Methods of Advanced Data Engineering Project: The Great Cricket Conundrum

Presented By

Vaishnav Negi
M.Sc in Data Science
Mat. no 23114574
Idm Id ba98gepe

Introduction

Methods

Results

Discussion/Future Works

The end

Introduction | Some background

- What is **CRICKET** ?
  - A **sports** to some……
  - A **religion** to many……

- Why is it **IMPORTANT**?
  - Cricket is a wildly popular sport, **second only to football** as the most-watched sport on the globe.
  - The global Cricket market size was valued at **USD 289.78** million in 2021 and is expected to expand at a CAGR of 3.59% during the forecast period, reaching USD 358.15 million by 2027.
  - The Board of Council for Cricket in India(BCCI) is the richest cricket board in the world with a staggering net worth of **$2.25 billion**.
  - The **IPL's valuation** jumped about 28% to reach a whopping **$10.7 billion** in 2023 against $8.4 billion in 2022. The total brand value of the IPL system has surged by 433% since its 2008 launch. The aggregated total assets of the Bundes liga as of 30 June 2021 exceeded €3.9 billion.

- How is it **RELEVANT** to a **DATA SCIENCE** student?
  - **Role in modern cricket**: providing teams with valuable insights for strategic decision-making, performance optimization, injury prevention, and talent identification. Cricket has become increasingly data-intensive, and the use of analytics has grown significantly in recent years.
  - **Gold mine of data**: In comparison to other sports like football or basketball, cricket can be considered more data-intensive due to its multifaceted nature. Cricket involves various formats (Test matches, One Day Internationals, T20s), different playing conditions (day-night matches, pitches of varying characteristics), and a wide range of player roles (batsmen, bowlers, all-rounders). This complexity makes cricket an excellent field for data science applications.

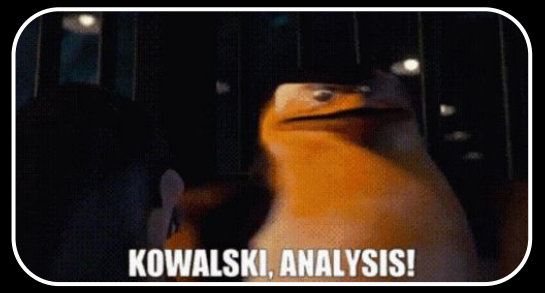
Introduction | Let's get this out of the way

- **What is Cricket? (more theory and less sentimental this time)**
  - Cricket is a bat-and-ball game played between two teams.
  - Each team has 11 players on a circular field with a central rectangular pitch.
- **Objective of the Game**
  - The objective is to score more runs than the opposing team.
  - Runs are scored by hitting the ball and running between wickets.
- **Key Roles in Cricket**
  - Batsmen: Score runs by hitting the ball.
  - Bowlers: Deliver the ball to dismiss batsmen.
  - Fielders: Prevent runs and take catches.
- **Cricket Formats**
  - Test Matches: Longest format, played over a maximum of five days.
  - One Day Internationals (ODIs): Limited to 50 overs per side.
  - Twenty20 (T20): Shortest format, each team bowls and bats for 20 overs.
- **Basic Rules - Batting**
  - Batsmen aim to protect their wickets while scoring runs.
  - Runs scored by running between wickets, boundaries, and sixes.
- **Basic Rules - Bowling**
  - Bowlers aim to dismiss batsmen by hitting the stumps or inducing mistakes.
  - Different types of deliveries: fast, spin, swing.
- **Fielding Rules**
  - Fielders aim to stop runs, take catches, and effect run-outs.
  - Strategic field placements based on the game situation.
- **Dismissals**
  - Batsmen can be dismissed in various ways, e.g., bowled, caught, lbw, runout.
  - Each dismissal has specific rules and conditions.





Introduction | The motivation and the primary question.



