

Exploratory Data Analysis on IPL Dataset



Submitted by

Shoumik Das [Regn.- 21213120026]

3rd year, student of B.Sc Mathematics


Sandipan Mondal [Regn.- 21213170008]

3rd year, student of B.Sc Mathematics

Debasmita Saha [Regn.- 21201220045]

3rd year, student of B.Sc Economics

Under the supervision of

 31/07/2024

Prof. Avishek Adhikari

Professor and Head of the Department, Mathematics
Presidency University, Kolkata

Abstract

This project highlights the use of **Exploratory Data Analysis (EDA)** on the **IPL (Indian Premier League)** dataset from 2020 - 2024. The aim was to clean, analyze, and draw some insights from the data, to identify the top performers and to select the best 15 players under different categories for team selection in the T20 format. The project revolves around various statistical measures and visualizations to summarize the trends and patterns in batting and bowling records. By conducting this analysis, we aimed to enhance our understanding of the IPL data, extract some meaningful insights, and provide further recommendations for player selection based on their performances.

Acknowledgement

We would like to express our deepest gratitude to **Prof. Avishek Adhikari**, Head of the Department of Mathematics, Presidency University, Kolkata, for his invaluable guidance, encouragement, and support throughout the duration of this project. His insightful feedback and suggestions greatly contributed to the success of this project.

We are immensely thankful to **Presidency University** for providing us with the necessary resources and a conducive environment to work on this project.

We would also like to extend our sincere thanks to all the faculty members of the Department of Mathematics for their constant support and guidance.

Lastly, we would like to acknowledge the support and encouragement from our fellow students, friends, and family members, without whom the completion of this project would not have been possible.

Sandipan Mondal, regn- 21213170008
Shoumik Das, regn- 21213120026
Debasmita Saha, regn- 21201220045
Presidency University, Kolkata

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 4 |
| 1.1 | About the Data | 4 |
| 1.2 | Objectives | 4 |
| 1.3 | Data Description | 4 |
| 1.3.1 | Dataset Overview | 4 |
| 1.4 | Data Cleaning | 5 |
| 1.5 | Statistical Data Analysis | 7 |
| 1.5.1 | Univariate Analysis | 7 |
| 1.5.2 | Bivariate Analysis | 13 |
| 1.5.3 | Correlation Coefficients between every data columns: | 15 |
| 1.6 | Team Composition Conditions | 18 |
| 1.6.1 | Roles | 18 |
| 2 | Insights, Findings and Selections | 19 |
| 2.0.1 | Analysis of Batting data : | 19 |
| 2.0.2 | Analysis of Wicket-Keepers data : | 20 |
| 2.0.3 | Analysis of All-Rounders data : | 21 |
| 2.0.4 | Analysis of Bowlers data : | 22 |
| 2.0.5 | Findings : | 23 |
| 3 | Final Selection | 27 |
| 3.1 | Selection based on user input | 27 |
| 3.1.1 | Introduction | 27 |
| 3.1.2 | Code for Selections | 28 |
| 3.2 | Conclusion | 33 |

Chapter 1

Introduction

1.1 About the Data

The **Indian Premier League (IPL)** is one of the most popular T20 cricket leagues in the world, attracting the best cricket players globally. The dataset consists of players performance under batting and bowling sections between the years 2008-2024.

This project seeks to explore the dataset and derive valuable insights into player performances using **Exploratory Data Analysis (EDA)** to select the best players for team selection. This includes analyzing batting, bowling etc. statistics, with a focus on selecting the best players who meet specific criteria for different roles in the team.

1.2 Objectives

This project has the following objectives:

- To clean and prepare the **IPL** dataset for analysis.
- To apply statistical techniques for analyzing each player's performances.
- To explore the relationships between various metrics.
- To select the best players based on predefined criteria for different roles (**Batsman, Wicket-keeper, All-rounder, and Bowler**).

1.3 Data Description

1.3.1 Dataset Overview

The dataset includes statistics of **IPL** players from **2008 to 2024**. The dataset contains **25 columns** representing various player statistics such as:

- Player Name
- Matches Played (both batted and bowled)
- Runs Scored, Batting Average, Balls Faced, Strike Rate
- Fours, Sixes, Centuries, Half Centuries

- Catches Taken, Stumpings
- Balls Bowled, Runs Conceded, Wickets Taken, Economy Rate, Bowling Average
- Four Wicket Hauls, Five Wicket Hauls

[illegible]

Figure : Overview of the data set

1.4 Data Cleaning

The data cleaning process involved:

- Filtering the data from year **2020 to 2024**

```
1 Data_2024 = df.loc[df["Year"]=="2024"]
2 Data_2023 = df.loc[df["Year"]=="2023"]
3 Data_2022 = df.loc[df["Year"]=="2022"]
4 Data_2021 = df.loc[df["Year"]=="2021"]
5 Data_2020 = df.loc[df["Year"]=="2020"]
6
```

- Removing unnecessary columns such as 'Year' and handling duplicates.

```
1 combined_df = combined_df.drop(["Not_Outs", "Year", '
Best_Bowling_Match', 'Four_Wicket_Hauls'], axis = 1)
2
```

- Converting **numeric columns** that were originally in **string format** into **numerical format** for analysis.

```
1 for col in numeric_columns:
2     Data_2024[col] = pd.to_numeric(Data_2024[col], errors='coerce')
3     Data_2023[col] = pd.to_numeric(Data_2023[col], errors='coerce')
4     Data_2022[col] = pd.to_numeric(Data_2022[col], errors='coerce')
5     Data_2021[col] = pd.to_numeric(Data_2021[col], errors='coerce')
6     Data_2020[col] = pd.to_numeric(Data_2020[col], errors='coerce')
```

- **Handling missing values by imputing** with median values or **removing** rows where critical data was missing.

```

1 for df in dataframes:
2     for col in columns_to_sum + columns_to_max:
3         df[col] = pd.to_numeric(df[col], errors='coerce')
4
5 # Merge DataFrames and compute required statistics
6 for df in dataframes[1:]:
7     combined_df = pd.merge(combined_df, df, on='Player_Name', how='outer',
8                             suffixes=('', '_y'))
9
10    # Sum specified columns across all years
11    for col in columns_to_sum:
12        combined_df[col] = combined_df[col].fillna(0) + combined_df[f'{col}_y'].fillna(0)
13
14    # Find the maximum for Highest_Score
15    for col in columns_to_max:
16        combined_df[col] = combined_df[[col, f'{col}_y']].max(axis=1)
17
18    # Update the Year column to the latest year
19    combined_df['Year'] = combined_df[['Year', f'Year_y']].max(axis=1)
20
21    # Drop unnecessary columns after merging
22    combined_df.drop(columns=[f'{col}_y' for col in columns_to_sum +
23                             columns_to_max] + [f'Year_y'], inplace=True)
24
25 # Calculate Batting_Average as total Runs_Scored / total Matches_Batted
26 combined_df['Batting_Average'] = combined_df['Runs_Scored'] /
27     combined_df['Matches_Batted']
28
29 # Calculate Bowling_Average as total Runs_Conceded / total Wickets_Taken
30 combined_df['Bowling_Average'] = combined_df['Runs_Conceded'] /
31     combined_df['Wickets_Taken']
32
33 # Calculate Economy_Rate as total Runs_Conceded / (Balls_Bowled / 6)
34 combined_df['Economy_Rate'] = combined_df['Runs_Conceded'] / (
35     combined_df['Balls_Bowled'] / 6)
36
37 # Calculate Batting_Strike_Rate as (Runs_Scored / Balls_Faced) * 100
38 combined_df['Batting_Strike_Rate'] = (combined_df['Runs_Scored'] /
39     combined_df['Balls_Faced']) * 100
40
41 combined_df['Bowling_Strike_Rate'] = combined_df['Balls_Bowled'] /
42     combined_df['Wickets_Taken']
43
44 combined_df['Batting_Average'].fillna(0, inplace=True)
45 combined_df['Bowling_Average'].fillna(0, inplace=True)
46 combined_df['Economy_Rate'].fillna(0, inplace=True)
47 combined_df['Batting_Strike_Rate'].fillna(0, inplace=True)
48 combined_df['Bowling_Strike_Rate'].fillna(0, inplace=True)
49
50 # Original Columns to Keep
51 final_columns = [
52     'Player_Name', 'Matches_Batted', 'Not_Outs', 'Runs_Scored', '
53     Highest_Score',
54     'Batting_Average', 'Balls_Faced', 'Batting_Strike_Rate', 'Centuries'
55 ]

```

```

47     'Half_Centuries', 'Fours', 'Sixes', 'Catches_Taken', 'Stumpings',
48     'Matches_Bowled', 'Balls_Bowled', 'Runs_Conceded', 'Wickets_Taken',
49     'Best_Bowling_Match', 'Bowling_Average', 'Economy_Rate',
50     'Bowling_Strike_Rate', 'Four_Wicket_Hauls', 'Five_Wicket_Hauls'
51 ]
52
53 combined_df = combined_df[final_columns]
54

```

- **Creating** derived features such as '**Balls per Boundary**' for evaluating strike power of the batsman.

```

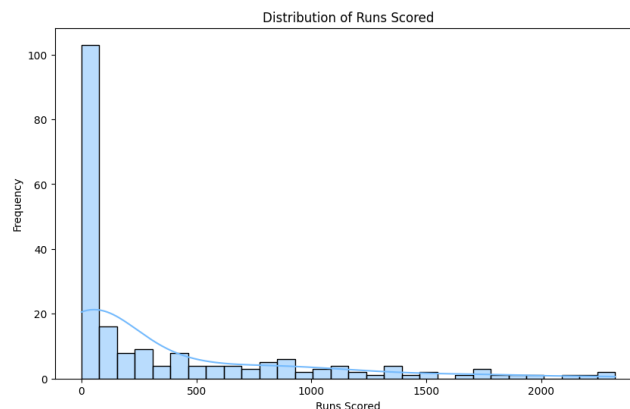
1 Boundary_Per_Ball = combined_df["Balls_Faced"]/(combined_df["Fours"] +
2   combined_df["Sixes"])

```

1.5 Statistical Data Analysis

1.5.1 Univariate Analysis

Univariate analysis focuses on understanding the distribution of individual features in the dataset. Some key observations include:



Analysis of the Distribution of Runs Scored

The histogram depicts the distribution of runs scored by players in the dataset.

Observations:

- The distribution appears to be right-skewed (positively skewed), indicating that most players scored a relatively smaller number of runs, while a few players scored significantly more runs.
- There is a peak in the histogram, showing the majority of players' run scores fall within a certain range.
- The long tail to the right suggests that there might be some exceptional players who have performed exceptionally well in terms of run-scoring.

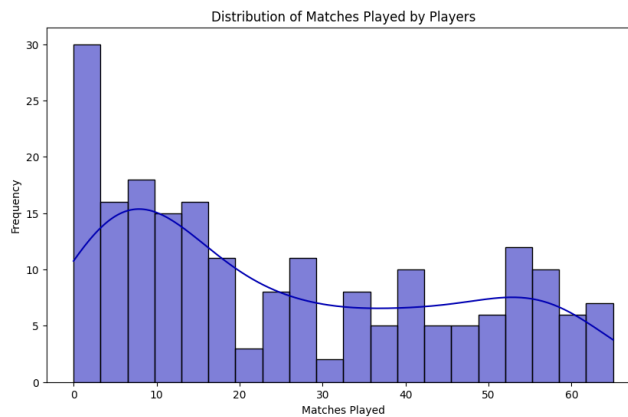
Descriptive Statistics:

| Statistic | Value |
|--------------------|-------------|
| Count | 204.000000 |
| Mean | 369.838235 |
| Standard Deviation | 543.002779 |
| Minimum | 0.000000 |
| 25% | 13.000000 |
| 50% (Median) | 75.000000 |
| 75% | 558.750000 |
| Maximum | 2322.000000 |

Insights from Descriptive Statistics:

- **Mean:** The average number of runs scored by a player is 369.84.
- **Median:** The middle value of the distribution is 75.0, which is generally less than the mean, confirming the right-skewness.
- **Standard Deviation:** The standard deviation of 543.0 indicates the spread of runs scored around the mean, showcasing the variability in player performance.
- **Minimum & Maximum:** The minimum and maximum values highlight the range of runs scored, with the maximum value likely representing the best-performing batsmen in the dataset.
- **Quantiles:** The 25th, 50th, and 75th percentiles (Q1, Q2, and Q3) reveal the distribution of run scores across different player performance levels. For instance, 75% of players scored below 558.75 runs.

The analysis of the distribution of runs scored demonstrates a right-skewed distribution, with the majority of players scoring a moderate number of runs. A few exceptional players likely influence the long tail on the right, and this highlights the presence of a high level of variability in batting performance across players.



Descriptive Statistics of Matches Batted

The descriptive statistics for the number of matches batted are as follows:

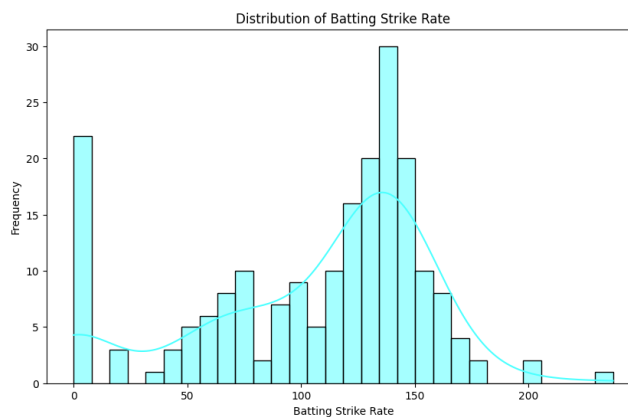
| Statistic | Value |
|--------------------|------------|
| Count | 204.000000 |
| Mean | 25.004902 |
| Standard Deviation | 20.162516 |
| Minimum | 0.000000 |
| 25% | 7.000000 |
| 50% (Median) | 19.000000 |
| 75% | 42.250000 |
| Maximum | 65.000000 |

Analysis and Insights of Matches Batted Distribution

The distribution of matches batted appears to be right-skewed, indicating that the majority of players have played a relatively smaller number of matches.

- The mean (average) number of matches batted is approximately 25.0, while the median is 19.0. This further supports the right-skewed nature of the distribution.
- The standard deviation of 20.16 indicates the extent of the spread in the number of matches played.
- The minimum number of matches batted is 0.0, and the maximum is 65.0.
- The 25th percentile (Q1) shows that 25% of players have batted in 7.0 matches or fewer.
- The 75th percentile (Q3) reveals that 75% of players have batted in 42.25 matches or fewer.

These statistics together provide insights into the playing experience and participation levels of cricket players in the dataset.



Descriptive Statistics of Batting Strike Rate

The descriptive statistics for the Batting Strike Rate are as follows:

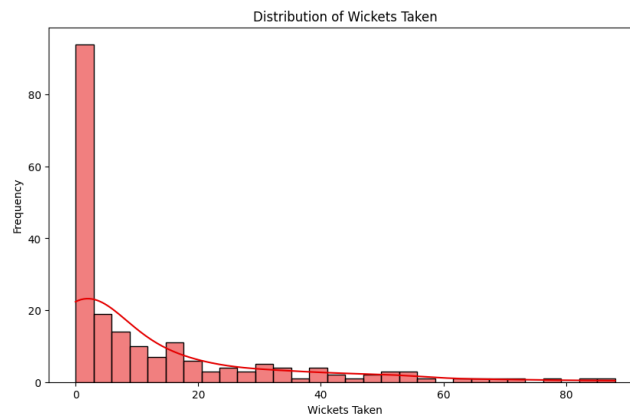
| Statistic | Value |
|--------------------|------------|
| Count | 204.000000 |
| Mean | 106.168598 |
| Standard Deviation | 50.944337 |
| Minimum | 0.000000 |
| 25% | 75.457317 |
| 50% (Median) | 123.259173 |
| 75% | 140.389740 |
| Maximum | 237.142857 |

Analysis and Insights of Batting Strike Rate Distribution

The distribution of Batting Strike Rate appears to be right-skewed, indicating that the majority of players have a relatively lower Batting Strike Rate.

- The mean (average) Batting Strike Rate is approximately 106.17, while the median is 123.26. This further supports the right-skewed nature of the distribution.
- The standard deviation of 50.94 indicates the extent of the spread in the Batting Strike Rate.
- The minimum Batting Strike Rate is 0.0, and the maximum is 237.14.
- The 25th percentile (Q1) shows that 25% of players have a Batting Strike Rate of 75.46 or lower.
- The 75th percentile (Q3) reveals that 75% of players have a Batting Strike Rate of 140.39 or lower.

These statistics together provide insights into the aggressive batting styles and ability to score runs quickly of cricket players in the dataset.



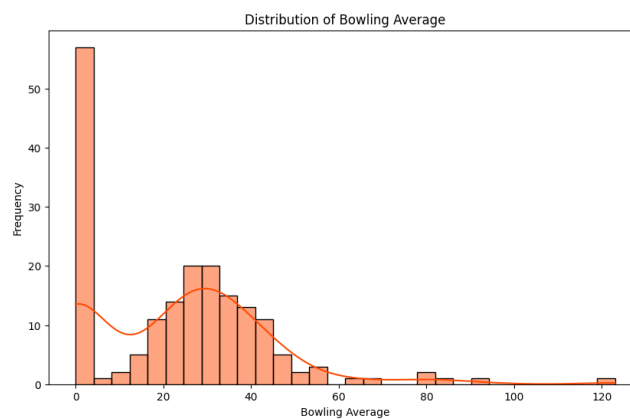
Descriptive Statistics for Wickets Taken

The descriptive statistics for Wickets Taken are as follows:

| Statistic | Value |
|--------------------|------------|
| Count | 204.000000 |
| Mean | 12.362745 |
| Standard Deviation | 18.429623 |
| Minimum | 0.000000 |
| 25% | 0.000000 |
| 50% (Median) | 3.000000 |
| 75% | 16.250000 |
| Maximum | 88.000000 |

Observations

- The average number of wickets taken is higher than the median, indicating a right-skewed distribution.
- The standard deviation of 18.43 indicates the degree of variability in the number of wickets taken by players.
- The histogram shows that most players have taken a relatively small number of wickets, with a long tail towards higher wicket counts, suggesting a right-skewed distribution.
- The maximum number of wickets taken is 88.0, which might be an outlier indicating exceptional performance by a few players.
- The right-skewed distribution suggests that taking a large number of wickets is a relatively uncommon feat, highlighting the importance of excellent bowling skills for achieving such results.
- The standard deviation can be used to identify players who are significantly better or worse than the average in terms of wickets taken.



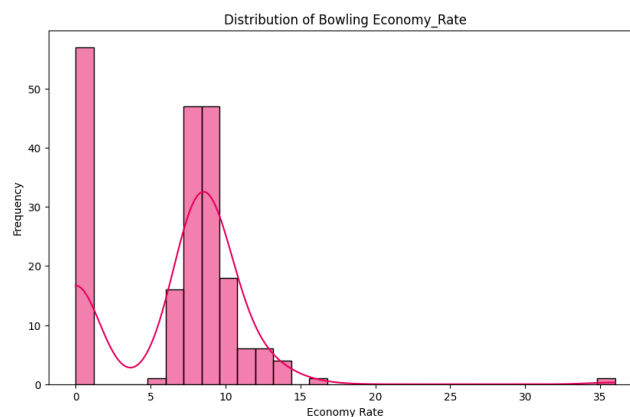
Descriptive Statistics for Bowling Average

The descriptive statistics for Bowling Average are as follows:

| Statistic | Value |
|--------------------|-----------|
| Count | 204.00000 |
| Mean | ∞ |
| Standard Deviation | NaN |
| Minimum | 0.00000 |
| 25% | 0.00000 |
| 50% (Median) | 27.32672 |
| 75% | 38.81250 |
| Maximum | ∞ |

Observations

- The average bowling average is higher than the median, indicating a right-skewed distribution.
- The standard deviation of NaN indicates the degree of variability in the bowling average of players.
- The histogram shows that most players have a relatively high bowling average, with a long tail towards lower bowling averages, suggesting a right-skewed distribution.
- The minimum bowling average is 0.0, which might be an outlier indicating exceptional bowling performance by a few players.
- The right-skewed distribution suggests that achieving a very low bowling average is a relatively rare feat, highlighting the importance of excellent bowling skills for achieving such results.
- The standard deviation can be used to identify players who are significantly better or worse than the average in terms of bowling average.



Descriptive Statistics for Bowling Economy Rate

The descriptive statistics for Bowling Economy Rate are as follows:

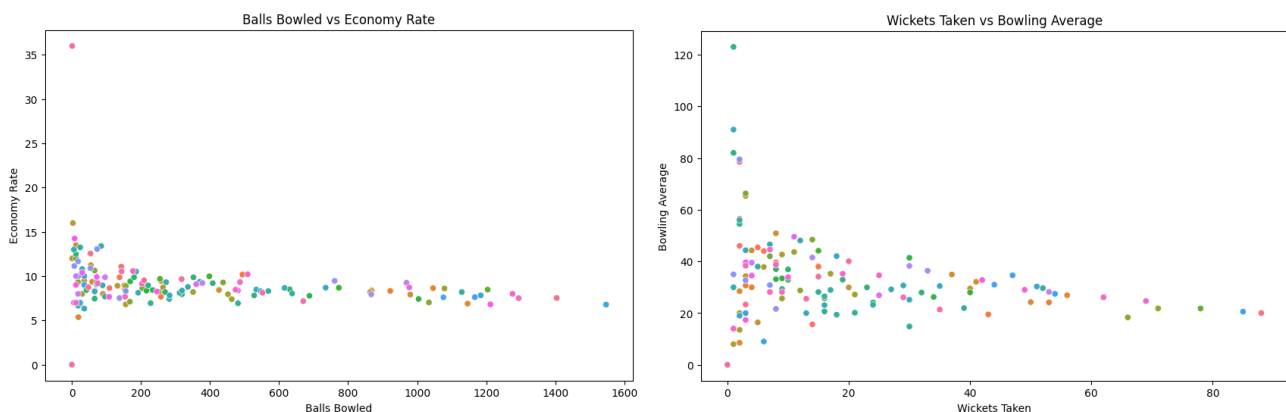
| Statistic | Value |
|--------------------|------------|
| Count | 204.000000 |
| Mean | 6.569210 |
| Standard Deviation | 4.734003 |
| Minimum | 0.000000 |
| 25% | 0.000000 |
| 50% (Median) | 8.078068 |
| 75% | 9.209967 |
| Maximum | 36.000000 |

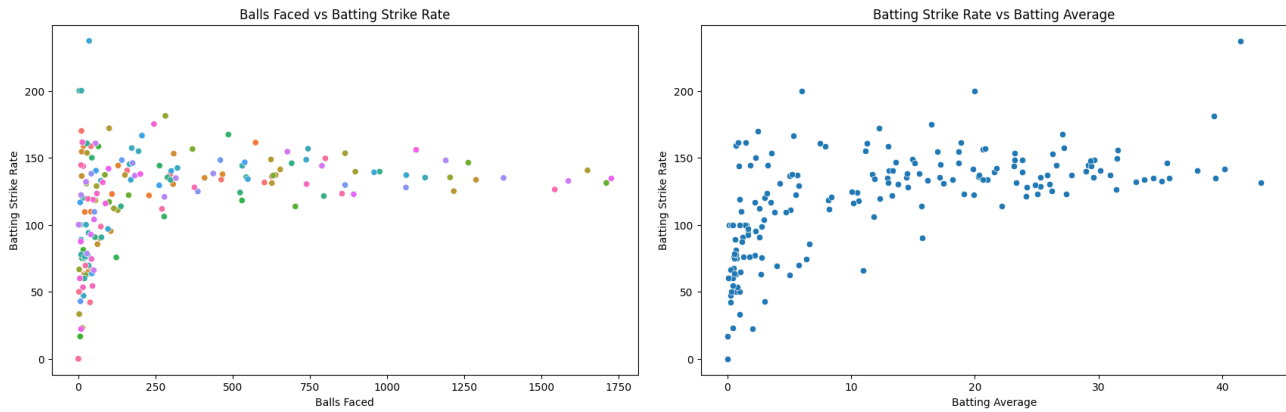
Observations

- The average economy rate is lower than the median, indicating a left-skewed distribution.
- The standard deviation of 4.73 indicates the degree of variability in the economy rate of players.
- The histogram shows that most players have a relatively high economy rate, with a long tail towards lower economy rates, suggesting a right-skewed distribution.
- The minimum economy rate is 0.0, which might be an outlier indicating exceptional bowling performance by a few players.

1.5.2 Bivariate Analysis

In this section, we analyzed relationships between two variables:





Analysis: Batting Strike Rate vs Batting Average

Players with higher batting averages tend to have a moderate batting strike rate. This suggests that players who score consistently also tend to score runs at a relatively controlled pace. Players with higher strike rates may not always have the highest averages, as they might take more risks. Finding the balance between strike rate and average is crucial for batting success.

- We can identify players who are good at scoring runs consistently while maintaining a good strike rate.
- Players with a high strike rate and average could be considered valuable assets for a team.
- Further analysis can be done to study how the strike rate and average vary over time and in different match situations.

Analysis: Balls Faced vs Batting Strike Rate

Players with higher strike rates tend to face fewer balls. This could be due to their ability to score runs quickly, which allows them to occupy the crease for a shorter period.

- We can identify players who have the ability to score quickly.
- Such players can be valuable in certain match situations (e.g., chasing a target or during the death overs).
- Further analysis can be done to study how the strike rate and balls faced vary based on player position and opponent team.

Analysis: Wickets Taken vs Bowling Average

Players who take more wickets tend to have a lower bowling average. This suggests that those who are successful in taking wickets also tend to be more economical bowlers.

- We can identify players who are good wicket-takers and are capable of controlling the run rate.
- Players with low bowling averages and high wicket counts are likely valuable assets for a team.
- Further analysis can be done to study the correlation between wickets taken, bowling average, and economy rate for different bowling types.

Analysis: Balls Bowled vs Economy Rate

Bowlers who bowl more often tend to have a moderately higher economy rate. This could be because they bowl more balls and may be subjected to greater scoring pressure.

- We can identify bowlers who are capable of maintaining a low economy rate despite bowling a large number of balls.
- Such players are valuable for a team as they can be relied upon to bowl a lot of overs without giving away many runs.

Mathematical analysis of those scatter plots :

Correlation between Batting Strike Rate and Batting Average

The correlation coefficient is approximately 0.628. This indicates a positive correlation between Batting Strike Rate and Batting Average. Players with higher batting averages tend to have higher strike rates, suggesting that those who are consistent in scoring runs also score them at a faster pace.

Correlation between Balls Faced and Batting Strike Rate

The correlation coefficient is approximately 0.403. This shows a positive correlation between Balls Faced and Batting Strike Rate. Players who face more balls tend to have higher strike rates, indicating that players who are more comfortable facing the ball also tend to score runs more quickly.

Correlation between Wickets Taken and Bowling Average

The correlation coefficient is *nan*, suggesting there is no significant correlation between Wickets Taken and Bowling Average.

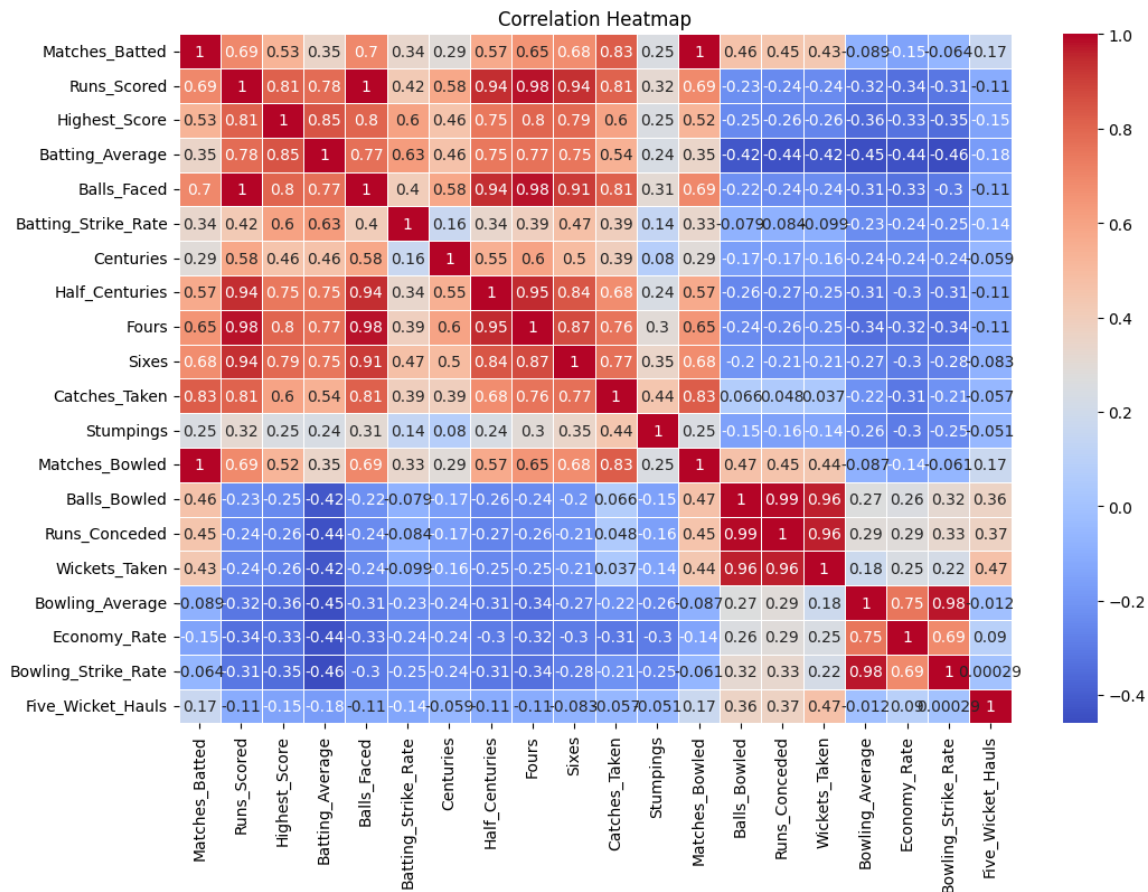
Correlation between Balls Bowled and Economy Rate

The correlation coefficient is approximately 0.257. This indicates a positive correlation between Balls Bowled and Economy Rate. Players who bowl more balls tend to have a higher economy rate.

Thus as we can see there is relations between different data columns we can find the correlation coefficients between each columns of data.

1.5.3 Correlation Coefficients between every data columns:

Correlation Coefficient measures the strength and direction of a linear relationship between two variables. We have used that concept in finding the relation between two data like relationship between "Matches Played" vs "Total Runs"



Correlation Coefficients Data-chart

Analysis of Correlation Heat map

Key Observations

- Variables with strong positive correlations (close to 1) indicate that they tend to increase or decrease together.
- For example, if there is a strong positive correlation between 'Runs_Scored' and 'Batting_Average', it means that players who score more runs also tend to have a higher batting average.
- Variables with strong negative correlations (close to -1) indicate that as one increases, the other tends to decrease.
- For example, if there is a strong negative correlation between 'Economy_Rate' and 'Wickets_Taken', it means that bowlers with lower economy rates tend to take more wickets.
- Values close to 0 indicate little to no correlation between the variables.
- For example, if there is a weak correlation between 'Matches_Batted' and 'Bowling_Average', it means there is no strong linear relationship between the number of matches a player has batted and their bowling average.

Strong Positive Correlations:

- **Runs Scored and Batting Average:** Players who score more runs tend to have higher batting averages.
- **Runs Scored and Batting Strike Rate:** Players with more runs scored often have higher strike rates.
- **Balls Faced and Runs Scored:** Players who face more balls tend to score more runs.
- **Wickets Taken and Balls Bowled:** Players who bowl more overs tend to take more wickets.
- **Wickets Taken and Bowling Average:** Players who take more wickets tend to have lower bowling averages.

Moderate Positive Correlations:

- **Centuries and Half-Centuries:** Players with more centuries also tend to have more half-centuries, indicating consistency in scoring big runs.

Strong Negative Correlations:

- **Bowling Average and Wickets Taken:** Players who have lower bowling averages tend to take more wickets.
- **Economy Rate and Wickets Taken:** Players who have lower economy rates tend to take more wickets.

Weak or No Correlations:

- Some variables have weak or no significant correlation with each other, indicating they might not be directly related or influencing each other.

summary of observations of correlation heatmap:

- **Batting Performance:** Strong positive correlations exist between runs scored, batting average, and batting strike rate, implying that players who score more runs are likely to have better batting averages and strike rates.
- **Bowling Performance:** Similar to batting, strong correlations are observed between wickets taken, bowling average, and economy rate, suggesting that bowlers with lower averages and economy rates are more effective in taking wickets.
- **Balanced Approach:** Players with both good batting and bowling statistics may show stronger correlations between their scores and performances in both departments, indicating a balanced approach to the game.
- **Outliers and Exceptional Players:** Certain players demonstrating exceptional performance in either batting or bowling might not follow the general trends captured by the correlation analysis.

1.6 Team Composition Conditions

1.6.1 Roles

Batsmen

- **Openers (2):** Select aggressive openers who can dominate the powerplay. Consider their performance against both pace and spin.
- **Middle-order (3):** Pick reliable players who can anchor innings or accelerate when needed. At least one of these should be able to handle pressure situations and be a finisher.

Wicket-Keeper(1-2)

- Choose a wicket-keeper who not only excels behind the stumps but can also contribute with the bat, either as an opener or middle-order batsman.

All-Rounders (2-3)

- Pick versatile all-rounders who can adapt to different match situations. Ensure at least one can bat in the lower order to add depth to the lineup.

Bowlers (4-5)

- Bowlers should be carefully selected to effectively restrict the opponent's batsmen from scoring heavily.

Chapter 2

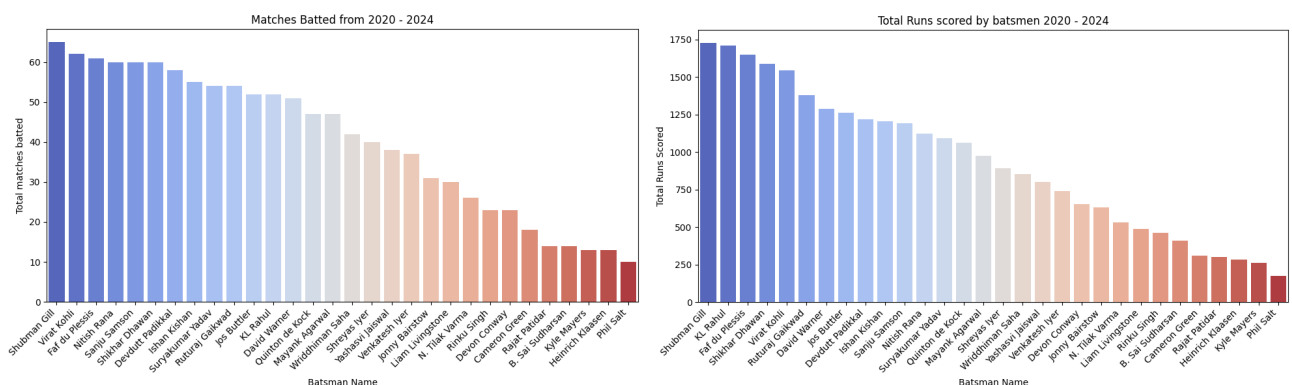
Insights, Findings and Selections

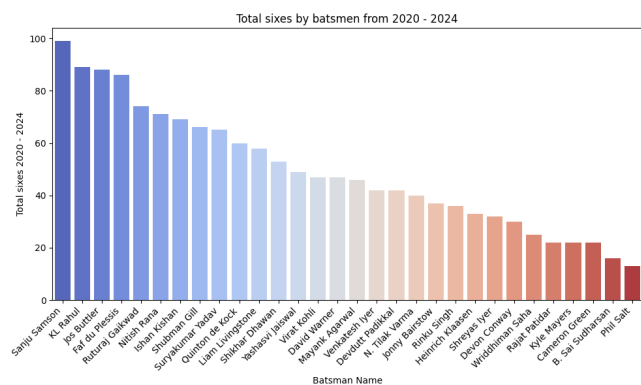
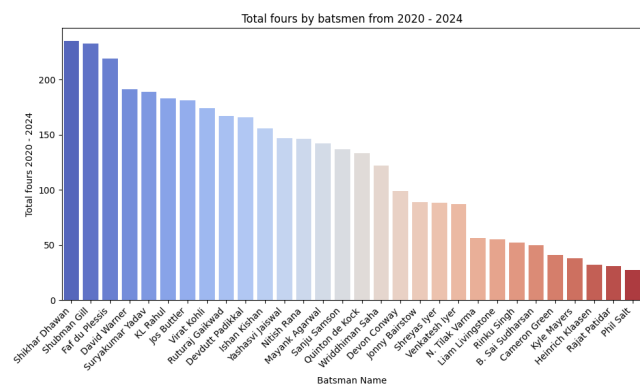
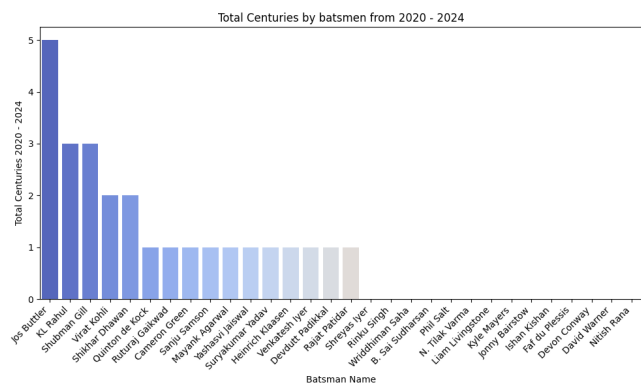
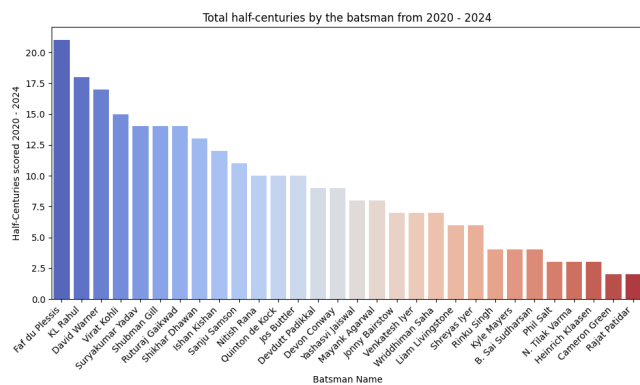
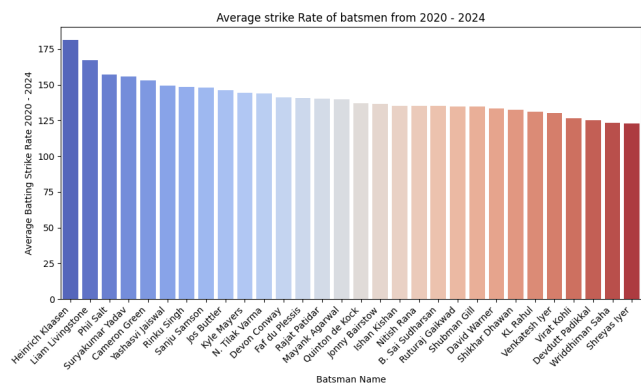
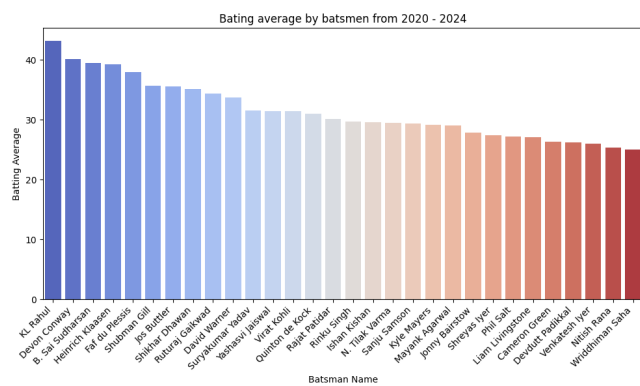
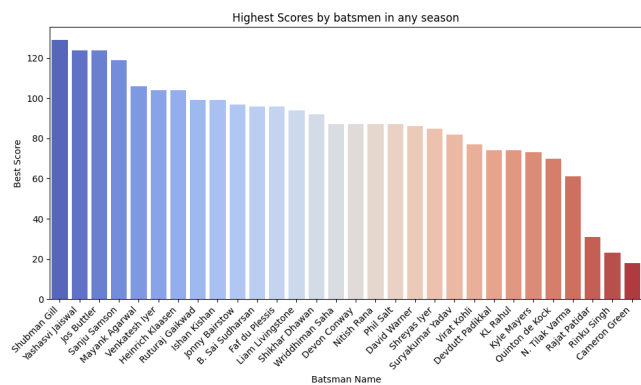
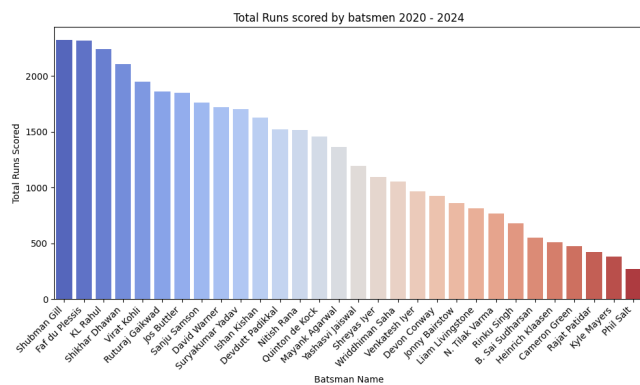
The key analysis and findings from the exploratory data analysis are as follows:

- **Top batsmen** consistently have high strike rates, with their performance correlating strongly with boundary-hitting ability (low Balls per Boundary).
- **Wicket Keepers** should have quality of good batsman along with keeping qualities e.g. catching, stumping etc
- **All-rounders** who contributed significantly in both batting and bowling were relatively rare but valuable.
- **Middle-order batsmen** showed a strong balance between strike rates and averages, indicating their role in stabilizing innings.

2.0.1 Analysis of Batting data :

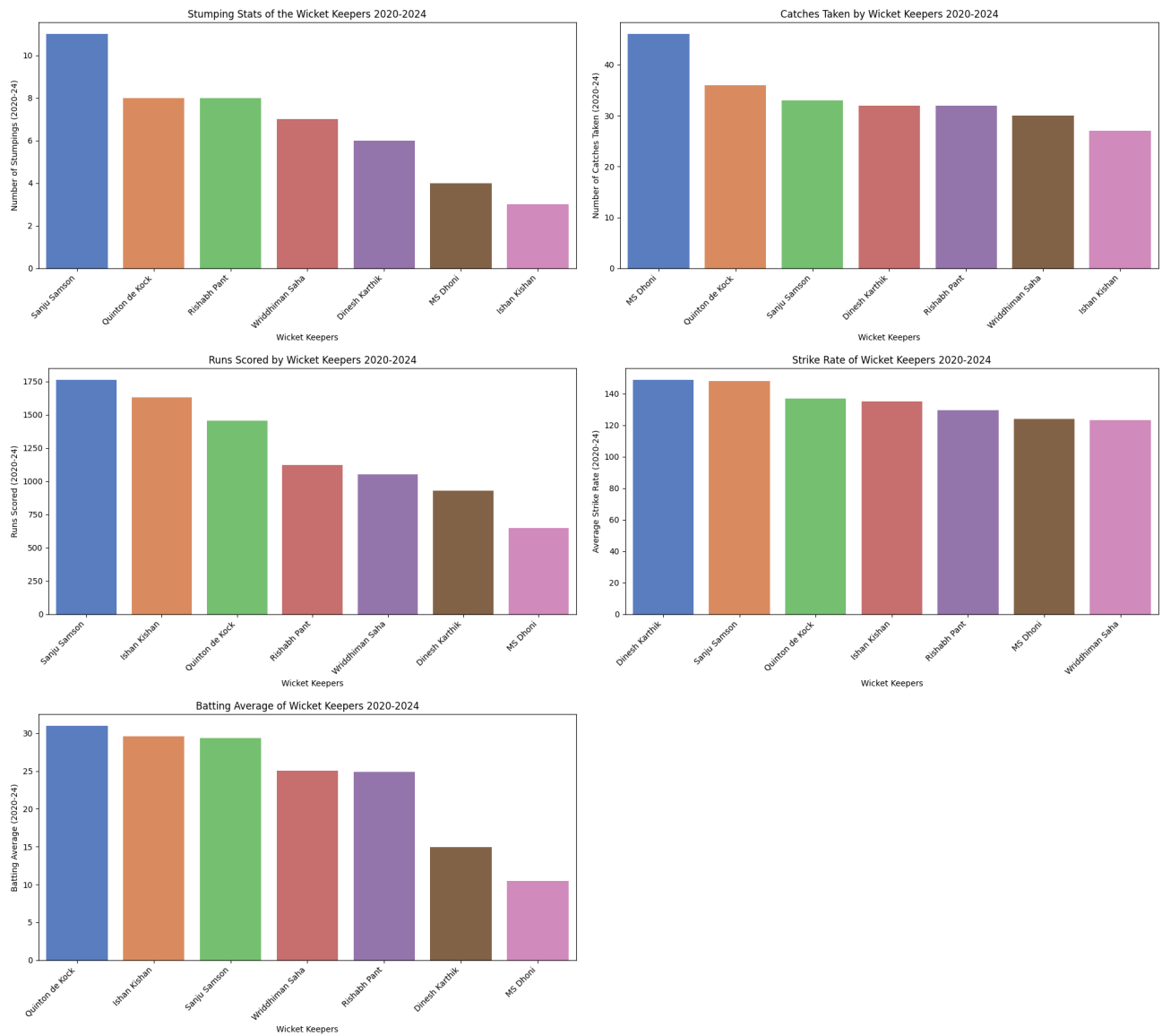
The analysis though visualisation highlighted top batsmen based on matches batted, runs scored, highest score, batting average, strike rate, half-centuries, centuries, fours, and sixes. This presents a comprehensive understanding of their performance as run-scorers.





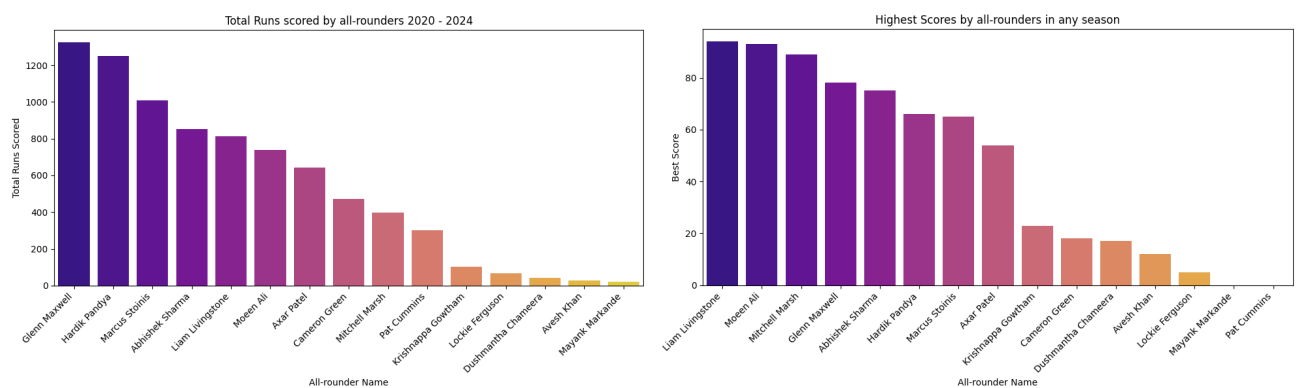
2.0.2 Analysis of Wicket-Keeper's data :

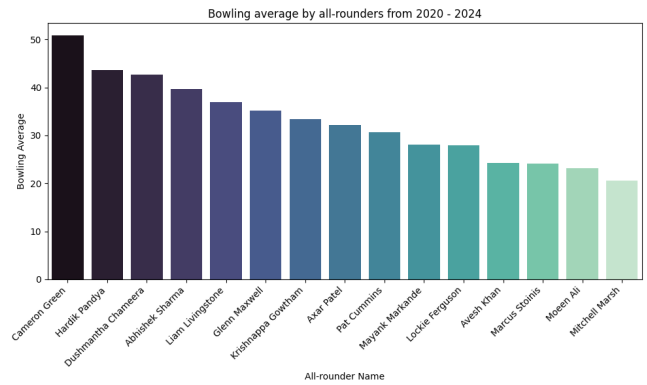
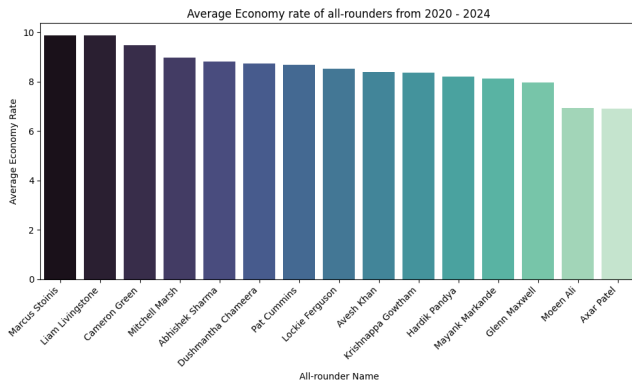
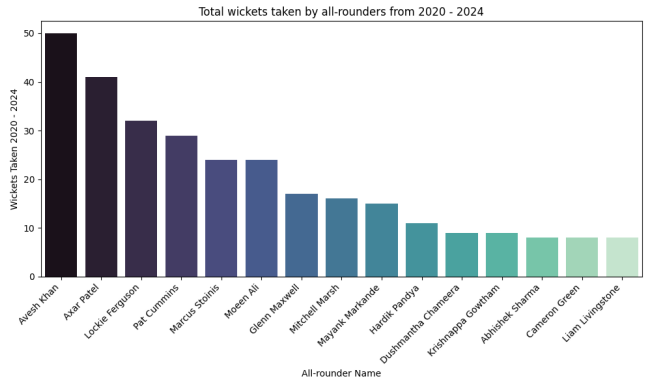
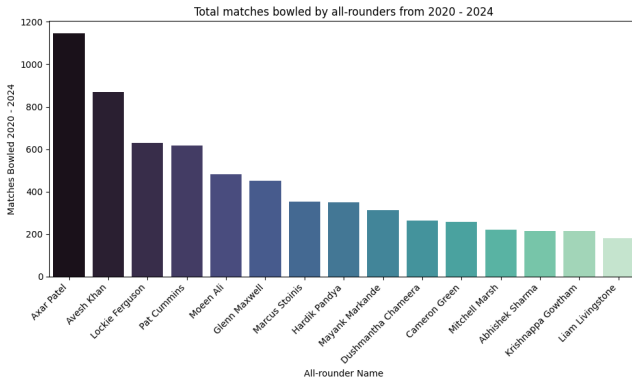
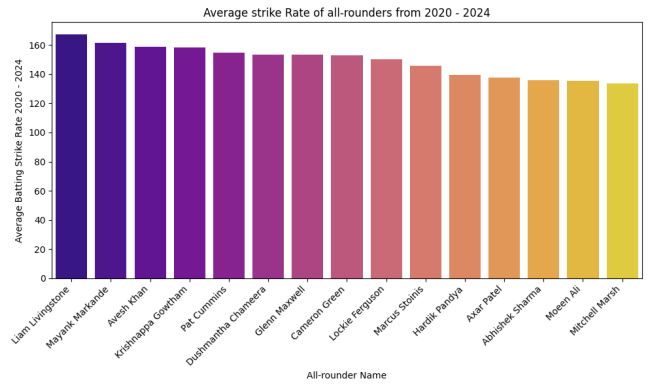
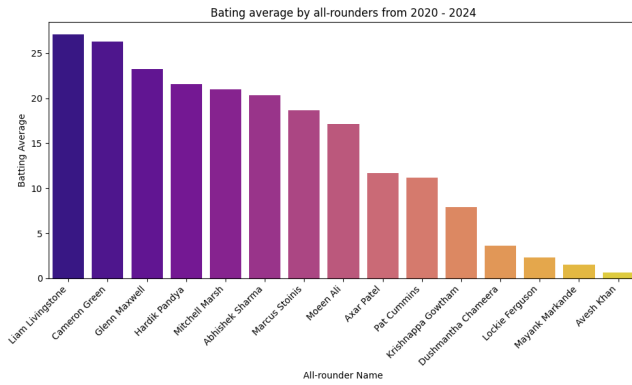
We identified the top wicket-keepers based on their performance in catches, stumpings, runs scored, batting strike rate, and batting average. This provides insights into their overall contribution to the team as both batsmen and wicket-keepers.



2.0.3 Analysis of All-Rounders data :

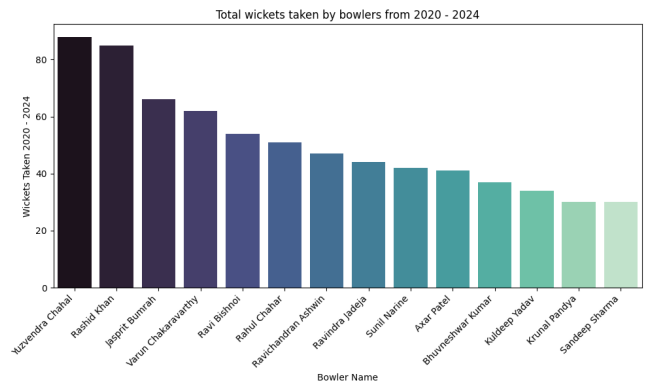
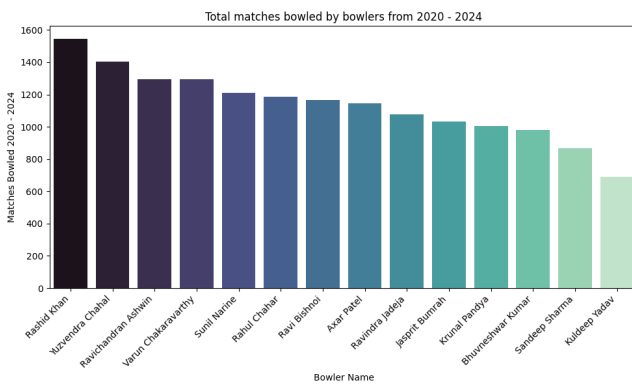
The analysis through visualisation highlighted the best all-rounders based on their batting and bowling performance. We looked at total runs scored, batting average, strike rate, balls bowled, wickets taken, bowling economy, and average. The findings provide a clear picture of their effectiveness as both batsmen and bowlers in the game.

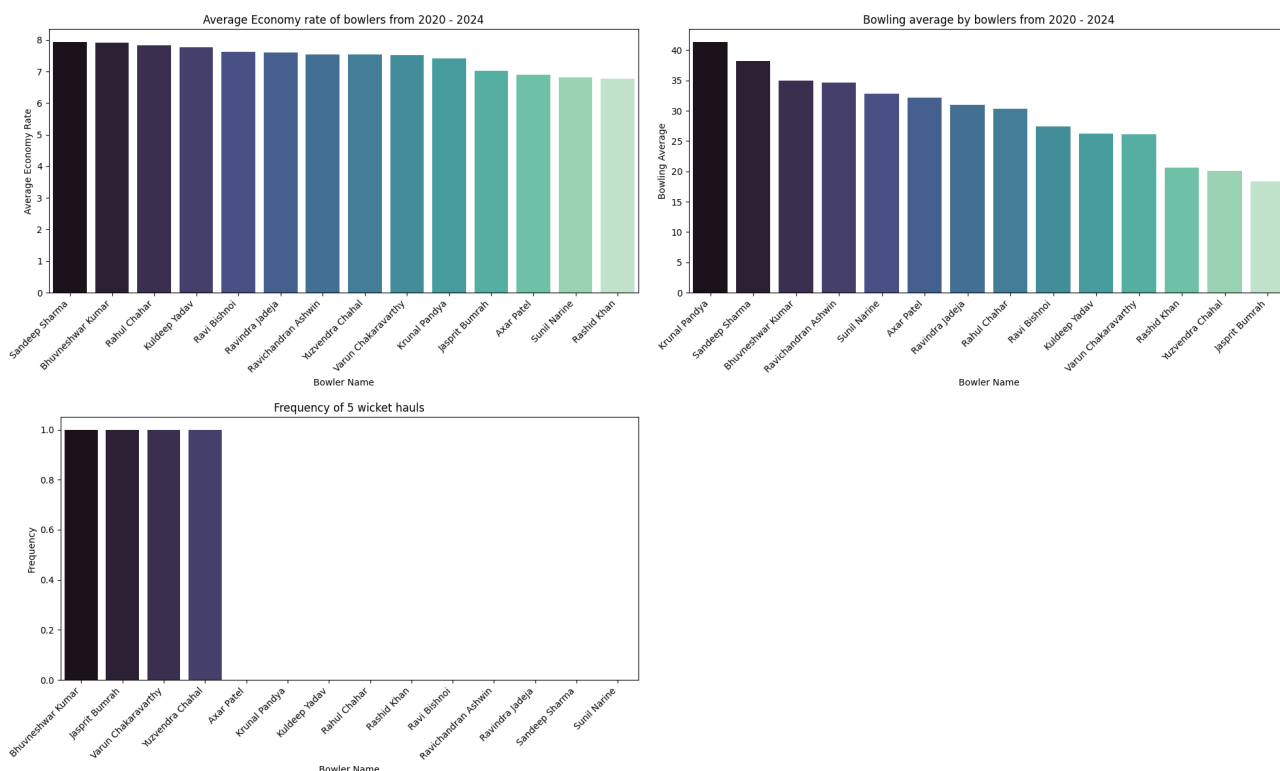




2.0.4 Analysis of Bowlers data :

The top bowlers were identified based on the balls they bowled, wickets they took, bowling economy, bowling average , and five-wicket hauls. These insights give a strong view of their impact on the game.





2.0.5 Findings :

Batsmen

This section highlights the top-performing batsmen across various metrics such as runs scored, batting averages, strike rates, and the number of centuries, half-centuries, fours, and sixes.

Top Run Scorers

Shubman Gill leads the pack with 2322 runs, closely followed by Faf du Plessis with 2318 runs. Other top scorers include KL Rahul (2244), Shikhar Dhawan (2105), and Virat Kohli (1949).

Top Batting Averages

KL Rahul has the highest batting average of 43.15, followed by Devon Conway (40.17), B. Sai Sudharsan (39.43), Heinrich Klaasen (39.31), and Faf du Plessis (38.00).

Top Strike Rates

Heinrich Klaasen boasts the best strike rate at 181.21, with Liam Livingstone (167.28) and Phil Salt (157.23) maintaining aggressive play. Suryakumar Yadav (155.85) and Cameron Green (153.07) follow.

Most Centuries

Jos Buttler is the top century maker with 5 centuries, while KL Rahul and Shubman Gill have 3 centuries each. Virat Kohli and Shikhar Dhawan have 2 centuries each.

Most Half-Centuries

Faf du Plessis has accumulated the most half-centuries with 21. KL Rahul (18), David Warner (17), Virat Kohli (15), and Suryakumar Yadav (14) also appear frequently on the scoreboard.

Most Fours

Shikhar Dhawan has hit the most boundaries with 235 fours. He is followed closely by Shubman Gill (233) and Faf du Plessis (219).

Most Sixes

Sanju Samson leads with 99 sixes, while KL Rahul (89), Jos Buttler (88), Faf du Plessis (86), and Ruturaj Gaikwad (74) have also showcased their power-hitting abilities.

Bowlers

This section reviews the top bowlers based on wickets taken, bowling averages, economy rates, and five-wicket hauls.

Most Wickets Taken

Yuzvendra Chahal is the top wicket-taker with 88 wickets, followed closely by Rashid Khan with 85 wickets. Jasprit Bumrah (66), Varun Chakaravathy (62), and Ravi Bishnoi (54) also rank among the top bowlers.

Best Bowling Averages

Jasprit Bumrah has the best bowling average at 18.35. Yuzvendra Chahal (20.03), Rashid Khan (20.58), Varun Chakaravathy (26.13), and Kuldeep Yadav (26.24) demonstrate strong performance.

Best Economy Rates

Rashid Khan leads with an economy rate of 6.78, followed closely by Sunil Narine (6.82) and Axar Patel (6.90). Jasprit Bumrah (7.03) and Krunal Pandya (7.42) also maintain respectable figures.

Most Five-Wicket Hauls

Bhuvneshwar Kumar, Jasprit Bumrah, Varun Chakaravathy, and Yuzvendra Chahal each have one five-wicket haul, showcasing their ability to take key wickets in crucial matches.

Wicket-Keepers

This section focuses on the performances of wicket-keepers in terms of stumpings, catches taken, batting averages, strike rates, and runs scored.

Most Stumpings

Sanju Samson leads with 11 stumpings, followed by Quinton de Kock and Rishabh Pant, each with 8. Wriddhiman Saha has 7, while Dinesh Karthik has 6 stumpings.

Most Catches Taken

MS Dhoni tops the list with 46 catches. Quinton de Kock follows with 36 catches, while Sanju Samson (33), Dinesh Karthik, and Rishabh Pant (32 each) are also notable contributors.

Best Batting Averages

Quinton de Kock has the highest batting average among wicket-keepers at 30.96. Ishan Kishan (29.62), Sanju Samson (29.35), Wriddhiman Saha (25.05), and Rishabh Pant (24.89) follow closely.

Best Strike Rates

Dinesh Karthik has the best strike rate at 148.72, with Sanju Samson (147.98) and Quinton de Kock (137.01) also maintaining high strike rates. Ishan Kishan (135.30) and Rishabh Pant (129.63) round out the top performers.

Most Runs Scored

Sanju Samson leads the wicket-keepers with 1761 runs. Ishan Kishan follows with 1629 runs, while Quinton de Kock (1455), Rishabh Pant (1120), and Wriddhiman Saha (1052) complete the list.

All-Rounders

This section summarizes the performances of all-rounders in terms of runs scored, wickets taken, batting averages, bowling averages, strike rates, and economy rates.

Top Run Scorers

Shubman Gill leads all-rounders with 2322 runs, followed closely by Faf du Plessis (2318) and KL Rahul (2244). Shikhar Dhawan (2105) and Virat Kohli (1949) also rank highly.

Most Wickets Taken

Yuzvendra Chahal is the top all-rounder in wickets taken with 88, followed by Rashid Khan (85), Jasprit Bumrah (66), Varun Chakaravathy (62), and Ravi Bishnoi (54).

Best Batting Averages

KL Rahul has the highest batting average among all-rounders at 43.15, with Devon Conway (40.17), B. Sai Sudharsan (39.43), Heinrich Klaasen (39.31), and Faf du Plessis (38.00) following.

Best Bowling Averages

Jasprit Bumrah leads with a bowling average of 18.35, with Yuzvendra Chahal (20.03), Rashid Khan (20.58), Varun Chakaravathy (26.13), and Kuldeep Yadav (26.24) not far behind.

Top Strike Rates

Heinrich Klaasen stands out with a batting strike rate of 181.21, followed by Liam Livingstone (167.28), Phil Salt (157.23), Suryakumar Yadav (155.85), and Cameron Green (153.07).

Best Economy Rates

Rashid Khan leads with an economy rate of 6.78, with Sunil Narine (6.82) and Axar Patel (6.90) maintaining competitive rates. Jasprit Bumrah (7.03) and Krunal Pandya (7.42) also perform well.

Chapter 3

Final Selection

3.1 Selection based on user input

3.1.1 Introduction

Through our analysis, we identified various metrics to evaluate the performance of each player, allowing us to determine the best players within each category. However, it is important to note that a player who excels in one area, such as having the highest strike rate, may not necessarily have a strong batting average.

Our model provides users with the flexibility to assign different weights to specific categories or criteria, enabling a tailored assessment based on individual preferences.

User Input Summary

The user provided weights for different criteria to select the best players for each role:

- **Batsmen:**
 - Strike rate
 - Total runs
 - Highest score
 - Balls per boundary
- **Wicket-Keepers:**
 - Stumpings
 - Catches
 - Batting average
 - Strike rate
- **All-Rounders:**
 - Batting average
 - Strike rate
 - Wickets taken

- Economy rate
- **Bowlers:**
 - Wickets taken
 - Economy rate
 - Bowling average
 - 5-wicket hauls

Selection Process

Based on the user's weights, a '**Weighted Score**' is calculated for each player in their respective role. The players are then ranked based on this weighted score, and the top players according to the weightings are selected.

Example

If a user prioritizes runs scored above strike rate for batsmen, players with high run tallies will be ranked higher, even if they have a slightly lower strike rate. Similarly, a user who prioritizes wicket-taking over economy rate for bowlers will see the players with the most wickets selected, even if their economy rate is a bit higher.

This approach allows for a customizable selection process where the user can fine-tune the criteria to fit their specific needs and preferences for team composition.

3.1.2 Code for Selections

Batsman selection

```

1 def select_best_batsmen(data_bat, strike_rate_weight, total_runs_weight,
2   Batting_Average_weight, balls_per_boundary_weight, num_batsmen=5):
3
4   data_bat['Weighted_Score'] = (
5       (data_bat['Batting_Strike_Rate'] * strike_rate_weight +
6        data_bat['Runs_Scored'] * total_runs_weight +
7        data_bat['Batting_Average'] * Batting_Average_weight +
8        data_bat['Balls_per_Boundary'] * balls_per_boundary_weight) / (
9           strike_rate_weight + total_runs_weight + Batting_Average_weight +
10          balls_per_boundary_weight)
11   )
12
13   selected_batsmen = data_bat.sort_values('Weighted_Score', ascending=False)
14   .head(num_batsmen)
15   return selected_batsmen
16
17 strike_rate_weight = float(input("Enter weight for strike rate : "))
18 total_runs_weight = float(input("Enter weight for total runs : "))
19 Batting_Average_weight = float(input("Enter weight for Batting Average : "))
20 balls_per_boundary_weight = float(input("Enter weight for balls per boundary : "))

```

```

20 best_batsmen = select_best_batsmen(data_bat, strike_rate_weight,
    total_runs_weight, Batting_Average_weight, balls_per_boundary_weight)
21
22 # Print the selected batsmen
23 print("\nSelected Batsmen:")
24 print(best_batsmen[['Player_Name', 'Batting_Strike_Rate', 'Runs_Scored', '
    Batting_Average', 'Balls_per_Boundary', 'Weighted_Score']])

```

Sample Output :

User Input Weights

Selected Batsmen Based on Weighted Scores

Weights:

- Batting Strike Rate: 50
- Total Runs: 20
- Batting Average: 20
- Balls per Boundary: 10

Selected Batsmen:

| Player Name | Strike Rate | Runs Scored | Batting Average | Balls per Boundary | Weighted Score |
|----------------|-------------|-------------|-----------------|--------------------|----------------|
| Faf du Plessis | 140.57 | 2318 | 38.00 | 5.41 | 542.03 |
| Shubman Gill | 134.53 | 2322 | 35.72 | 5.77 | 539.39 |
| KL Rahul | 131.23 | 2244 | 43.15 | 6.29 | 523.67 |
| Shikhar Dhawan | 132.64 | 2105 | 35.08 | 5.51 | 494.89 |
| Virat Kohli | 126.31 | 1949 | 31.44 | 6.98 | 459.94 |

Table 3.1: Selected Batsmen and Their Weighted Scores

Wicket-Keeper Selection

```

1 def select_best_wicket_keepers(data_wk, stumpings_weight, catches_weight,
    batting_avg_weight, strike_rate_weight, num_wicket_keepers=5):
2
3     data_wk['Weighted_Score'] = (
4         (data_wk['Stumpings'] * stumpings_weight +
5         data_wk['Catches_Taken'] * catches_weight +
6         data_wk['Batting_Average'] * batting_avg_weight +
7         data_wk['Batting_Strike_Rate'] * strike_rate_weight)/(stumpings_weight
8         + catches_weight+batting_avg_weight+strike_rate_weight)
9     )
10
11     selected_wicket_keepers = data_wk.sort_values('Weighted_Score', ascending=
12         False).head(num_wicket_keepers)
13     return selected_wicket_keepers
14
15 # Get user input for weights
16 stumpings_weight = float(input("Enter weight for stumpings (e.g., 0.3): "))
17 catches_weight = float(input("Enter weight for catches (e.g., 0.3): "))

```

```

17 batting_avg_weight = float(input("Enter weight for batting average (e.g.,
    0.2): "))
18 strike_rate_weight = float(input("Enter weight for strike rate (e.g., 0.2):
    "))
19
20
21 # Select the best wicket-keepers based on user input
22 best_wicket_keepers = select_best_wicket_keepers(data_wk, stumpings_weight,
    catches_weight, batting_avg_weight, strike_rate_weight)
23
24 # Print the selected wicket-keepers
25 print("\nSelected Wicket-Keepers:")
26 print(best_wicket_keepers[['Player_Name', 'Stumpings', 'Catches_Taken', '
    Batting_Average', 'Batting_Strike_Rate', 'Weighted_Score']])
27
28 best_wicket_keepers.head(5)

```

Analysis of Selected Wicket-Keepers Based on Weighted Scores

Weights:

- Stumpings: 20
- Catches Taken: 20
- Batting Average: 40
- Batting Strike Rate: 20

Selected Wicket-Keepers:

| Player Name | Stumpings | Catches Taken | Batting Average | Batting Strike Rate | Weighted Score |
|-----------------|-----------|---------------|-----------------|---------------------|----------------|
| Sanju Samson | 11.0 | 33.0 | 29.35 | 147.98 | 50.14 |
| Quinton de Kock | 8.0 | 36.0 | 30.96 | 137.01 | 48.58 |
| Ishan Kishan | 3.0 | 27.0 | 29.62 | 135.30 | 44.91 |
| Rishabh Pant | 8.0 | 32.0 | 24.89 | 129.63 | 43.88 |
| Dinesh Karthik | 6.0 | 32.0 | 14.97 | 148.72 | 43.33 |

Table 3.2: Selected Wicket-Keepers and Their Weighted Scores

All-rounders Selection

```

1 def select_best_all_rounders(data_ar, batting_avg_weight, strike_rate_weight
    , wickets_weight, economy_rate_weight, num_all_rounders=5):
2
3     data_ar['Weighted_Score'] = (
4         (data_ar['Batting_Average'] * batting_avg_weight +
5         data_ar['Batting_Strike_Rate'] * strike_rate_weight +
6         data_ar['Wickets_Taken'] * wickets_weight +
7         (10 - data_ar['Economy_Rate']) * economy_rate_weight) / (batting_avg +
8         strike_rate_weight + wickets_weight + economy_rate_weight)
9     )
10     selected_all_rounders = data_ar.sort_values('Weighted_Score', ascending=
    False).head(num_all_rounders)

```

```

11     return selected_all_rounders
12
13 # Get user input for weights
14 batting_avg_weight = float(input("Enter weight for batting average (e.g.,
    0.25): "))
15 strike_rate_weight = float(input("Enter weight for strike rate (e.g., 0.25):
    "))
16 wickets_weight = float(input("Enter weight for wickets taken (e.g., 0.25): ")
    )
17 economy_rate_weight = float(input("Enter weight for economy rate (e.g.,
    0.25): "))
18
19 # Select the best all-rounders based on user input
20 best_all_rounders = select_best_all_rounders(data_ar, batting_avg_weight,
    strike_rate_weight, wickets_weight, economy_rate_weight)
21
22 # Print the selected all-rounders
23 print("\nSelected All-Rounders:")
24 print(best_all_rounders[['Player_Name', 'Batting_Average', '
    Batting_Strike_Rate', 'Wickets_Taken', 'Economy_Rate', 'Weighted_Score']])
25
26
27 best_all_rounders.head(5)

```

Analysis of Selected All-Rounders Based on Weighted Scores

Weights:

- Batting Average: 20
- Batting Strike Rate: 30
- Wickets Taken: 10
- Economy Rate: 40

Selected All-Rounders:

| Player Name | Batting Average | Strike Rate | Wickets Taken | Economy Rate | Weighted Score |
|------------------|-----------------|-------------|---------------|--------------|----------------|
| Liam Livingstone | 27.10 | 167.28 | 8 | 9.87 | 56.46 |
| Avesh Khan | 0.68 | 158.82 | 50 | 8.38 | 53.43 |
| Glenn Maxwell | 23.25 | 153.36 | 17 | 7.95 | 53.18 |
| Cameron Green | 26.28 | 153.07 | 8 | 9.47 | 52.19 |
| Pat Cummins | 11.19 | 154.87 | 29 | 8.67 | 52.13 |

Table 3.3: Selected All-Rounders and Their Weighted Scores

Bowlers Selection

```

1 def select_best_bowlers(data_bowler, wickets_weight, economy_rate_weight,
    bowling_avg_weight, five_wicket_haul_weight, num_bowlers=5):
2
3
4     data_bowler['Weighted_Score'] = (
5         (data_bowler['Wickets_Taken'] * wickets_weight +

```



```

6         (10 - data_bowler['Economy_Rate']) * economy_rate_weight +
7         (1 / data_bowler['Bowling_Average']) * bowling_avg_weight +
8         data_bowler['Five_Wicket_Hauls'] * five_wicket_haul_weight))/
9         (wickets_weight+economy_rate_weight+bowling_avg_weight+
10         five_wicket_haul_weight)
11     )
12     selected_bowlers = data_bowler.sort_values('Weighted_Score', ascending=
13         False).head(num_bowlers)
14     return selected_bowlers
15
16 # Get user input for weights
17 wickets_weight = float(input("Enter weight for wickets taken (e.g., 0.3): ")
18 )
19 economy_rate_weight = float(input("Enter weight for economy rate (e.g., 0.3)
20 : "))
21 bowling_avg_weight = float(input("Enter weight for bowling average (e.g.,
22 0.2): "))
23 five_wicket_haul_weight = float(input("Enter weight for 5-wicket hauls (e.g
24 ., 0.2): "))
25
26 # Select the best bowlers based on user input
27 best_bowlers = select_best_bowlers(data_bowler, wickets_weight,
28     economy_rate_weight, bowling_avg_weight, five_wicket_haul_weight)
29
30 # Print the selected bowlers
31 print("\nSelected Bowlers:")
32 print(best_bowlers[['Player_Name', 'Wickets_Taken', 'Economy_Rate', '
33     Bowling_Average', 'Five_Wicket_Hauls', 'Weighted_Score']])
34 best_bowlers.head(5)

```

Analysis of Selected Bowlers Based on Weighted Scores

Weights:

- Wickets Taken: 20
- Economy Rate: 30
- Bowling Average: 40
- 5-Wicket Hauls: 10

Selected Bowlers:

| Player Name | Wickets Taken | Economy Rate | Bowling Average | 5-Wicket Hauls | Weighted Score |
|--------------------|---------------|--------------|-----------------|----------------|----------------|
| Yuzvendra Chahal | 88 | 7.53 | 20.03 | 1 | 18.46 |
| Rashid Khan | 85 | 6.78 | 20.58 | 0 | 17.98 |
| Jasprit Bumrah | 66 | 7.03 | 18.35 | 1 | 14.21 |
| Varun Chakaravathy | 62 | 7.51 | 26.13 | 1 | 13.26 |
| Ravi Bishnoi | 54 | 7.62 | 27.46 | 0 | 11.53 |

Table 3.4: Selected Bowlers and Their Weighted Scores

3.2 Conclusion

The **EDA** on the **IPL** dataset provided valuable insights into player performances over the last five seasons (**2020-2024**). By analyzing various batting and bowling metrics, we were able to identify top performers and select a balanced team of **15 players**, including **batsmen**, **all-rounders**, **wicket-keepers**, and **bowlers**. This analysis highlighted the importance of **strike rates**, **economy rates**, and **consistency in both batting and bowling**.

Further analysis could involve using **machine learning models** to predict future player performances based on historical data or exploring team-level strategies.

References

- Patrick B. (2020). IPL Complete Dataset (2008-2020). Retrieved from Kaggle.
- Gujarati, D. N. (2020). *Basic Econometrics*. McGraw-Hill Education.
- Ross, S. M. (2019). *Introduction to Probability*. Academic Press.