Cricket Analytics: Performance Evaluation of Players in Indian Premier League

Bhavik Nikulbhai Patel.
Data Analytics
San Jose State University
San Jose, California
bhavik.n.patel@sjsu.edu

Sourab Saklecha
Data Analytics
San Jose State University
San Jose, California
sourabrajendra.saklecha@sjsu.edu

Smeet Sheth
Data Analytics
San Jose State University
San Jose, California
Smeetpiyush.sheth@sjsu.edu

Swetha Neha Kutty Sivakumar
Data Analytics
San Jose State University
San Jose, California
swetaneha.kutty.sivakumar@sjsu.edu

Kashish Thakur
Data Analytics
San Jose State University
San Jose, California
kashish.thakur@sjsu

***Abstract*— With the advent of the fast-paced T20 format in cricket and the financially lucrative business model that it brings along, analytics in cricket is now more important than ever. When a single ball, a run or even an extra conceded can prove to be the difference between the winner and the loser--refined, deep and real-time analytics is priceless and the need-of-hour. Analytics is not just used during a match or the tournament, but also right from scouting for players to building the squad during an auction. Big-pocket franchises like the Rajasthan Royals and Perth Scorchers now carry around a full-fledged team of analysts who work round the year in preparation for the lucrative two-month summer season tournament-further showcasing the importance of data analytics in the sport. With this project, we intend to demonstrate how refined analytics that goes beyond just conventional strike rates and averages, can help teams make more sound off-the-field decisions that can give them an edge in a player vs player on-field battle. We are using data from all Indian Premier League matches played from its inception in 2008 until the 2022 season. This project we have analyzed individual player performance based on conditions like pitch type, stadium dimensions, opposition player, bowler type, ball type in specific match scenarios. This analysis can be used further to decide which player to consider in critical match conditions in order to win the match.**

Keywords—Data Analysis, NoSQL Database, Data Visualizations, Cricket Analytics

I. Introduction

Cricket is one of the most popular games in the world played around 106-member states, which has 1.5 billion worldwide fans according to Indian Cricket Council(ICC). Currently there are three major formats, One Day International, T20 Cricket and Test Matches. With the advent of IPL in 2008, its popularity has increased even more and it has now become a billion-dollar industry.[5] Our study aim to analyze individual player performance at each ball for which different parameter are considered and utilized like Bowler Type, the Stadium capacity, pitch type, Ball type, Opposition player. All these parameters are used to see how an individual plays for a particular ball, the historical data has been collected from multiple sources to provide the best results for the analysis. The dataset mainly includes IPL data from all the matches played from 2008 to 2022 season. The dataset consists of 23 different tables and more than 2,25,000 records.

The analysis is conducted on both SQL and NoSQL databases which can help us to perform a comparative study for the results achieved. For the SQL database we have used MYSQL server and for NOSQL database we have utilized MongoDB. MongoDB is a widely used NOSQL database. It is an open source database which is document oriented and is highly compatible with BSON(Binary JSON) file type. A document in it can be any foundational data type like date, strings, array or even an embedded document. Even though the mongoDB platform does not provide joins like RDBMS it provided improved latency through the embedding of the document.[7]. We have also utilized an ETL tool which is Amazon Redshift, which is a columnar database that has been completely set up and along with the execution of SQL queries and visualization in Power BI(or Tableau).

II. Motivation

The current approach in cricket analytics is based on manual calculations and derives very few insights i.e., average, strike rate, economy, etc. We are trying to develop a model by using the data of the IPL tournament of the last 15 years. With the help of this data, we will try to predict the current performance of the player based on its records. We are trying to gain insights into how an individual player performs against a particular opposing team under given conditions. We are trying to get an end-to-end result by only inserting values in the fields and then getting the player's performance analytics based on the given conditions. We will then match the current performance with the expected performance and measure the accuracy of the model. Coaches and team managers can use that insight before the game and make necessary adjustments for the improvement in the performance of their team. Running SQL queries on the big data every time to gain insights can be tiresome and time-consuming. Instead of that we are trying to develop a real-time performance model which uses past data and gives the information needed.

III. Literature Review

Mazhar Awan *et al* conducted a study to investigate the use of big data analytics in the field of cricket. The aim is to utilize the technology for the prediction of a team's scores with the help of the Sparks ML framework. The dataset used in the research is mainly focused on one of the cricket formats which is one-day international (ODI) cricket and consists of ODI match data from 2006-2017. A prediction model is used which will predict the match's winner. They constructed two separate models and selected the best model based on the accuracy of the results. The prediction results are quantified and analyzed using Linear regression in Spark ML with multiple metrics like MSE (mean square error), MAE (mean absolute error), and Accuracy and they have obtained 95% accuracy with 1350.34 MSE and 28.2 MAE. [1]

Dibyojyoti Bhattacharjee *et al* have focused on the use of statistical analysis & data mining in Cricket. It utilizes analytical and data mining tools to measure the performance of cricketers. These methods can also be used in other aspects of the game like player selection and evaluation, impact prediction based on home advantage, toss's decision, and match's target reset in case of rain interruption. It emphasizes the use of techniques like Bayesian network, neural network, and regression which can be applied to

the game of cricket in form of measure of performance, pattern detection, and predicting the outcome.[2]

Wei yin *et al* have proposed a comprehensive estimation tool to evaluate the player's overall performance in cricket. Three different estimations are made to check the efficiency of the player in the different formats of the game (For batting, bowling, and fielding). They have used the dataset of international cricketers from 1877-2019 to find the most efficient player (batter, bowler, fielder) for all the formats of the game. The model used is the DEA Super SBM model to evaluate the efficiency of the player. The indices used by the study produced more clear and more accurate results when compared with the traditional parameter.[3]

Anurag Sinha *et.al* have studied the key factors which influence the match prediction for which they have used a regression model that works best for the dataset considered and provides the best prediction results. They have built an ML prediction approach known as IPL match predictor which works on the data sets and the previous stats by training it in different dimensions. The dimensions considered include the toss results, Captains, Opposition Team, Home Ground etc. Each of these factors have shown varying strengths which is determined using KNIME TOOL and added intelligence of Naive Bayes Algorithm and using Euler Strength Calculation Formula. Their study aims to improve the team performance and increase the team's chance of winning.[4]