- **MOTIVATION:**
  - Cricket's global impact extends beyond the game, influencing cultures and fostering unity.
  - Tradition meets innovation as data analysis reshapes cricket strategy.
- **THE INVESTIGATION:**
  - This report focuses on Ball-by-ball data for International Men's Cricket Matches (2003 - 2023) and Indian Premier League match data (2008 - 2023).
  - It unveils a meticulously crafted data engineering pipeline that downloads, structures, and transforms raw cricketing data into accessible and analyzable CSV files.
- **PRIMARY QUESTION:**
  - Guiding Query: "What methods of dismissal hold significance in both the international format and the IPL? Do trends show similarity, and what can be inferred?"
  - Objective: Unravel dynamics of player dismissals, identify patterns, and discern trends to inform strategies and team compositions.
- **SIGNIFICANCE:**
  - **Strategic Insights:**
    - Player performance evaluation.
    - Tailoring bowling and team strategies.
  - **Informed Decision-Making:**
    - Analysis of opponents.
    - In-game decisions grounded in statistical evidence.
  - **Team Improvement:**
    - Guiding training sessions.
    - Scouting potential talent.
    - Addressing weaknesses for overall team enhancement.

Introduction

Methods

Results

Discussion/Future Works

The end

Methods | The Data Engineering Pipeline

Downloading and Extracting JSON Data

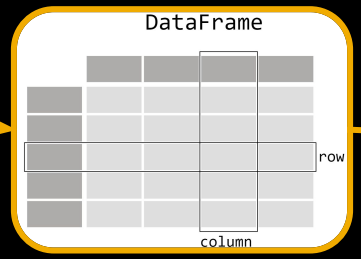
+> Downloads ZIP files containing JSON data for International Twenty over format matches and Indian Premier League matches.
+> It then extracts the contents of the ZIP files into respective directories ‘t20s_male_json/’ and ‘ipl_json/’.

Data Loading Functions

+> Two functions ‘load_data_it20’ and ‘load_data_ipl’ are defined to load JSON files into lists.
+> These functions iterate through JSON files, load each file's data, and append it to a list.
+> Match IDs are extracted from filenames and added to the data.

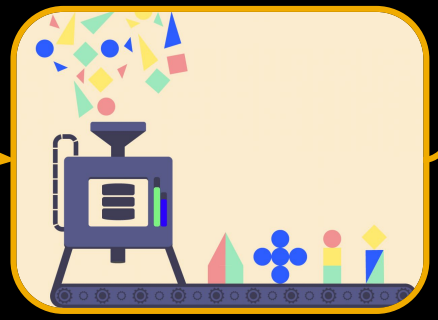
Data Processing

+> ‘create_df’ function processes the loaded data and creates a Pandas DataFrame (DF).
+> The DF includes details like match ID, date, venue, teams, innings details, ball-by-ball information, and outcome.
+> Extracts relevant match statistics from the DF, focusing on completed matches with a winner and target score.



Feature Engineering

+> Calculates additional features such as runs to target, balls remaining, and whether a chase was successful, total batter runs, non-striker runs, and bowler runs conceded.
+> Iterates over rows to calculate cumulative runs and balls faced for batters and non-strikers.
+> Handles cases where a player gets out, updating final stats and resetting runs and balls faced.



Saving Processed Data

+> The processed data for International Twenty over matches and Indian Premier League matches are saved as CSV files (‘it20.csv’ and ‘ipl.csv’).

Methods | JSON Data

The JSON structure provides a detailed representation of a cricket match, including match metadata, outcomes, and ball-by-ball information for each inning, capturing essential details such as runs, wickets, and specific events during each delivery. Below is a breakdown of the JSON file structure, explaining the relevant fields in detail(Some fields are missing for brevity):

- **meta** (*Object*): contains metadata for each cricket match.
- **info** (*Object*):Contains metadata and information about the cricket match. Relevant Subfields:
  - balls_per_over (*Integer*): 6 balls per over.
  - city (*String*): City in which the match took place.
  - **dates** (*Array of Strings*): Contains the date or dates when the cricket match took place. Example: [“2022-01-01”]
  - match_type (*String*): Format of the game. “T20” in our case.
  - outcome (*Object*): Contains details about the match outcome. The winner and the margin of the win.
  - overs (*Integer*): Total overs per inning (20 in our case).
  - players (*Array of objects*): Contains 2 lists of players from each team.
  - venue (*String*): Represents the venue where the match was played.
- **innings** (*Array of Objects*): Contains details about each inning of the cricket match. There are 2 objects, each representing an inning played by one of the 2 teams: Subfields:
  - team (*String*): Represents the team playing the inning.
  - **overs** (*Array of Objects*): Contains information about each over in the inning (< = 20 items of 6 deliveries each). Important Subfields:
    - **deliveries** (*Array of Objects*): Contains information about each delivery in an over. Subfields:
      - **batter** (*String*): Represents the batsman facing the delivery.
      - **non_striker** (*String*): Represents the non-striker batsman.
      - **bowler** (*String*): Represents the bowler delivering the ball.
      - **runs** (*Object*): Contains information about runs scored in the delivery. Subfields:
        - **batter** (*Integer*): Runs scored by the batsman.
        - **extras** (*Integer*): Extra runs scored (e.g., wides, no-balls).
        - **total** (*Integer*): Total runs from the delivery.
      - **wickets** (*Array of Objects*): Contains information about wickets taken in the delivery. Subfields:
        - **kind** (*String*): Type of dismissal (e.g., “bowled,” “caught”).
        - **player_out** (*String*): Batsman who got out.
        - **fielders** (*String*): Name of the fielder contributing in dismissal.

