ODI CRICKET MATCH ANALYSIS FROM 2013 – 2019

Name – Md Sahil

Date – 30/11/2022

Introduction:

This Assignment given by ineuron.ai . The dataset contains data of ODI cricket matches from 2013 to 2019. The dataset is completely raw and uncleaned data. The main focus of this assignment is cleaning the data and preparing the data for analysis.

PHASE 1: Identifying business task

Goal:

The Goal of this analysis is to clean the data and preparing the data for analysis. Finding useful Insights.

Key Business tasks:

As There is no specific business task given by team. I am assuming business task is to analyze Match and players data of Indian Cricket Team to find out effective gaming strategy.

PHASE 2: Preparing the Data

Data Source:

The data is given by ineuron.ai team.

Data details:

There is 3 csv file contains batting information, bowling information and match information respectively.

Following is the description of the dataset.

Batsmen_Data:

1. **Row ID:** contains row number.
2. **Bat1** – Score of the batsmen, DNB means that batsman has not participated in the match
3. **Runs** – Runs made in the match by the batsmen
4. **BF** – Balls Faced
5. **SR** – Strike rate
6. **4s** – No. of 4's
7. **6s** – No of 6's
8. **Opposition** – The opponent team the batsman played with
9. **Batsman** – Name of the batsman
10. **Ground** – The location of the cricket stadium
11. **Start Date** – The date on which the ODI was played
12. **MatchID** – unique id of the match
13. **Player ID** – Unique ID of the player

Bowlers_Data:

1. **Row ID:** contains row number.
2. **Overs –** The Over bowled by the Person
3. **Mdns –** Maiden balls
4. **Runs –** Runs given
5. **Wkts –** Wickets taken
6. **Econ –** Economy
7. **Ave –** Average Ball Speed
8. **SR –** Strike rate
9. **Opposition** – The opponent team the Bowler played with

10. **Bowler** – Name of the Bowler
11. **Ground** – The location of the cricket stadium
12. **Start Date** – The date on which the ODI was played
13. **MatchID** – unique id of the match

ODI_Match_Results

1. **Row ID:** contains row number.
2. **Result –** the result of the match – Won or lost or draw or Not played
3. **Margin –** The margin by which the team has won
4. **BR – Irrelavant column**
5. **Toss –** Information on if the team won or lost the toss
6. **Opposition** – The opponent team the batsman played with
7. **Ground** – The location of the cricket stadium
8. **Start Date** – The date on which the ODI was played
9. **MatchID** – unique id of the match
10. **Country –** The main Country
11. **Country ID –** The Unique id Of the country

PHASE 3: Process and Analyzing the Data

For the cleaning and transforming process I used Excel and SQL in SNOWFLAKE Data Platform.

For Details Data Cleaning Steps Click the following link

https://github.com/mdsahilmca20/iNeuron-assignment/blob/main/PowerBI%20sample%20Assignment/ODI_CRICKET_DATA_CLEANING.sql

The following steps were taken for cleaning purpose:

1	Remove row_id column from csv files in MS-Excel
2	Creating a custom warehouse ”SAHIL_DEMO_WAREHOUSE” in Snowflake
3	Creating Database for this project named “CRICKET_ODI”
4	Creating BATSMEN_DATA table according given data set of batsmen_data
	In given data both match_id and player_id are repeated many times so we can not make them as primary key individually but together match_id and player_id form primary_key
	Load respective CSV file into table
5	In Score column there are many other values rather than any numeric one like ‘DNB’, ‘TDNB', 'sub', 'absent' OR '-'.
	Create a new column PLAYER_STATUS, if score is not any numeric then value will be 'NOT PLAYED' else 'PLAYED'
6	Update column Runs, BALLS_FACED, STRIKE_RATE, FOURS, SIXS. Replace '-' with NULL
7	Trim column OPPOSITION, GROUND, BATSMAN, MATCH_ID, PLAYER_ID
8	Removing v and space from prefix of column OPPOSITION
9	Try to check if length of MATCH_ID is equal or not in the column
10	Now select required data in appropriate format which are required for visualization, and download the result data of the query in csv format.
11	Creating BOWLER_DATA table according given data set of Bowler_data
	In given data both match_id and player_id are repeated many times so we can not make them as primary key individually but together match_id and player_id form primary_key
	Load respective CSV file into table
12	Update column Overs, Maiden_balls, Runs_given, Wickets_taken, Economy, Average_Ball_Speed, Strike_Rate. Replace '-' with NULL
13	Trim column OPPOSITION, GROUND, BOWLER, MATCH_ID, PLAYER_ID
14	Removing v and space from prefix of column OPPOSITION
15	Try to check if length of MATCH_ID is equal or not in the column
16	Now select required data in appropriate format which are required for visualization, and download the result data of the query in csv format.
17	Creating MATCH_RESULT table according given data set of Match_data
	In given data both match_id are repeated many times so we can not make it as primary key but together match_id and country_id form primary_key
	Load respective CSV file into table
18	Make country into uppercase letters.