Vistro *et.al* have conducted a research on the prediction of the match winner before it begins based on the different selected features like batsman's performance, team strengths , venues etc. They have built a model by applying different machine learning algorithms like Random Forest, SVM, Naive Bayes and Decision Trees. For their study they have considered SEMMA methodology in order to analyze the IPL T20 match winner dataset. They have achieved the maximum accuracy of 94.23% for XGBoost machine learning algorithm without the parameter tuning whereas for the decision learning approach they initially achieved an accuracy of 76.9%, in order to improve the performance they have fine tuned the parameters which then enhanced the accuracy to 94.87%. The prediction model devised will benefit the cricketing boards while evaluating the team's strengths, as well as cricket analysis.[5]

Bedekar *et.al* have devised a novel paradigm of research for cricket analytics which is termed as Timing Index. This index is based on real life IOT angle, max bat speed and impact bat speed. It provides a holistic view of the player's shots and his playing abilities and is further divided into five categories like Early, Late, Missed, Timed, Well-Left that helps IOT sensors. These sensors are fed to a multiple class label classifier implementing different machine learning algorithms like k-nearest neighbors, decision trees and ANN. The timing index devised can be very important to a batter who aims to improve their footwork or their playing technique while timing their middle and time certain deliveries. They aim to incorporate varying pitch lengths, spinners, types of balls, and check if substantial factors can influence the timing of the player. [6]

Gupta *et.al* have performed an experiment to investigate the query performance of MongoDB using IPL dataset. Their methodology included the query formulation and empirical study in which they conducted analysis of the results achieved. The analysis indicated that certain parameters like winning a toss, playing at home grounds and application of Duckworth Lewis Rule has played an important role while determining which team will be the one which wins the match. [7]

IV. METHODOLOGY

For the implementation of the whole project, we have followed Agile Methodology using Scrum. The Scrum board is created using Azure Dev-Ops. The link to the Agile Project and the Scrum Board.

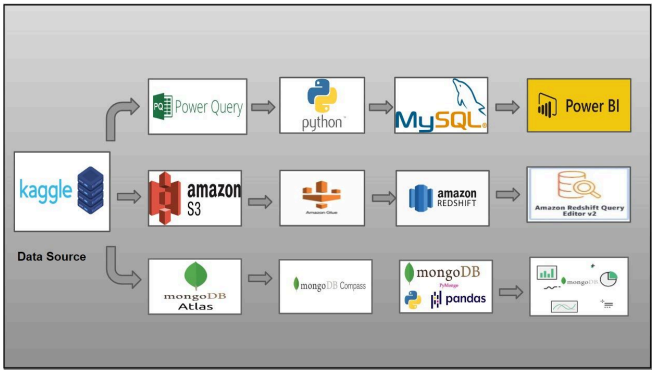


Fig. 1. Project Flow

The above diagram represents the project approach we have adopted. There are four major phases starting with the Data Collection & cleansing phase, the Data Analysis phase, the Result Analysis Phase, and Deployment phase. Below are the tools we have worked on for this project.

A. Data Collection & Cleaning Phase

a) Data Sources

The initial dataset is taken from Kaggle. The remaining data is collected from different sources on the web to import information into 23 different tables used for the analysis. The data present in a few tables is also derived data which is generated through Stored Procedures. We have created three different Stored Procedures, Master_Update, update_player_stats, and update_team_stats.

b) Data Transformation

The initial dataset from Kaggle contained only the two fact tables with transactional data for matches and ball_by-ball. All the other twenty-one dimension tables had to be built around them using the unique values that were present in the fact tables. We also had to fill in the NULL values from open sources and ensure that the data was consistent. Data that was misspelled that to be corrected. Normalization was performed on the table by transposing the column-oriented data (like seen in the seasons table) into a row-oriented table (like seen in seasons_team_history). The Primary Key-Foreign Key relationships had to be established and the string data replaced with its equivalent numerical key to make the queries more efficient.

B. *Data Analysis Phase*

a) AWS Tools workflow

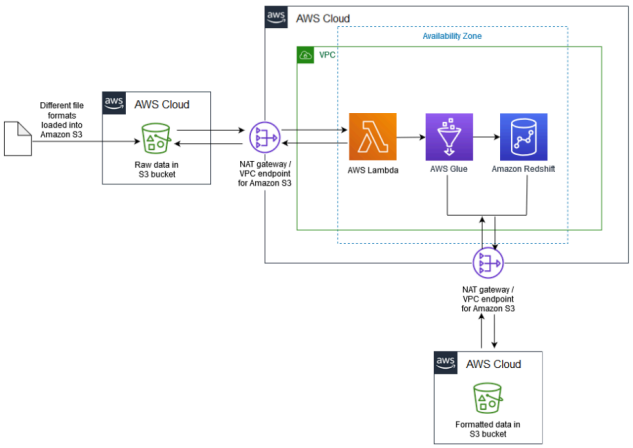


Fig. 2. AWS workflow (Source: AWS Redshift Documentation)

- *S3*

  A user may store and access any volume of data from anywhere on the internet with Amazon S3(Simple Storage Service), a cloud-based object storage service. It is frequently utilized for website hosting, content distribution, data lakes, and big data analytics.

- *Glue*

  Moving data between data stores is simple with the help of the fully-managed extract, transform, and load (ETL) service AWS Glue. It streamlines the procedure for getting data ready and adapted for analytics, machine learning, and other applications. The process of creating and managing data pipelines is automated, and it can scale to handle massive datasets.

- RedShift

  Amazon offers AWS Redshift, a cloud-based data warehousing service. With the aid of SQL-based tools, it is used for cost-effectively storing, querying, and analyzing massive amounts of data. Redshift offers high-performance querying capabilities as well as the ability to scale up to petabytes of data.

- IAM roles

  A user may securely manage access to AWS resources with the aid of the IAM (Identity and Access Management) service from AWS. IAM roles Make sure that only those with permission to access your AWS resources is possible through the creation and management of users, groups, and permissions.

