PAPER • OPEN ACCESS

Analysis and Predicting Results of IPL T20 Matches

To cite this article: Praveen Banasode *et al* 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1065** 012040

View the article online for updates and enhancements.

You may also like

Analysis and Predicting Results of IPL T20 Matches

Praveen Banasode, Minal Patil, SupriyaVerma

Department of Master of Computer Applications, Jain College of Engineering Belagavi-14

Affiliated to Visvesvaraya Technological University Belagavi, Karnataka state, India.

Email: [1]praveenb.jce@gmail.com

Abstract. The craze of Indian Premier League (IPL) is always there in millions of people including Indian and our work is about the analysis of the data and prediction of the IPL matches. IPL Data Analysis is all about the analysing the data that is al- ready present in data set using data science, machine learning and python. This is an application design for the purpose of analysing the data by fetching the attribute from the data set and predicting the future of the match and as well as of the players .This will help in the selection of the IPL team that the team should perform good and win the match. Prediction is done for anything like which player will play well in tomorrow's match, which team will win toss and even match etc. The prediction can be done with the help of the analysis on that data set collected and by displaying proper data that is useful for the future prediction. The algorithms have given accuracy over 95%.

Keywords: Indian Premier League, IPL, Winner Prediction, Analyzation.

1. Introduction

Cricket is the biggest tournament played in all most all the countries. It is the game between two teams in which the each team has 11players the final result will be either loss or win or at the rare cases points will be shared with both teams which mean no team has lost or won. Sometimes the game is unpredictable because of that game keeps on changing each and every time.

The madness of cricket in people is like anything by looking into this, the main objective of our work is predicting the match result before the game starts based on the past statistic data that is present in the form of data set .In this the study of Indian Premier League (IPL) is done using that past 12 seasons played till the date.

The Indian Premier League (IPL) is one of the biggest T20 Cricket League in the world. It started in the year 2008, started with 8 teams initially and later added two more teams, for two year (2016 & 2017), each team plays two matches against each other the highest points earned by winning maximum matches with good net run rate will progress to the play-offs, where top 4 teams will be considered the table topper will get two chances if they lose their play-off they can play against the second eliminator and still have a chance to qualify for the final match. IPL attracts many foreign players due to its popularity and fast-paced action. Many aspiring cricketers are shaping with this league and have got a chance to play for their country because of their good performance.

The goal of work is gathering past matches data and analysing them to find out the useful information, which will help in predicting the match result. There is much other way to do prediction; it can be done by consideration of the player's performance and also the performance of the whole team.

There can be many other scenarios where it suddenly starts raining or the player get injured and not able to play. These factors will affect the prediction of the match.

2. Related Work

Deep prakash et al. [7] proposed reading the match result based on 5 features of IPL career and 5 features of International T20 career have been taken into considerations for both batsman and bowler where it gives an idea about the previous 6-8 months performance which gives a good combination of players. Sanket et al. [10] proposed dynamic winner prediction in a Twenty20 cricket match by comparing strength of playing team and also the score that the team batting first is expected to score based on the this the team can plan in setting up a good total on scoreboard and defending the score thereby increasing the chances of win percentage. Shilpi Agrawal et al. [12] proposed predicting the uncertainty of which team will IPL match based on teamwork of whole team on past data, instead of concentrating on a single player they worked on the best possible 11 in team by choosing the parameter of teamwork. Shubra singh and Parmeet kaur [13] addressed the problem of predicting the outcome of an IPL cricket also the player profilingsystem which can be a great help for the team leaders on the auction day, the auction brings a very good player into the team by bidding and making the team more strong by picking right player in terms of batting and bowling or an all-rounder. We have taken all the data from cricinfo.com from 2008 to 2019 season of total 12 IPL seasons for predicting the IPL match results based on players previous performance.

