zrxgz0f1s

December 22, 2024

```
[1]: import pandas as pd
```

```
[2]: import seaborn as sns
```

```
[3]: from matplotlib.pyplot import show
```

```
[4]: import matplotlib.pyplot as plt
```

```
[5]: a=pd.read_csv("C:/Users/suraj/Documents/Python Classes/PV22/IPL_Dataset.csv")
```

```
[47]: a.isnull().sum()
```

```
[47]: 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
      player_dismissed    0
      dismissal_kind      0
      fielder             0
      Total Extras        0
      dtype: int64
```

```
[7]: a.fillna('No Information Available',inplace=True)
```

```
[8]: # What is the total number of runs scored by each batting team
     a.groupby('batting_team')['total_runs'].sum()
```

```
[8]: batting_team
     Chennai Super Kings            20899
     Deccan Chargers                11463
     Delhi Daredevils               21953
     Gujarat Lions                   4862
     Kings XI Punjab                23068
     Kochi Tuskers Kerala            1901
     Kolkata Knight Riders          21965
     Mumbai Indians                 24521
     Pune Warriors                   6358
     Rajasthan Royals               17703
     Rising Pune Supergiant          2470
     Rising Pune Supergiants         2063
     Royal Challengers Bangalore    23436
     Sunrisers Hyderabad            11652
     Name: total_runs, dtype: int64
```

```
[9]: # How many wides have been bowled by each bowling team
     a.groupby('bowling_team')['wide_runs'].sum()
```

```
[9]: bowling_team
     Chennai Super Kings            526
     Deccan Chargers                328
     Delhi Daredevils               614
     Gujarat Lions                  107
     Kings XI Punjab                668
     Kochi Tuskers Kerala            56
     Kolkata Knight Riders          658
     Mumbai Indians                 786
     Pune Warriors                  174
     Rajasthan Royals               586
     Rising Pune Supergiant          69
     Rising Pune Supergiants         77
     Royal Challengers Bangalore    725
     Sunrisers Hyderabad            268
     Name: wide_runs, dtype: int64
```

```
[10]: # What is the average number of runs scored per inning by each team
      a.groupby(by=['batting_team','inning'])['total_runs'].mean()
```

```
[10]: batting_team                 inning
      Chennai Super Kings          1         1.339207
                                   2         1.307128
                                   3         1.800000
```

```
Deccan Chargers              1         1.281250
                             2         1.251465
Delhi Daredevils             1         1.257512
                             2         1.293123
                             4         1.833333
Gujarat Lions                1         1.313441
                             2         1.413210
                             4         0.750000
Kings XI Punjab              1         1.304182
                             2         1.317853
                             3         2.142857
                             4         2.500000
Kochi Tuskers Kerala         1         1.151826
                             2         1.263456
Kolkata Knight Riders        1         1.279835
                             2         1.269295
                             3         2.000000
Mumbai Indians               1         1.310780
                             2         1.274842
                             3         2.200000
Pune Warriors                1         1.214461
                             2         1.130217
Rajasthan Royals             1         1.275026
                             2         1.267466
                             4         2.500000
Rising Pune Supergiant       1         1.311871
                             2         1.286976
Rising Pune Supergiants      1         1.322733
                             2         1.285910
Royal Challengers Bangalore  1         1.368046
                             2         1.284413
                             3         2.500000
                             4         2.500000
Sunrisers Hyderabad          1         1.308332
                             2         1.246616
                             3         2.857143
Name: total_runs, dtype: float64
```

[11]:

```
# Which batsman has scored the highest number of runs
a.groupby('batsman')['total_runs'].sum().head(1)
```

[11]:

```
batsman
A Ashish Reddy    288
Name: total_runs, dtype: int64
```

[12]:

```
# What is the total runs scored by each over
a.groupby('over')['total_runs'].sum()
```

[12]:
```
over
1      7733
2      8993
3      9852
4     10207
5     10227
6     10397
7      8413
8      8966
9      9247
10     9047
11     9456
12     9694
13     9713
14     9900
15    10240
16    10397
17    10817
18    10899
19    10469
20     9647
Name: total_runs, dtype: int64
```

[13]:
```python
# Which bowler has taken the most wickets
a.groupby('bowler')['player_dismissed'].nunique().head(1)
```

