Spark Foundation

Task 4:Perform 'Exploratory Data Analysis' on dataset 'Indian Premier League'

Author:Sujata Sambhaji Gaikwad

In [1]:
```
#Import libraries in python
import pandas as pd
import numpy as np
```

In [2]:
```
# Loading dataset
data=pd.read_csv(r"C:\Users\HP\Downloads\deliveries.csv")
data
```

Out[2]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan	TS Mills
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan	TS Mills
...	...	...	...	...	...	...	...	...	...
179073	11415	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	SR Watson	SL Malinga
179074	11415	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	RA Jadeja	SL Malinga
179075	11415	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	RA Jadeja	SL Malinga
179076	11415	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	RA Jadeja	SL Malinga
179077	11415	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	RA Jadeja	SL Malinga

179078 rows × 21 columns

In [3]:
```
# Slicing useful data
data1=data.iloc[:,0:18]
data1
```

Out[3]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan	TS Mills
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan	TS Mills
...	...	...	...	...	...	...	...	...	...
179073	11415	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	SR Watson	SL Malinga
179074	11415	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	RA Jadeja	SL Malinga
179075	11415	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	RA Jadeja	SL Malinga
179076	11415	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	RA Jadeja	SL Malinga
179077	11415	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	RA Jadeja	SL Malinga

179078 rows × 18 columns

In [4]:
```
# find isnull
data1.isnull()
```

Out[4]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowl
0	False	False	False	False	False	False	False	False	Fals
1	False	False	False	False	False	False	False	False	Fals
2	False	False	False	False	False	False	False	False	Fals
3	False	False	False	False	False	False	False	False	Fals
4	False	False	False	False	False	False	False	False	Fals
...	...	...	...	...	...	...	...	...	
179073	False	False	False	False	False	False	False	False	Fals
179074	False	False	False	False	False	False	False	False	Fals
179075	False	False	False	False	False	False	False	False	Fals
179076	False	False	False	False	False	False	False	False	Fals
179077	False	False	False	False	False	False	False	False	Fals

179078 rows × 18 columns

In [5]:
```
# Cheack the null values
data1.isnull().sum()
```

Out[5]:
```
match_id         0
inning           0
batting_team     0
bowling_team     0
over             0
ball             0
batsman          0
non_striker      0
bowler           0
is_super_over    0
wide_runs        0
bye_runs         0
legbye_runs      0
noball_runs      0
penalty_runs     0
batsman_runs     0
extra_runs       0
total_runs       0
dtype: int64
```

In [6]:
```
# find the unique
data["batting_team"].unique()
```

Out[6]:
```
array(['Sunrisers Hyderabad', 'Royal Challengers Bangalore',
       'Mumbai Indians', 'Rising Pune Supergiant', 'Gujarat Lions',
       'Kolkata Knight Riders', 'Kings XI Punjab', 'Delhi Daredevils',
       'Chennai Super Kings', 'Rajasthan Royals', 'Deccan Chargers',
       'Kochi Tuskers Kerala', 'Pune Warriors', 'Rising Pune Supergiants',
       'Delhi Capitals'], dtype=object)
```

In [7]:
```
# convert string to numeric
from sklearn.preprocessing import LabelEncoder
l1=LabelEncoder()
data1["batting_team"]=l1.fit_transform(data1["batting_team"])
data1["bowling_team"]=l1.fit_transform(data1["bowling_team"])
data1["batsman"]=l1.fit_transform(data1["batsman"])
data1["non_striker"]=l1.fit_transform(data1["non_striker"])
data1["bowler"]=l1.fit_transform(data1["bowler"])
```

In [8]:
```
data1
```

Out[8]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler
0	1	1	14	13	1	1	112	391	379
1	1	1	14	13	1	2	112	391	379
2	1	1	14	13	1	3	112	391	379
3	1	1	14	13	1	4	112	391	379
4	1	1	14	13	1	5	112	391	379
...	...	...	...	...	...	...	...	...	...
179073	11415	2	0	8	20	2	361	442	340
179074	11415	2	0	8	20	3	444	360	340
179075	11415	2	0	8	20	4	444	360	340
179076	11415	2	0	8	20	5	437	360	340
179077	11415	2	0	8	20	6	437	360	340