3. Proposed Model

By collecting the previous season data of each player and team, we perform training on the data by developing data analyzation. The information makes use of to calculate points of each player and also calculate the overall strength of each team based on past performance. The output is displayed with the help of various types of graphs where user can predict the match result or take decision in selecting the players for a particular match.
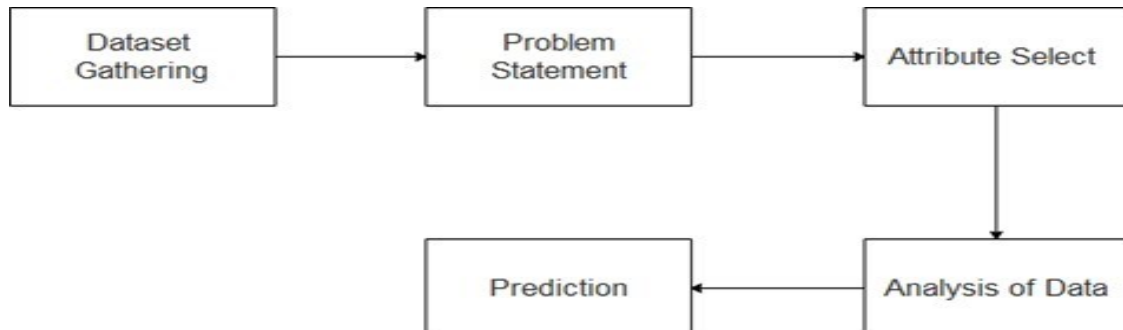

Fig. 1. Block Diagram for Proposed Model

Gathering Dataset: The first step is to collect dataset of past matches which is enough to perform the operations.

Understanding Problem Statement: Problem statement consist of the data to be displayed like, in how many matches team opted batting first after winning toss and won the match and how many choose fielding and won the match.

Attribute Selection: Base on the problem statement the data will be fetched from the.csv File particular data is fetched.

Analysing: The raw data is converted into useful information by performing certain operations. This is done to provide information in appropriate manner.

Prediction: The future is predicated whenever that situation occurs with the help of the information the user can predict the result of the match.

There are total 756 rows and 18 columns, 756 rows imply that there were 756 IPL matches held between 2008 and 2019.

- 75% of the victorious teams that bat first won by a margin of 19 runs.
- 75% of the victorious teams that bat second won by a margin of 6 wickets.

4. Results and Discussions

This section discusses findings of our proposed model. The results are found by performing the calculations on individual players like batsman, bowler, all-rounder and fielder on different dataset. Achieved results will help the team management to decide the best playing 11 to win the match.

The Dataset used for analysis was collected from https://kaggle.com and has been scrapped from https://stats.espncricinfo.com, where data from 2008 to 2019 was available. Datasets used here are highest scores by a batsman, maximum runs scored in a particular venue, highest average score, against which bowler batsman strike rate is good and about the bowler the most dismissal kind of wickets, the bowling economy and also the Toss factor for every match played in IPL.

TABLE I: Match Attributes

Attribute	
id	dl_applied
season	matchwinner
city	result
date	matchplayer
firstteam	umpire1
secondteam	umpire2
toss	umpire3
result	Match_refree

TABLE II: Individual player's performance

Attribute	
match_id	bowler
batting_team	batsman_runs
bowling_team	extra_runs
over	total_runs
batsman	player_dismissed
non_striker	dismissal_kind

Table II consist of the twelve attribute one thousand thirty-seven instance of it. It is stored in the .csv file.

i. Run Scored By Player in All Season

batsman_run_data = bats_man_data.groupby (['season','batting_team'])
[['batsman_runs']].sum().reset_index ()   (1)

In above Equation batsman_run is the variable created where the season and the team is grouped (i.e. for that particular season the player played for the particular team) by using the **groupby** clause of the particular player (batsman_data) , the **sum of the run** is calculated for the every season using in-built function **sum**. Once it is done then using **reset_index ()** method the new index value is set and used.