- Cloud watch

  Amazon Web Services offers AWS CloudWatch as a monitoring and logging solution. It is used to gather and monitor metrics, gather and watch over log files, and set resource use alarms. It enables system administrators and developers to learn more about the functionality and performance of their infrastructure and applications.

b) *Working with the SQL Database*

- *MySQL Workbench*

  The CSV files were imported into the schema we created in SQL.

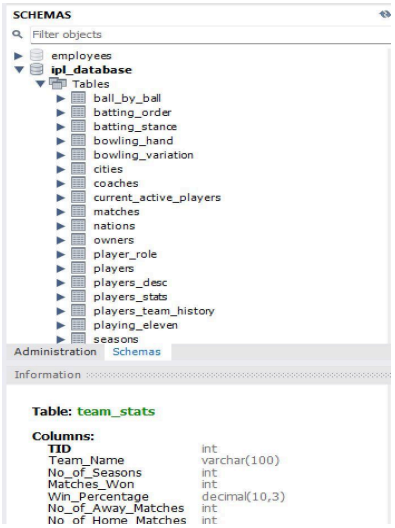


Fig. 3. SQL Tables

- *Stored Procedures*

Due to their precompilation and memory storage, we have used stored procedures to improve database speed by reducing the time required for query optimization and execution.

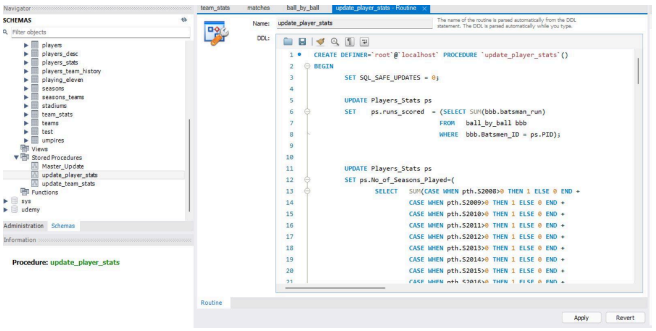


Fig. 4. Three Stored procedures

c) *Working with the NoSQL Database*

- *MongoDB*

  It is an open source database which is document oriented and is highly compatible with BSON(Binary JSON). We are using Aggregation Pipelines to perform analytics

- *PyMongo*

  We have formed the connection between Python and MongoDB using a native Python Driver known as PyMongo with the help of which we can interact with the MongoDB database.

C. *Result Analysis Phase*

a) *Query Execution*

The queries are performed in different platforms and databases like SQL server , MongoDB and Amazon Redshift. The query results are depicted and explained further in the Query Results section further.

D. *Deployment Phase*

a) *AWS Cloud Setup*

We have setup the AWS Cloud using the following procedure:

- *S3 Bucket Creation*

  We created buckets for each CSV file, we did it because it is easy for crawlers to read and load all the files into the crawler tables.

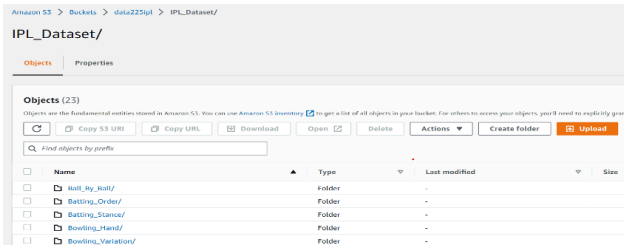


Fig. 5. *S3 Bucket named :IPL Dataset*

- *Import CSV files*

  Created an IPL-Dataset folder and then uploaded all the CSV files into differently assigned sub-folders. Fig 4 displays an example of a bucket called ball_by_ball_new/, which has loaded the CSV file.

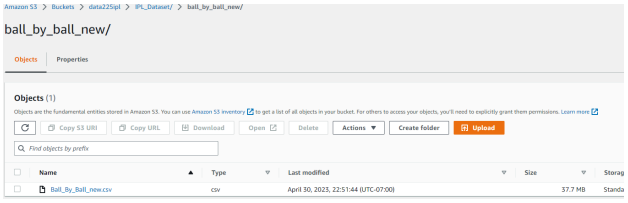


Fig. 6. *Imported CSV File IPL Dataset Bucket*

- *Virtual Private Cloud Creation*

  Creating Virtual Private Cloud (VPC) endpoints for the ETL process of the data present in an S3 bucket.

Fig. 7. Virtual Private Cloud

- *Role Setup and Crawler Creation*

  Details about the crawler and assigned Identity and Access Management (IAM) role. IAM roles were set up to get the permission to access various services of AWS resources. It also provides the logs which can be used to check who accessed the database and what type of actions were performed.

Fig. 8. Crawler Creation with setting of IAM roles

- *Crawler Tables*

  Once all the csv files have been uploaded into S3 we created and ran the crawler. Doing so we got the data catalog with different tables for each of the files in a single run.

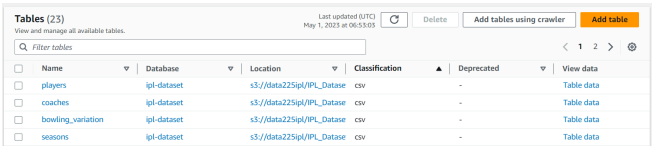


Fig. 9. Crawler Tables

- *Job Creation*

  Creating jobs for all the tables in the AWS glue where the AWS glue data catalog is the data source and the data target is the Redshift database. The Amazon redshift connection is used. Fig 8 shows an example of a job creation. Total 23 jobs are created for each CSV file, which is shown in Fig.9.

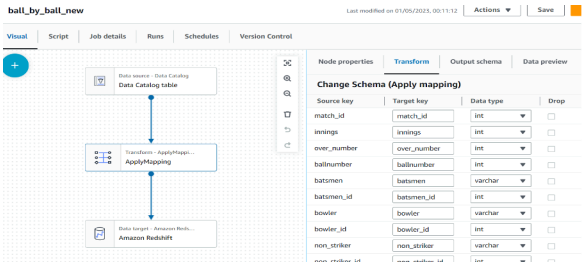


Fig. 10. Verifying the details before the Job is created

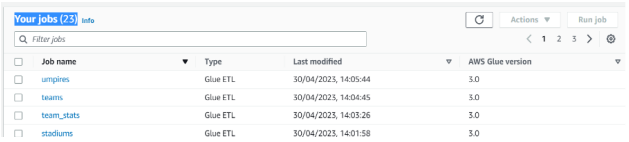


Fig. 11. Total 23 different jobs are created

- *Cluster Creation*

  Created a Redshift cluster to store the transformed data. Connected to the database. Performed complex queries for analysis.