179078 rows × 18 columns

In [9]:
```
# description of dataset
data1.describe()
```

Out[9]:

	match_id	inning	batting_team	bowling_team	over	b
count	179078.000000	179078.000000	179078.000000	179078.000000	179078.000000	179078.0000
mean	1802.252957	1.482952	6.918002	6.936223	10.162488	3.6155
std	3472.322805	0.502074	4.365744	4.364309	5.677684	1.8069
min	1.000000	1.000000	0.000000	0.000000	1.000000	1.0000
25%	190.000000	1.000000	3.000000	3.000000	5.000000	2.0000
50%	379.000000	1.000000	7.000000	7.000000	10.000000	4.0000
75%	567.000000	2.000000	10.000000	10.000000	15.000000	5.0000
max	11415.000000	5.000000	14.000000	14.000000	20.000000	9.0000

In [10]:
```
# correlation of dataset
data1.corr()
```

Out[10]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman
match_id	1.000000	0.003958	0.040812	0.039681	0.008268	-0.001349	-0.014857
inning	0.003958	1.000000	0.003734	-0.005977	-0.050076	-0.003943	0.000271
batting_team	0.040812	0.003734	1.000000	-0.107414	-0.002806	0.000173	-0.001028
bowling_team	0.039681	-0.005977	-0.107414	1.000000	0.000798	0.000337	-0.023301
over	0.008268	-0.050076	-0.002806	0.000798	1.000000	-0.007424	-0.021045
ball	-0.001349	-0.003943	0.000173	0.000337	-0.007424	1.000000	0.002939
batsman	-0.014857	0.000271	-0.001028	-0.023301	-0.021045	0.002939	1.000000
non_striker	-0.014577	-0.003418	-0.008615	-0.026033	-0.030445	-0.001176	-0.140019
bowler	-0.011718	-0.008651	-0.012497	0.031542	-0.006412	-0.000190	0.007195
is_super_over	-0.009150	0.084154	0.004189	0.006509	-0.034329	-0.001143	-0.003411
wide_runs	-0.007549	0.001201	-0.000101	0.002570	-0.010003	-0.004665	0.000345
bye_runs	0.000905	-0.000757	-0.000763	0.001498	0.012111	0.006602	-0.006761
legbye_runs	-0.012429	-0.001996	-0.003725	-0.002853	-0.004764	-0.002727	-0.005258
noball_runs	-0.004623	-0.000904	-0.000809	-0.002763	0.016984	0.000567	0.000530
penalty_runs	-0.001475	0.003442	-0.002999	-0.001866	-0.000979	0.000711	0.002474
batsman_runs	0.033510	-0.005362	0.001222	0.001176	0.086701	0.007950	-0.007038
extra_runs	-0.013323	-0.000531	-0.002766	0.000110	-0.002479	-0.002576	-0.004808
total_runs	0.030727	-0.005485	0.000634	0.001202	0.086326	0.007414	-0.008076