19	Try to identify distinct results. Results are 'won', 'lost', 'n/r', 'tied', 'aban', 'canc' and '-'.
20	From resulut, where result =’-‘, we can conclude that match is played but we can not predict the result, so here we assume that the match is n/r(no result). Update result '-' to 'n/r'
21	From the data where result='aban', we can not predict anything, that we can replace, so here we assume that the match is not played. Update result 'aban' to 'not played'.
22	From the data where result='canc', we can not predict anything, that we can replace, so here we assume that the match is not played. Update result 'canc' to 'not played'.
23	Trim column OPPOSITION, GROUND, BAT, TOSS, MATCH_ID, COUNTRY, COUNTRY_ID
24	Removing v and space from prefix of column OPPOSITION
25	Try to check if length of MATCH_ID is equal or not in the column. FOUND THAT TWO TYPES OF LENGTH 10,11 EXISTS. Observing records where length of MATCH_ID is 11, found that in suffix an extra 'a' exists. Update the column to remove extra 'a' from from suffix.
26	Try to identify distinct values from TOSS. Observing the result of the match where '-' found in TOSS, results are either 'n/r' or 'NOT PLAYED'. Update column TOSS, replace '-' with NULL.
27	Try to identify distinct values from BAT. Observing the result of the match where '-' found in BAT, results are either 'n/r' or 'NOT PLAYED'. Update column BAT, replace '-' with NULL.
28	Try to identify distinct values from MARGIN. Observing the result of the match where '-' found in MARGIN, results are either 'n/r' or 'NOT PLAYED'. Update column MARGIN, replace '-' with NULL.
29	Now select required data in appropriate format which are required for visualization, and download the result data of the query in csv format.

Insights of Data:

1. Key in 3 tables is totally different. From Batsmen_data and Bowlers_data together match_id and player_id form primary_key, where as in Match_data together match_id and country_id form primary_key.
2. In all 3 tables Not, all batsmen do bowl whereas not all Bowlers do Batting and also not in all matches every player plays. So here making one-to-one, one-to-many or many-to-one cardinality ratio anyhow resulting data loss. Only many-to-

many cardinality ratio exists. So, joining tables results in some inconsistent result. That's why I make many-to-many cardinality ratio in powerBI.

PHASE 4: Visualizations

I used PowerBI to run some further analysis and generate visualizations that support the key findings in the analysis and also used Power Query to further transformation.

For full visualization follow the link:
https://www.novypro.com/project/odi-cricket-match-analysis