[13]:
```
bowler
A Ashish Reddy    17
Name: player_dismissed, dtype: int64
```

[14]:
```python
# # How many dismissals have occurred of each dismissal kind
a.groupby('dismissal_kind')['player_dismissed'].nunique()
```

[14]:
```
dismissal_kind
No Information Available      1
bowled                      293
caught                      374
caught and bowled           120
hit wicket                    9
lbw                         174
obstructing the field         1
retired hurt                  7
run out                     263
stumped                     126
Name: player_dismissed, dtype: int64
```

```
[15]: # How many super overs have been played in total
      a['is_super_over'].sum()
```

```
[15]: 81
```

```
[16]: # What is the average number of extras ,wide runs, bye runs, legbye runs,␣
       ↪noball runs, penalty runs) conceded per inning by each bowling team #
      a['Total Extras'] = a[['wide_runs', 'bye_runs', 'legbye_runs', 'noball_runs',␣
       ↪'penalty_runs','extra_runs']].sum(axis=1)
      average_extras = a.groupby('bowling_team')[['wide_runs', 'bye_runs',␣
       ↪'legbye_runs', 'noball_runs', 'penalty_runs','extra_runs', 'Total Extras']].
       ↪mean().reset_index()
      average_extras
```

```
[16]:                    bowling_team  wide_runs  bye_runs  legbye_runs  \
      0            Chennai Super Kings   0.033800  0.003791     0.023198
      1                Deccan Chargers   0.036287  0.006859     0.023786
      2               Delhi Daredevils   0.035909  0.004445     0.023920
      3                  Gujarat Lions   0.030183  0.002821     0.018054
      4                Kings XI Punjab   0.038408  0.004772     0.022539
      5           Kochi Tuskers Kerala   0.034696  0.003098     0.025403
      6          Kolkata Knight Riders   0.037792  0.005629     0.021940
      7                 Mumbai Indians   0.041634  0.005085     0.021929
      8                  Pune Warriors   0.031886  0.006964     0.017775
      9               Rajasthan Royals   0.041528  0.004819     0.024024
      10         Rising Pune Supergiant   0.035788  0.001556     0.019710
      11        Rising Pune Supergiants   0.047678  0.001858     0.014861
      12   Royal Challengers Bangalore   0.040458  0.005859     0.022377
      13           Sunrisers Hyderabad   0.030153  0.003263     0.018902

          noball_runs  penalty_runs  extra_runs  Total Extras
      0      0.003599      0.000000    0.064388      0.128775
      1      0.005421      0.000553    0.072906      0.145813
      2      0.005146      0.000000    0.069419      0.138839
      3      0.001975      0.000000    0.053032      0.106065
      4      0.005002      0.000000    0.070722      0.141444
      5      0.004957      0.000000    0.068154      0.136307
      6      0.003791      0.000000    0.069152      0.138303
      7      0.005085      0.000265    0.073998      0.147995
      8      0.004765      0.000000    0.061389      0.122778
      9      0.004606      0.000000    0.074977      0.149954
      10     0.000519      0.000000    0.057573      0.115145
      11     0.002477      0.000000    0.066873      0.133746
      12     0.004353      0.000000    0.073047      0.146094
      13     0.002475      0.000000    0.054793      0.109586
```

[17]:

```python
# How many runs have been scored by each batsman-bowler pair
a.groupby(['batsman', 'bowler'])['total_runs'].sum()
```

[17]:

```
batsman         bowler
A Ashish Reddy  A Nehra            8
                AB Dinda           9
                AD Mathews        25
                AD Russell         4
                Anureet Singh      2
                                  ..
Z Khan          SR Watson          4
                Shakib Al Hasan    3
                Sohail Tanvir      3
                VRV Singh          1
                YK Pathan          1
Name: total_runs, Length: 17151, dtype: int64
```

[18]:

```python
# What is the most common dismissal kind
a['dismissal_kind'].max()
```

[18]:

```
'stumped'
```

[19]:

```python
# How many matches have been played by each batting team
a.groupby(by='batting_team')['match_id'].nunique()
```