In [11]:
```
data1["total_runs"].unique()
```

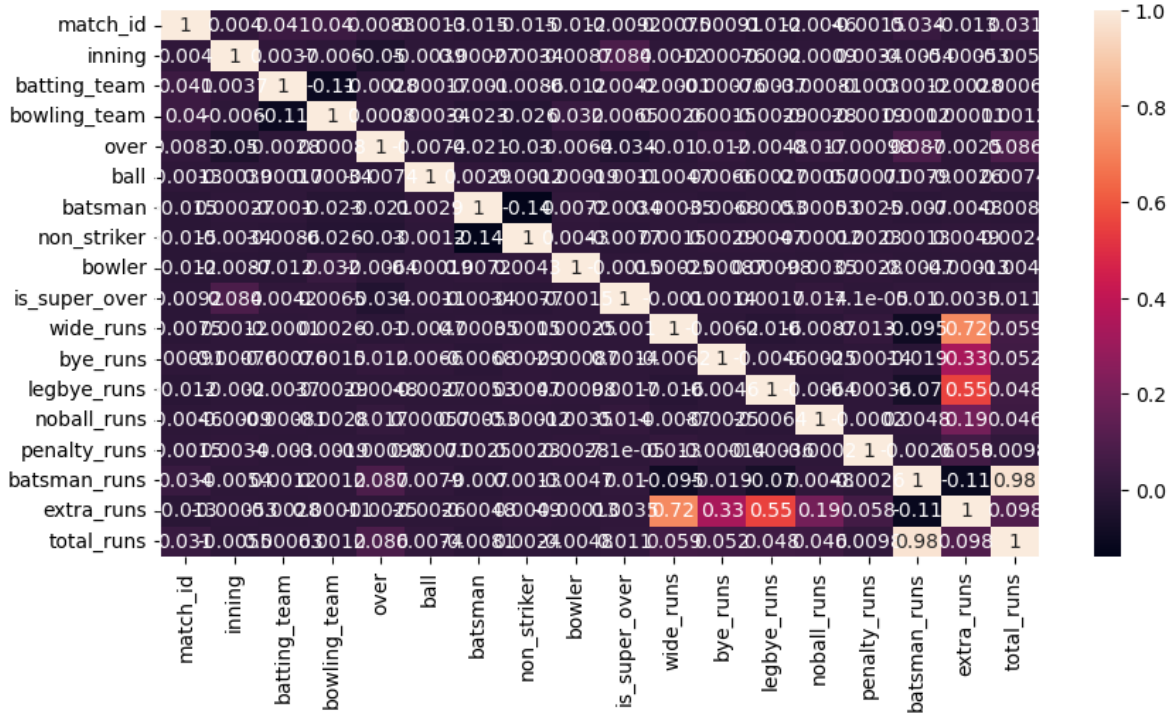
Out[11]:
```
array([ 0,  4,  2,  1,  6,  3,  5,  7, 10,  8], dtype=int64)
```

In [12]:
```
data1["total_runs"].value_counts()
```

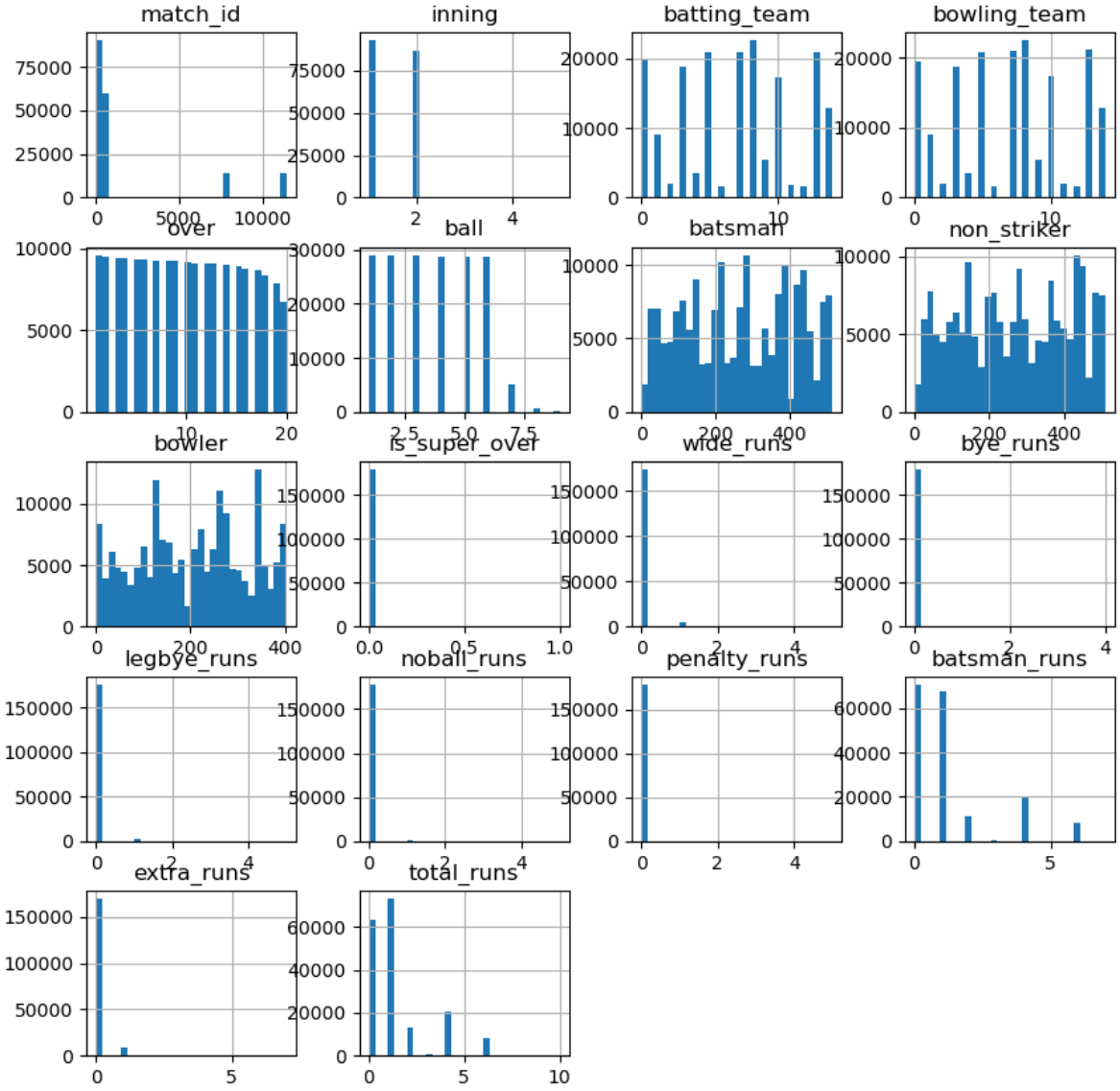
Out[12]:
```
1     73059
0     63002
4     20599
2     13125
6      8148
3       688
5       339
8        64
7        38
10       16
Name: total_runs, dtype: int64
```

In [13]:
```
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(10,5))
sns.heatmap(data1.corr(),annot=True)
```

