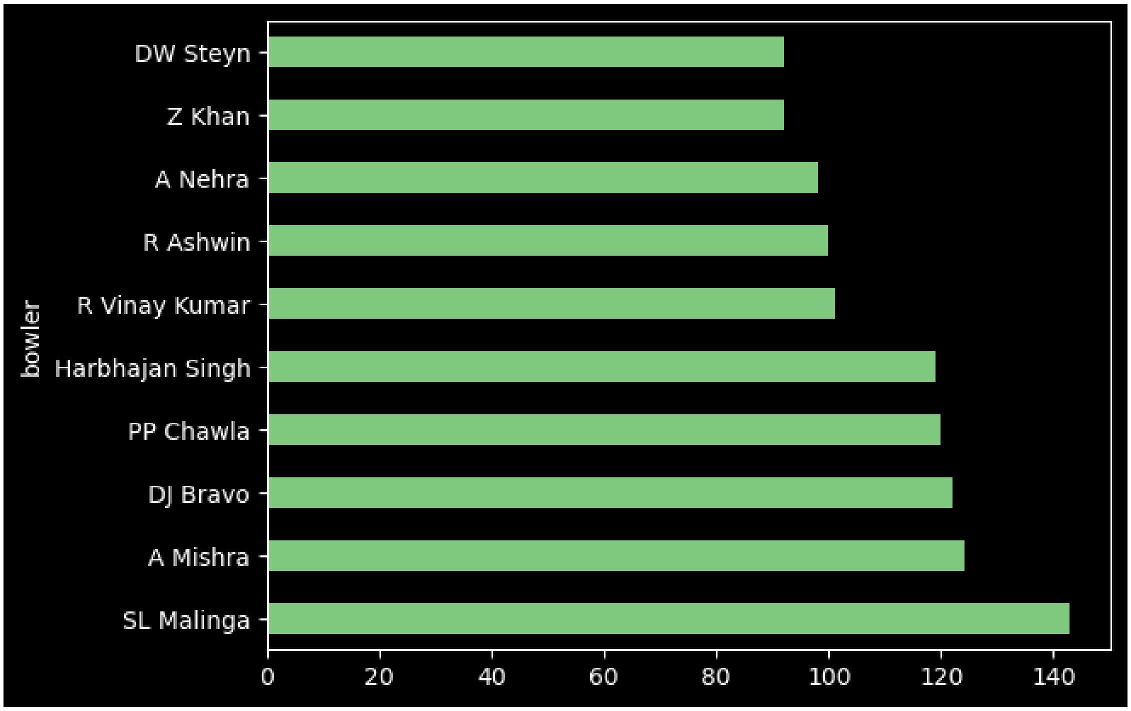


3.2.6 Performance of top bowlers over seasons

Malinga is the highest wicket taken in IPL so far

```
[ ]: bowlers_wickets = bowlers.groupby(['bowler'])['wickets'].sum()
     bowlers_wickets.sort_values(ascending = False, inplace = True)
     bowlers_wickets[:10].plot(x= 'bowler', y = 'runs', kind = 'barh', colormap =␣
      ↪'Accent')
```

```
[ ]: <AxesSubplot: ylabel='bowler'>
```



3.2.7 Extra runs conceded by bowlers

```
[ ]: bowlers_extras = bowlers.groupby(['season', 'bowler'])['extras'].sum().
       ↪unstack().T
     bowlers_extras['Total'] = bowlers_extras.sum(axis=1)
     #bowlers_extras('Total', ascending = False, inplace = True)
     bowlers_extras.head()
```