For example, if we select any player from any team it shows his runs scored from 2008 to 2019. If the player has scored consistently in every season with good batting strike rate then that player is a X factor for the team. The following stat shows in graphical representation of a player scored in every season of IPL. This analysis helps in selecting a good batsman for the match.
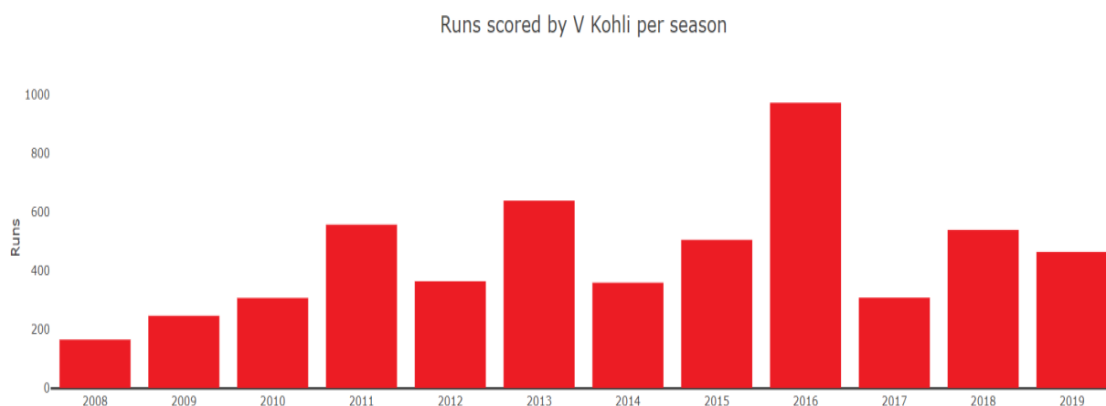

Fig. 2. Runs scored by a player in all seasons, for every team we had set a unique colour (i.e. Team Jersey Colour) in dictionary format. In the bar graph, the x-axis stores the coordinates as season and y-axis as runs. Text stores the name of the batting team

ii. Find FavouriteVenue:

venue_count = venue.groupby ('venue') [['batsman_runs']].sum ().reset_index()
.sort_values('batsman_runs',ascending=False)   (2)

In the above equation **venue_count** is a variable created, **venue** is the variable that stores the name of the player, the run of the player is calculated of every season and is grouped by venue using **groupby** clause then the index is reset (new value to the data set) using **reset_index (),** values of run are sorted in descending order and stored in the variable.

Every player has his favourite venue where they enjoyed playing against any team and has consistently performed well in all the matches so far they have played, may be because of the positive vibes of the city and the ground. To find out such player we did analysis of every venue.

Most runs scored by V Kohli by venue

Venue	Runs by batsman
M Chinnaswamy Stadium	2114
Feroz Shah Kotla	430
Wankhede Stadium	401
Rajiv Gandhi International Stadium, Uppal	340
Eden Gardens	307
MA Chidambaram Stadium, Chepauk	291
M.Chinnaswamy Stadium	234
Saurashtra Cricket Association Stadium	164
Maharashtra Cricket Association Stadium	156
Sawai Mansingh Stadium	131

Fig. 3. Most runs scored by a player in a particular venue (stadium). Cells display the data of the table i.e. all venues of the batsman in one column and the runs scored in an- other column.

iii. Economy Rate of Bowler:

bowling_data ['economy'] = bowling_data.batsman_runs / (bowling_data.ball / 6) (3)

To calculate the economy rate, formula we used is - Run Conceded/Overs bowled, using same the economy rate is calculated, bowling_data is the variable which stores the name of the bowler and it fetches the run given (run scored from that bowler ball) by the particular bowler i.e. batsman_runs .It is the divided by the total ball (of the bowler).

The economy rate of bowler plays a crucial role in selecting a good player against the strong batting opponent team. The bowler possesses skills in bowling death over and taking wickets at regular interval and restricting the team total. In our analysis we considered 50 best bowlers and we generated a scattered graph of an individual bowler.
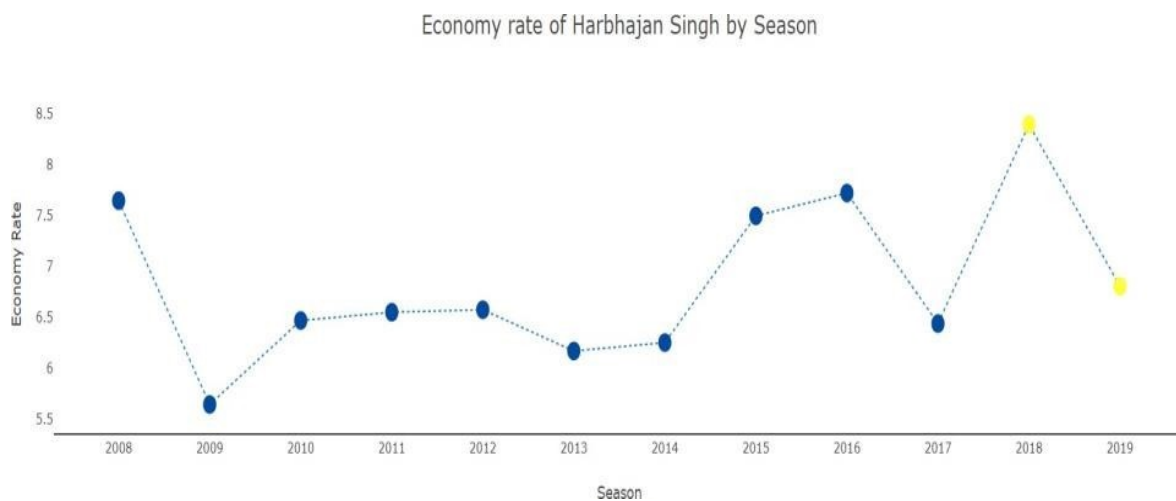
Fig. 4. Economy rate of bowler is displayed in scatter graph format, where x-axis stores the season and y-axis as economy. Text stores the name of the bowling team.

