

```
In [24]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as ple
```

Data Loading

```
In [11]: dfbat=pd.read_csv("df_batting.csv")
```

```
In [12]: dfbowl=pd.read_csv("df_bowling.csv")
```

```
In [10]: dfmatch=pd.read_csv("df_match.csv")
dfplayers=pd.read_csv("df_players.csv")
```

Data OverView

```
In [14]: dfplayers.head()
```

	name	team	image	battingStyle	bowlingStyle	playingRole	description
0	Najmul Hossain Shanto	Bangladesh	NaN	Left hand Bat	Right arm Offbreak	Top order Batter	Nazmul Hossain Shanto emerged from an unusual ...
1	Soumya Sarkar	Bangladesh	NaN	Left hand Bat	Right arm Medium fast	Middle order Batter	A rarity among Bangladesh allrounders, top-ord...
2	Liton Das	Bangladesh	NaN	Right hand Bat	NaN	Wicketkeeper Batter	Liton Das is the first wicketkeeper-batsman in...
3	Shakib Al Hasan(c)	Bangladesh	NaN	Left hand Bat	Slow Left arm Orthodox	Allrounder	When the annals of Bangladesh cricket are sift...
4	Afif Hossain	Bangladesh	NaN	Left hand Bat	Right arm Offbreak	Allrounder	Bangladesh left-hander Afif Hossain made his T...

```
In [15]: dfmatch.head()
```

	team1	team2	winner	margin	ground	matchDate	match_id
0	Namibia	Sri Lanka	Namibia	55 runs	Geelong	Oct 16, 2022	T20I # 1823
1	Netherlands	U.A.E.	Netherlands	3 wickets	Geelong	Oct 16, 2022	T20I # 1825
2	Scotland	West Indies	Scotland	42 runs	Hobart	Oct 17, 2022	T20I # 1826
3	Ireland	Zimbabwe	Zimbabwe	31 runs	Hobart	Oct 17, 2022	T20I # 1828
4	Namibia	Netherlands	Netherlands	5 wickets	Geelong	Oct 18, 2022	T20I # 1830

In [13]: `dfbat.head()`

	match	teamInnings	battingPos	batsmanName	runs	balls	4s	6s	SR	out/not_out	ma
0	Namibia Vs Sri Lanka	Namibia	1	Michael van Lingen	3	6	0	0	50.00	out	
1	Namibia Vs Sri Lanka	Namibia	2	Divan la Cock	9	9	1	0	100.00	out	
2	Namibia Vs Sri Lanka	Namibia	3	Jan Nicol Loftie-Eaton	20	12	1	2	166.66	out	
3	Namibia Vs Sri Lanka	Namibia	4	Stephan Baard	26	24	2	0	108.33	out	
4	Namibia Vs Sri Lanka	Namibia	5	Gerhard Erasmus(c)	20	24	0	0	83.33	out	

In [16]: `dfbowl.head()`

	match	bowlingTeam	bowlerName	overs	maiden	runs	wickets	economy	0s	4s	6s	wid
0	Namibia Vs Sri Lanka	Sri Lanka	Maheesh Theekshana	4.0	0	23	1	5.75	7	0	0	
1	Namibia Vs Sri Lanka	Sri Lanka	Dushmantha Chameera	4.0	0	39	1	9.75	6	3	1	
2	Namibia Vs Sri Lanka	Sri Lanka	Pramod Madushan	4.0	0	37	2	9.25	6	3	1	
3	Namibia Vs Sri Lanka	Sri Lanka	Chamika Karunaratne	4.0	0	36	1	9.00	7	3	1	
4	Namibia Vs Sri Lanka	Sri Lanka	Wanindu Hasaranga de Silva	4.0	0	27	1	6.75	8	1	1	

Exploratory Data Analysis (EDA)

Let's start with some basic exploratory data analysis to understand the distributions and relationships in the data.

In [17]: `#checking for null values
dfplayers.isna().sum()`

```
Out[17]: name          0
         team          0
         image        188
         battingStyle  0
         bowlingStyle  20
         playingRole   0
         description   70
         dtype: int64
```

```
In [19]: dfplayers.shape
```

```
Out[19]: (219, 7)
```

```
In [20]: dfmatch.isna().sum()
```

```
Out[20]: team1          0
         team2          0
         winner          0
         margin          4
         ground          0
         matchDate       0
         match_id        0
         dtype: int64
```

```
In [21]: dfbat.isna().sum()
```

```
Out[21]: match          0
         teamInnings     0
         battingPos      0
         batsmanName      0
         runs            0
         balls           0
         4s              0
         6s              0
         SR              0
         out/not_out      0
         match_id         0
         dtype: int64
```

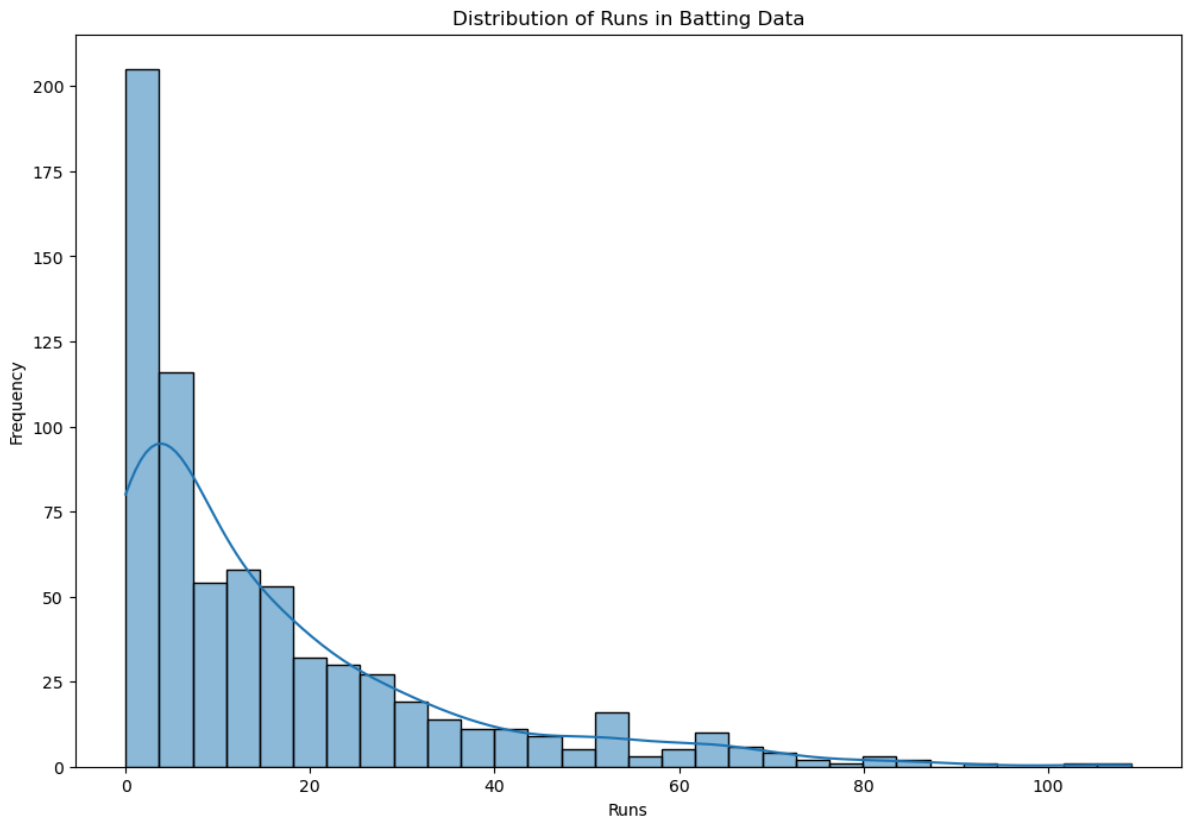
```
In [22]: dfbowl.isna().sum()
```

```
Out[22]: match          0
         bowlingTeam     0
         bowlerName      0
         overs           0
         maiden          0
         runs            0
         wickets         0
         economy         0
         0s              0
         4s              0
         6s              0
         wides           0
         noBalls         0
         match_id        0
         dtype: int64
```