Methods | Resulting CSV file

- **Match ID**: Unique identifier for each cricket match.
- **Date**: Date of the match.
- **Venue**: Venue where the match took place.
- **Bat First**: Team batting first in the innings.
- **Bat Second**: Team batting second in the innings.
- **Innings**: Inning number (1 or 2).
- **Over**: Over number in the innings.
- **Ball**: Ball number in the over.
- **Batter**: Player facing the ball.
- **Non Striker**: Non-striker player.
- **Bowler**: Player bowling the ball.
- **Batter Runs**: Runs scored by the batter from the ball.
- **Extra Runs**: Extra runs scored from the ball (e.g., wides, no-balls).
- **Runs From Ball**: Total runs scored from the ball (including batter runs and extra runs).
- **Ball Rebowled**: Binary indicator (0 or 1) representing whether the ball had to be rebowled.
- **Extra Type**: Type of extra runs (e.g., wides, no-balls).
- **Wicket**: Binary indicator (0 or 1) representing whether a wicket fell in that ball.
- **Method**: Method of dismissal (if a wicket fell).
- **Player Out**: Player who got out (if a wicket fell).
- **Innings Runs**: Total runs scored in the innings.
- **Innings Wickets**: Total wickets fallen in the innings.
- **Target Score**: Target score for the team batting second.
- **Runs to Get**: Runs required to win (relevant for the team batting second).
- **Balls Remaining**: Number of balls remaining in the innings.
- **Winner**: Team that won the match.
- **Chased Successfully**: Binary indicator (0 or 1) representing whether the chasing team won.
- **Total Batter Runs**: Total runs scored by the batter in the match.
- **Total Non Striker Runs**: Total runs scored by the non-striker in the match.
- **Batter Balls Faced**: Number of balls faced by the batter in the match.
- **Non Striker Balls Faced**: Number of balls faced by the non-striker in the match.
- **Player Out Runs**: Runs scored by the player who got out (if a wicket fell).
- **Player Out Balls Faced**: Balls faced by the player who got out (if a wicket fell).
- **Bowler Runs Conceded**: Runs conceded by the bowler in that ball (including batter runs and extra runs).
- **Valid Ball**: Binary indicator (0 or 1) representing whether the ball was valid (not rebowled).