iv. Batsman Run Against Team:

runs_against_team=batsman_data.groupby ('bowling_team') [['batsman_runs']].sum(). reset_index ().sort_values ('batsman_runs', ascending=False)   (4)

In the above equation runs_against_team is variable created, batsman_data is the variable that stores the name of the player, the run of the player is calculated of every season and is grouped by bowling team using groupby clause then the index is reset (new value to the data set) using reset_index (), values of run are sorted in descending order and stored in the variable.

Some batsman has performed well against few teams where they have scored consistently and contributed very well in team success, this helps the team to a pick a right player for the match.
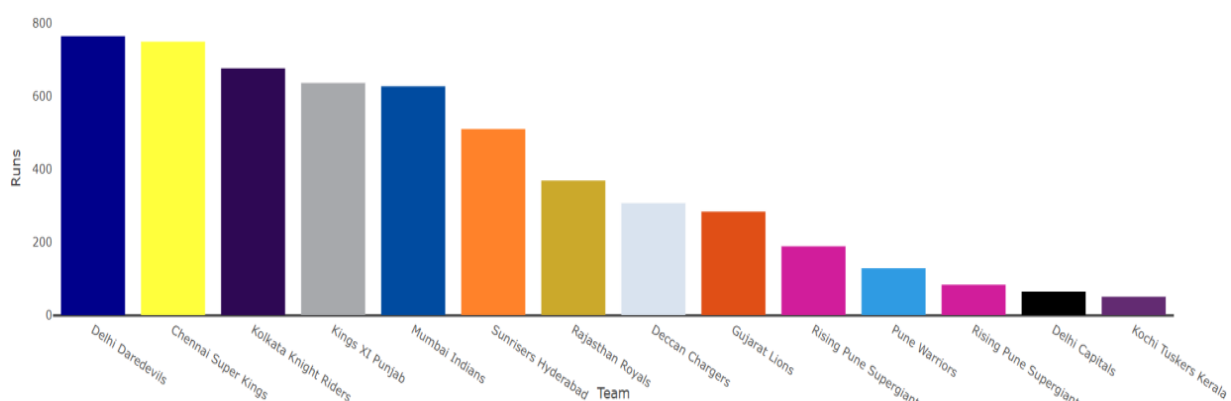

Fig. 5. Maximum runs scored by a batsman against all team, the x-axis stores the coordinates as bowling team and y-axis as runs.

v. Run Scored against Bowlers

batsman_fav_bowler = batsman_data.groupby ('bowler').agg ({'ball': len, 'batsman_runs': sum}) [['ball','batsman_runs']].reset_index()batsman_fav_bowler.sort_values   ('batsman_runs', ascending=False) (5)

In equation batsman_data is the variable that stores the name of batsman and grouped by the bowler's name against which all the batsman had played, the aggregation is calculated based on total number of

ball and the run scored by the batsman. The index value is reset and the sorting of the run is done of every bowler in descending order.

The Following table helps in finding a runs scored against a bowler faced a minimum 6 balls, this will help in sending a right player at crucial time to accelerate the team score by facing a minimum balls and striking at good run rate per ball.