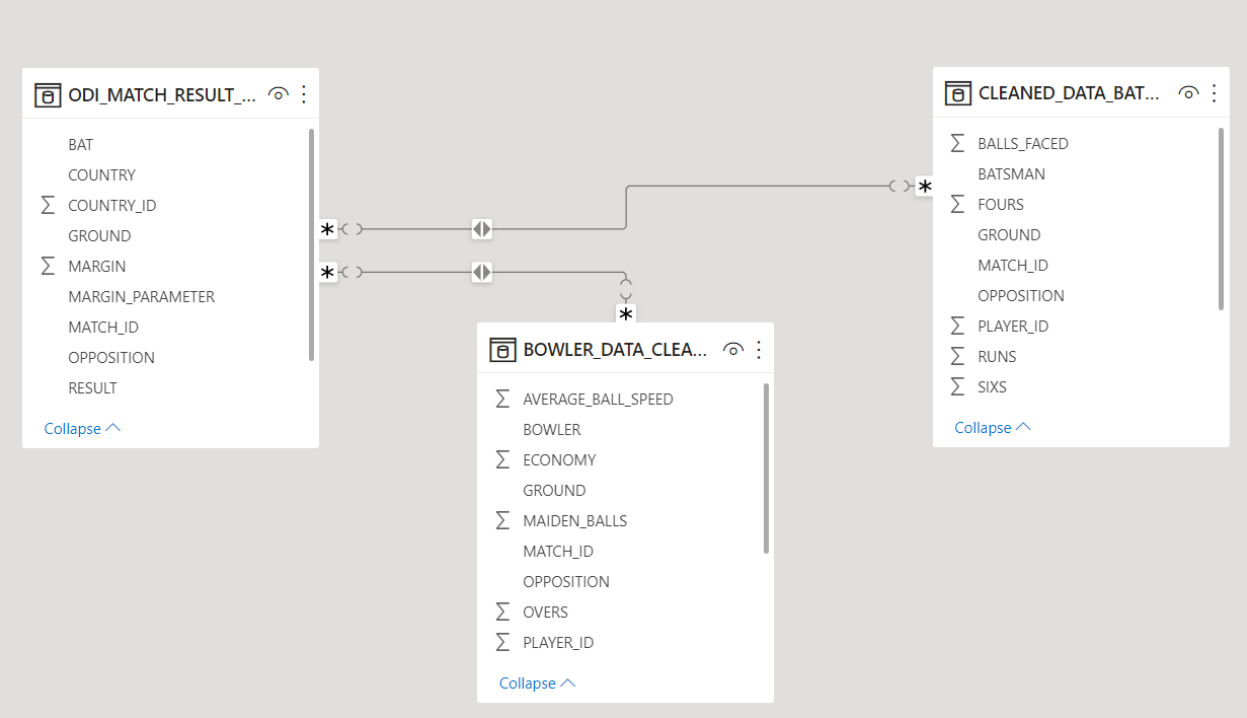
Transformation:

1. Load the cleaned csv files into PowerBI.
2. Click on transform data.
3. Changing Column data types accordingly.
4. Split the Margin column of Match data by space. So Margin Column splits into 2 columns one contains numeric values other contains by runs or wicket. Renamed numeric column as Margin and also renamed other ones as Margin_parameter.

Model:

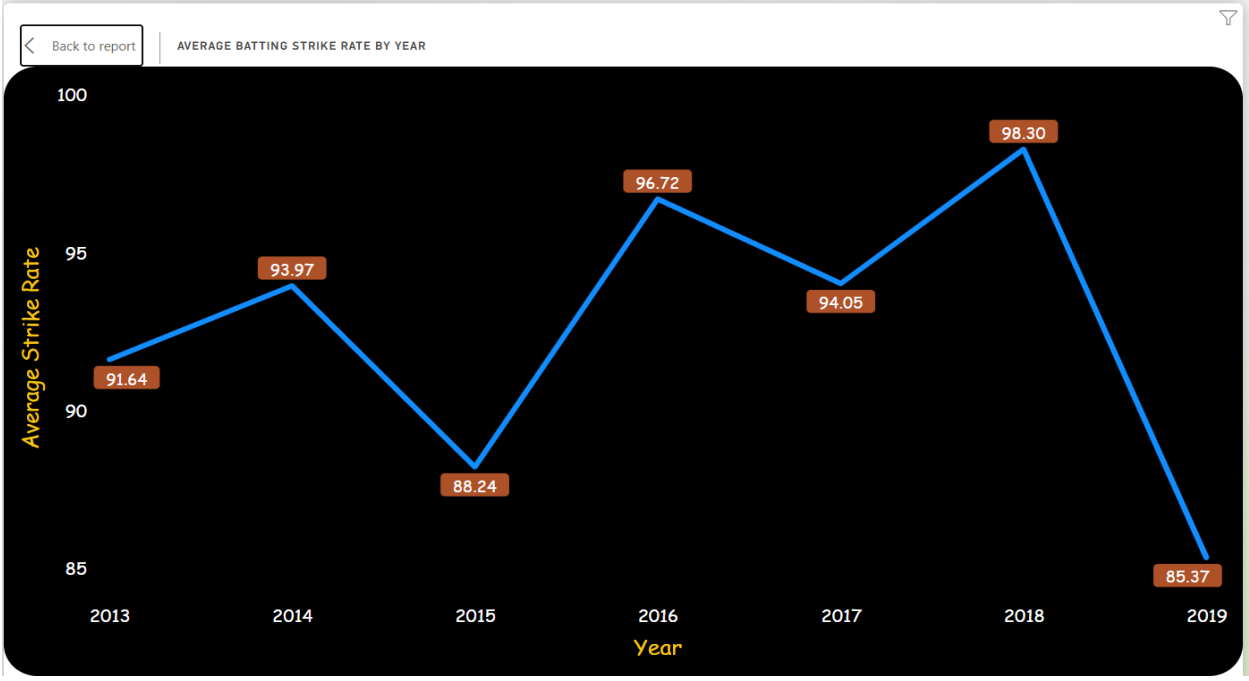
As we previously discussed one-to-one, one-to-many or many-to-one cardinality ratio are not possible for this dataset. Only many-to-many cardinality ratio exists. So create 2 many-to-many relationships one from match_result to batsmen_data and another one is match_result to bowler_data by match_id field.