Introduction

Methods

Results

Discussion/Future Works

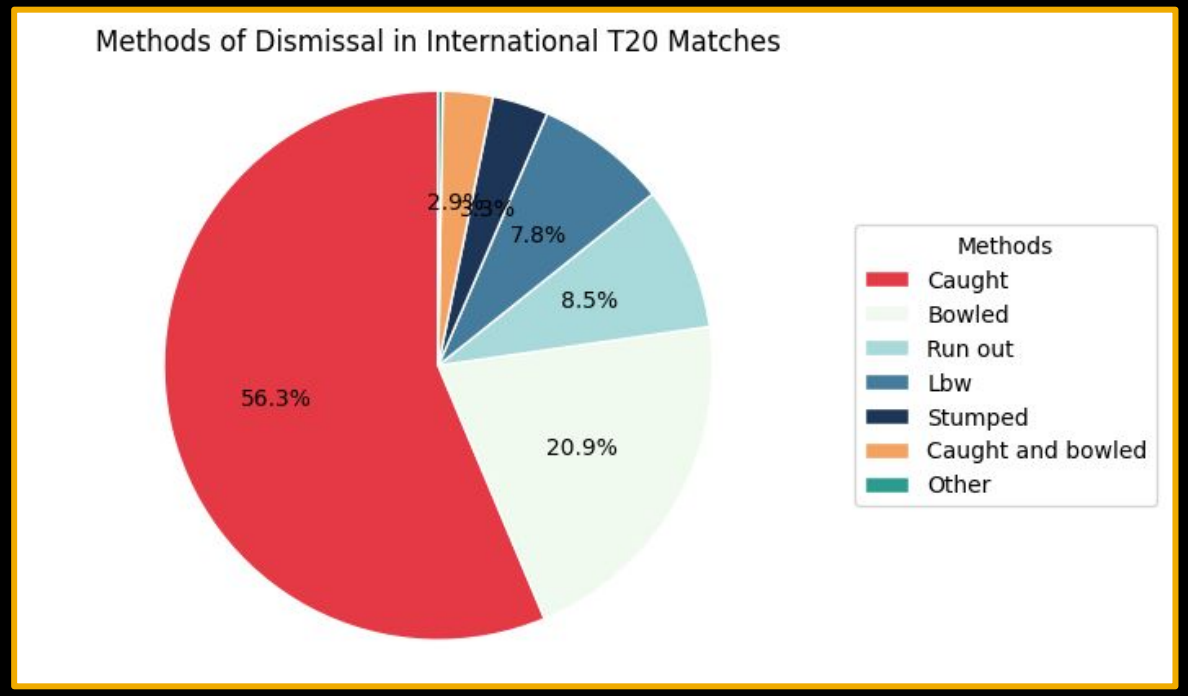
The end

Results | Answering the primary questions:

To answer the primary question, “What methods of dismissal hold significance in both the international format of the game and the league format, specifically the IPL? Do the trends show similarity and what can be inferred from this.”. Let’s look at the primary visualization code.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

t20_data = pd.read_csv('it20.csv')
ipl_data = pd.read_csv('ipl.csv')

filtered_df = t20_data[t20_data['Method'] != 'N/A']
value_counts = filtered_df['Method'].value_counts()
threshold = 50
filtered_value_counts = value_counts[value_counts >= threshold]
other_count = value_counts[value_counts < threshold].sum()
filtered_value_counts['Other'] = other_count
labels = filtered_value_counts.index.str.capitalize()
colors = ['#E63946', '#F1FAEE', '#A8DADC', '#457B9D', '#1D3557', '#F4A261', '#2A9D8F']
fig, ax = plt.subplots()

def autopct_filter(pct):
    return f'{pct:.1f}%' if pct >= 1 else ''

wedges, texts, _ = ax.pie(
    filtered_value_counts,
    labels=None,
    autopct=autopct_filter,
    startangle=90,
    colors=colors,
    wedgeprops={'edgecolor': 'white'},
)