Runs by V Kohli against bowlers (min. 6 balls)

Bowler	Runs
A Mishra	159
R Ashwin	152
UT Yadav	141
DJ Bravo	130
PP Chawla	123
JJ Bumrah	112
RA Jadeja	104
DS Kulkarni	101
SP Narine	99
IK Pathan	92

Fig.6. Run Scored against Bowlers displayed in table format, the table header as Bowler and Runs, Cells display the data of the table i.e. bowler in one column and the run scored in another column.

vi. Total Number of Wickets and Its Kind

Wickets=bowling_data.groupby(['dismissal_kind','season']).size().reset_index()   (6)

In equation bowling data is the variable that stores the name of bowler and grouped by 'dismissal_kind'(the type of which the bowler took wicket like LBW, Caught etc.) group it by season , size is find i.e. number of time the wicket is taken and its kind and the index is reset to provide new index value.

The following graph shows the bowling statics of a particular bowler, which gives clear idea about the kind of dismissal of wickets.
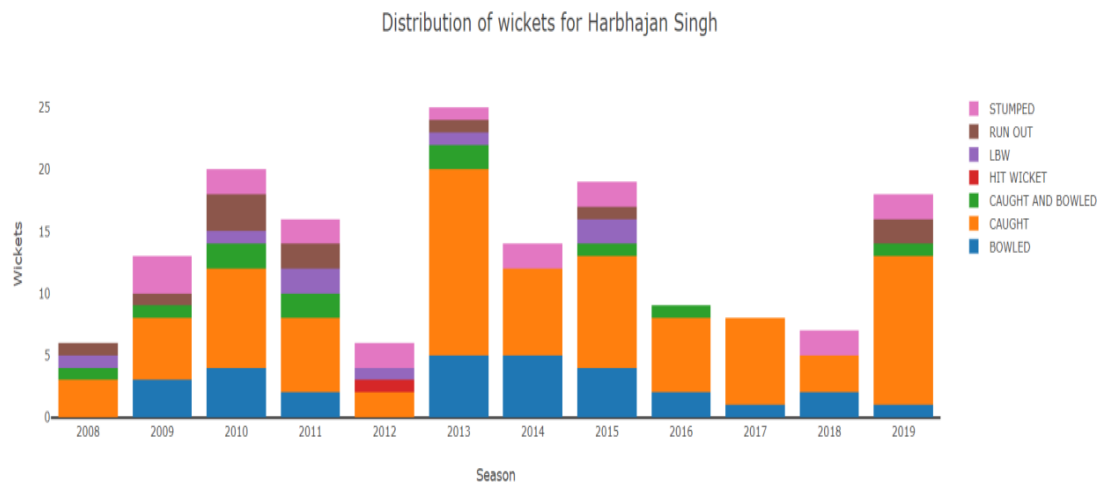
Fig.7. Total Number of Wickets and Its Kind, x-axis stores the coordinates as season and y-axis coordinate as dismissal kind.

The Toss factor plays a crucial role in deciding the match result, the analysis shows win percentage is more when the team bats first and sets up a very good total the average winning percentage is 58. And in some venues bowling first is favourable because of pitch nature where the moisture plays an important factor in swinging the ball and picking up the wickets or not allowing the batsman to score and restricting the team below average total score.
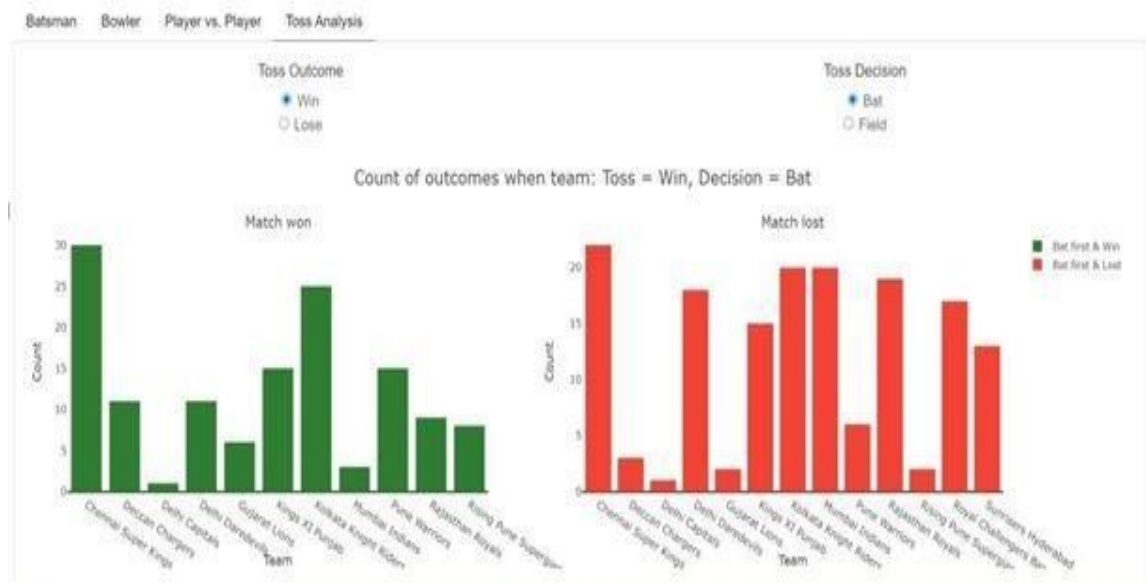