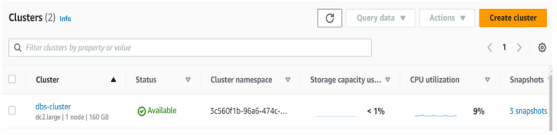


Fig. 12. Creating Cluster for running queries

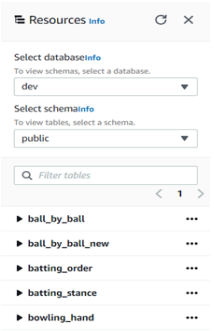


Fig. 13. Resources in our Redshift Database

b) *Visualization Dashboard*

- *PowerBI*

  Created aesthetically appealing visuals to compare various attributes in order to conclude some data driven decisions that can be beneficial to the overall team performance as well as management.

- *MongoDB Atlas*

Created a visualization dashboard in MongoDB Atlas using its charts feature to provide a comprehensive view of the analysis results.

V. ER DIAGRAM

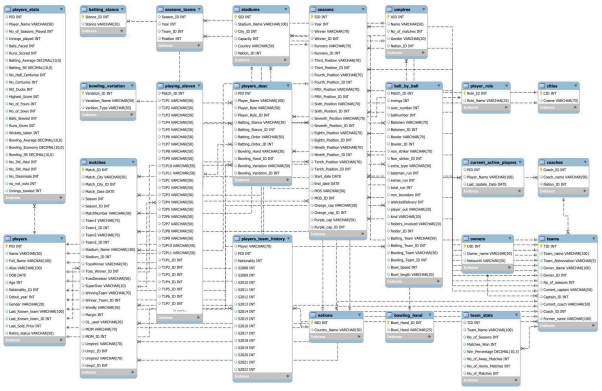


Fig. 14. ER Diagram for the Database Setup

Original Image is provided in the following link : [Project ER Diagram]

VI. QUERY RESULTS

A. Analyzing with AWS

a) AWS Query 1

Query Analysis: The query gives the number of times the bowler has taken the wicket of the same batsman. Based on this analysis the batsman can figure out how to play against that particular bowler.

```
SELECT Batsmen, Bowler, COUNT(*) AS Count FROM ball_by_ball_new WHERE isWicketDelivery = 1 GROUP BY Batsmen, Bowler ORDER BY Count DESC;
```

Result 1 (100)

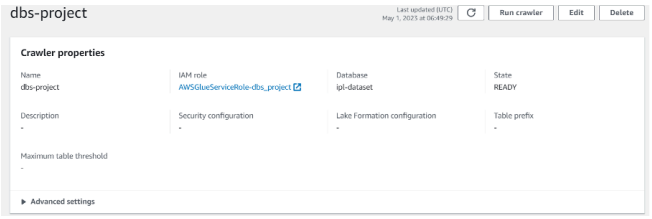
batsmen	bowler	count
RG Sharma	SP Narine	8
V Kohli	Sandeep Sharma	7
PA Patel	B Kumar	7
MS Dhoni	PP Ojha	7
MS Dhoni	Z Khan	7
RV Uthappa	R Ashwin	7

Fig. 15. Query Results for AWS Query 1

b) AWS Query 2

Query Analysis: The following query gives the maximum runs scored by the batsman in 18th, 19th and 20th over combined. From this query the coach can get the idea about which batsman to place next depending on his ability to score the most runs in death overs.

```
SELECT batsmen, SUM(batsman_run) AS total_runs FROM ball_by_ball_new WHERE over_number IN (18, 19, 20) AND bowler IN ( SELECT DISTINCT bowler FROM ball_by_ball_new WHERE over_number IN (18, 19, 20) AND CAST(bowl_speed AS FLOAT) > 140) GROUP BY batsmen ORDER BY total_runs DESC LIMIT 5;
```



Result 1 (5)

batsmen	total_runs
MS Dhoni	569
AB de Villiers	367
RA Jadeja	272
KA Pollard	269
HH Pandya	242

Fig. 16. Query Results forAWS Query 2

c) AWS Query 3

Query Analysis:The query gives the count of all the teams who had won the toss and ended up winning the match and also the count of all the teams who had lost the toss and still ended up winning the match. This gives us the correlation between the toss factor and the match results.

```
SELECT CASE WHEN Team1 = tossWinner AND team1 = winningteam THEN team1 WHEN team2 = tosswinner AND team2 = winningteam THEN team2 END AS Toss_Winning_ and match_ winning_ Team, CASE WHEN team1 = tosswinner AND team1 <> winningteam THEN team1 WHEN team2 = tosswinner AND team2 <> winningteam THEN team2 END AS Toss Losing and match winning Team, COUNT (*) AS Count_Of_Teams FROM matches WHERE tosswinner = winningteam OR tosswinner <> winningteam GROUP BY Toss Winning_and match_winning_ Team, Toss_Losing_and match_winning_Team;
```

count_of_teams	toss_winning_and_match_winning_team	toss_losing_and_match_winning_team
70	Mumbai Indians	NULL
64	Kolkata Knight Riders	NULL
41	NULL	Chennai Super Kings
4	Lucknow Super Giants	NULL
56	Delhi Capitals	NULL
3	NULL	Lucknow Super Giants
34	Sunrisers Hyderabad	NULL
40	Punjab Kings	NULL
54	NULL	Punjab Kings
8	Rising Pune Supergiants	NULL
5	NULL	Gujarat Lions
5	NULL	Rising Pune Supergiants
10	NULL	NULL
49	Rajasthan Royals	NULL
7	Gujarat Titans	NULL
3	NULL	Gujarat Titans
53	NULL	Mumbai Indians

Fig. 17. Query Result for the AWS Query 3

B. Analyzing with MySQL

a) MySQL Query 1

Query Analysis: This query gives the list of all the stadiums and the leading scorer in that particular stadium. The output is then ordered by the total number of runs in descending order.

```
SELECT Stadium_Name,Batsmen,Total_Runs FROM (SELECT stadium_name, Stadium_Name, Batsmen, Total_Runs, Rank() OVER(PARTITION BY Stadium_Name ORDER BY Stadium_Name,Total_Runs DESC) ranking FROM
```

(SELECT stadium_name, batsmen, SUM(batsman_run) Total_Runs FROM ball_by_ball b INNER JOIN matches m ON b.Match_ID = m.Match_ID
GROUP BY batsmen, stadium_name
ORDER BY stadium_name, Total_Runs DESC) runs_ranking) top_batsmen
WHERE ranking=1
ORDER BY Total_Runs DESC

Stadium_Name	Batsmen	Total_Runs
M Chinnaswamy Stadium	V Kohli	2346
Wankhede Stadium	RG Sharma	1840
Rajiv Gandhi International Stadium	DA Warner	1602
MA Chidambaram Stadium	SK Raina	1506
Eden Gardens	G Gambhir	1407
Sawai Mansingh Stadium	AM Rahane	1100
Punjab Cricket Association IS Bindra Stadium	SE Marsh	1064
Feroz Shah Kotla	V Sehwag	933
Dubai International Cricket Stadium	KL Rahul	612
Sheikh Zayed Stadium	SA Yadav	500
Maharashtra Cricket Association Stadium	SPD Smith	469
Subrata Roy Sahara Stadium	RV Uthappa	418
Dr DY Patil Sports Academy	RV Uthappa	359
Himachal Pradesh Cricket Association Stadium	SE Marsh	334
Saurashtra Cricket Association Stadium	SK Raina	328

Fig. 18. Query Results SQL Query 1

b) *MySQL Query 2*

Query Analysis: The query gives the top 3 leading wicket taker player in each season of the IPL from 2008-2022 with the count of their wickets. This can be used to analyze the consistency of the players over the years.

SELECT season ,Bowler,Total_Wickets,ranking FROM (SELECT bowler,season,total_Wickets,rank() OVER (PARTITION BY season ORDER BY season , total_Wickets DESC) AS ranking FROM (SELECT b.bowler, season, COUNT(*) total_Wickets FROM ball_by_ball b INNER JOIN matches mON b.match_Id=m.match_ID WHERE isWicketDelivery=1 GROUP BY b.bowler,season ORDER BY total_Wickets DESC , season DESC) AS a) AS b WHERE ranking=1 OR ranking=2 OR ranking=3;

Result Grid | Filter Rows:

season	Bowler	Total_Wickets	ranking
2011	S Aravind	22	2
2011	MM Patel	22	2
2012	M Morkel	30	1
2012	SP Narine	29	2
2012	SL Malinga	25	3
2013	DJ Bravo	34	1
2013	JP Faulkner	33	2
2013	R Vinay Kumar	27	3
2014	MM Sharma	26	1
2014	SP Narine	22	2
2014	B Kumar	21	3
2015	DJ Bravo	28	1
2015	SL Malinga	26	2
2015	A Nehra	25	3
2016	B Kumar	24	1
2016	SR Watson	23	2
2016	YS Chahal	22	3
2017	B Kumar	28	1
2017	JD Unadkat	27	2
2017	JJ Bumrah	23	3
2018	AJ Tye	28	1

Result 12

Fig. 19. Query Results SQL Query 2

c) *MySQL Query 3*

Query Analysis: This query uses Common Table Expression (CTE) to rank the batsmen with the highest runs and then filtering the top 3 batsmen from each season.. It then displays the top three leading run scorers of each season from 2008-2022. It can be used to observe the consistency of the batsmen over the years.

WITH Players_TotalRun AS(SELECT Batsmen, Total_Runs, Season, RANK() OVER (partition by Season ORDER BY Season DESC, Total_Runs DESC) Ranking FROM (SELECT Batsmen, SUM(batsman_run) Total_Runs, season Season FROM ball_by_ball b INNER JOIN matches m ON b.Match_ID = m.Match_ID GROUP BY Batsmen,season ORDER BY Total_Runs) a)

SELECT Batsmen, Total_Runs, Season, Ranking FROM Players_TotalRun WHERE Ranking <4;

Result Grid | Filter Rows: | Exp

Batsmen	Total_Runs	Season	Ranking
SE Marsh	616	2008	1
G Gambhir	534	2008	2
ST Jayasuriya	514	2008	3
ML Hayden	572	2009	1
AC Gilchrist	495	2009	2
AB de Villiers	465	2009	3
SR Tendulkar	618	2010	1
JH Kallis	572	2010	2
SK Raina	528	2010	3
CH Gayle	608	2011	1
V Kohli	557	2011	2
SR Tendulkar	553	2011	3
CH Gayle	733	2012	1
G Gambhir	590	2012	2
S Dhawan	569	2012	3
MEK Hussey	733	2013	1

Fig. 20. Query Results SQL Query 3

d) *MySQL Query 4*

Query Analysis: This query shows that there is a 33% chance of winning the match when choosing to bat first. While on the other side there is a 67% chance that there is a probability of 2/3 the team will win if they choose to field first. This shows that the chasing team has a higher scope of winning the match as compared to defending the total.

SELECT Tossdecesion,CONCAT(FORMAT(COUNT(CASE WHEN Tossdecesion ='BAT' THEN1 ELSE 0 END) /(SELECT COUNT(*) FROM matches WHERE winningteam= tosswinner) * 100,0),'%') Winning_Chances FROM matches WHERE winningteam= tosswinner GROUP BY Tossdecesion;

Result Grid | Filter Rows:

Tossdecesion	Winning_Chances
bat	33%
field	67%

Fig. 21. Query Results SQL Query 4

C. *Analyzing with MongoDB*

a) *MongoDB Query 1*

Query Analysis: This pipeline aggregation joins the 'ball by ball' table with 'matches' table, sort and limit the result data to identify the Top Five Wicket Takers of all Season. This helps to identify the best bowlers across various seasons.