Distribution of Runs in Batting Data

```
In [26]: plt.figure(figsize=(12,8))
         sns.histplot(dfbat['runs'],bins=30, kde=True)
         plt.title('Distribution of Runs in Batting Data')
```

```
plt.xlabel('Runs')
plt.ylabel('Frequency')
plt.show()
```



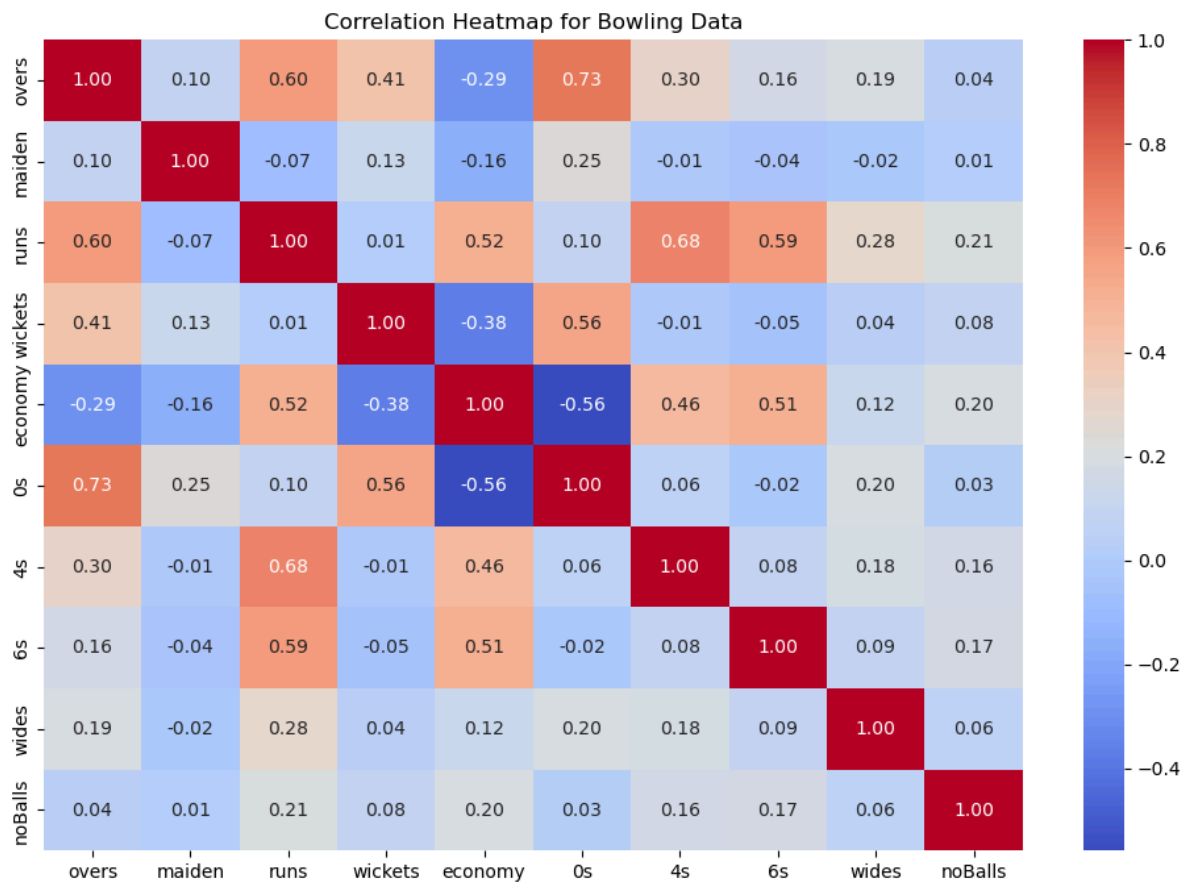
```
In [47]: top_bowlers = dfbowl.groupby('bowlerName')['wickets'].sum().sort_values(ascending=False)
top_bowlers
```

```
Out[47]: bowlerName
Wanindu Hasaranga de Silva    15
Bas de Leede                  13
Sam Curran                    13
Blessing Muzarabani           12
Paul van Meekeren              11
Josh Little                    11
Shadab Khan                    11
Shaheen Shah Afridi            11
Anrich Nortje                  11
Sikandar Raza                  10
Name: wickets, dtype: int64
```

Correlation Heatmap for Bowling Data

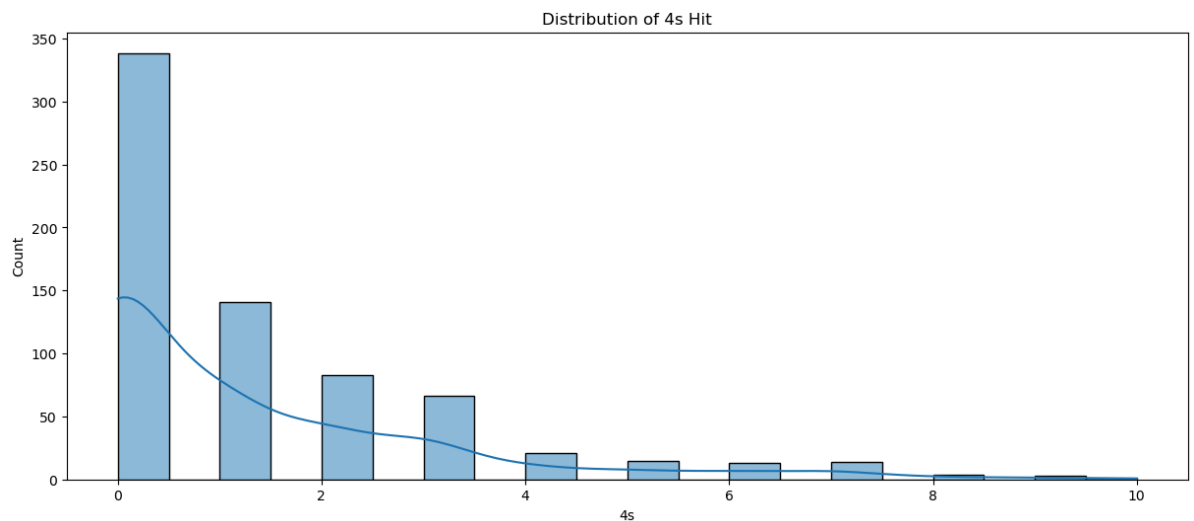
Let's see if there are any interesting correlations in the bowling data.

```
In [29]: numeric_df = dfbowl.select_dtypes(include=[np.number])
plt.figure(figsize=(12, 8))
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm', fmt='.2f')
plt.title('Correlation Heatmap for Bowling Data')
plt.show()
```



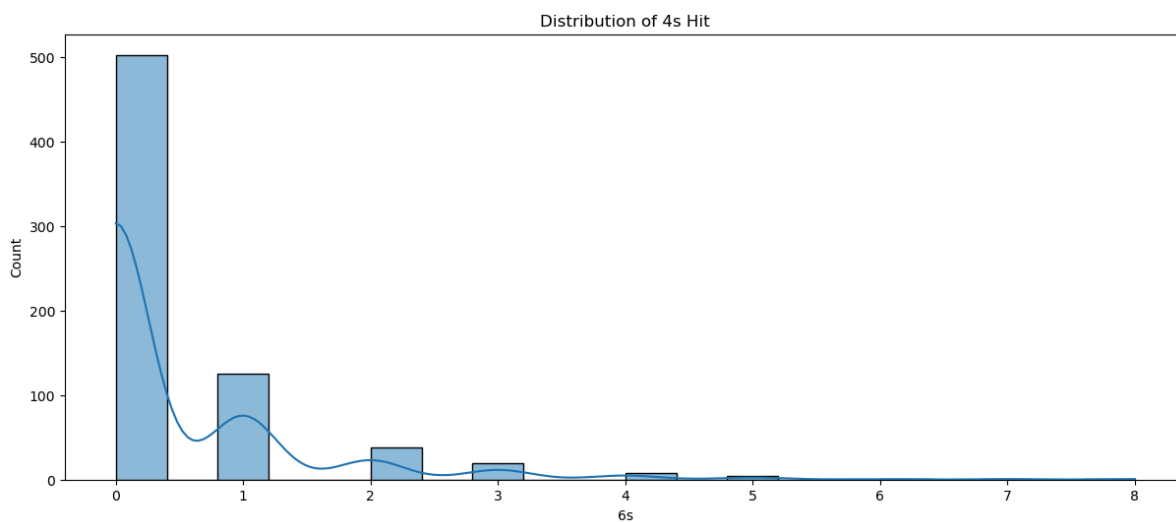
```
In [56]: plt.figure(figsize=(50, 20))
plt.subplot(3, 3, 4)
sns.histplot(dfbat['4s'], kde=True, bins=20)
plt.title('Distribution of 4s Hit')
```

Out[56]: Text(0.5, 1.0, 'Distribution of 4s Hit')



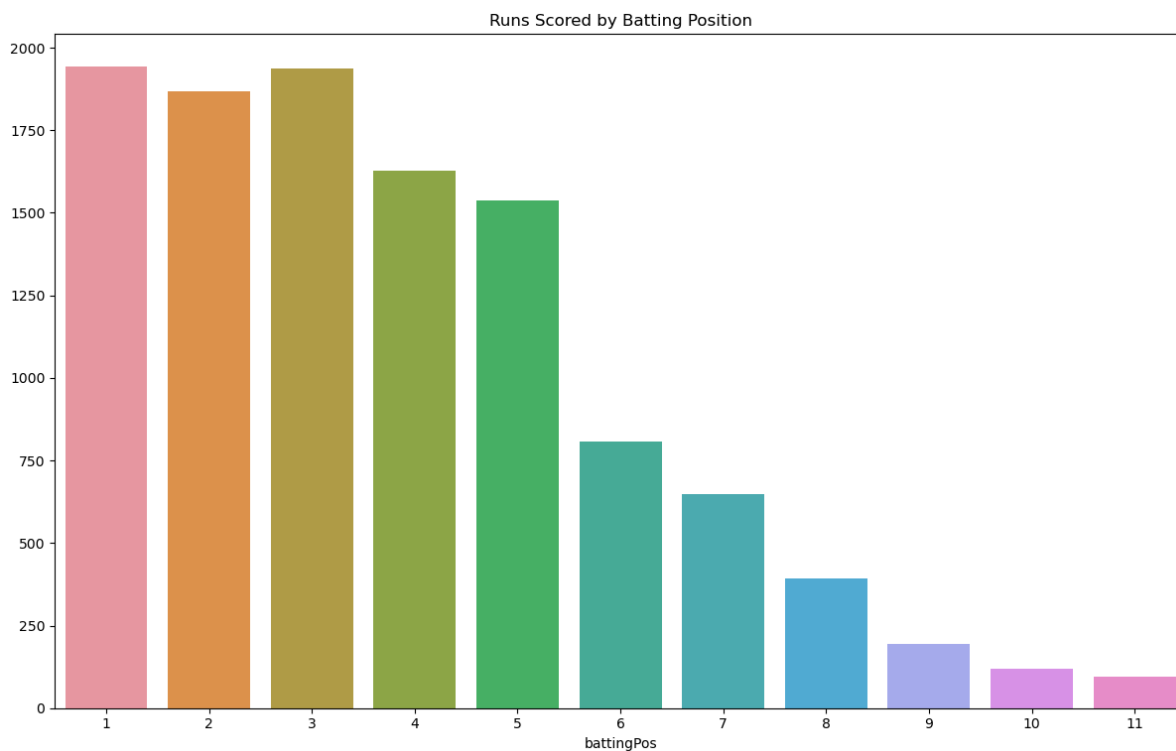
```
In [58]: plt.figure(figsize=(50, 20))
plt.subplot(3, 3, 4)
sns.histplot(dfbat['6s'], kde=True, bins=20)
plt.title('Distribution of 4s Hit')
```

Out[58]: Text(0.5, 1.0, 'Distribution of 4s Hit')



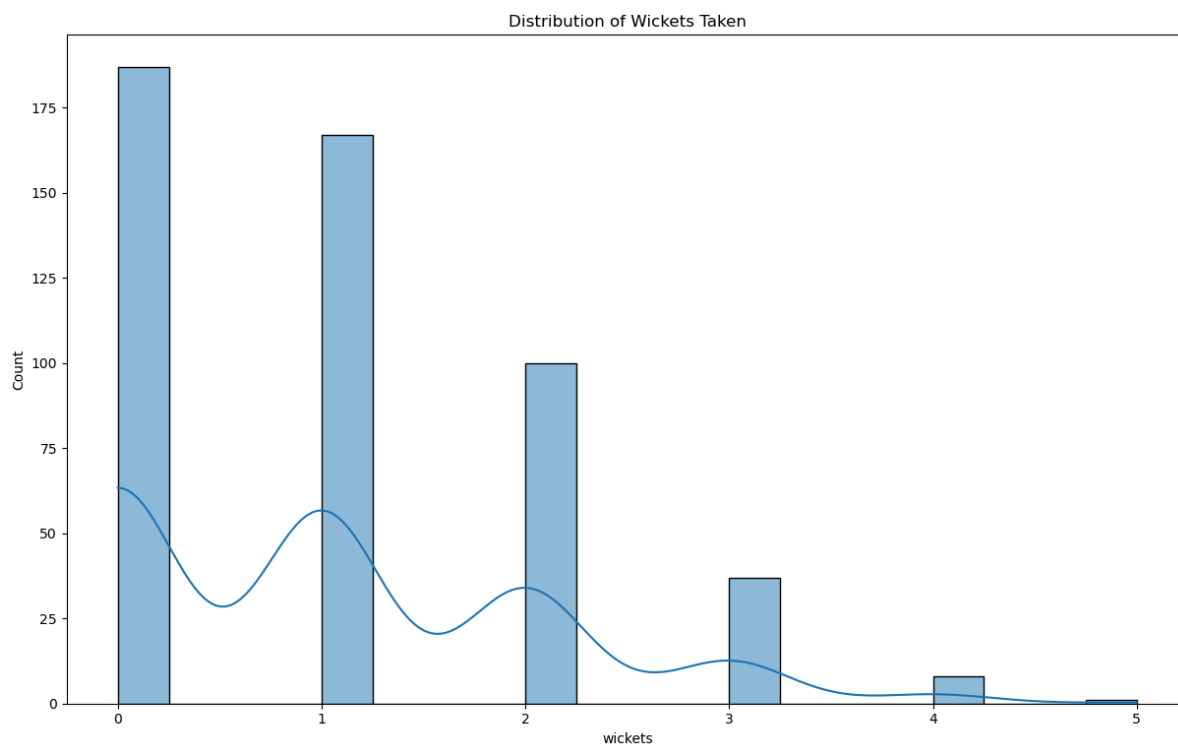
```
In [60]: plt.figure(figsize=(50, 30))
batting_pos_runs = dfbat.groupby('battingPos')['runs'].sum().sort_values(ascending=
plt.subplot(3, 3, 6)
sns.barplot(x=batting_pos_runs.index, y=batting_pos_runs.values)
plt.title('Runs Scored by Batting Position')
```

```
Out[60]: Text(0.5, 1.0, 'Runs Scored by Batting Position')
```



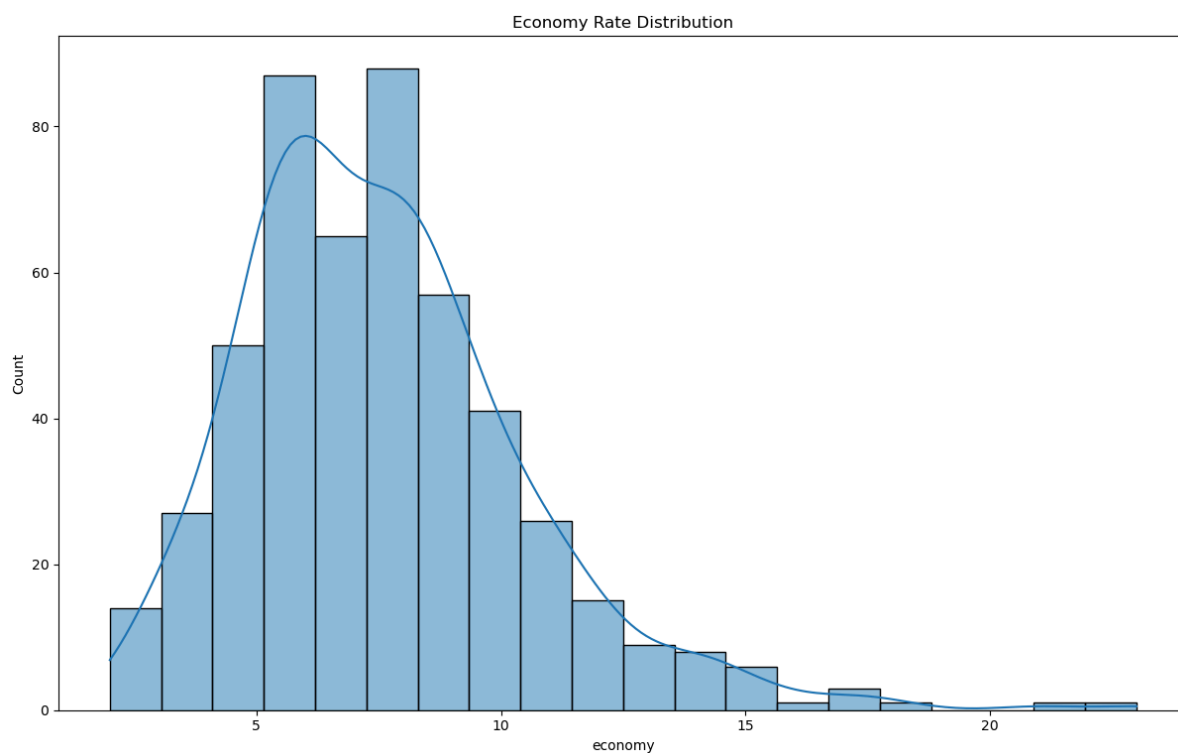
```
In [62]: plt.figure(figsize=(50, 30))
plt.subplot(3, 3, 7)
sns.histplot(df bowl['wickets'], kde=True, bins=20)
plt.title('Distribution of Wickets Taken')
```

```
Out[62]: Text(0.5, 1.0, 'Distribution of Wickets Taken')
```

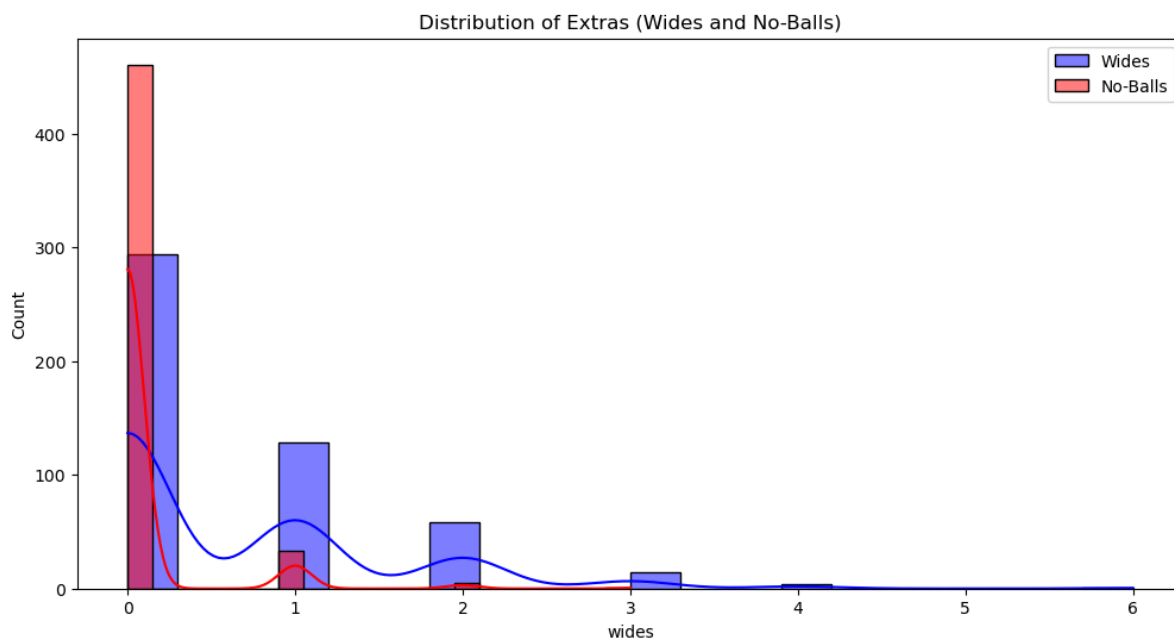


```
In [63]: plt.figure(figsize=(50, 30))
plt.subplot(3, 3, 8)
sns.histplot(dfbow1['economy'], kde=True, bins=20)
plt.title('Economy Rate Distribution')
```

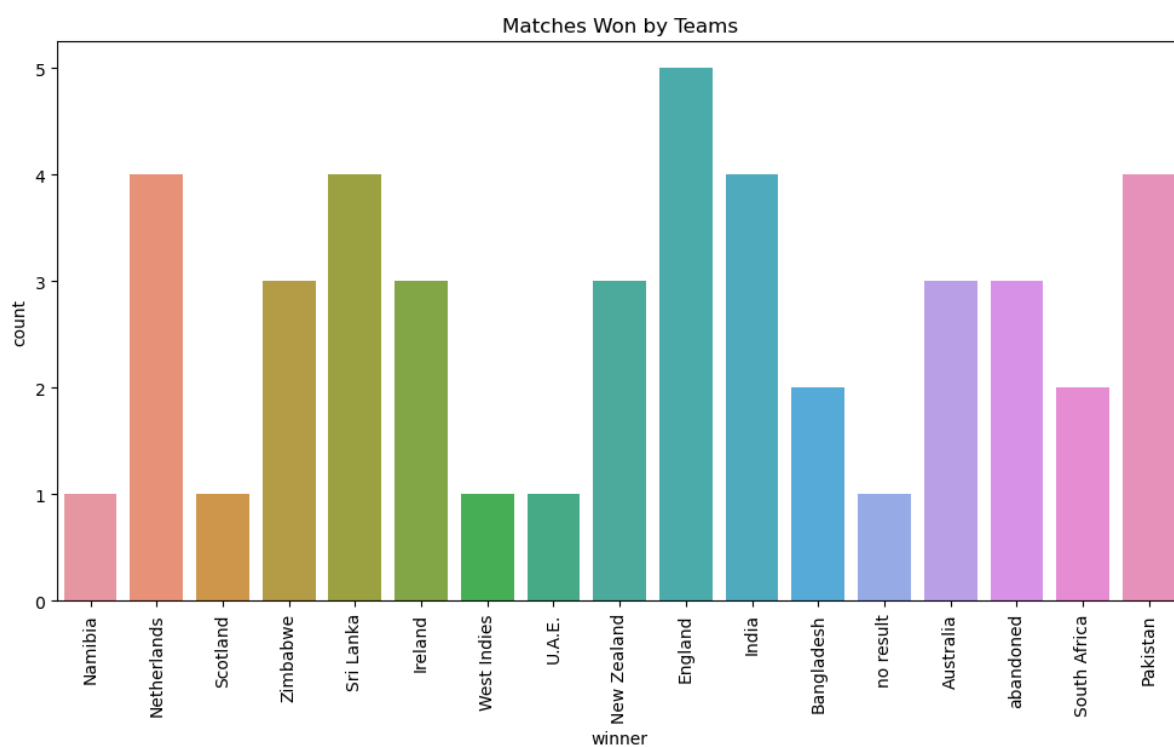
```
Out[63]: Text(0.5, 1.0, 'Economy Rate Distribution')
```



```
In [64]: plt.figure(figsize=(12, 6))
sns.histplot(dfbow1['wides'], kde=True, color='blue', label='Wides', bins=20)
sns.histplot(dfbow1['noBalls'], kde=True, color='red', label='No-Balls', bins=20)
plt.title('Distribution of Extras (Wides and No-Balls)')
plt.legend()
plt.show()
```