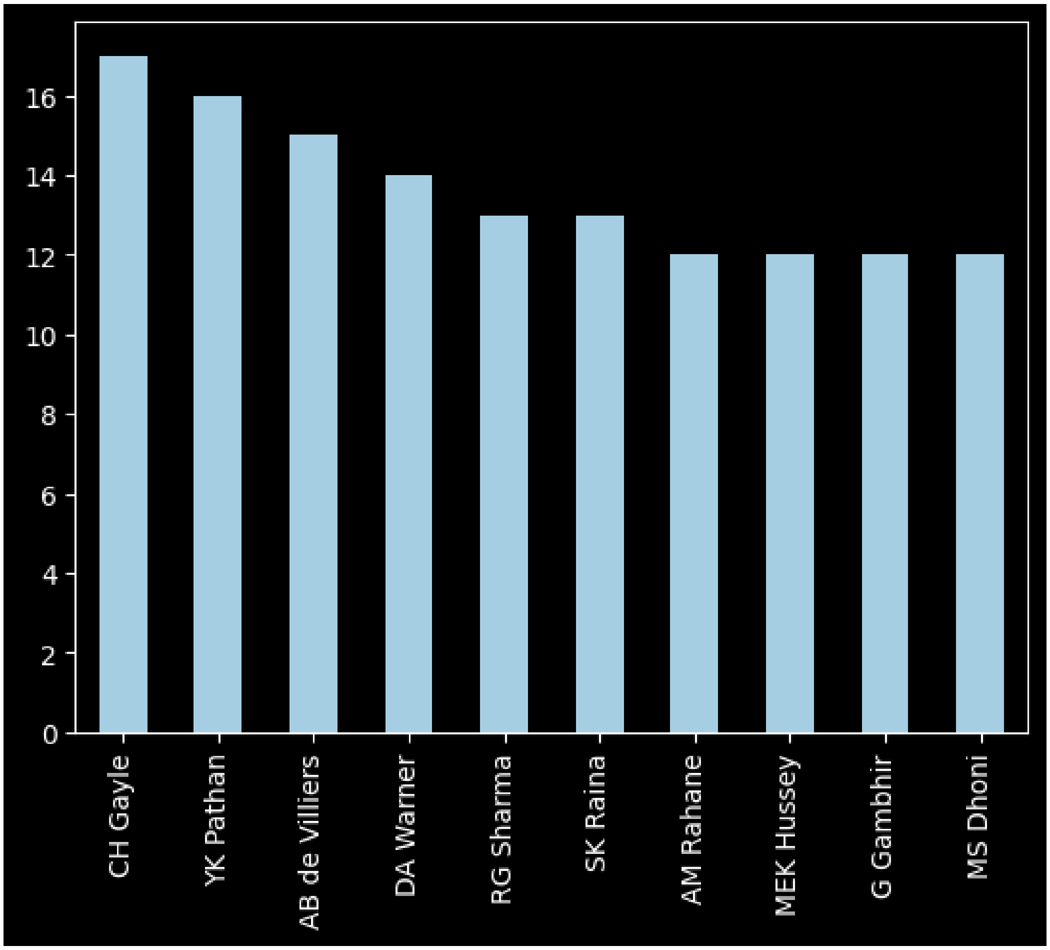
```
[ ]: season             2008  2009  2010  2011  2012  2013  2014  2015  2016  Total
     bowler
     A Ashish Reddy      NaN   NaN   NaN   NaN   8.0   1.0   NaN   1.0   0.0   10.0
     A Chandila          NaN   NaN   NaN   NaN   0.0   0.0   NaN   NaN   NaN    0.0
     A Flintoff          NaN   0.0   NaN   NaN   NaN   NaN   NaN   NaN   NaN    0.0
     A Kumble           10.0   7.0  14.0   NaN   NaN   NaN   NaN   NaN   NaN   31.0
     A Mishra            4.0   9.0  11.0   4.0   8.0   4.0   5.0   9.0  12.0   66.0
```

3.2.8 Player of the match

See who has won the Player of the match award maximum number of times

```
[ ]: matches['player_of_match'].value_counts()[:10].plot(kind = 'bar')
```

```
[ ]: <AxesSubplot: >
```



4 Analysis On IPL Data 2008-2020

[ ]:

```
deliveries2=pd.read_csv("/home/blackheart/Documents/DATA SCIENCE/PROJECT/IPL␣
 ↪Analysis/IPL Ball-by-Ball 2008-2020.csv")
deliveries2.head()
```