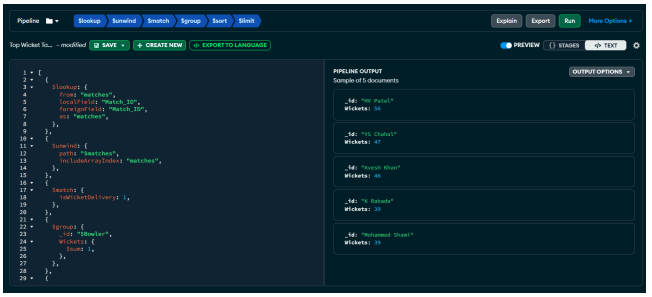


Fig. 22. Query Results MongoDB Query 1

b) *MongoDB Query 2*

Query Analysis:This pipeline aggregation displays the winning team with the highest margin by grouping the winning team and identifying the maximum margin. Knowing this detail shows the team with a great run rate.

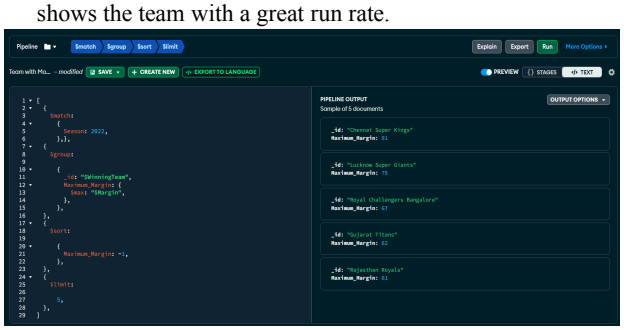


Fig. 23.. Query Results MongoDB Query 2

c) *MongoDB Query 3*

Query Analysis: This aggregation pipeline showcases the impact of bowling length on wickets taken by knowing the cumulative sum of wickets lost by the bowling length. The trainers and bowlers can use this information to determine the pattern for wicket keeping.

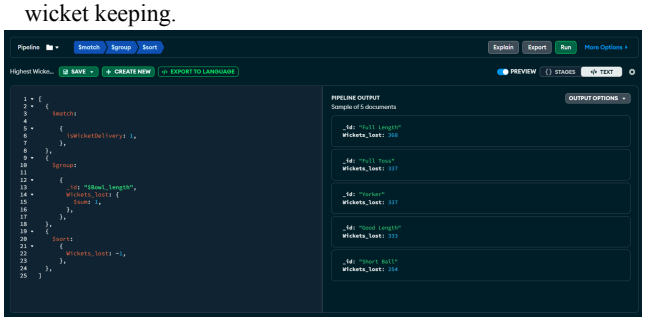


Fig. 24. Query Results MongoDB Query 3

d) *PyMongo Query 1*

Query Analysis: Updating the Batsmen_Order for the Player YBK Jaiswal

```
updatequery = { "Batsmen":"YBK Jaiswal"} newvalues ={
"$set": { "Batsmen_Order" :"Middle Order"} }
x=ball_by_ball_collection.update_many(updatequery,newvalue)
print(x.modified_count,"Documents updated")
```

```
PS C:\Users\swetha\Documents\DB_systems_for_Analytics\Project\MongoDB> python Ipl_pymongo.py
Please Enter the db password:**************
366 Documents updated
PS C:\Users\swetha\Documents\DB_systems_for_Analytics\Project\MongoDB>
```

Fig. 25. Query Results PyMongo Query 1

VII. Data Visualizations

A. *Visualization Dashboard of MongoDB*

Fig. 26. Dashboard displaying the query results on MongoDB Atlas

B. *Visualization Dashboard for Power BI*

a) *Batsman Stats*

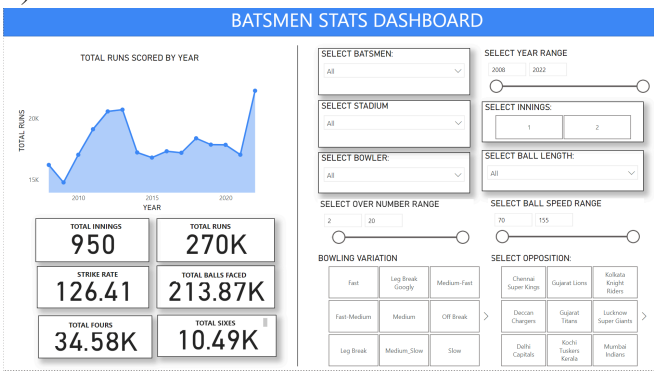


Fig. 23. Dashboard displaying the Batsman Stats

b) *Bowler Stats*

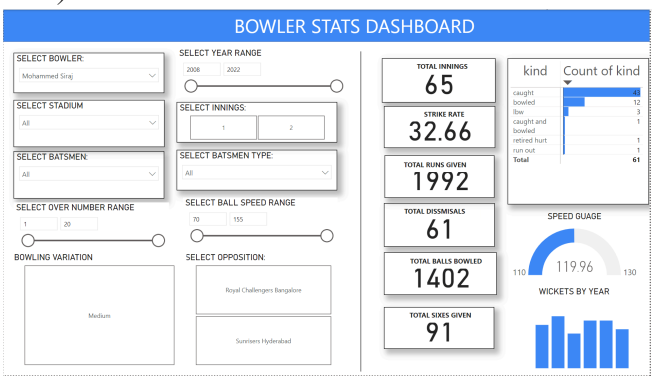


Fig. 27. Dashboard displaying the Bowler Stats

c) *Bowler Speed VS Runs*

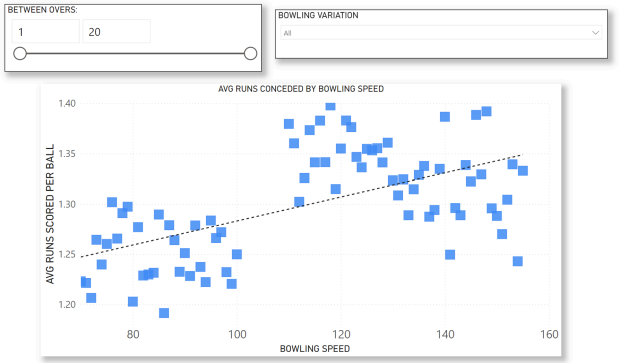


Fig. 28. Visualization of Bowling Speed V/s Runs

VIII. KEY FINDINGS

i) For fast bowler bowling between overs 16-20 , the best length is short ball as it has yielded the least amount(1.24 runs/ball) of runs going against the common notion the best option in the death overs is the Yorker (1.31 runs/ball).