[19]:

```
batting_team
Chennai Super Kings            131
Deccan Chargers                 75
Delhi Daredevils               147
Gujarat Lions                   30
Kings XI Punjab                148
Kochi Tuskers Kerala            14
Kolkata Knight Riders          148
Mumbai Indians                 157
Pune Warriors                   45
Rajasthan Royals               117
Rising Pune Supergiant          16
Rising Pune Supergiants         14
Royal Challengers Bangalore    152
Sunrisers Hyderabad             76
Name: match_id, dtype: int64
```

[20]:

```python
# What is the average number of runs scored in each over
a.groupby(by='over')['total_runs'].mean()
```

[20]:

```
over
1     0.955635
```

```
2     1.121601
3     1.242057
4     1.291698
5     1.298832
6     1.321933
7     1.074869
8     1.149635
9     1.189325
10    1.170981
11    1.233177
12    1.269347
13    1.275509
14    1.313694
15    1.373206
16    1.418031
17    1.495300
18    1.561685
19    1.600031
20    1.707736
Name: total_runs, dtype: float64
```

[21]:

```
# Which fielder has been involved in the most dismissals
b = a.groupby('fielder')['dismissal_kind'].count().reset_index()
b[['fielder','dismissal_kind']].
 ↪sort_values(by='dismissal_kind',ascending=False).head(2)
```

[21]:

```
                      fielder  dismissal_kind
273  No Information Available          145091
189                KD Karthik             127
```

[22]:

```
# How does the number of runs scored vary with the over number
a.groupby('over')['total_runs'].var().reset_index()
```

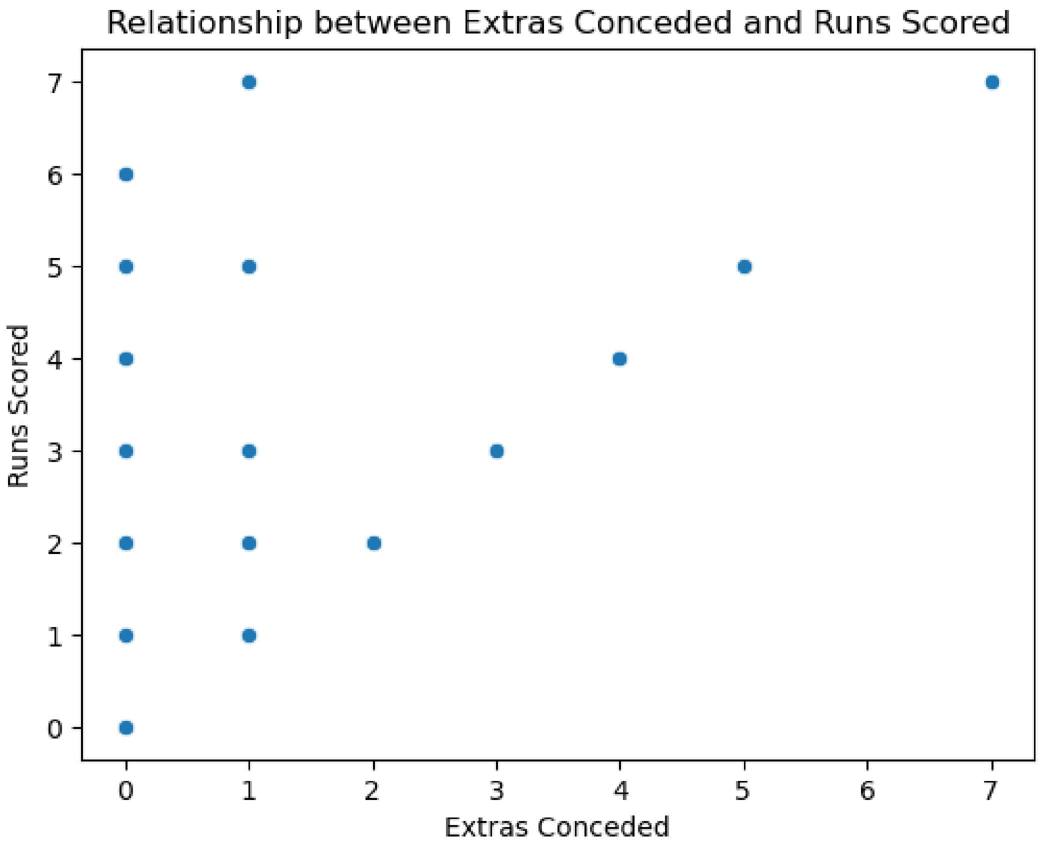
[22]:

```
    over  total_runs
0      1    2.062671
1      2    2.467312
2      3    2.714822
3      4    2.817952
4      5    2.840575
5      6    2.852094
6      7    1.790203
7      8    1.964142
8      9    2.077607
9     10    1.932833
10    11    2.163699
11    12    2.212802
12    13    2.327027
```

```
13    14    2.396871
14    15    2.523386
15    16    2.637531
16    17    2.786996
17    18    3.006620
18    19    3.076778
19    20    3.274871
```

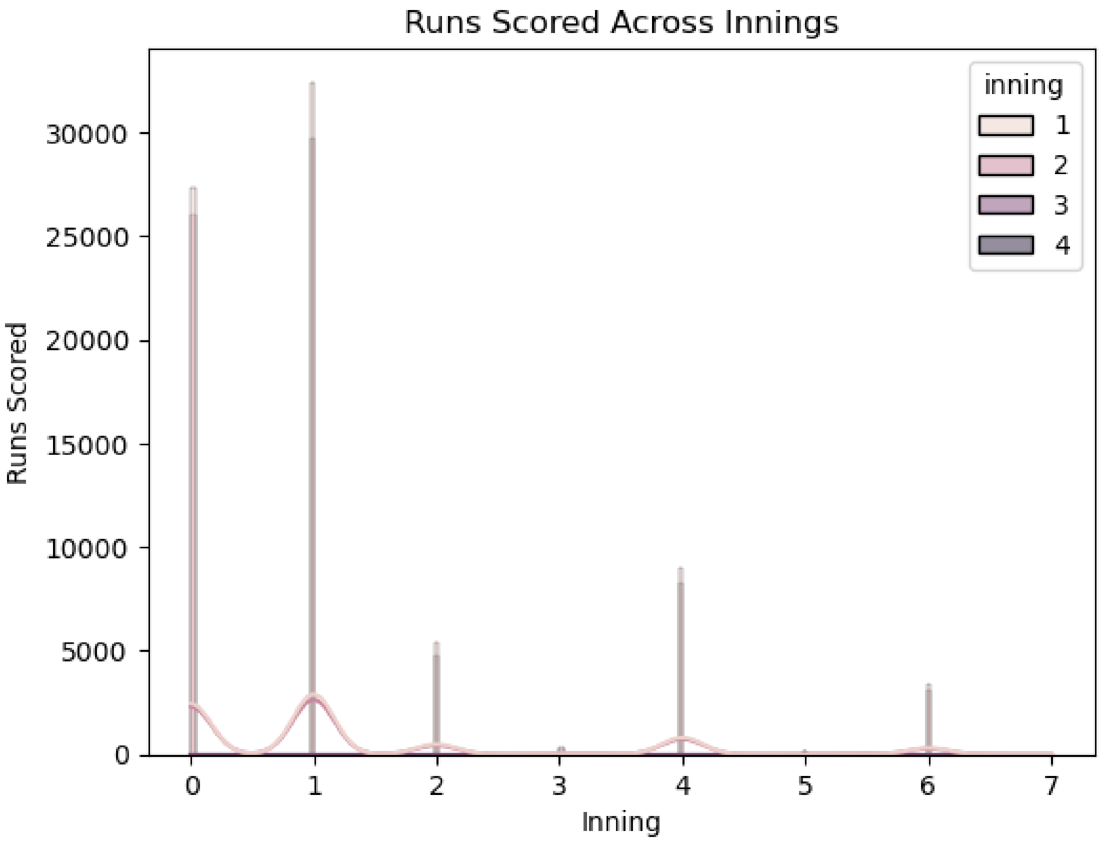
[49]:

```
# how is the relationship between the number of runs scored and the number of␣
 ↪extras conceded
sns.scatterplot(data=a, x='extra_runs', y='total_runs')
plt.title('Relationship between Extras Conceded and Runs Scored')
plt.xlabel('Extras Conceded')
plt.ylabel('Runs Scored')
plt.show()
```



[65]:

```
# Are there any notable trends in the distribution of runs scored by batting␣
 ↪teams across innings
sns.histplot(data=a,x='total_runs',hue='inning',kde=True)
plt.title('Runs Scored Across Innings')
plt.xlabel('Inning')
plt.ylabel('Runs Scored')
plt.show()
```



```
[73]: # Which batting team has the highest average runs per match
      b=a.groupby('batting_team')['total_runs'].mean().reset_index()
      b.loc[b['total_runs'].idxmax()]
```

```
[73]: batting_team    Gujarat Lions
      total_runs           1.363432
      Name: 3, dtype: object
```

```
[75]: # What is the total runs scored by each batsman
      a.groupby('batsman')['total_runs'].sum().reset_index()
```

```
[75]:             batsman  total_runs
      0    A Ashish Reddy         288
      1        A Chandila           4
      2          A Chopra          58
      3       A Choudhary          25
```

```
4        A Flintoff          72
..              …         …
456     YV Takawale         207
457   Yashpal Singh          49
458     Younis Khan           3
459    Yuvraj Singh        2710
460          Z Khan         121

[461 rows x 2 columns]
```

[79]:
```
# How does the average number of runs scored per inning vary across different␣
  ↪bowling teams
a.groupby(['bowling_team', 'inning'])['total_runs'].mean().reset_index()
```

[79]:
```
                   bowling_team  inning  total_runs
0           Chennai Super Kings       1    1.307204
1           Chennai Super Kings       2    1.244825
2           Chennai Super Kings       4    2.500000
3               Deccan Chargers       1    1.306978
4               Deccan Chargers       2    1.268249
5              Delhi Daredevils       1    1.319155
6              Delhi Daredevils       2    1.286621
7              Delhi Daredevils       3    2.500000
8                 Gujarat Lions       1    1.411168
9                 Gujarat Lions       2    1.464331
10                Gujarat Lions       3    2.200000
11              Kings XI Punjab       1    1.364118
12              Kings XI Punjab       2    1.311530
13              Kings XI Punjab       3    1.800000
14              Kings XI Punjab       4    1.500000
15         Kochi Tuskers Kerala       1    1.176190
16         Kochi Tuskers Kerala       2    1.289406
17        Kolkata Knight Riders       1    1.254968
18        Kolkata Knight Riders       2    1.277148
19        Kolkata Knight Riders       4    2.900000
20               Mumbai Indians       1    1.283357
21               Mumbai Indians       2    1.242813
22               Mumbai Indians       4    0.750000
23                Pune Warriors       1    1.267641
24                Pune Warriors       2    1.241454
25             Rajasthan Royals       1    1.278066
26             Rajasthan Royals       2    1.246365
27             Rajasthan Royals       3    2.050000
28       Rising Pune Supergiant       1    1.300416
29       Rising Pune Supergiant       2    1.226708
30      Rising Pune Supergiants       1    1.165138
31      Rising Pune Supergiants       2    1.476447
```

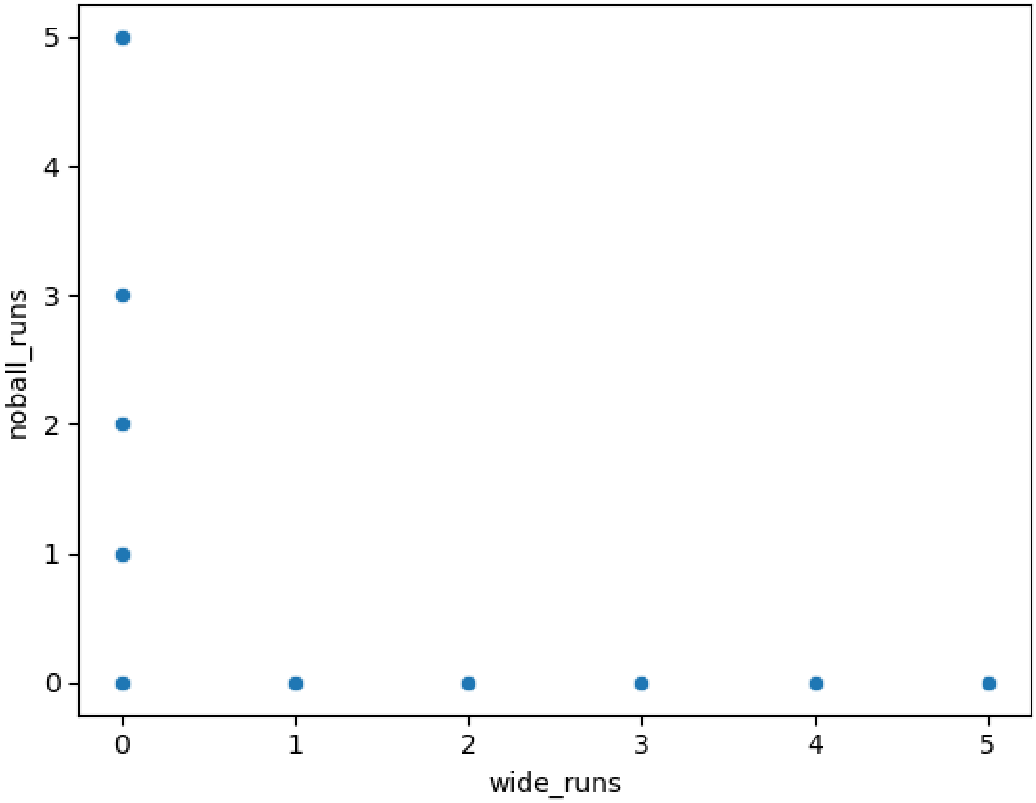
```
32  Royal Challengers Bangalore       1    1.328971
33  Royal Challengers Bangalore       2    1.303023
34  Royal Challengers Bangalore       3    2.857143
35  Royal Challengers Bangalore       4    1.833333
36          Sunrisers Hyderabad       1    1.263612
37          Sunrisers Hyderabad       2    1.307425
38          Sunrisers Hyderabad       4    2.500000
```

[81]:

```
# how is the relationship between the number of wides and the number of no␣
  ↪balls bowled
sns.scatterplot(data=a,x='wide_runs',y='noball_runs')
```

[81]:

```
<Axes: xlabel='wide_runs', ylabel='noball_runs'>
```



[209]:

```
# Which batting team has the highest number of centuries
century_data = a.groupby(['match_id', 'batsman',␣
  ↪'batting_team'])['batsman_runs'].sum().reset_index()
centuries = century_data[century_data['batsman_runs'] >= 100]
team_centuries = centuries['batting_team'].value_counts()
top_team = team_centuries.idxmax()
century_count = team_centuries.max()
```

```
print(f"Team with the highest number of centuries: {top_team} ({century_count}␣
  ↪centuries)")
```

```
Team with the highest number of centuries: Royal Challengers Bangalore (12
centuries)
```

[147]:
```
# How does the number of wides vary with the over number
a.groupby('over')['wide_runs'].var().reset_index()
```

[147]:
```
    over  wide_runs
0      1   0.115240
1      2   0.088267
2      3   0.086086
3      4   0.061003
4      5   0.056791
5      6   0.077189
6      7   0.093290
7      8   0.065692
8      9   0.053336
9     10   0.057519
10    11   0.036218
11    12   0.057840
12    13   0.061646
13    14   0.055387
14    15   0.057370
15    16   0.056091
16    17   0.051758
17    18   0.058824
18    19   0.053401
19    20   0.074004
```

[161]:
```
# What is the average number of runs scored per inning in matches with a super␣
  ↪over
matches_with_super_over = a[a['is_super_over'] == True]['match_id'].unique()
super_over_matches_data = a[a['match_id'].isin(matches_with_super_over) &␣
  ↪(a['inning'] != 'is_super_over')]
average_runs = super_over_matches_data['is_super_over'].mean()
print(f"Average runs scored per inning in matches with a super over:␣
  ↪{average_runs:.2f}")
```

```
Average runs scored per inning in matches with a super over: 0.04
```

[163]:
```
# How does the number of extras vary with the over number
a.groupby('over')['extra_runs'].var().reset_index()
```

[163]:
```
    over  extra_runs
0      1    0.184045
```

```
1      2    0.163345
2      3    0.149933
3      4    0.123786
4      5    0.104646
5      6    0.149661
6      7    0.150770
7      8    0.114552
8      9    0.099550
9     10    0.108516
10    11    0.080901
11    12    0.094627
12    13    0.112725
13    14    0.107488
14    15    0.094257
15    16    0.097857
16    17    0.113437
17    18    0.123617
18    19    0.111671
19    20    0.154124
```

[167]:
```
# Which bowler has the highest economy rate
a['economy_rate'] = a['total_runs'] / a['over']
highest_economy = a.groupby('bowler')['economy_rate'].mean().reset_index()
worst_bowler = highest_economy.loc[highest_economy['economy_rate'].idxmax()]
worst_bowler
```

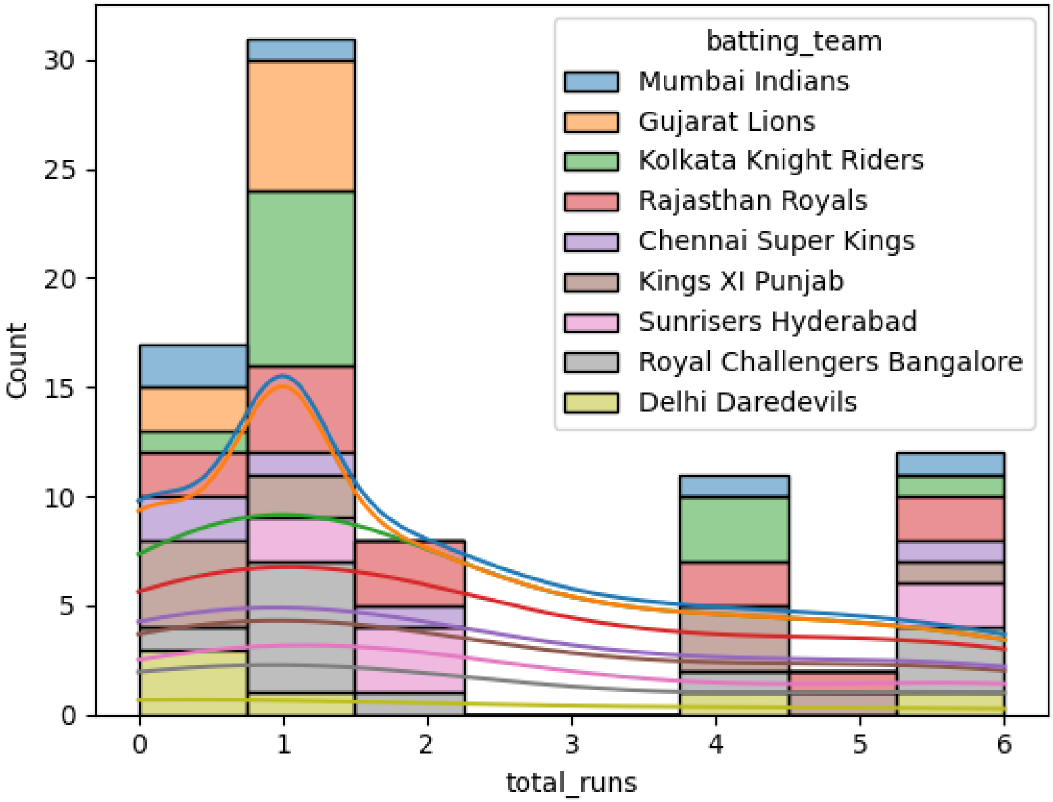
[167]:
```
bowler          SA Yadav
economy_rate    1.333333
Name: 281, dtype: object
```

[173]:
```
# How does the number of dismissals vary across different dismissal kinds
dismissal_counts = a['dismissal_kind'].value_counts().reset_index()
dismissal_counts.columns = ['dismissal_kind', 'player_dismissed']
dismissal_counts
```

[173]:
```
             dismissal_kind  player_dismissed
0  No Information Available            143022
1                    caught              4373
2                    bowled              1382
3                   run out               755
4                       lbw               455
5                   stumped               243
6         caught and bowled               211
7              retired hurt                 9
8                hit wicket                 9
9     obstructing the field                 1
```

[203]:
```
# What is the distribution of runs scored by each batting team in super overs
super_over_data = a[a['is_super_over'] == True]
sns.histplot(data=super_over_data, x='total_runs', hue='batting_team',␣
 ↪multiple='stack',kde=True)
```

[203]:
```
<Axes: xlabel='total_runs', ylabel='Count'>
```


batting_team
Mumbai Indians
Gujarat Lions
Kolkata Knight Riders
Rajasthan Royals
Chennai Super Kings
Kings XI Punjab
Sunrisers Hyderabad
Royal Challengers Bangalore
Delhi Daredevils
Count
total_runs