Models diagram given below:



Finding Key insight for India:

1. Average batting Strike throughout year

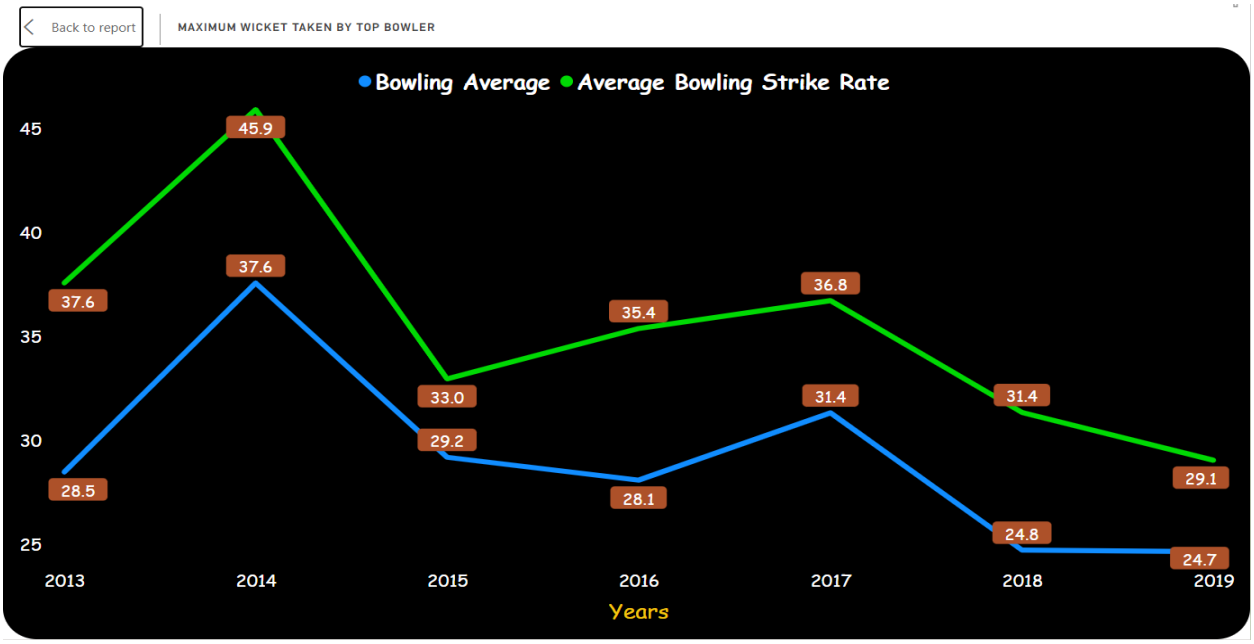


So batting strike rate Average 2013 to 2014 and 2016 to 2018 is good but it is very less in 2019.

Top 2 batsman whose batting strike rate is high are Rohit Sharma and Virat Kohli whereas Rohit Sharma's batting Strike rate is down falling, Virat Kohli's batting strike rate is high and growing.

Whereas Virat Kohli is the most run scorer of ODI match, Rohit Sharma Has high Score of ODI.