ii) When it comes to bowling speeds across the history of the tournament, we see that slower bowlers (spin and pace-off)-bowling between 80 kmph and 110 kmph have done far better job at containing runs across the 20 overs than the speedsters by giving away lesser runs-per-ball. It can be attributed to the fact that most of the IPL has been played in the subcontinent when the pitches are conducive to spin bowling.

iii) While comparing the impact of bowling length, we can clearly observe that 'Full length' bowling style has the highest wicket losses.

iv) When it comes to debutants-bowlers have done much better and batters. The average and Strike rate of debutant bowlers is much closer to the average of that season. The delta between the average runs scored by debutants and all other batsmen is significantly bigger.

v) In overs 19 and 20 of a run chase- most of the fielding captains have turned to fast bowlers when the runs needed per over is larger than 10 and to spinner when the required run rate is less than 10.

IX. Technical Difficulties

i) In contrast to relational databases, which allow us to link two tables, MongoDb does not allow this. The joining of the saved documents was an intricate task. subsequently is not impossible because we can manually code it, but it took a long time, which ultimately decreased performance.

ii) Tracking players' performances across teams during IPL seasons has been difficult because it requires extensive data analysis. To track the movement of players between teams, we looked at information from a variety of sources, including official IPL records, team websites, and sports websites. The information was then manually entered into the necessary attributes.

iii) During the AWS setup, creating jobs required changing the default data types that the system assigned and even while There were a lot of issues while creating VPC endpoints and assigning file paths while loading data into crawlers from buckets and assigning right IAM roles while running ETL jobs.

X. Team Members And Their Responsibilities

TABLE I. Responsibility Breakdown

ID	Members' Names	Responsibilities
016800550	Bhavik Patel	Project Proposal, Analysis stage 1, Evaluation of Results, Deployment phase(AWS), Final Report
016919383	Kashish Thakur	Finding the Dataset, Setting Evaluation Metrics, Setting up the Database in SQL, Final Report, Creating a project Blog
016786133	Smeet Sheth	Understanding Tools, Deployment Phase(AWS),, Preparing Presentation Slides
016605420	Sourab Saklecha	Data cleaning, Creating Intermediate Status Report, Setting up the Database in SQL, Beta Testing
016780712	Swetha Neha Kutty Sivakumar	Data cleaning, Setting up the Database in NoSQL, Final Data Analysis and Data Visualization

XI. Significance To The Real World

The brand value of IPL accounts up to 8.5 billion dollars at the end of the 2022 season. With each year passing the revenue it is generating is tremendous with substantial growth factor.

A. Team Owner

After seeing the detailed analysis of the team performance as well as the player performance the owner can decide :
1. The value of the player.
2. Return on investment of any particular player.

B. Team Coach

The coach can strategically use all the players present in the squad. Categorize them and can use particular players under the given conditions where the player can give the best output. The coach can use all the resources available and use them accordingly wherever and whenever required.

C. Player

Players can get the idea at what point he is lacking or what are the areas that need improvement. Based on the past records, players can analyze and improve their skills so that they can give their best on field and also increase their individual brand value. Improving their performance can help in excelling their career and also lead them to play for a bigger format internationally.With so much stake on board investors, sponsors and franchise owners are willing to bet their money on the best players and team which all comes down to minute analysis performed at the ground level.

D. Fan Engagement

Situations where insights are arduous to understand just by looking at the data, we can run queries those are provided in the report with it's visualization, we can help admirer to understand game and augment their interest and engagement to the game, the more engaged the follower the more is the return on investment for the IPL brand.

XII. Conclusion

This cricket analytics project has shown the effectiveness of a data-driven approach in comprehending and enhancing cricket performance. We have discovered factors that have a substantial impact on individual and team performance by studying a range of cricketing statistics, which has given us invaluable insights into the game of cricket.Our analysis has highlighted the significance of numerous crucial elements, including strike rates, economy rates, ball speed and bowling length. We have also identified additional variables that affect match outcomes, including the opposition,pitch, stadium and team makeup.Coaches, players, and teams can all benefit from the knowledge gained through this project by performing better and getting better outcomes. Cricket

teams may make wise judgments by utilizing the power of data and analytics.

XIII. Key Learnings

- Implementing the course's concepts has involved experimenting with a variety of technologies, including PowerBI, Amazon Redshift and Glue, MongoDB Atlas, and various presentation tools like Prezi and Grammarly, all of which have an impact on our overall development.
- Implementing analytics on SQL,NoSQL and AWS made us understand the flexibility of AWS on computational power as and when required on executing complex queries and optimizing the performance with columnar storage.
- We worked collaboratively as a team from the research to the project's development phase. As a result, we were more productive and imparted knowledge rapidly.
- Following the Agile methodology, we made sure that our project was well-planned and that we were aware of any upcoming challenges and came up with appropriate action items in advance.

XIV. Innovation

Approaching data analytics in cricket by bottom up instead of top down as is usually the case in industries. We approached this analytic task by starting from the fact table with the transaction of each ball and built our way up to the final aggregated stats table. This way analytics should be performed on minute factors like balling position, ball length, ball speed, and pitch condition. We have further provided provision to include women's IPL data which can then be used to compare with the existing data from men's IPL.

XV. Future Scope

i) Considering that we are working on just Male IPL we can include Female IPL, which has recently started from this year. information for analysis.
ii) The research can be conducted on other NOSQL databases as well like Column-Based (Cassandra), key-value pair (Aerospike) and Graph databases (Orient-Base and NEO4J). The analysis performance of all the three databases can be compared. The pros and cons of each can be identified using a comparative study conducted on all the three databases together.
iii) We aim to include Auction data to calculate Return on Investment for each player.
iv) We intend to come up with one composite score which can be derived from multiple metrics that can be used as a one stop performance evaluator by Fantasy Cricket Players.
v) Further improving the performance of queries by using joins in place of sub queries.

XVI. Use Of Grammarly

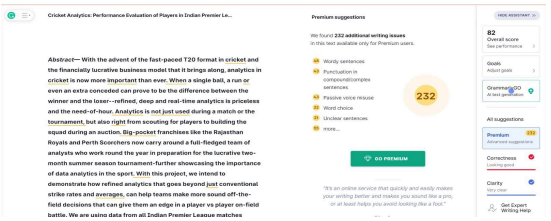


Fig. 26. Generated Grammarly Report

References

[1] Awan, M.J.; Gilani, S.A.H.; Ramzan, H.; Nobanee, H.; Yasin, A.; Zain, A.M.; Javed, R. Cricket Match Analytics Using the Big Data Approach. Electronics 2021, 10, 2350. https://doi.org/10.3390/electronics10192350

[2] Bhattacharjee, Dibyojyoti, et al. Cricket Performance Management: Mathematical Formulation and Analytics. 1 ed., Springer Singapore, 2019, https://link.springer.com/book/10.1007/978-981-15-1354-1 #about-authors. Accessed 8 March 2023.

[3] Yin W, Ye Z, Shah WUH.Indices Development for Player's Performance Evaluation through the Super-SBM Approach in Each Department for All Three Formats of Cricket. Sustainability. 2023; 15(4):3201. https://doi.org/10.3390/su15043201

[4] Sinha, A. Application of Machine Learning in Cricket and Predictive Analytics of IPL 2020. Preprints.org 2020, 2020100436. https://doi.org/10.20944/preprints202010.0436.v1.

[5] Vistro, D.M., Rasheed, F., & David, L.G. (2019). The Cricket Winner Prediction With Application Of Machine Learning And Data Analytics. International Journal of Scientific & Technology Research, 8, 985-990.

[6] Bedekar, M. V., Vishwarupe, V., Joshi, P. M., Pawar, V. & Shingote, P. (2022).Data Analytics in the Game of Cricket: A Novel Paradigm Procedia Computer Science, 204:937–944. doi: 10.1016/j.procs.2022.08.114

[7] K. Chaudhary, M. Gupta and P. kaur, "Analyzing IPL dataset with MongoDB," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019, pp. 212-216,doi:10.1109/CONFLUENCE.2019.8776979.

APPENDIX

CRITERIA	PTS	EXPLANATION
Presentation Skills Includes Time Management	5Pts	
Code Walkthrough	3Pts	The whole code and SQL scripts are uploaded on Github. The link to the Github document is uploaded on Canvas.
Discussion/ Q&A	4Pts	
Demo	5Pts	
Version Control Use of Git / GitHub or equivalent; must be publicly accessible	3Pts	The link to our document is the following: GitHub Document
Significance to the real world	5Pts	Most cricket analytical projects do not go beyond conventional statistical models. This is why this project tries to establish insights based on player-to-player match-ups which are much more useful in real-world scenarios.
Lessons learned Included in the report and presentation? How substantial and unique are they?	5Pts	The key learnings are included in both report and presentation.
Innovation	5Pts	The innovation part is mentioned in the report as a separate section.
Teamwork	5Pts	The complete Team Responsibility breakdown is provided in the report in tabular form
Technical Difficulty	4Pts	Mentioned in the report as a separate section.
Practiced Pair Programming?	2Pts	The whole team has worked on the project both physically and virtually using Zoom meetings. The MOM for all the virtual meetings is provided on Canvas.
Practiced agile / scrum (1-week sprints)? Submit evidence on Canvas - meeting minutes, other artifacts	3Pts	Working for the project is done in sprints, the tasks and issues are created using Azure Devops and the Scrum Boards are created for the project. The link to the Project in Azure Devops is mentioned in the Methodology Section.
Used Grammarly / other tools for language? Grammarly free version is sufficient; can use other tools as well. Submit report screenshot on Canvas.	2Pts	Grammarly Report has been generated for the Project Report and has been uploaded on canvas
Slides	5Pts	The presentation slides are submitted on canvas
Report Format, completeness, language, plagiarism, whether turnItIn could process it (no unnecessary screenshots), etc	5Pts	Report is created in IEEE format with the help of Google Docs and is provided as well on the Canvas.
Used unique tools E.g.: LaTeX for writing reports (submit .tex that is not generated from another format such as .docx; generating from .lyx and similar LaTeX editor outputs is fine.	5Pts	Report is created in IEEE format with the help of Google Docs and the presentation slides are very Interactive and visually pleasing which are made using the Prezi tool.

Also checkout https://www.overleaf.com/LinksLinksLinks Links to an external site. to an external site. to an external site. to an external site.) Unique features of Prezi or powerpoint, etc		
Performed substantial analysis using database techniques Project must include an analytics component	3Pts	Aggregation Pipelines used for analytics in MongodB Compass. We have used Windows functions for ranking and aggregation in MYSQL. Visualizations are created using Power BI and MongoDB Atlas.
Used a new database or data warehouse tool not covered in the HW or class	3Pts	Amazon S3 as a data lake, AWS Glue is used to perform ETL, for warehouse AWS Redshift is used where analytics is done.
Used appropriate data modeling techniques	5Pts	For the data modeling we have used a combination of MySQL reverse engineering(ER diagram -the link of which is mentioned in the report) and microsoft powerpoint features to create diagrams for the report..
Used ETL Tool	1Pts	We have used Amazon Glue as the ETL tool for the project.
Demonstrated how Analytics support business decisions	3Pts	An interactive Dashboard in Power BI has been created which gives the analysis results, we can observe the player(Batsman, Bowler) and Team performance analysis across different seasons which can aid in decision making of which player to consider for playing in different match conditions.
Used RDBMS Idea is to exercise as many topics from the course as possible	1Pts	For the Aggregation and Ranking we have used the Windows function in MYSQL, for populating the database for derived data we have created different Stored Procedures, Common table expressions and sub queries.
Used Data Warehouse Idea is to exercise as many topics from the course as possible	1Pts	Amazon Redshift is our cloud data warehouse to run SQL queries and run additional analysis.
Includes DB Connectivity / API calls Possibly using Python	1Pts	A connection has been established between Python and MongoDB using Pymongo for running the queries in Python. The screenshots are provided in the Query results section.
Used NOSQL	1Pts	We have performed the analysis of the queries using MongoDB Compass (Aggregation Pipeline) along with the creation of Visualization Dashboard in MongoDB Atlas
Total Points	85Pts	