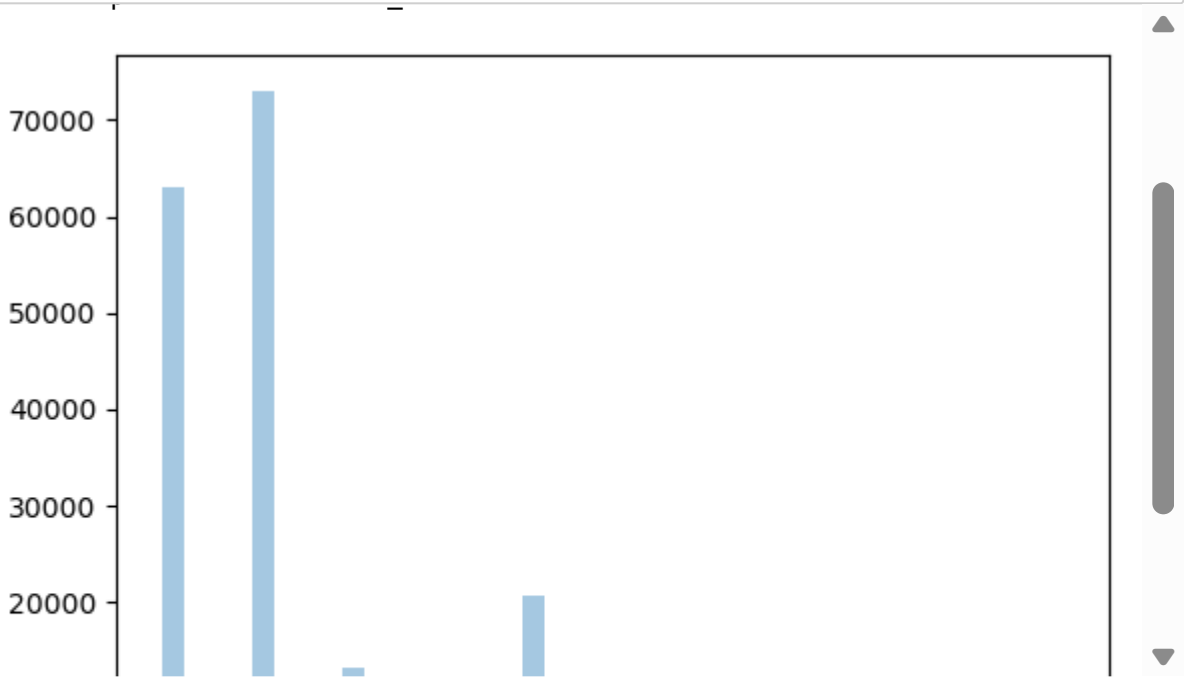
Out[13]:
```
<AxesSubplot:>
```



In [14]:
```
data1.hist(bins=30,figsize=[10,10])
plt.show()
```



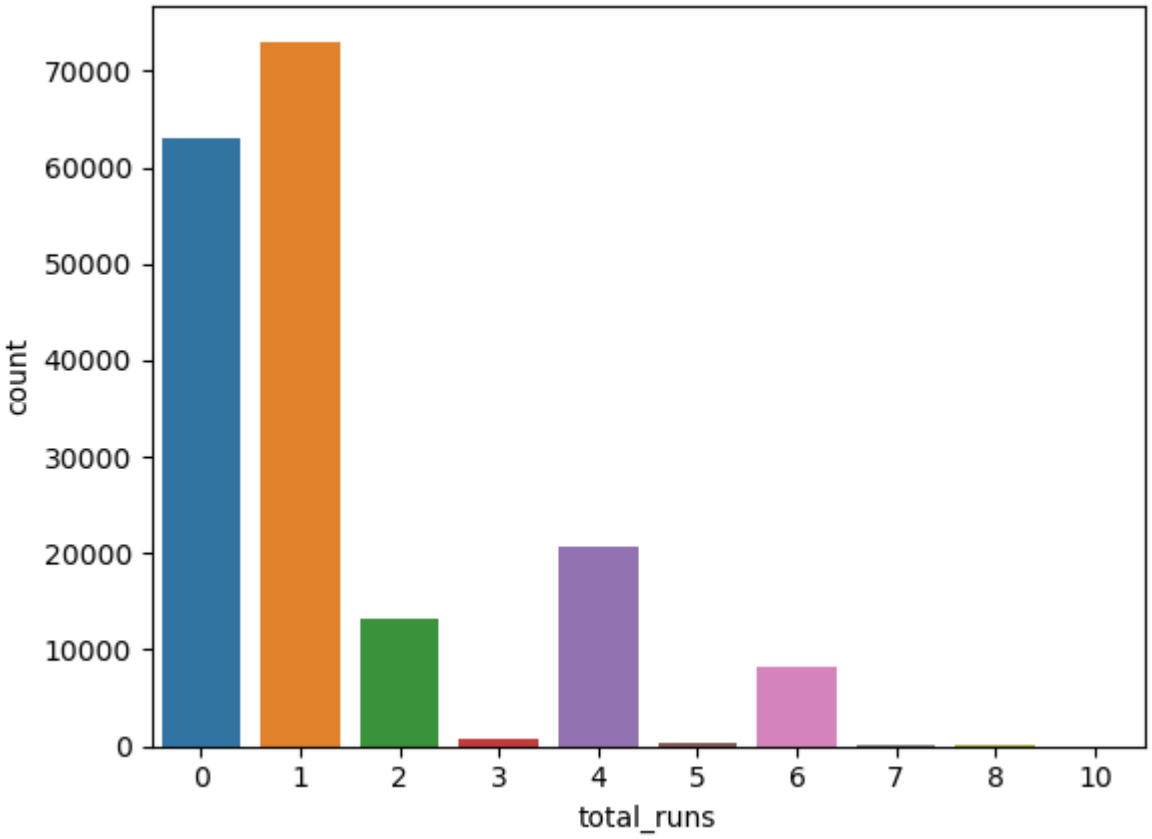
In [15]:
```
sns.distplot(data1["total_runs"],kde=False,bins=40)
```