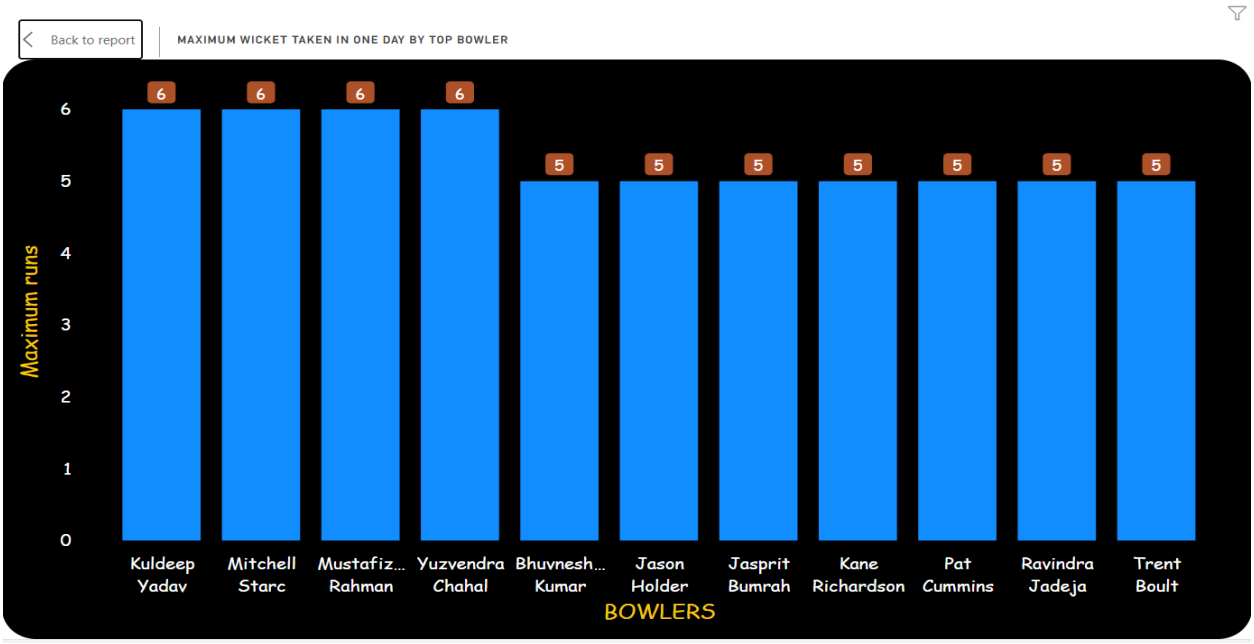
2. Average bowling Strike and bowling average throughout year



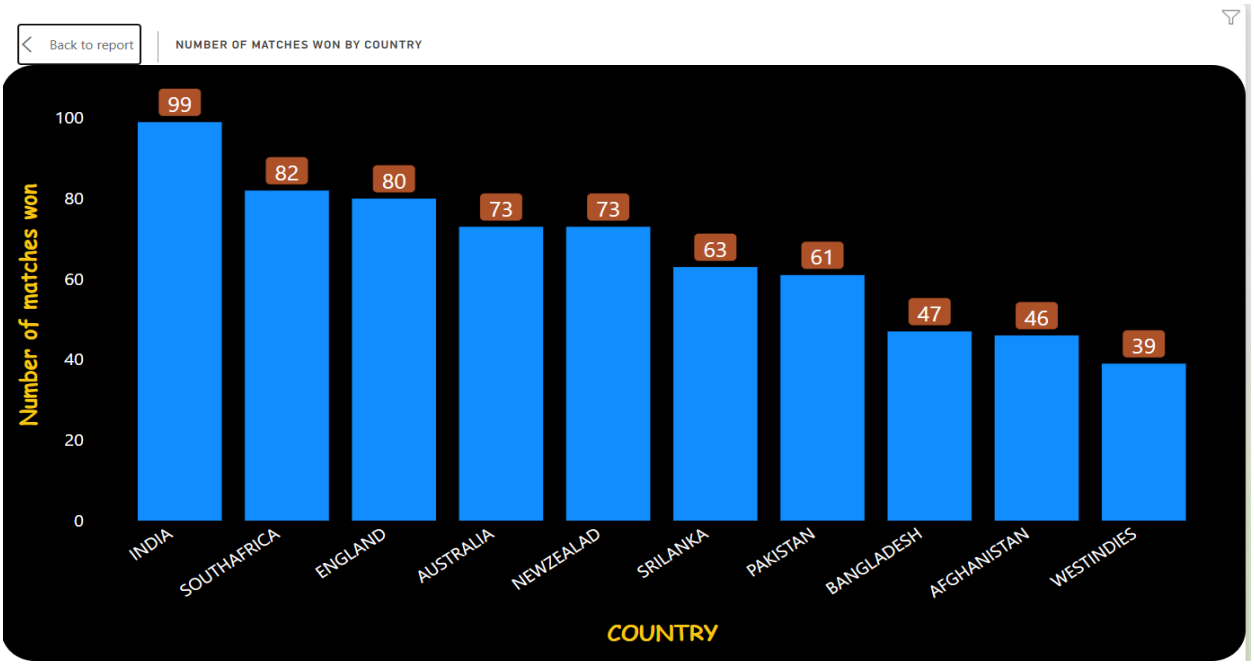
So Both bowling average and average Bowling Strike rate is down falling for India.

Most Wicket taken in throughout ODI matches are

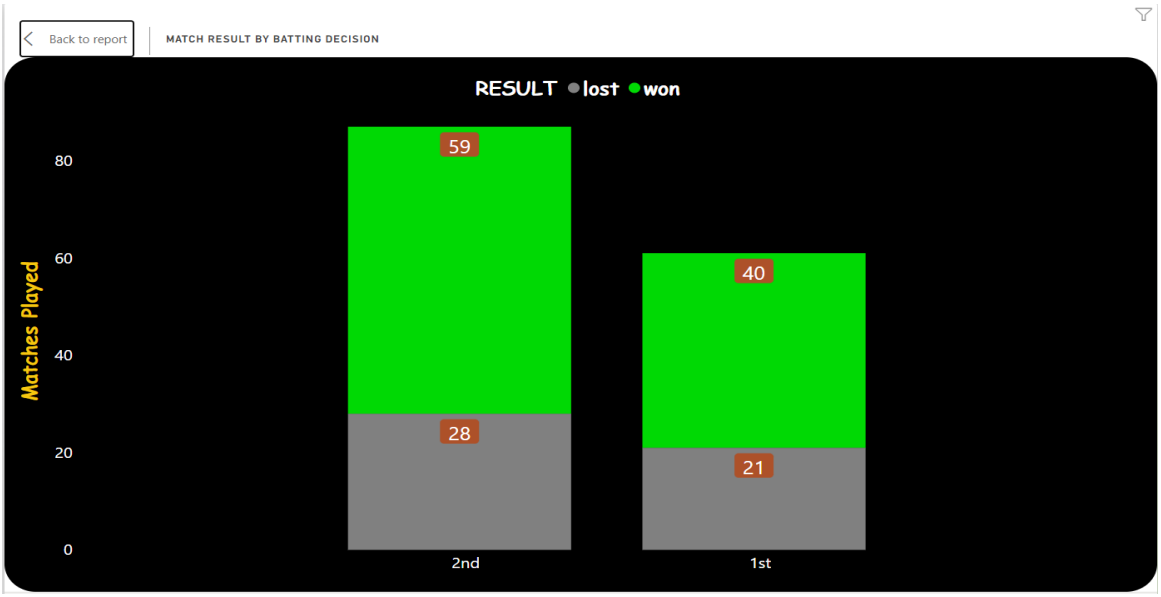
And maximum wicket taken in one day matches (including opposition team member also) are: Kuldeep Yadav and Yuzvendra Cahal.



3. Match Analysis



Although India has won maximum matches, still we try to find out relation between won and lost matches.



So we can say that in maximum cases where India is going for bat in second places India has won.

PHASE 5: Act

My top 4 key findings are:

1. Top batsman (in term of Average Strike Rate and Most run Scored) in ODI is Virat Kohli and Rohit Sharma.
2. India should be more emphasizing in bowling as Both bowling average and average Bowling Strike rate is down falling for India throughout year.
3. Top bowlers (in term of Average Strike Rate, Bowling average and Most Wicket taken) in ODI is Bhuvneshwar Kumar, Ravindra Jadeja, Kuldeep Yadav and Yuzvendra Cahal.
4. India should try to go for batting in 2nd places.