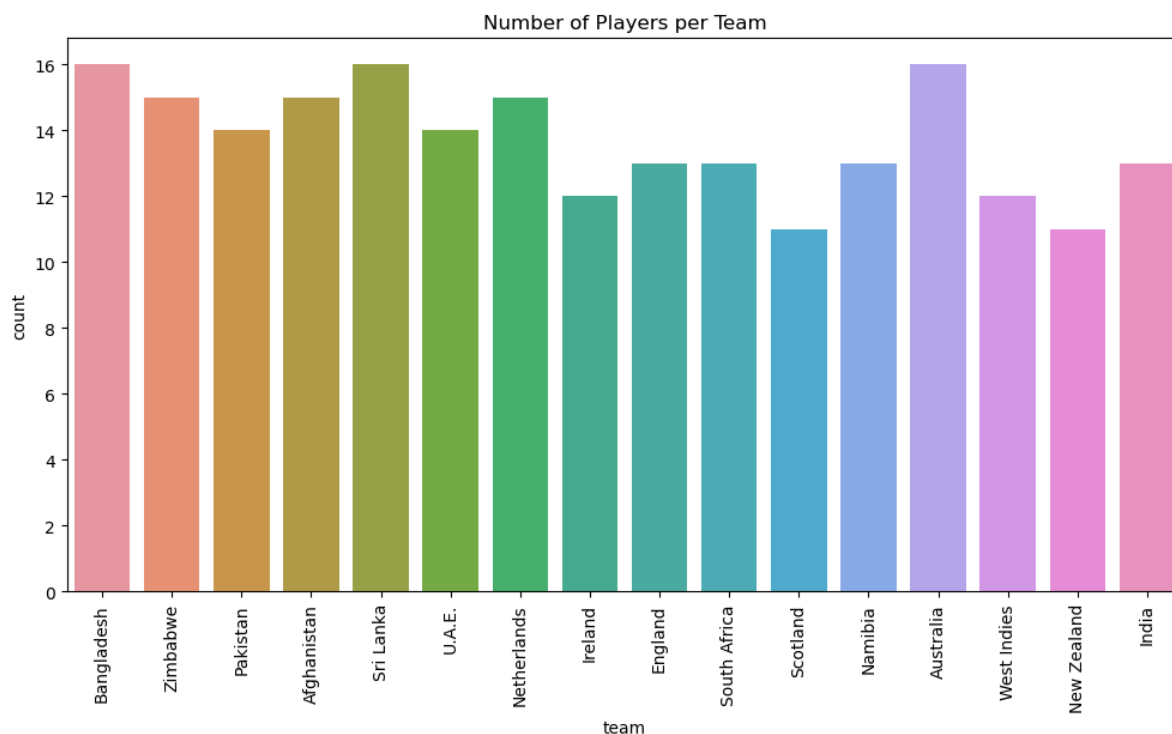


```
In [65]: plt.figure(figsize=(12, 6))
sns.countplot(data=dfmatch, x='winner')
plt.title('Matches Won by Teams')
plt.xticks(rotation=90)
plt.show()
```