Fig.8. Toss Decision and Result

Mumbai Indians is the most successful team as they have won more than 100 plus matches
and also won the most toss till 2019 followed by Chennai Super Kings and Kolkata Knight Riders.
 Time taken to calculate the result for all these equation is 2 seconds.

5. Conclusion

The IPL data analysis is the most famous technology in today's world. With the help of the data analysis we can find out the pros and cons of the player and work on the performance of the individual player and also the whole, it will help in taking decision. The application can be used for the selection commission to select the best player including bowler, batsman and even the fielder for the team and to

perform well on the field during match. It is helpful for all type of game to work on the performance and predict the future performance of the player and team. The following methodology helps in other T20 leagues around the world like PKL,LPL,CBL,BBL,SuperSmash,T20 Blast, MSL, BPL, APL, and World League CLT20.

References

[1] Aditya Mittal, Amutha Manavalan "The IPL Model: Sports Marketing and Product Placement Sponsorship". International Journal of Humanities and Social Science Invention ISSN (Online): 2319 – 7722, ISSN (Print): 2319 – 7714 www.ijhssi.org ||Volume 6 Issue 2||February. 2017 ||PP.44-61.

[2] Amal Chaminda Kaluarachchi, Aparna S. Varde"CricAI: A classification based tool to predict the outcome in ODI cricket". Conference Paper · IEEE December2010.

[3] Amit Gupta "India and the IPL: Cricket's Globalized Empire". TheCommonwealthJournal of InternationalAffairs.

[4] Ananda Bandulasiri, "Predicting the Winner in One DayInternationalCricket". Journal of Mathematical Sciences & Mathematics Education, Vol. 3, No.1.

[5] Deep Prakash, C. Patvardhan"Data Analytics based Deep Mayo Predictor for IPL-9". International Journal of Computer Applications · October2016.

[6] Jayant Prakash, Mayank Khandelwal, and Tribikram Pradhan "Evaluation of IPL Teams and Players using Association, Correlation and Classification Rules" IEEE International Conference on Computer, Communication and Control(IC4-2015).

[7] Madan Gopal Jhawar, Vikram Pudi "Predicting the Outcome of ODI Cricket Matches: A Team Composition Based Approach". European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD2016).

[8] Sasank Viswanadha, Vikram Pudi "Dynamic Winner Prediction in Twenty20 Cricket: Based on Relative Team Strengths". Conference Paper · September2017.

[9] Sachi Priyanka "Prediction of Indian Premier League-IPL 2020 using Data Mining Algorithms". International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177. Volume 8 Issue II Feb 2020 Matches using MachineLearning".

[10]   Shilpi Agrawal, Suraj Pal Singh, Jayash Kumar Sharma "Predicting Results of Indian Premier League T-20. Conference Paper · November2018.

[11]   Shubhra Singh, Parmeet Kaur " IPLVisualization and Prediction Using HBase" . Information Technology and Quantitative Management(ITQM2017).

[12]   https://kaggle.com

[13]   https://statscricinfo.com