Machine Learning Laboratory

(410302)

BE Sem I Honors in AI/ML

Academic Year: 2021-22

Lab Assignment No. 5

Name: Aboli Marathe

Roll Number: 41301

Branch: Department of Computer Engineering

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
match = pd.read_csv('IPL Matches 2008-2020.csv')
ball = pd.read_csv('IPL Ball-by-Ball 2008-2020.csv')
```

```
match.head(5)
```

	id	city	date	player of match	venue	neutral venue	team1

```
ball.head(5)
```

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

```
match.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 816 entries, 0 to 815
Data columns (total 17 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   id               816 non-null    int64  
 1   city             803 non-null    object 
 2   date             816 non-null    object 
 3   player_of_match  812 non-null    object 
 4   venue            816 non-null    object 
 5   neutral_venue    816 non-null    int64  
 6   team1            816 non-null    object 
 7   team2            816 non-null    object 
 8   toss_winner      816 non-null    object 
 9   toss_decision    816 non-null    object 
 10  winner           812 non-null    object 
 11  result           812 non-null    object 
 12  result_margin    799 non-null    float64
 13  eliminator       812 non-null    object 
 14  method           19 non-null     object 
 15  umpire1          816 non-null    object 
 16  umpire2          816 non-null    object 
dtypes: float64(1), int64(2), object(14)
memory usage: 108.5+ KB
```

```
ball.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 193468 entries, 0 to 193467
```

```
Data columns (total 18 columns):
 #   Column            Non-Null Count   Dtype  
---  ------            --------------   -----  
 0   id                193468 non-null  int64  
 1   inning            193468 non-null  int64  
 2   over              193468 non-null  int64  
 3   ball              193468 non-null  int64  
 4   batsman           193468 non-null  object 
 5   non_striker       193468 non-null  object 
 6   bowler            193468 non-null  object 
 7   batsman_runs      193468 non-null  int64  
 8   extra_runs        193468 non-null  int64  
 9   total_runs        193468 non-null  int64  
 10  non_boundary      193468 non-null  int64  
 11  is_wicket         193468 non-null  int64  
 12  dismissal_kind    9495 non-null    object 
 13  player_dismissed  9495 non-null    object 
 14  fielder           6784 non-null    object 
 15  extras_type       10233 non-null   object 
 16  batting_team      193468 non-null  object 
 17  bowling_team      193277 non-null  object 
dtypes: int64(9), object(9)
memory usage: 26.6+ MB
```

```
match.isnull().sum()
```

```
id                  0
city               13
date                0
player_of_match     4
venue               0
neutral_venue       0
team1               0
team2               0
toss_winner         0
toss_decision       0
winner              4
result              4
result_margin      17
eliminator          4
method            797
umpire1             0
umpire2             0
dtype: int64
```

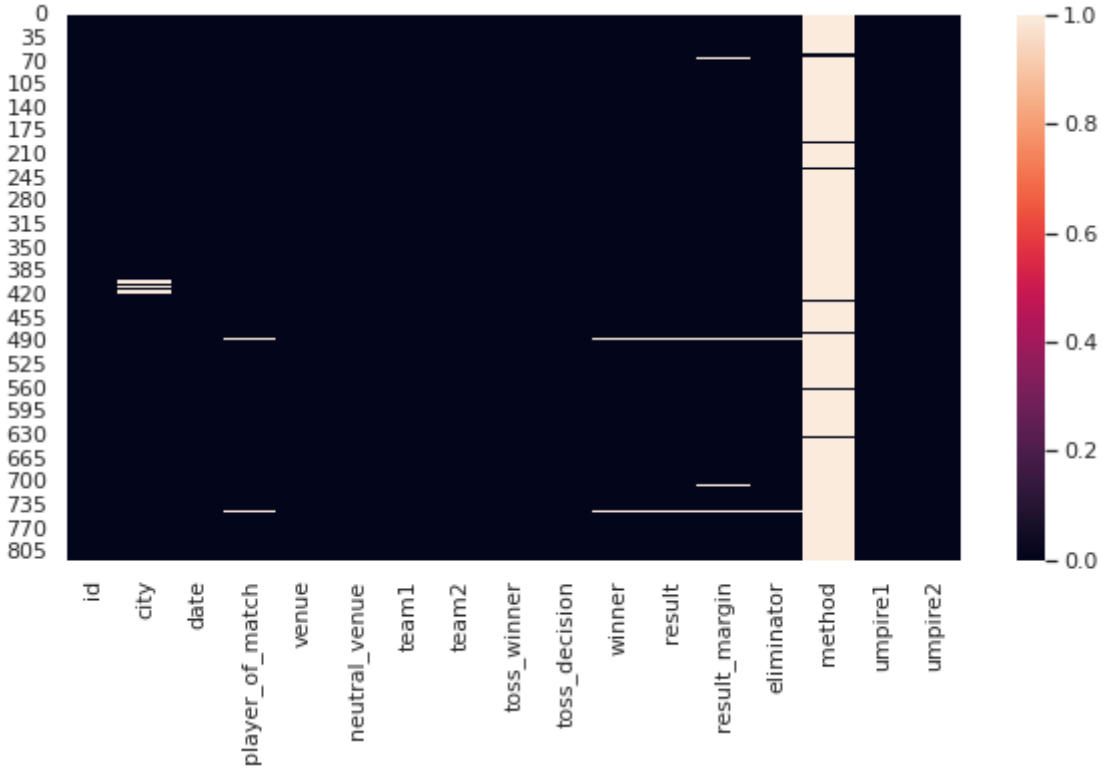
```
ball.isnull().sum()
```

```
id                    0
inning                0
over                  0
ball                  0
batsman               0
non_striker           0
bowler                0
batsman_runs          0
extra_runs            0
total_runs            0
non_boundary          0
is_wicket             0
```

```
dismissal_kind      183973
player_dismissed    183973
fielder             186684
extras_type         183235
batting_team             0
bowling_team           191
dtype: int64
```

```
sns.heatmap(match.isnull())
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f67f930cbd0>
```




```
sns.heatmap(ball.isnull())
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f67f9269250>
```