In [16]:
```
sns.countplot(data1['total_runs'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[16]:
```
<AxesSubplot:xlabel='total_runs', ylabel='count'>
```

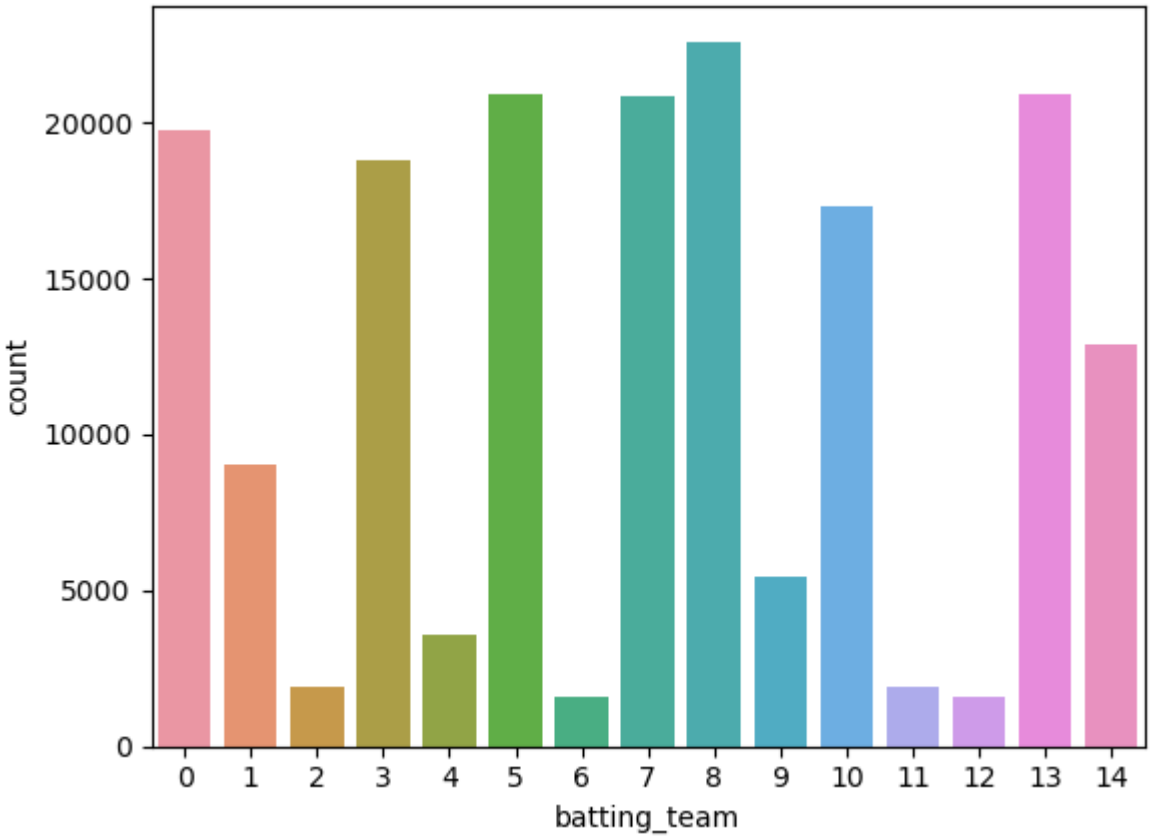


In this dataset Rising Pune Supergiant this most successful teams.

In [17]:
```
sns.countplot(data1['batting_team'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[17]: <AxesSubplot:xlabel='batting_team', ylabel='count'>



In [ ]: