****

**VIRGINIA COMMONWEALTH UNIVERSITY**

**Statistical analysis and modelling (SCMA 632)**

**A1a: Preliminary preparation and analysis of data- Descriptive statistics**

**Bala Vignesh Aravindan**

**V01106579**

**Date of Submission: 16-06-2024**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title in R** | **Page No.** |
| **1.** | Introduction | **1** |
| **2.** | Results | **2** |
| **3.** | Interpretations | **3** |
| **4.** | Recommendations | **3** |
| **5.** | Codes | **4-8** |
| **6.** | References | **8** |

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title in Python** | **Page No.** |
| **1.** | Introduction | **9** |
| **2.** | Results | **9** |
| **3.** | Interpretations | **10** |
| **4.** | Recommendations | **11** |
| **5.** | Codes | **12-13** |

**Exploratory Analysis of Rahul Tewatia's Performance in IPL Matches**

**Introduction:**

Cricket enthusiasts and stakeholders in the Indian Premier League (IPL) are constantly seeking ways to evaluate player performance effectively. This analysis delves into the performance of Rahul Tewatia, a prominent figure in IPL cricket, aiming to uncover insights into his contributions to his team's success and his overall impact on match outcomes.

The analysis begins with the collection and preprocessing of IPL match data, including player statistics, match details, and team information. Data cleaning techniques are employed to handle missing values, outliers, and inconsistencies, ensuring the integrity and reliability of the dataset.

**Results:**

1. **Identification of Matches Involving Rahul Tewatia:**
   * Rahul Tewatia's participation in IPL matches is meticulously tracked, with a focus on his roles as a batsman or a bowler. This allows for a comprehensive assessment of his on-field involvement and performance.
2. **Isolation of Matches Involving Rahul Tewatia Alone:**
   * Instances where Rahul Tewatia's name appears without any other player sharing the same name are identified. These occurrences provide valuable insights into matches where Tewatia's performance data stands uniquely.
3. **Grouping Data by IPL Rounds:**
   * The dataset is organized and grouped based on IPL rounds, enabling a detailed examination of player performance trends across different phases of the tournament. This segmentation facilitates the identification of patterns and fluctuations in performance over time.
4. **Performance Metrics Summary:**
   * Key performance metrics such as runs scored, wickets taken, strike rates, and economy rates are summarized for Rahul Tewatia across various IPL matches. These summaries offer a quantitative overview of Tewatia's contributions and effectiveness on the field.
5. **Top Performers Identification:**
   * The top three run-getters and wicket-takers in each IPL round are determined through comprehensive analysis. By identifying standout performers in different stages of the tournament, this analysis highlights players who have made significant contributions to their team's success.

**Interpretations:**

* Rahul Tewatia's consistent participation in IPL matches underscores his importance within his team's lineup and strategy.
* Matches where Rahul Tewatia appears as the sole player with his name indicate the uniqueness of his performance data, potentially offering valuable insights into his individual contributions.
* Grouping the data by IPL rounds enables a nuanced understanding of player dynamics and performance variations throughout the tournament, allowing teams and stakeholders to adapt strategies accordingly.
* Identifying top performers in each IPL round emphasizes the critical role played by standout players in shaping match outcomes and team success.
* As per the salary ,if Rahul Tewatia’s performance is good automatically his salary increases

**Recommendations:**

* The insights gained from this analysis can be leveraged by IPL teams and stakeholders to make informed decisions regarding player selection, strategy formulation, and resource allocation.
* Strategies should be devised to optimize player utilization and maximize the impact of top performers like Rahul Tewatia, potentially leading to improved team performance and competitiveness in the IPL.

**Code:**

.libPaths()

# Load necessary libraries

library(dplyr)

library(readr)

# Read the CSV file

data <- read\_csv("C:\\Users\\Bala Vignesh.A\\Desktop\\SCMA 632\\Cricket\_data.csv")

spec(data)

data <- read\_csv("C:\\Users\\Bala Vignesh.A\\Desktop\\SCMA 632\\Cricket\_data.csv", col\_types = cols(

home\_key\_batsman = col\_character(),

home\_key\_bowler = col\_character(),

home\_playx1 = col\_character(),

away\_playx1 = col\_character(),

away\_key\_batsman = col\_character(),

away\_key\_bowler = col\_character()

))

columns\_to\_check <- c('home\_key\_batsman', 'home\_key\_bowler', 'home\_playx1', 'away\_playx1', 'away\_key\_batsman', 'away\_key\_bowler')

library(stringr)

# Create a filter mask

filter\_mask <- data %>%

select(all\_of(columns\_to\_check)) %>%

mutate(across(everything(), ~str\_detect(., 'Rahul Tewatia'))) %>%

rowwise() %>%

mutate(filter\_mask = any(c\_across(everything()))) %>%

pull(filter\_mask)

# Filter the data

rahul\_tewatia\_data <- data[filter\_mask, ]

# View the filtered data

print(rahul\_tewatia\_data)

# Count the number of rows where 'Rahul Tewatia' appears alone

num\_alone <- nrow(rahul\_tewatia\_data)

print(rahul\_tewatia\_data)

# Calculate the number of rows where 'Rahul Tewatia' appears alone

num\_alone <- nrow(rahul\_tewatia\_data)

cat("Number of rows where 'Rahul Tewatia' appears alone: ", num\_alone, "\n")

# Arrange data round-wise and batsman, ball, runs, and wickets per player per match

data\_arranged <- data %>%

mutate(round = ifelse(match\_days <= 15, "Round 1",

ifelse(match\_days > 15 & match\_days <= 30, "Round 2", "Round 3")),

Balls = home\_overs \* 6) %>%

select(round, season, id, name, short\_name, description, home\_team, away\_team, toss\_won, decision, `1st\_inning\_score`, `2nd\_inning\_score`, winner, result, start\_date, end\_date, venue\_id, venue\_name, home\_captain, away\_captain, pom, points, super\_over, home\_overs, home\_runs, home\_wickets, home\_boundaries, Player = home\_key\_batsman, match\_days, umpire1, umpire2, tv\_umpire, referee, reserve\_umpire, Runs = home\_runs, Wickets = home\_wickets) %>%

filter(!is.na(Player)) %>%

arrange(round, season, match\_days, Player)

# Indicate the top three run-getters and top three wicket-takers in each round

# Indicate the top three run-getters and top three wicket-takers in each round

data\_top\_players <- data\_arranged %>%

group\_by(round) %>%

top\_n(n = 3, wt = Runs) %>%

arrange(round, desc(Runs)) %>%

select(round, Player, Runs) %>%

rename(Batsman = Player) %>%

bind\_rows(

data\_arranged %>%

group\_by(round) %>%

top\_n(n = 3, wt = Wickets) %>%

arrange(round, desc(Wickets)) %>%

select(round, Player, Wickets) %>%

rename(Bowler = Player)

) %>%

arrange(round, desc(Runs), desc(Wickets))

# Print the top three run-getters and top three wicket-takers in each round

print(data\_top\_players)

install.packages("ggplot2")

# Load necessary libraries

library(ggplot2)

# Create a dataframe with the player's details

player\_data <- data.frame(

Player = "Rahul Tewatia",

Salary = 900,

Runs = 1000 # assume 1000 runs for Rahul Tewatia

)

# Plot the relationship between a player's performance and the salary he gets

ggplot(player\_data, aes(x = Runs, y = Salary, label = Player)) +

geom\_point() +

geom\_text(nudge\_y = 0.1, size = 3) +

labs(x = "Runs", y = "Salary", title = "Relationship between a player's performance and salary")

**References:**

* The analysis utilizes R packages such as dplyr, readr, and ggplot2 for data manipulation, CSV file parsing, and visualization. Additionally, external cricket data sources may have been consulted to enhance the accuracy and comprehensiveness of the analysis.

**Exploratory Analysis of Rahul Tewatia's Performance in IPL Matches**

**Introduction:**

In this analysis, we delve into the performance of Rahul Tewatia within the context of matches where his salary is 9 crore. By focusing on these specific matches, we aim to gain insights into Tewatia's average runs scored, wickets taken, and boundaries hit, thereby evaluating his impact on the field when his remuneration aligns with this particular threshold.

**Results:**

The analysis reveals compelling statistics for matches where Rahul Tewatia's salary stands at 9 crore:

* The average runs scored by Tewatia in these matches is approximately 2878.78.
* Tewatia's average wickets taken is approximately 101.11.
* The average number of boundaries hit by Tewatia amounts to approximately 359.11.

**Interpretations:**

The noteworthy findings from the analysis prompt several interpretations:

* Tewatia's consistently high average runs scored in matches with a 9 crore salary underscore his value as a dependable batsman. This indicates his capability to contribute significantly to his team's total runs, thereby bolstering their chances of success.
* Additionally, Tewatia's remarkable average wickets taken highlights his versatility as an all-rounder. His ability to take wickets consistently adds depth to his contributions, impacting both the batting and bowling aspects of the game positively.
* The average number of boundaries hit by Tewatia further emphasizes his effectiveness as a batsman capable of scoring runs quickly and decisively. His knack for finding the boundaries adds a crucial dimension to his team's scoring potential, putting pressure on the opposition.

**Recommendations:**

Based on the compelling results obtained:

* It is recommended that Tewatia continues to be deployed in matches where his salary aligns with the 9 crore threshold. His demonstrated ability to perform consistently across batting, bowling, and boundary-hitting aspects makes him a valuable asset to his team in such matches.
* Furthermore, to enhance understanding and optimize Tewatia's performance, additional analysis could explore specific factors contributing to his success in matches with a 9 crore salary. This deeper dive could uncover insights into conditions, opponent strategies, or team dynamics that further amplify Tewatia's effectiveness on the field.

By leveraging these recommendations, teams can capitalize on Tewatia's strengths and optimize their strategies to maximize his impact in matches where his salary corresponds to 9 crore, ultimately enhancing their chances of success in the IPL and similar cricket tournaments.

**Codes:**

import pandas as pd data = pd.read\_csv('C:\Users\Bala Vignesh.A\Desktop\SCMA 632\Cricket\_data.csv') print(data) print(ct\_data) (data['home\_key\_batsman'] == 'Rahul Tewatia') (data['home\_playx1'] == 'Rahul Tewatia') (data['away\_playx1'] == 'Rahul Tewatia') (data['away\_key\_batsman'] == 'Rahul Tewatia') (data['away\_key\_bowler'] == 'Rahul Tewatia') columns\_to\_check = ['home\_key\_batsman', 'home\_key\_bowler', 'home\_playx1', 'away\_playx1', 'away\_key\_batsman', 'away\_key\_bowler']

filter\_mask = data[columns\_to\_check].apply(lambda x: x.str.contains('Rahul Tewatia')).any(axis=1)

rahul\_tewatia\_data = data[filter\_mask]

print(rahul\_tewatia\_data)

rahul\_tewatia\_alone\_data = data[filter\_mask]

num\_alone = rahul\_tewatia\_alone\_data.shape[0]

print(f"Number of rows where 'Rahul Tewatia' appears alone: {num\_alone}")

(data['home\_key\_bowler'] == 'Rahul Tewatia') columns\_to\_check = ['home\_key\_batsman', 'home\_key\_bowler', 'home\_playx1', 'away\_playx1', 'away\_key\_batsman', 'away\_key\_bowler'] filter\_mask = data[columns\_to\_check].apply(lambda x: x.str.count('Rahul Tewatia') == 1).any(axis=1) print(data.columns) data.columns = data.columns.str.strip() print(data.columns) Assuming 'data' is the DataFrame containing the IPL data Create a new DataFrame with the required columns ipl\_data = data[['season', 'id', 'name', 'short\_name', 'description', 'home\_team', 'away\_team', 'toss\_won', 'decision', '1st\_inning\_score', '2nd\_inning\_score', 'winner', 'result', 'start\_date', 'end\_date', 'venue\_id', 'venue\_name', 'home\_captain', 'away\_captain', 'pom', 'points', 'super\_over', 'home\_overs', 'home\_runs', 'home\_wickets', 'home\_boundaries', 'away\_overs', 'away\_runs', 'away\_wickets', 'away\_boundaries', 'highlights', 'home\_key\_batsman', 'home\_key\_bowler', 'home\_playx1', 'away\_playx1', 'away\_key\_batsman', 'away\_key\_bowler', 'match\_days', 'umpire1', 'umpire2', 'tv\_umpire', 'referee', 'reserve\_umpire']] print(ipl\_data) Extract the batsman, ball, runs, and wickets per player per match ipl\_data\_batting = ipl\_data[['season', 'id', 'name', 'short\_name', 'description', 'home\_team', 'away\_team', 'toss\_won', 'decision', '1st\_inning\_score', '2nd\_inning\_score', 'winner', 'result', 'start\_date', 'end\_date', 'venue\_id', 'venue\_name', 'home\_captain', 'away\_captain', 'pom', 'points', 'super\_over', 'home\_overs', 'home\_runs', 'home\_wickets', 'home\_boundaries', 'home\_key\_batsman', 'match\_days', 'umpire1', 'umpire2', 'tv\_umpire', 'referee', 'reserve\_umpire']].rename(columns={'home\_key\_batsman': 'Player'}) ipl\_data\_bowling = ipl\_data[['season', 'id', 'name', 'short\_name', 'description', 'home\_team', 'away\_team', 'toss\_won', 'decision', '1st\_inning\_score', '2nd\_inning\_score', 'winner', 'result', 'start\_date', 'end\_date', 'venue\_id', 'venue\_name', 'home\_captain', 'away\_captain', 'pom', 'points', 'super\_over', 'home\_overs', 'home\_runs', 'home\_wickets', 'home\_boundaries', 'home\_key\_bowler', 'match\_days', 'umpire1', 'umpire2', 'tv\_umpire', 'referee', 'reserve\_umpire']].rename(columns={'home\_key\_bowler': 'Player'}) ipl\_data\_batting['Runs'] = ipl\_data\_batting['home\_runs'] ipl\_data\_bowling['Wickets'] = ipl\_data\_bowling['home\_wickets'] ipl\_data\_batting['Balls'] = ipl\_data\_batting['home\_overs'] \* 6 ipl\_data\_bowling['Balls'] = ipl\_data\_bowling['home\_overs'] \* 6 Combine the batting and bowling data ipl\_data\_player = pd.concat([ipl\_data\_batting, ipl\_data\_bowling], axis=0) Group by IPL round, player, and player type, and calculate total runs and wickets ipl\_data\_grouped = ipl\_data\_player.groupby(['season', 'Player']).agg({'Runs': 'sum', 'Wickets': 'sum', 'Balls': 'sum'}).reset\_index() kets ipl\_data\_grouped = ipl\_data\_player.groupby(['season', 'Player']).agg({'Runs': 'sum', 'Wickets': 'sum', 'Balls': 'sum'}).reset\_index() Find the top three run-getters and top three wicket-takers in each IPL round top\_run\_getters = ipl\_data\_grouped.groupby('season').apply(lambda x: x.nlargest(3, 'Runs')).reset\_index(drop=True) top\_wicket\_takers = ipl\_data\_grouped.groupby('season').apply(lambda x: x.nlargest(3, 'Wickets')).reset\_index(drop=True) Print the results print("Top Run-Getters in Each IPL Round:") print(top\_run\_getters) print("\nTop Wicket-Takers in Each IPL Round:") print(top\_wicket\_takers) import scipy.stats as stats import matplotlib.pyplot as plt Extract the top three batsmen and bowlers in the last three IPL tournaments print("Top Run-Getters in Each IPL Round:") print(top\_run\_getters) print(top\_run\_getters.head(3)) print("\nTop Wicket-Takers in Each IPL Round:") print(top\_wicket\_takers) print(top\_wicket\_takers.head(3)) Calculate the performance metrics (e.g., runs, wickets, boundaries) print(rahul\_tewatia\_data.columns) rahul\_tewatia\_performance = rahul\_tewatia\_data.groupby('season')[['home\_runs', 'home\_wickets', 'home\_boundaries', 'away\_runs', 'away\_wickets', 'away\_boundaries']].sum() Calculate the average performance metrics average\_performance = rahul\_tewatia\_performance.mean() Print the relationship between performance and salary print("Relationship between Rahul Tewatia's performance and salary:") print("Salary: 9 crore") print("Average runs: ", average\_performance['home\_runs'] + average\_performance['away\_runs']) print("Average wickets: ", average\_performance['home\_wickets'] + average\_performance['away\_wickets']) print("Average boundaries: ", average\_performance['home\_boundaries'] + average\_performance['away\_boundaries'])