```
match['Season'] = pd.DatetimeIndex(match['date']).year #Set year from one column of date i
match.head(3)
```

	id	city	date	player_of_match	venue	neutral_venue	team1
0	335982	Bangalore	2008-04-18	BB McCullum	M Chinnaswamy Stadium	0	Roya Challengers Bangalore
1	335983	Chandigarh	2008-04-19	MEK Hussey	Punjab Cricket Association Stadium, Mohali	0	Kings X Punjab
2	335984	Delhi	2008-04-19	MF Maharoof	Feroz Shah Kotla	0	Delh Daredevils

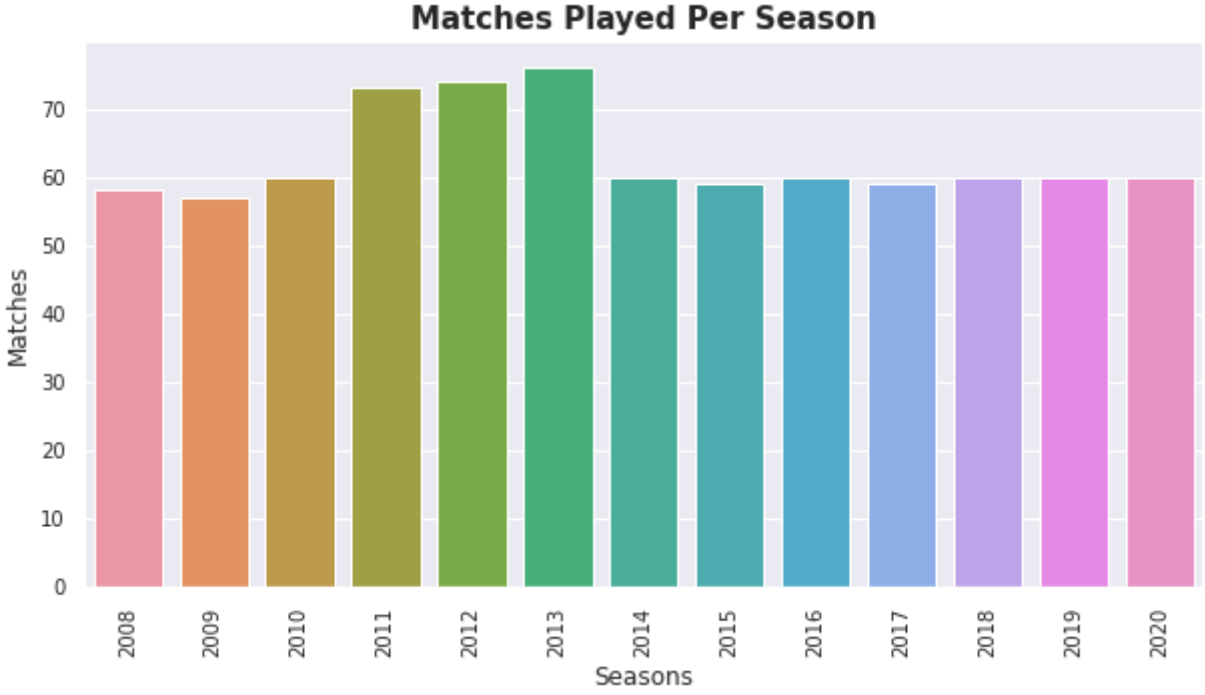
```
#Total matches played per season.
match_per_season = match.groupby(['Season'])['id'].count().reset_index().rename(columns={'
match_per_season
```

	Season	Matches
0	2008	58
1	2009	57
2	2010	60
3	2011	73
4	2012	74
5	2013	76
6	2014	60
7	2015	59
8	2016	60
9	2017	59
10	2018	60
11	2019	60
12	2020	60

```
#Total matches played per season in a plot.
sns.countplot(match['Season']) #Plot countplot as it is.
plt.xticks(rotation = 90, fontsize = 10) #Stylize on X axis.
plt.yticks(fontsize = 10) #Stylize X axis.
```

```
plt.xlabel('Seasons', fontsize  = 12 ) #Stylize X header.
plt.ylabel('Matches', fontsize = 12) #Stylize X header.
plt.title('Matches Played Per Season', fontsize = 15, fontweight = 'bold') #Add and styliz
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
Text(0.5, 1.0, 'Matches Played Per Season')
```