[ ]:

```
       id  inning  over  ball      batsman  non_striker     bowler  \
0  335982       1     6     5   RT Ponting  BB McCullum  AA Noffke
1  335982       1     6     6  BB McCullum   RT Ponting  AA Noffke
2  335982       1     7     1  BB McCullum   RT Ponting     Z Khan
3  335982       1     7     2  BB McCullum   RT Ponting     Z Khan
4  335982       1     7     3   RT Ponting  BB McCullum     Z Khan

   batsman_runs  extra_runs  total_runs  non_boundary  is_wicket  \
0             1           0           1             0          0
1             1           0           1             0          0
2             0           0           0             0          0
```

```
3             1           0           1             0          0
4             1           0           1             0          0

  dismissal_kind player_dismissed fielder extras_type           batting_team  \
0            NaN              NaN     NaN         NaN  Kolkata Knight Riders
1            NaN              NaN     NaN         NaN  Kolkata Knight Riders
2            NaN              NaN     NaN         NaN  Kolkata Knight Riders
3            NaN              NaN     NaN         NaN  Kolkata Knight Riders
4            NaN              NaN     NaN         NaN  Kolkata Knight Riders

                  bowling_team
0  Royal Challengers Bangalore
1  Royal Challengers Bangalore
2  Royal Challengers Bangalore
3  Royal Challengers Bangalore
4  Royal Challengers Bangalore
```

```
[ ]: matches2=pd.read_csv("/home/blackheart/Documents/DATA SCIENCE/PROJECT/IPL␣
      ↪Analysis/IPL Matches 2008-2020.csv")
     matches2.head()
```

```
[ ]:        id        city        date player_of_match  \
     0  335982   Bangalore  2008-04-18     BB McCullum
     1  335983  Chandigarh  2008-04-19      MEK Hussey
     2  335984       Delhi  2008-04-19     MF Maharoof
     3  335985      Mumbai  2008-04-20      MV Boucher
     4  335986     Kolkata  2008-04-20       DJ Hussey

                                             venue  neutral_venue  \
     0                       M Chinnaswamy Stadium              0
     1  Punjab Cricket Association Stadium, Mohali              0
     2                            Feroz Shah Kotla              0
     3                            Wankhede Stadium              0
     4                                Eden Gardens              0

                              team1                        team2  \
     0  Royal Challengers Bangalore        Kolkata Knight Riders
     1              Kings XI Punjab          Chennai Super Kings
     2             Delhi Daredevils             Rajasthan Royals
     3               Mumbai Indians  Royal Challengers Bangalore
     4        Kolkata Knight Riders              Deccan Chargers

                        toss_winner toss_decision                       winner  \
     0  Royal Challengers Bangalore         field        Kolkata Knight Riders
     1          Chennai Super Kings           bat          Chennai Super Kings
     2             Rajasthan Royals           bat             Delhi Daredevils
     3               Mumbai Indians           bat  Royal Challengers Bangalore
```

```
4               Deccan Chargers           bat        Kolkata Knight Riders

    result  result_margin eliminator method    umpire1         umpire2
0     runs          140.0          N    NaN  Asad Rauf     RE Koertzen
1     runs           33.0          N    NaN  MR Benson      SL Shastri
2  wickets            9.0          N    NaN  Aleem Dar  GA Pratapkumar
3  wickets            5.0          N    NaN   SJ Davis       DJ Harper
4  wickets            5.0          N    NaN  BF Bowden     K Hariharan
```

- **In December 2018 the team changed their name from Delhi Daredevils to Delhi Capitals and Sunrisers Hyderabad replaced Deccan Chargers in 2012 and debuted in 2013. But I consider them to be the same in this IPL analysis task. Now let's start with some data preparation:**

[ ]:

```
x=['Sunrisers Hyderabad', 'Mumbai Indians', 'Gujarat Lions',
    'Rising Pune Supergiant', 'Royal Challengers Bangalore',
    'Kolkata Knight Riders', 'Delhi Daredevils', 'Kings XI Punjab',
    'Chennai Super Kings', 'Rajasthan Royals', 'Deccan Chargers',
    'Kochi Tuskers Kerala', 'Pune Warriors', 'Rising Pune Supergiants', 'Delhi␣
 ↪Capitals']

y =␣
 ↪['SRH','MI','GL','RPS','RCB','KKR','DC','KXIP','CSK','RR','SRH','KTK','PW','RPS','DC']

matches2.replace(x,y,inplace = True)
deliveries2.replace(x,y,inplace = True)
```