ax.legend(labels, title="Methods", loc="center left", bbox_to_anchor=(1, 0, 0.5, 1))
ax.axis('equal')
plt.title('Methods of Dismissal in International T20 Matches')
plt.show()
```

Results | Visualizations:



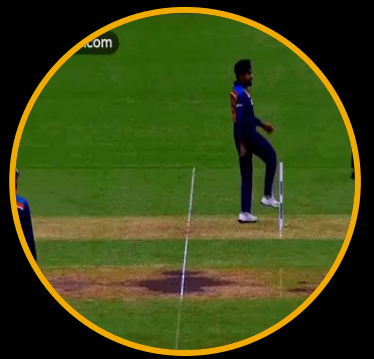
L.B.W.

Stumped

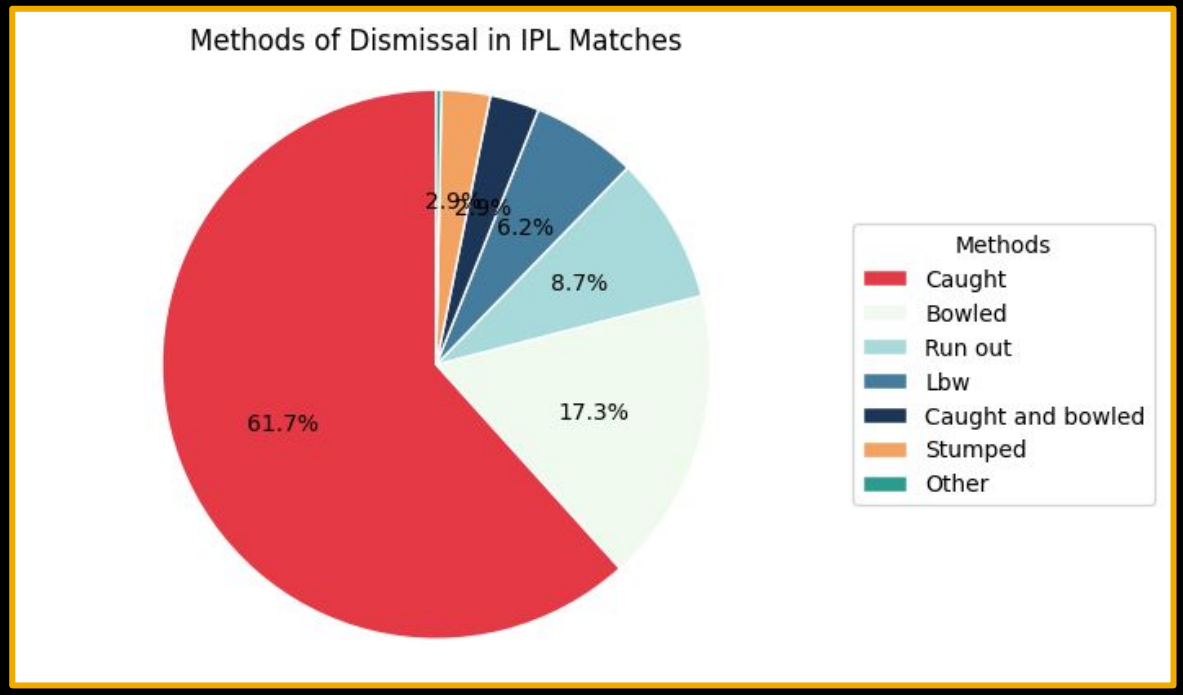
Caught & Bowled

Caught

Bowled



Run-Out



Results | Discussion and Inferences:

	IT20	IPL
Caught	56.3 %	61.7 %
Bowled	20.9 %	17.3 %
Run Out	8.5 %	8.7%
Leg Before Wicket	7.8 %	6.2 %
Stumped	3.3 %	2.9 %
Caught and Bowled	2.9 %	2.9 %

- The commonality in the top dismissal methods between IT20 and league matches underscores the consistency of certain strategies across different formats.
- Teams in both IT20 and league matches should prioritize fielding skills, especially catching, given its dominant role in dismissals.
- Bowlers who excel in getting batsmen out through bowled and LBW dismissals can be considered valuable assets in both formats.
- Strategies around run-outs could be adapted based on the format, with teams in longer format matches having more opportunities to create and execute run-out chances.

Introduction

Methods

Results

Discussion/Future Works

The end

Challenges, Discussions & Future Works

- The structure of the data was notably complex, making it challenging to decipher and extract the relevant information efficiently. The nested nature of the JSON objects required meticulous navigation, resulting in a time-consuming and frustrating process to identify and structure the necessary details for our analysis. These websites helped a lot in visualizing the data hierarchy (site 1 and site 2).
- Moreover, the data processing task involved not only extracting existing information but also calculating additional features to enrich our dataset. This process required a considerable amount of effort, as it involved aggregating and computing various statistics and metrics at the granular level of individual deliveries in cricket matches. These calculations were essential for capturing nuanced aspects of the game, such as total runs scored by players, balls faced, and runs conceded by bowlers.
- The nature of the dataset hindered me from asking more complex questions as that would take up a lot of time and report space for explanation, therefore they are explored a little bit separately, linked at the end.
- Although the data engineering pipeline provides a reliable and efficient means of generating up-to-date datasets for analysis and machine learning tasks, it can be made a lot leaner by shedding unnecessary features and engineering features that better summarize two or more features together.
- Although the data has a lot to provide. This data is still missing a lot of details that allows for finer analysis, like field settings, type of balls, batsmens' handedness, bowlers' handedness, ball and shot placement etc. All this would allow for a much detailed in depth analysis and finding a raw data source to providing these could benefit our analysis exponentially.
- Future work could involve implementing target and chase predictors using machine learning algorithms to enhance strategic decision-making during matches.
- Advanced statistics could be calculated to analyze player performance from different perspectives, contributing to a deeper understanding of player dynamics in the game. (See a bit of this here in the EDA and further analysis python notebook)
  PLEASE CONSIDER VISITING THE LINK !