```
#Make a new data set season by putting Season from Match data in Ball data, by ID.
seasons = match[['id', 'Season']].merge(ball, left_on= 'id',right_on = 'id', how = 'left')
```

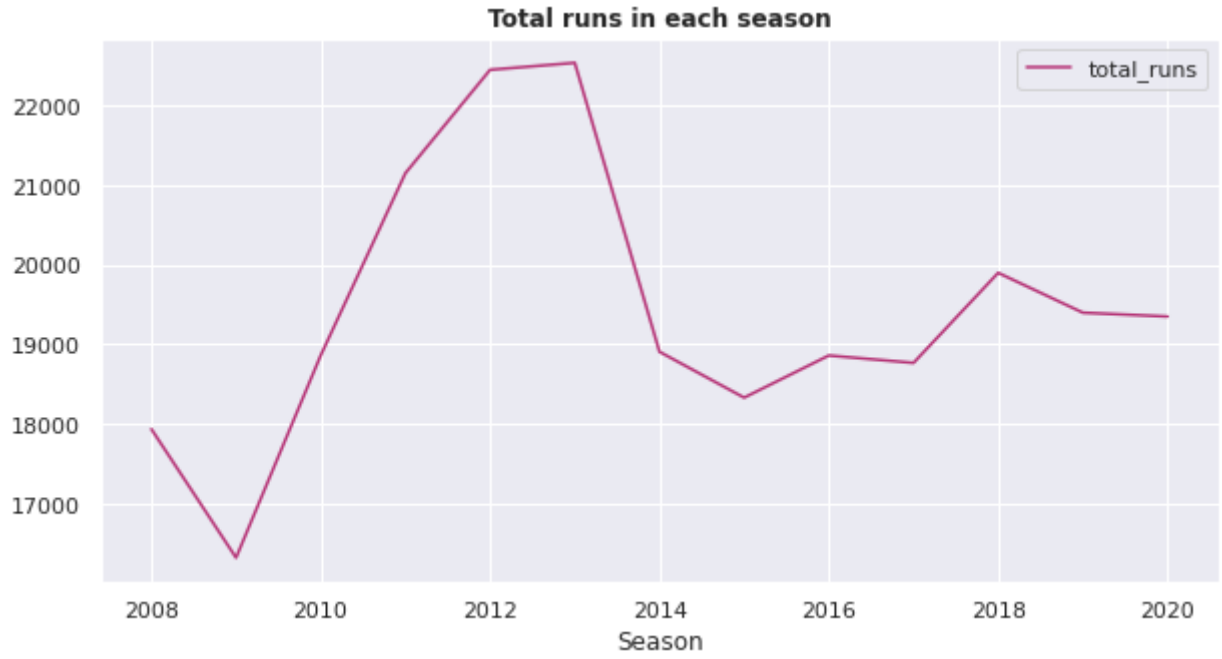
```
seasons.head(3)
```

	id	Season	inning	over	ball	batsman	non_striker	bowler	batsman_runs
0	335982	2008	1	6	5	RT Ponting	BB McCullum	AA Noffke	1
1	335982	2008	1	6	6	BB McCullum	RT Ponting	AA Noffke	1
2	335982	2008	1	7	1	BB McCullum	RT Ponting	Z Khan	0

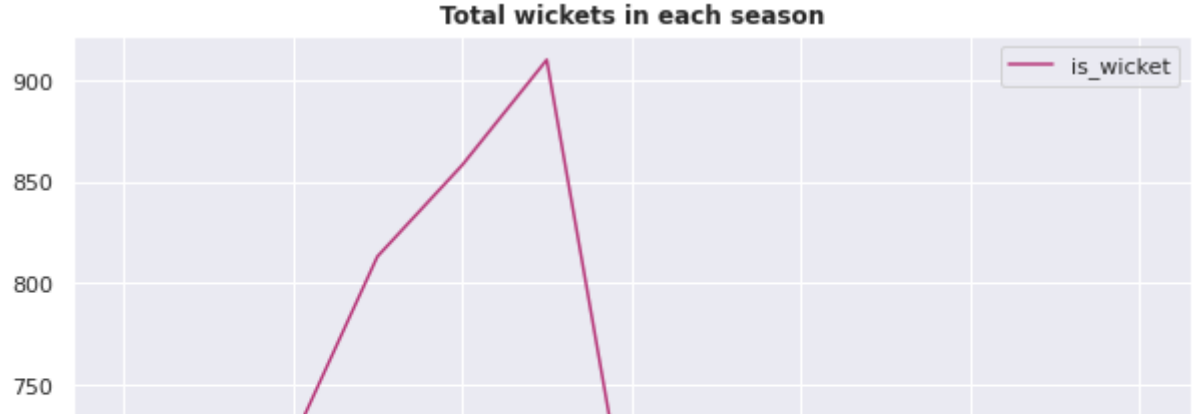
```
seasons = seasons.drop('id', axis=1) #Drop column with 'id'
#Could be done in one step - seasons = match[['id', 'Season']].merge(ball, left_on= 'id',r
seasons.head(3)
```

	Season	inning	over	ball	batsman	non_striker	bowler	batsman_runs	extra_ru
0	2008	1	6	5	RT Ponting	BB McCullum	AA Noffke	1	
1	2008	1	6	6	BB McCullum	RT Ponting	AA Noffke	1	
2	2008	1	7	1	BB McCullum	RT Ponting	Z Khan	0	

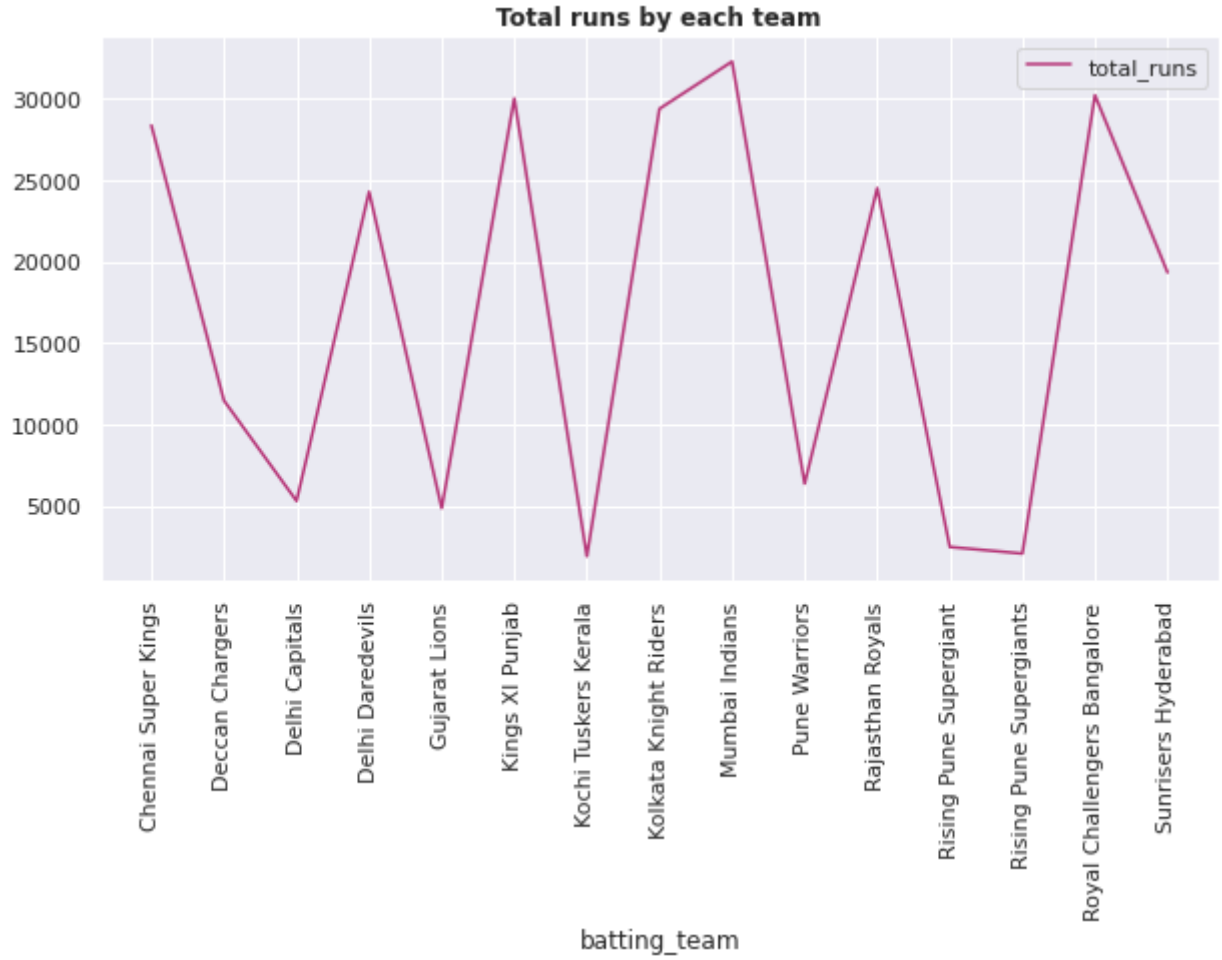
```
#Plot a lineplot to show runs scored across seasons.
runs_season=seasons.groupby(['Season'])['total_runs'].sum().reset_index() #Make a new list
rs=runs_season.set_index('Season') #Set_index to convert it to a dataframe.
ax = plt.axes()
sns.lineplot(data=rs,palette="magma") #Line colour
plt.title('Total runs in each season',fontsize=12,fontweight="bold") #Stylize Title
plt.show()
```