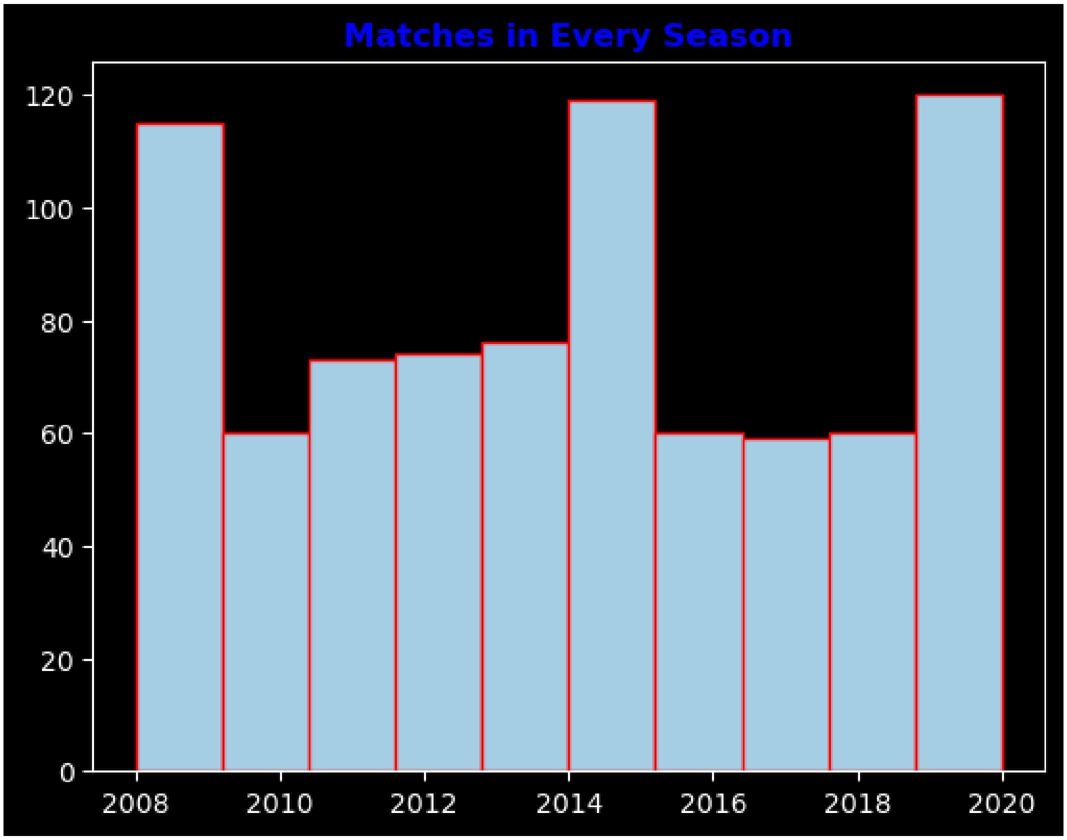
- **Let's start with looking at the number of matches played in every season of the IPL**

[ ]:

```
# Plotly to create interactive graph
import chart_studio.plotly as py
from plotly import tools
from plotly.offline import init_notebook_mode,iplot
init_notebook_mode(connected=False)
import plotly.figure_factory as ff
import plotly.graph_objs as go
```

[ ]:

```
d=matches2['date'].str[:4].astype(int)
plt.hist(d,edgecolor='red')
plt.title("Matches in Every Season",color='blue',weight='bold')
plt.show()
```



- **The year 2013 has the most matches, possibly due to super overs. Also, there are 10 teams in 2011, 9 in 2012 and 2013, this is another reason for the increase in the number of matches.**

4.0.1 Matches Played V/s Win

```
[ ]: matches_played=pd.concat([matches2['team1'],matches2['team2']])
     matches_played=matches_played.value_counts().reset_index()
     matches_played.columns=['Team','Total Matches']
     matches_played['wins']=matches2['winner'].value_counts().reset_index()['winner']

     matches_played.set_index('Team',inplace=True)
     totm = matches_played.reset_index().head(8)
     totm
```

```
[ ]:    Team  Total Matches  wins
     0    MI            203   120
     1   SRH            199   106
     2   RCB            195    99
     3    DC            194    95
     4   KKR            192    91
```

```
5  KXIP              190     88
6   CSK              178     86
7    RR              161     81
```

- **Now let's analyze the winning percentage of all IPL teams:**

[ ]:
```
trace1 = go.Bar(x=matches_played.index,y=matches_played['Total Matches'],
                name='Total Matches',opacity=0.4)

trace2 = go.Bar(x=matches_played.index,y=matches_played['wins'],
                name='Matches Won',marker=dict(color='red'),opacity=0.4)

trace3 = go.Bar(x=matches_played.index,
               y=(round(matches_played['wins']/matches_played['Total␣
 ↪Matches'],3)*100),
               name='Win Percentage',opacity=0.6,marker=dict(color='gold'))

data = [trace1, trace2, trace3]

layout = go.Layout(title='Match Played, Wins And Win␣
 ↪Percentage',xaxis=dict(title='Team'),
                   yaxis=dict(title='Count'),bargap=0.2,bargroupgap=0.1,␣
 ↪plot_bgcolor='rgb(245,245,245)')

fig = go.Figure(data=data, layout=layout)
iplot(fig)
```

- **So MI, SRH and RCB are the top three teams with the highest winning percentage. Let's look at the winning percentage of these three teams:**

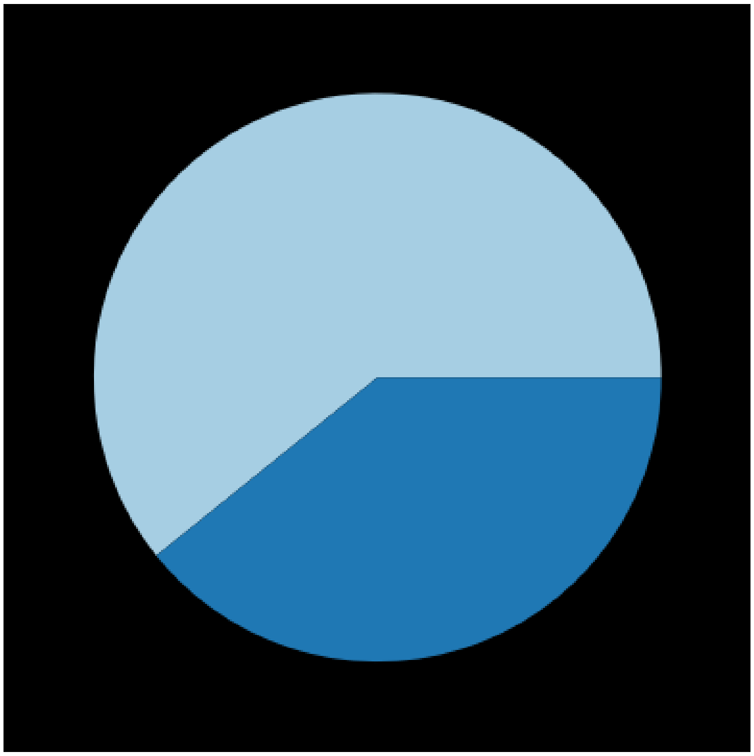
[ ]:
```
win_percentage = round(matches_played['wins']/matches_played['Total␣
 ↪Matches'],3)*100
win_percentage.head(3)
```

[ ]:
```
Team
MI     59.1
SRH    53.3
RCB    50.8
dtype: float64
```

- Now let's have a look the most prefered decision taken by teams after winning the toss:

[ ]:
```
x = matches2["toss_decision"].value_counts()
#y = matches2["toss_decision"].value_counts().values
plt.pie(x)
```

[ ]:
```
([<matplotlib.patches.Wedge at 0x7f6cd4ba1610>,
  <matplotlib.patches.Wedge at 0x7f6cd40c7a10>],
 [Text(-0.3655903556118915, 1.0374698510721025, ''),
Text(0.3655904527468272, -1.037469816843059, '')])
```



- More likely win after win toss
- **Higest Score runs by team**

```
[ ]: high_scores=deliveries2.groupby(['id',␣
      ↪'inning','batting_team','bowling_team'])['total_runs'].sum().reset_index()
     high_scores=high_scores[high_scores['total_runs']>=200]
     hss = high_scores.nlargest(10,'total_runs')

     trace = go.Table(
         header=dict(values=["Inning","Batting Team","Bowling Team", "Total Runs"],
                     fill = dict(color = 'red'),
                     font = dict(color = 'white', size = 14),
                     align = ['center'],
                    height = 30),
         cells=dict(values=[hss['inning'], hss['batting_team'], hss['bowling_team'],␣
      ↪hss['total_runs']],
                    fill = dict(color = ['lightsalmon', 'rgb(245, 245, 249)']),
                    align = ['center'], font_size=13))

     layout = dict(
         width=830,
         height=410,
         autosize=False,
         title='Highest scores of IPL',
         showlegend=False,
     )

     fig1 = dict(data=[trace], layout=layout)
     iplot(fig1)
```

```
[ ]: hss
```

```
[ ]:           id  inning batting_team bowling_team  total_runs
     700   598027       1          RCB           PW         263
     1116  980987       1          RCB           GL         248
     292   419137       1          CSK           RR         246
     1354 1136604       1          KKR         KXIP         245
     2     335983       1          CSK         KXIP         240
     1000  829795       1          RCB           MI         235
     472   501260       1         KXIP          RCB         232
     1480 1178422       1          KKR           MI         232
     398   501223       1           DC         KXIP         231
     850   733987       1         KXIP          CSK         231
```

5 Reference

- Aman Khawal
- Kaggle

6 Thank You