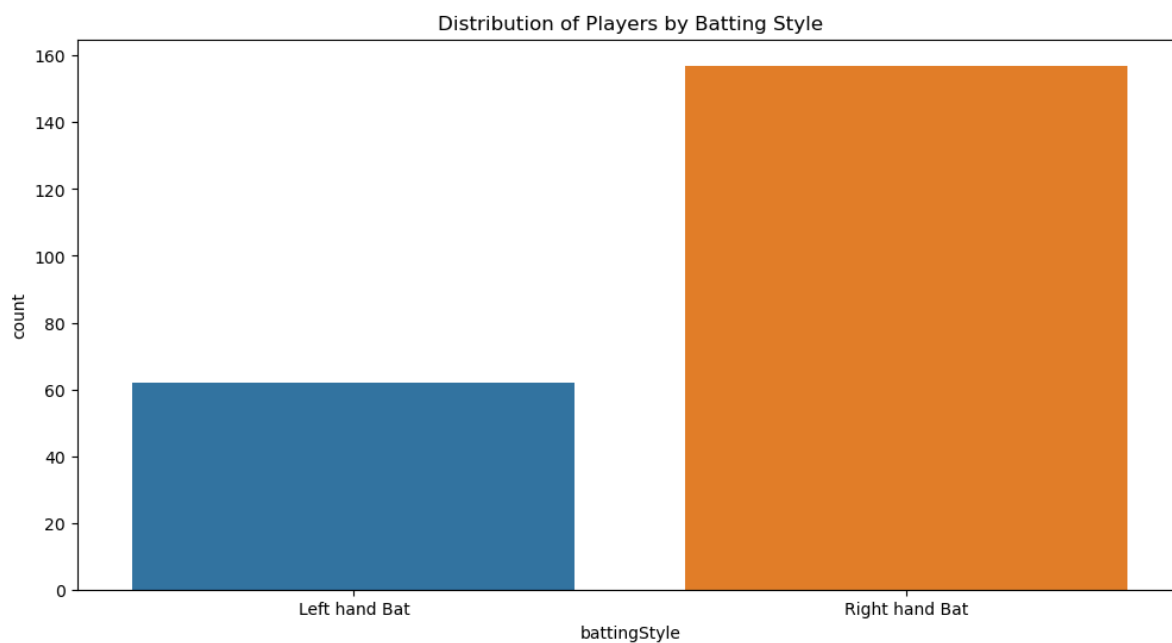


Player Analysis

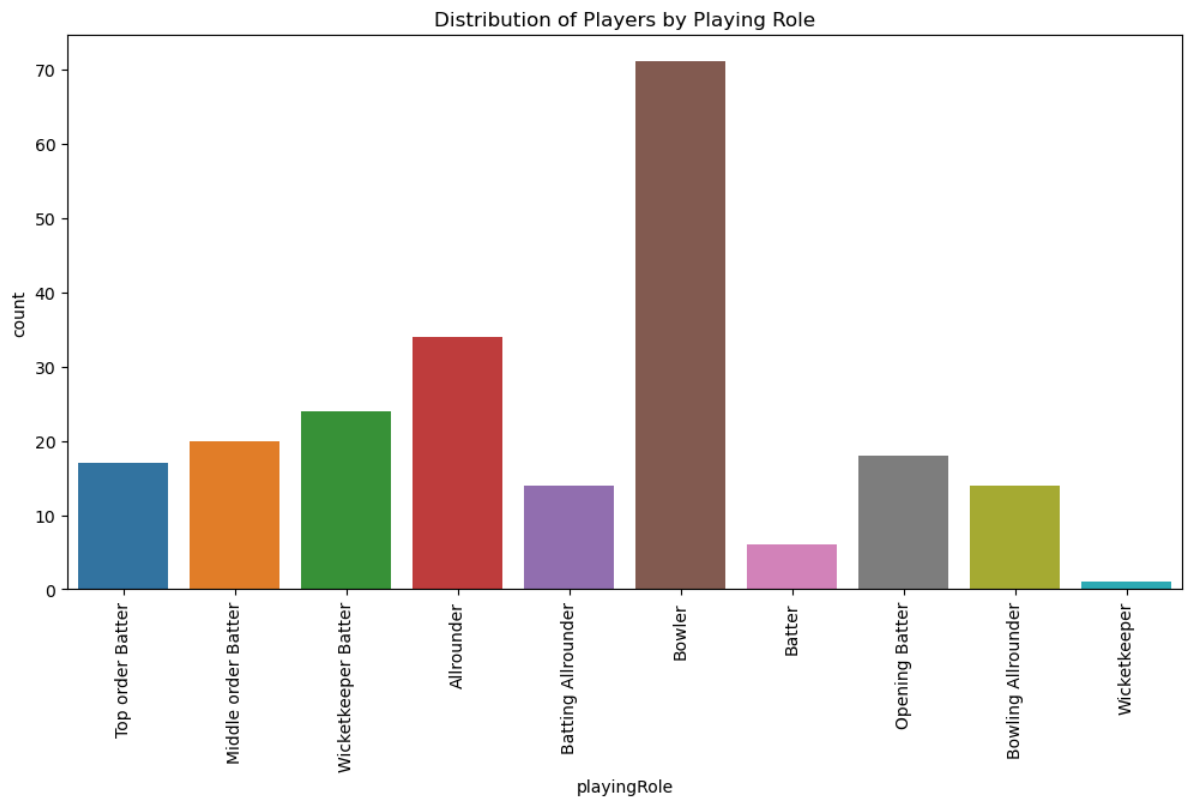
```
In [66]: plt.figure(figsize=(12, 6))
sns.countplot(data=dfplayers, x='team')
plt.title('Number of Players per Team')
plt.xticks(rotation=90)
plt.show()
```

```
In [67]: plt.figure(figsize=(12, 6))
sns.countplot(data=dfplayers, x='battingStyle')
plt.title('Distribution of Players by Batting Style')
plt.show()
```

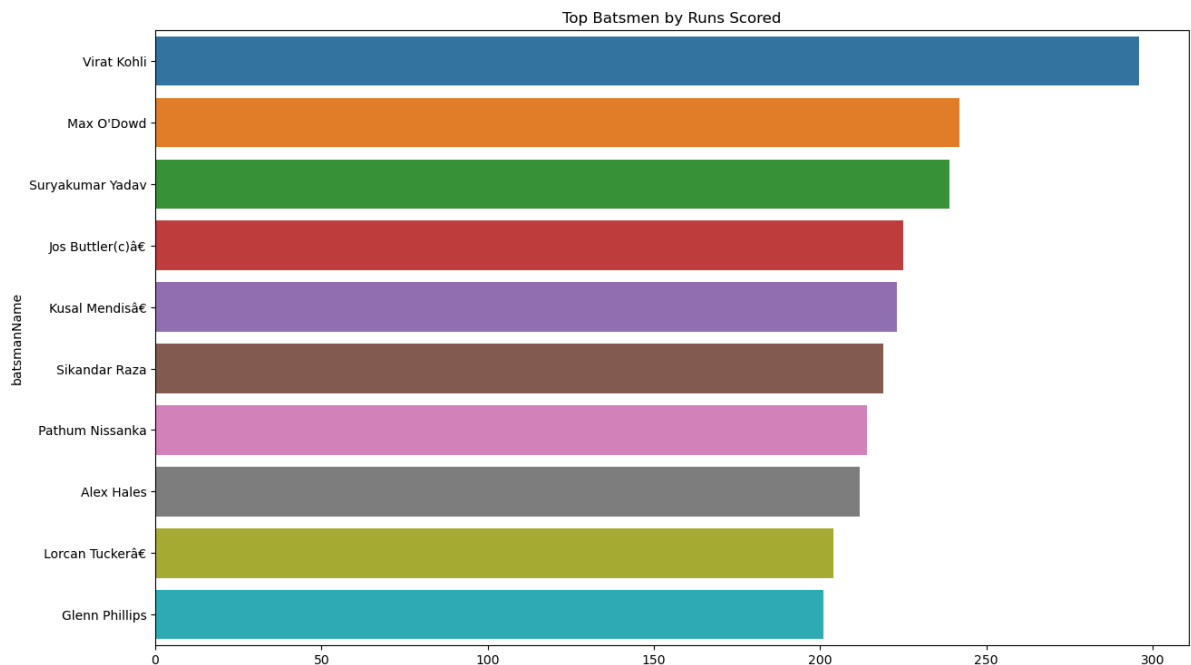


```
In [68]: plt.figure(figsize=(12, 6))
sns.countplot(data=dfplayers, x='playingRole')
plt.title('Distribution of Players by Playing Role')
plt.xticks(rotation=90)
plt.show()
```