```
#Plot a lineplot to show wickets across seasons.
wickets_season=seasons.groupby(['Season'])['is_wicket'].sum().reset_index() #Make a new li
ws=wickets_season.set_index('Season') #Set_index to convert it to a dataframe.
ax = plt.axes()
sns.lineplot(data=ws,palette="magma") #Line colour
plt.title('Total wickets in each season',fontsize=12,fontweight="bold") #Stylize Title
plt.show()
```



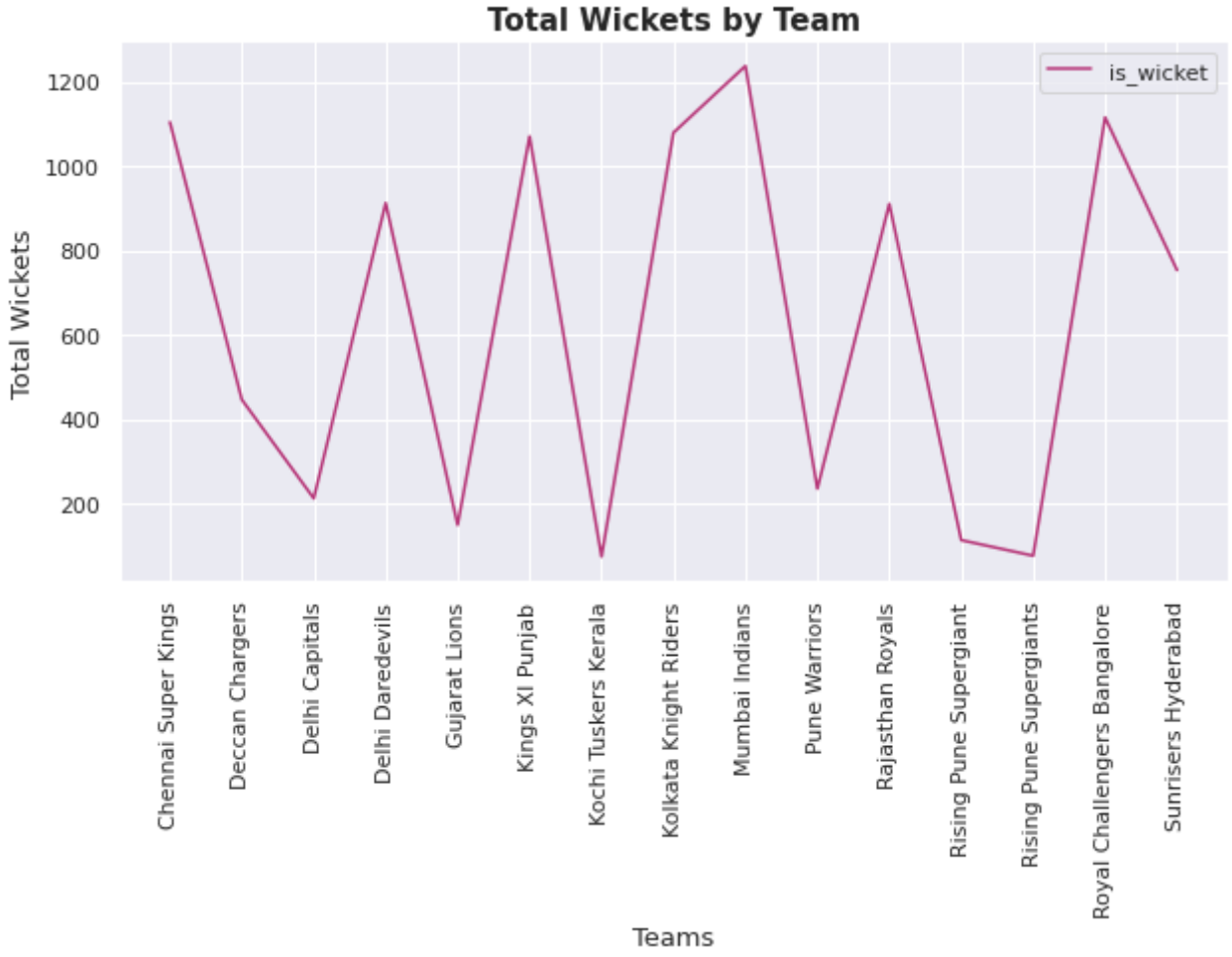

```
#Plot a lineplot to show runs across teams.
team_runs=seasons.groupby(['batting_team'])['total_runs'].sum().reset_index() #Make a new
tr=team_runs.set_index('batting_team') #Set_index to convert it to a dataframe.
ax = plt.axes()
sns.lineplot(data=tr,palette="magma") #Line colour
plt.xticks(rotation = 90)
plt.title('Total runs by each team',fontsize=12,fontweight="bold") #Stylize Title
plt.show()
```



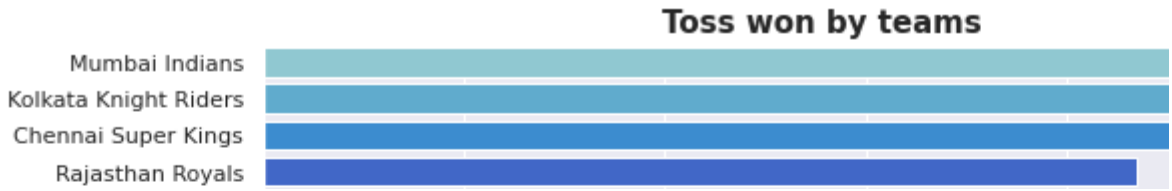

```
#Plot a lineplot to show runs across teams.
wickets_team = seasons.groupby(['bowling_team'])['is_wicket'].sum().reset_index()
wt = wickets_team.set_index('bowling_team')
ax = plt.axes()
sns.lineplot(data = wt, palette = 'magma')
plt.title('Total Wickets by Team', fontsize = 15, fontweight = 'bold')
plt.xticks(rotation=90)
```

```
plt.xlabel('Teams', fontsize = 13 )
plt.ylabel('Total Wickets', fontsize = 13)
```

```
Text(0, 0.5, 'Total Wickets')
```

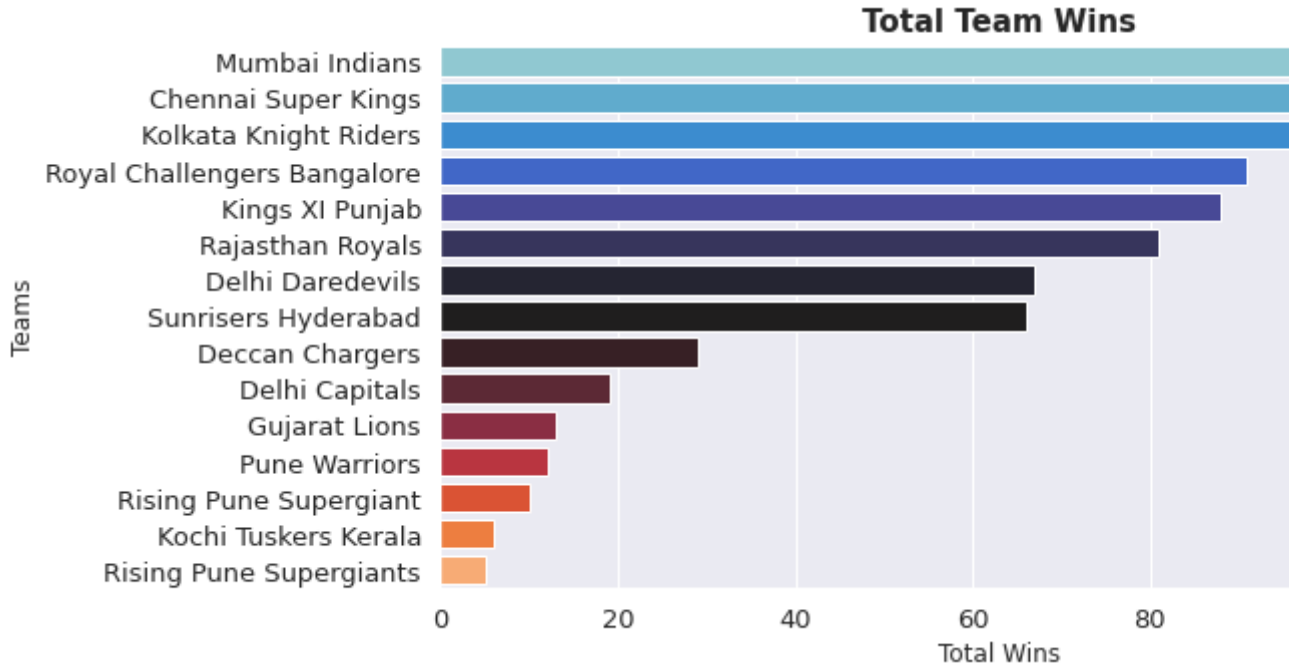



```
#Plot a barplot to show toss won per team.
toss = match['toss_winner'].value_counts()
ax = plt.axes()
sns.set(rc={'figure.figsize':(10,5)})
sns.barplot(y=toss.index, x=toss, orient='h', palette = 'icefire', saturation = 1)
plt.title('Toss won by teams', fontsize = 15, fontweight = 'bold')
plt.xlabel('Toss Won')
plt.ylabel('Teams')
plt.show()
```



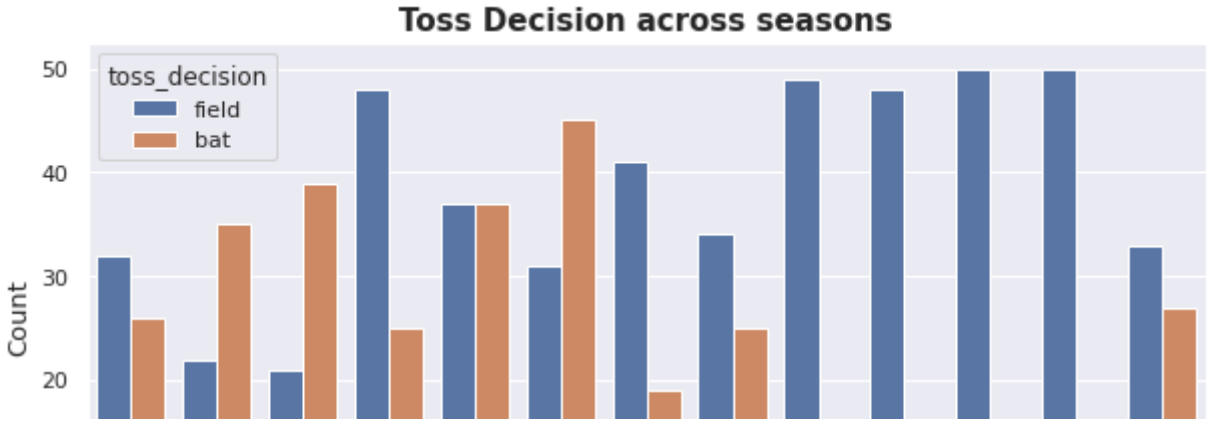

```
#Plot a barplot to show matches won per team.
win = match['winner'].value_counts()
ax = plt.axes()
sns.set(rc={'figure.figsize': (10,5)})
sns.barplot(y=win.index, x=win, orient='h', palette = 'icefire', saturation = 1)
plt.xticks(fontsize = 13)
plt.yticks(fontsize = 13)
plt.xlabel('Total Wins')
plt.ylabel('Teams')
plt.title('Total Team Wins', fontsize = 15, fontweight = 'bold')
```

```
Text(0.5, 1.0, 'Total Team Wins')
```



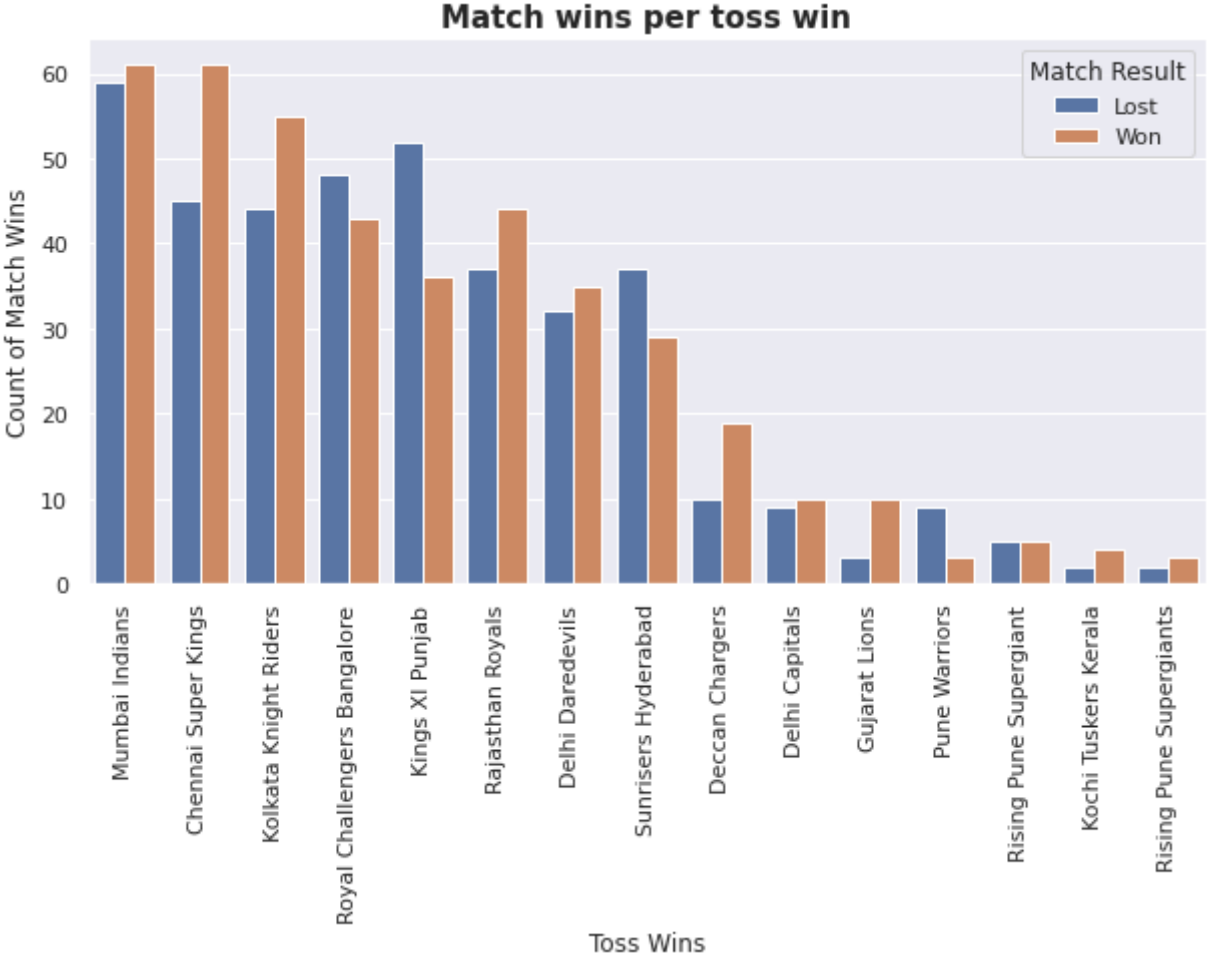

```
#Toss decision across seasons.
ax = plt.axes()
sns.set({'figure.figsize':(10,5)})
sns.countplot(x='Season', hue = 'toss_decision', data= match)
plt.xlabel('Seasons', fontsize = 13)
plt.ylabel('Count', fontsize = 13)
plt.title('Toss Decision across seasons', fontsize = 15, fontweight = 'bold')
plt.xticks(rotation = 90)
```

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12]),
 <a list of 13 Text major ticklabel objects>)
```



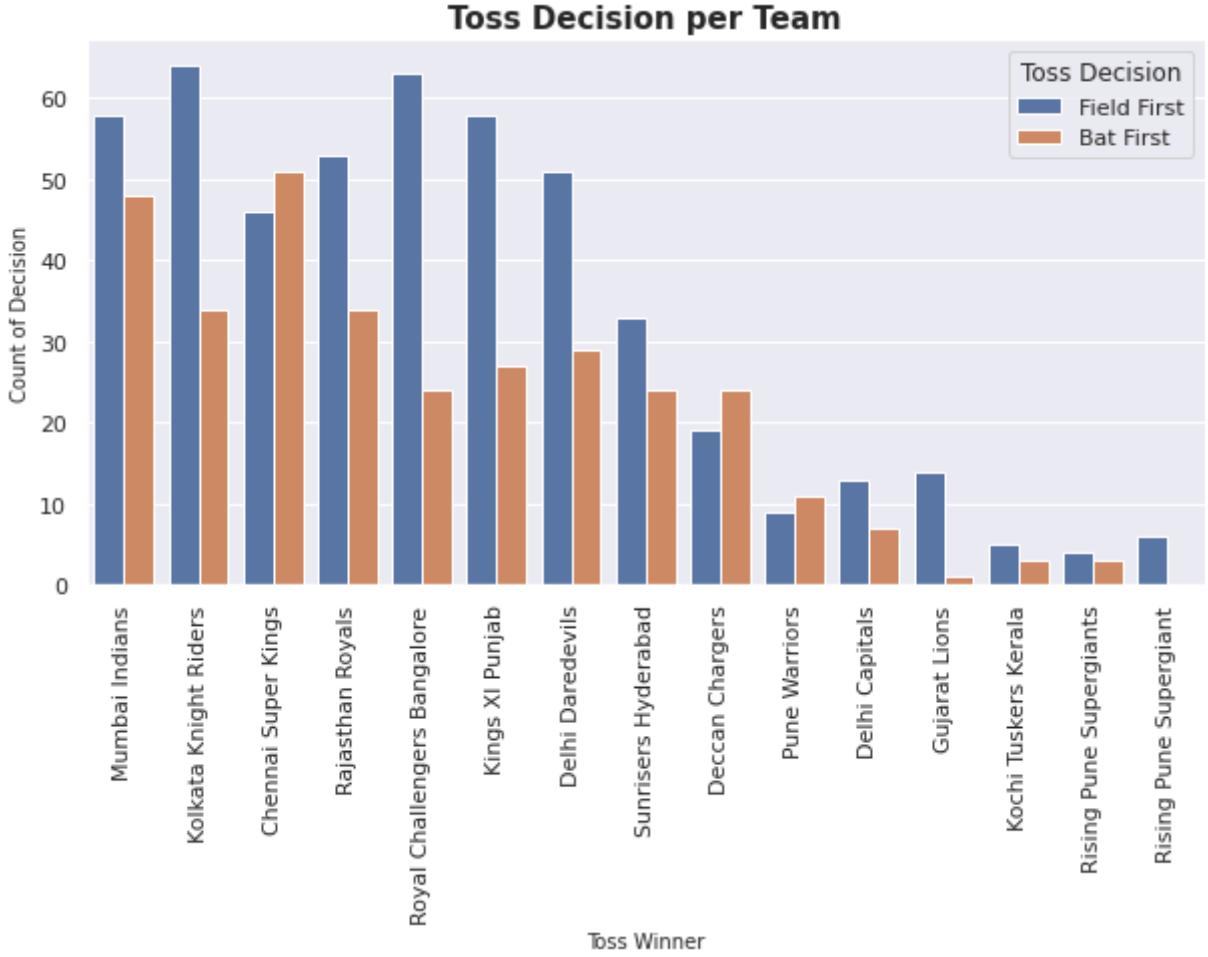

```
#How many matches each team won after winning and loosing toss.
twmw = match['toss_winner'] == match['winner']
ax = plt.axes()
sns.set({'figure.figsize':(10,5)})
sns.countplot(match['winner'], hue=twmw, order = match['winner'].value_counts().index) #Or
plt.xticks(rotation = 90)
plt.xlabel('Toss Wins')
plt.ylabel('Count of Match Wins')
plt.title('Match wins per toss win', fontsize = 15, fontweight = 'bold')
plt.legend(title='Match Result', loc='upper right', labels=['Lost', 'Won']) #Rename Legend
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
<matplotlib.legend.Legend at 0x7f67f9dd8f50>
```



```
#What was decision of teams after winning toss?
twmw = match['toss_winner'] == match['winner']
ax = plt.axes()
sns.set({'figure.figsize':(10,5)})
sns.countplot(match['toss_winner'], hue=match['toss_decision'], order = match['toss_winner
plt.xticks(rotation = 90)
plt.legend(title = 'Toss Decision', loc = 'upper right',labels = ['Field First', 'Bat Firs
plt.xlabel('Toss Winner', fontsize = 10)
plt.ylabel('Count of Decision', fontsize = 10)
plt.title('Toss Decision per Team', fontsize = 15, fontweight = 'bold')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
Text(0.5, 1.0, 'Toss Decision per Team')
```




```
#Place a countplot most wicket taking bowlers.
#First fill all NA in dismissal kind with 'Not a wicket'.
ball['dismissal_kind'].fillna('Not a Wicket', inplace = True)
ball.dismissal_kind.unique()
```

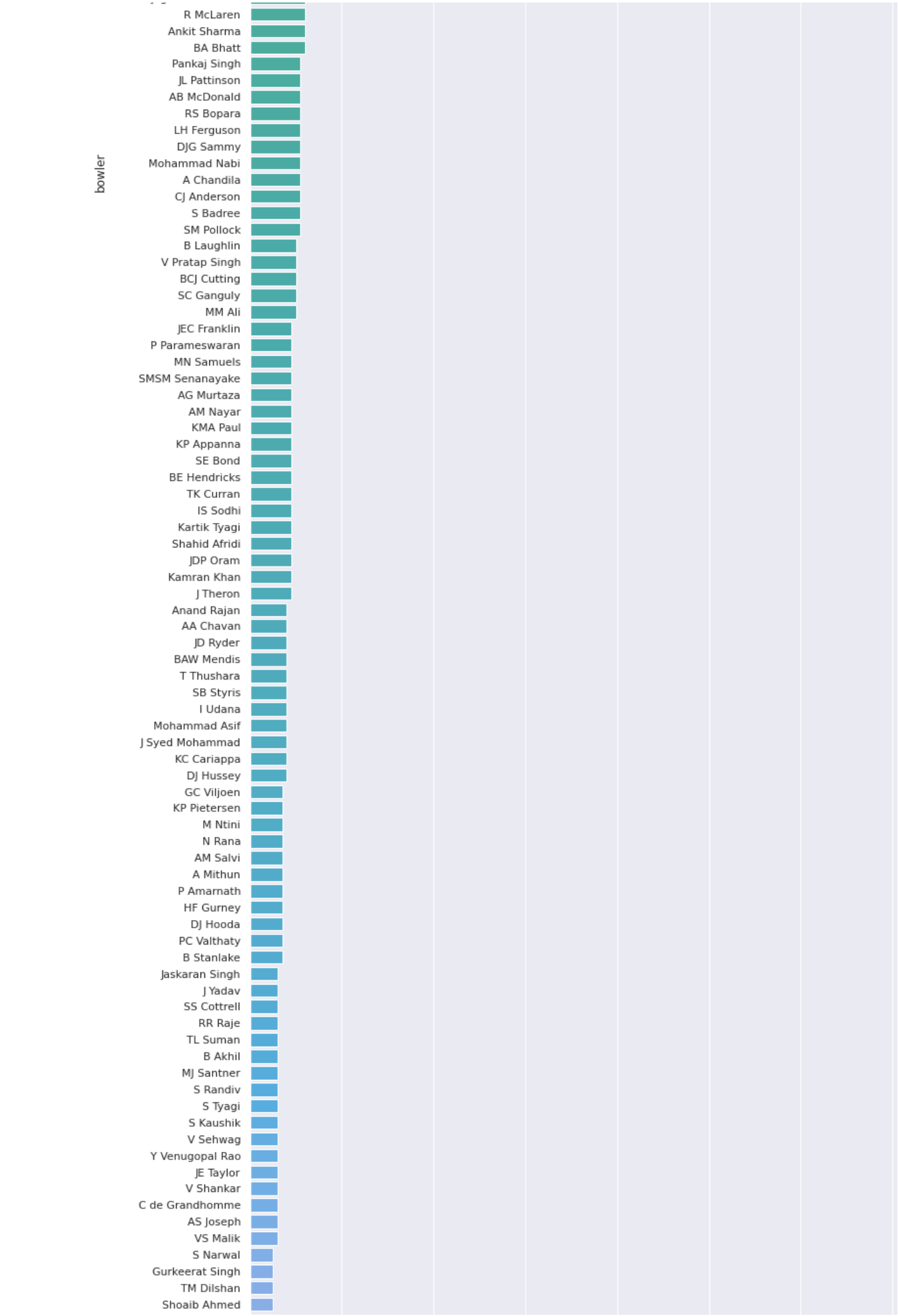
```
array(['Not a Wicket', 'caught', 'run out', 'bowled', 'lbw',
       'retired hurt', 'stumped', 'caught and bowled', 'hit wicket',
       'obstructing the field'], dtype=object)
```

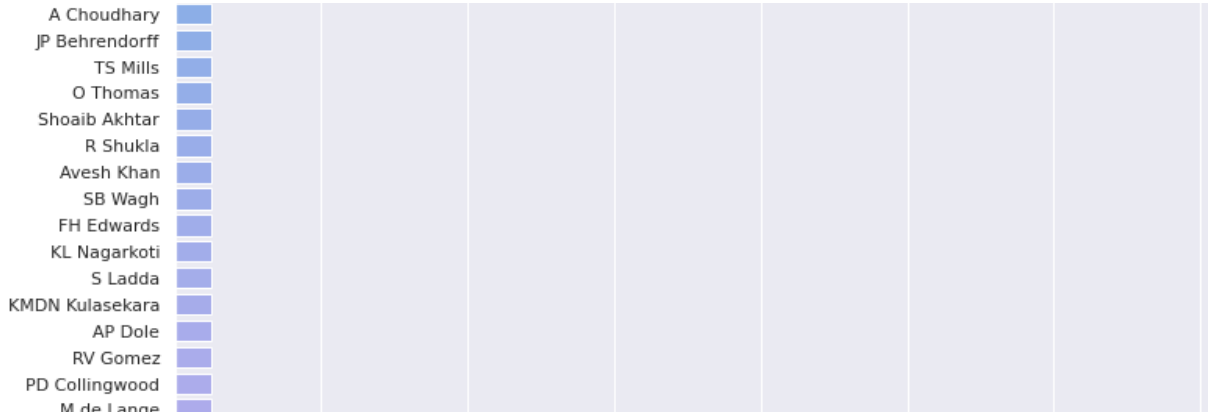
```
#Make a new data frame of all entries with wickets.
#Multiple Filter on a single column.
yes_bowler_wickets = ['caught', 'bowled', 'lbw', 'stumped', 'caught and bowled', 'hit wick
```

```
bowler_wickets = ball[ball['dismissal_kind'].isin(yes_bowler_wickets)]
print('\n Type of Dismissals', '\n' ,bowler_wickets['dismissal_kind'].unique())
```

```
 Type of Dismissals
 ['caught' 'bowled' 'lbw' 'stumped' 'caught and bowled' 'hit wicket']
```

```
plt.figure(figsize=(15,112))
sns.countplot(y=bowler_wickets['bowler'],
              order = bowler_wickets['bowler'].value_counts().index)
plt.title('Most wicket taking bowler!')
plt.show()
```


bowler
R McLaren
Ankit Sharma
BA Bhatt
Pankaj Singh
JL Pattinson
AB McDonald
RS Bopara
LH Ferguson
DJG Sammy
Mohammad Nabi
A Chandila
CJ Anderson
S Badree
SM Pollock
B Laughlin
V Pratap Singh
BCJ Cutting
SC Ganguly
MM Ali
JEC Franklin
P Parameswaran
MN Samuels
SMSM Senanayake
AG Murtaza
AM Nayar
KMA Paul
KP Appanna
SE Bond
BE Hendricks
TK Curran
IS Sodhi
Kartik Tyagi
Shahid Afridi
JDP Oram
Kamran Khan
J Theron
Anand Rajan
AA Chavan
JD Ryder
BAW Mendis
T Thushara
SB Styris
I Udana
Mohammad Asif
J Syed Mohammad
KC Cariappa
DJ Hussey
GC Viljoen
KP Pietersen
M Ntini
N Rana
AM Salvi
A Mithun
P Amarnath
HF Gurney
DJ Hooda
PC Valthaty
B Stanlake
Jaskaran Singh
J Yadav
SS Cottrell
RR Raje
TL Suman
B Akhil
MJ Santner
S Randiv
S Tyagi
S Kaushik
V Sehwag
Y Venugopal Rao
JE Taylor
V Shankar
C de Grandhomme
AS Joseph
VS Malik
S Narwal
Gurkeerat Singh
TM Dilshan
Shoaib Ahmed


A Choudhary
JP Behrendorff
TS Mills
O Thomas
Shoaib Akhtar
R Shukla
Avesh Khan
SB Wagh
FH Edwards
KL Nagarkoti
S Ladda
KMDN Kulasekara
AP Dole
RV Gomez
PD Collingwood