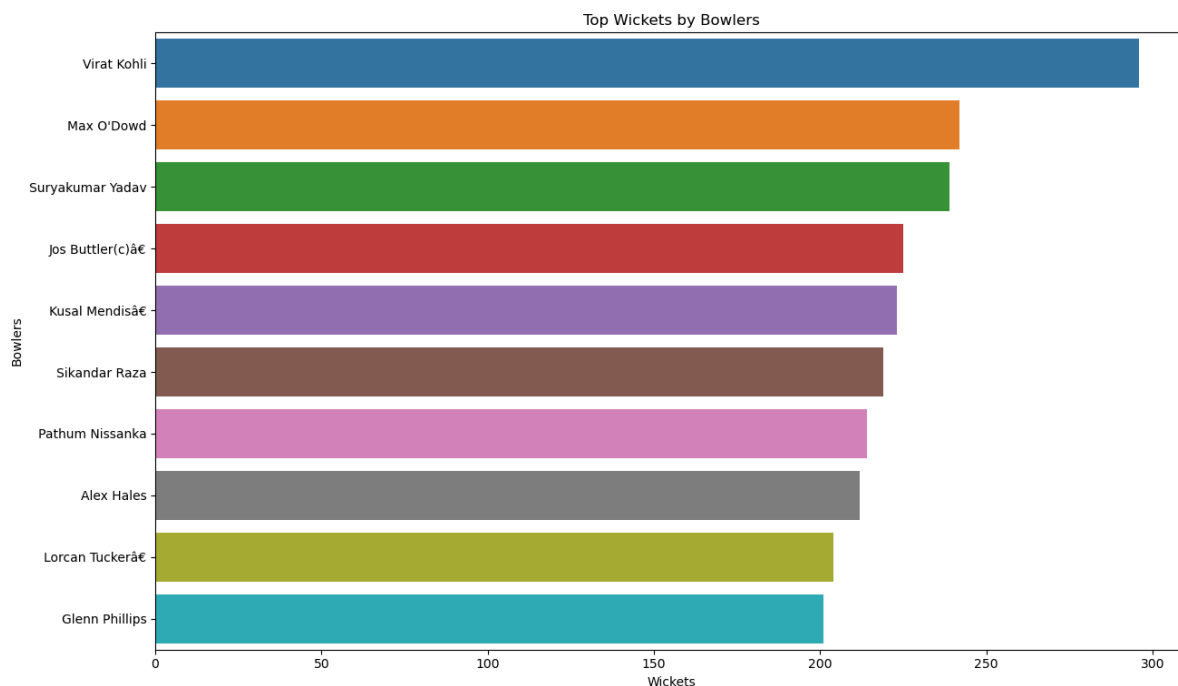


```
In [69]: plt.figure(figsize=(50, 30))
top_batsmen = dfbat.groupby('batsmanName')['runs'].sum().sort_values(ascending=False)
plt.subplot(3, 3, 3)
sns.barplot(x=top_batsmen.values, y=top_batsmen.index)
plt.title('Top Batsmen by Runs Scored')
```

Out[69]: Text(0.5, 1.0, 'Top Batsmen by Runs Scored')

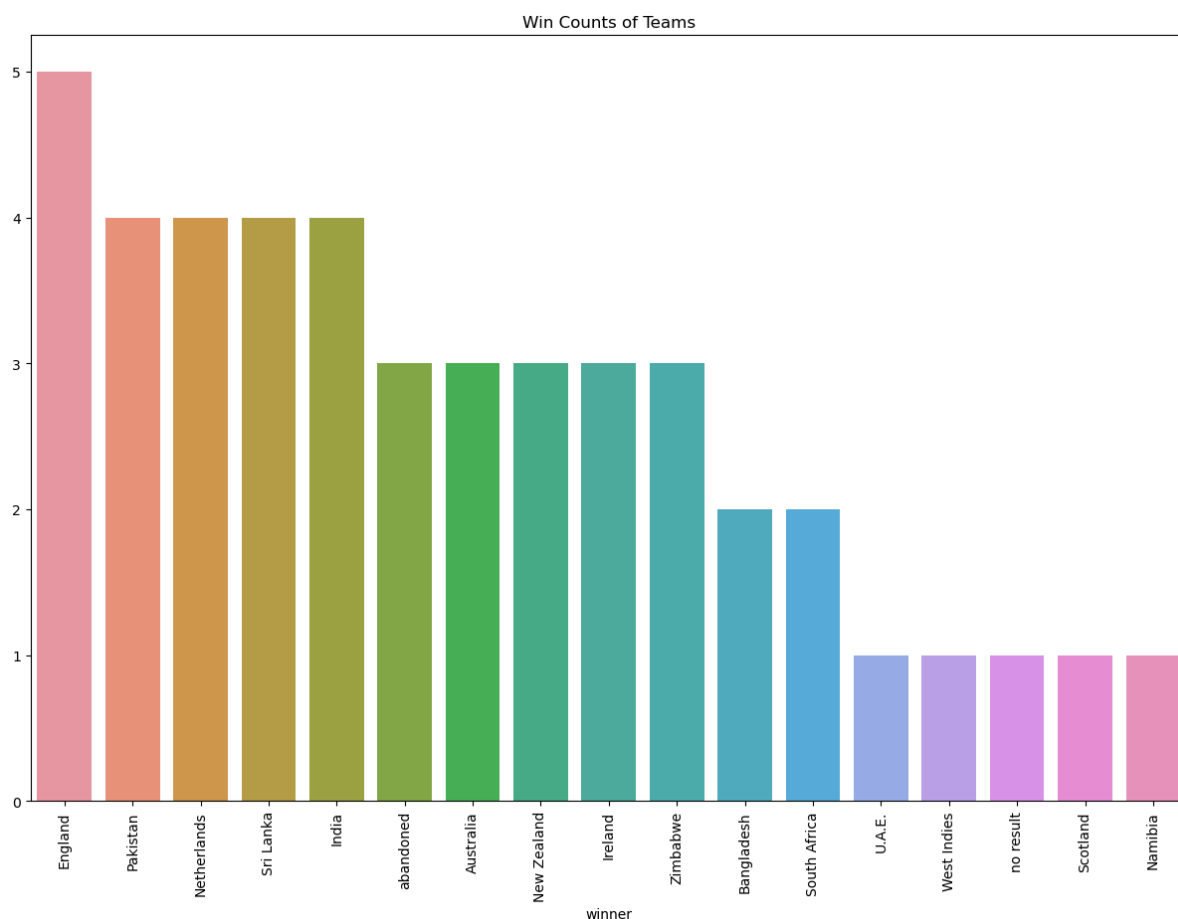


```
In [70]: plt.figure(figsize=(50, 30))
top_bowlers = dfbowl.groupby('bowlerName')['wickets'].sum().sort_values(ascending=False)
plt.subplot(3, 3, 3)
sns.barplot(x=top_bowlers.values, y=top_bowlers.index)
plt.title('Top Wickets by Bowlers')
plt.xlabel('Wickets')
plt.ylabel('Bowlers')
plt.show()
```



Team Performance Analysis

```
In [73]: win_loss_counts = dfmatch['winner'].value_counts()
plt.figure(figsize=(15, 10))
sns.barplot(x=win_loss_counts.index, y=win_loss_counts.values)
plt.title('Win Counts of Teams')
plt.xticks(rotation=90)
plt.show()
```



In []: