Data Analytics based Deep Mayo Predictor for IPL-9

3 authors, including:

Chellapilla Deep Prakash
Indian Institute of Management
7 PUBLICATIONS   **27** CITATIONS

SEE PROFILE

C. Patvardhan
Dayalbagh Educational Institute
118 PUBLICATIONS   **1,020** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Automatic Design of Quantum Circuits View project

Data Analytics based Deep Mayo Predictor for IPL-9

C. Deep Prakash
Dayalbagh Educational
Institute
Agra-282005, India

C. Patvardhan
Dayalbagh Educational
Institute
Agra-282005,India

C. Vasantha Lakshmi
Dayalbagh Educational
Institute
Agra-282005,India

ABSTRACT

This paper presents a Deep Mayo Predictor model for predicting the outcomes of the matches in IPL 9 being played in April – May, 2016. The model has three components which are based on multifarious considerations emerging out of a deeper analysis of T20 cricket. The models are created using Data Analytics methods from machine learning domain. The prediction accuracy obtained is high as the Mayo Predictor Model is able to correctly predict the outcomes of 39 matches out of the 56 matches played in the league stage of the IPL IX tournament. Further improvement in the model can be attempted by using a larger training data set than the one that has been utilized in this work. No such effort at creating predictor models for cricket matches has been reported in the literature.

General Terms

Machine Learning, Prediction, Cricket, Algorithms

Keywords

Mayo Predictor, Deep Analytics, IPL 9, Random Forest

1. INTRODUCTION

In 2003, T20 Cricket was first introduced in England. It became very popular among the cricket fans because of its shorter format which allowed one to enjoy the complete match in one evening. 20 overs are allowed in a T20 cricket game for each batting side to score as many runs as they can if they have wickets. Due to its high voltage action in every over and on every ball T20 cricket came to India also.

Board of Control of Cricket in India (BCCI) [1] started a T20 cricket league named Indian Premier League (IPL) in 2008 . In this league eight of the largest cities in India (Bangalore, Chennai, Delhi, Hyderabad, Jaipur, Kolkata, Mohali, Mumbai) were taken and eight franchises (Royal Challengers Bengaluru, Chennai Super Kings, Delhi Daredevils, Deccan Chargers, Rajasthan Royals, Kolkata Knight Riders, Kings XI Punjab, Mumbai Indians) were assigned to them. From a pool of available players selected by BCCI according to their rules, the franchises selected their players through competitive bidding. Every year BCCI has been organizing the IPL T20 cricket tournament . 9 IPL tournaments have been held till date .

The use of analytical methods in various aspects of cricket such as Batting, Bowling, Fielding, Team Selection, Result Prediction, Target Revision in a rain affected match are very important. There is a huge demand of various algorithms for each and every aspect of cricket because of its popularity and the huge amounts of money involved in the game. In India the followers of cricket are also followers of statistical records. Thus the analysis of a league like IPL becomes more important .

Some studies related to cricket reported in the literature are as follows. D Parker et al. worked on player evaluations in IPL [2]. S Singh presented a methodology for measuring the performance of team in IPL [3]. H Saikia et al. used Bayesian approach for classification of all-rounders in IPL [4]. LJA Lenten et al. presented a model for player wage determination in IPL [5]. SK Rastogi et al. presented a methodology of player pricing and valuation of cricketing attributes [6]. C Peterson et al. worked on performance evaluation of IPL season 1 [7].

Prediction of outcome of matches using algorithmic approaches is one of the very important aspects in cricket. No comprehensive attempt has been made in the literature to this end. This is an important problem because franchises invest huge amounts of money. With the help of a coin toss one can predict with 50% probability the winning team treating the game as one of perfect chance without even considering the relative merits of the respective teams playing against each other. In contrast to this records of past performance of players, their current form and other cricket related data can be analyzed statistically to create mathematical models with better probability of success in predicting the winning team.

Design of a predictor model for predicting the winners of individual matches can be based on different considerations. One approach is to check whether the team as a whole is better than the other team or not. This would require creating a composite index that comprehensively includes the batting, bowling and other skills of the team in a single statistic or number. If such an index is created it is only a matter of comparing the values for the opposing teams and one would be able to obtain the result. However, creating such a composite index is non-trivial as it has to bring in the consideration of multiple disparate considerations into a single metric.

Another alternative is to create a model free estimator based on machine learning or neural network techniques that takes values of individual performance parameters aggregated at the team level and is trained to output the result based on these parameters. Recent research has unearthed a variety of such approaches that can be employed for this purpose. However, relatively large amount of training data may be required to obtain meaningful prediction results.

A different approach could be based on comparing the players one on one at corresponding batting positions in the rival teams with respect to both batting and bowling strengths. The individual differences have to be then aggregated into a single index either through a composite index or by an estimator based on machine learning techniques as above. This approach is meaningful because there are players like Chris Gayle, Virat Kohli, Rohit Sharma , AB De Villiers etc. in batting and Sunil Narine, Lasith Malinga, R Ashwin etc. in bowling who can win matches for their team single handedly.

This is especially true in T20 because this shorter format does not give enough time for the team at the receiving end of a great batting display or a burst of inspired bowling to recover. In this format a couple of overs may swing the match either way.

The above two approaches predict the performance of each player and, thereby the team, going by his career records. But this has a limitation of not taking current form into consideration. Every player has ups and downs in a career in sports and this is particularly true of cricket. Sometimes the current form is awesome and it appears that the player can do no wrong. Sometimes the same player appears to be just a shadow of his own self. The current form has to be included for better chance of success. The model can then predict when a relatively weaker team that is playing well as a whole can beat the stronger team having great individual players with very good career statistics but not currently in good form.

Thus prediction cannot be achieved with just one model because various cricketing aspects in T20 are important and not commensurable. In this work, three different models have been constructed based on each of these three approaches. Machine learning techniques have been utilized with advantage for this purpose. The outcome of a match is predicted by taking a majority vote of these three models. This combined prediction model is named Mayo Predictor.

The rest of the paper is organized as follows. In section 2, a ranking methodology for ranking the batsmen and bowlers in fray in the IPL IX is described. This ranking forms the basis for all the three models. The three models are described in sections 3, 4 and 5. In section 6, the results and in section 7 the conclusion and future work are described. In section 8 references are given.

2. PERFORMANCE RANKING SCHEME FOR CRICKETERS IN IPL IX

A strong mathematical model for performance ranking evaluates the relative performance of the players and enables an ordering to be imposed for determining the better player in a pair-wise comparison. In this work, both overall T20 career data of the players (http://www.espncricinfo.com) [8] as well as their IPL career data (www.iplT20.com) [9] are taken into consideration. The often quoted index in the case of IPL is the Most Valuable Player Index (MVPI) popularized by Rediff Cricket [10]. The MVPI for a player is the sum of his batting points and bowling points. These metrics provide an overall picture of the performance but are typically biased towards senior players. There is a need for developing other metrics. Deep Performance Index [11] ranks the bowlers and batsmen based on their performance in the IPL up to season VII and their performance in T20 matches overall up to that point in time. However, T20 is an evolving game [12]. The ranking scheme has to take into account the latest performance statistics up to IPL season VIII and correspondingly the latest overall T20 performance statistics. In this paper, DPI is computed by taking current data.

In order to evaluate the batting capability in T20, five indices are considered as follows.

1) HardHitter = (4*Fours + 6*Sixes) / Balls faced by player
2) Finisher = Number of times Not Out/ Total number of Innings played
3) FastScorer =Total runs scored/ Total balls faced

4) Consistent = Total runs scored/ Total number of innings in which he got out
5) RunningBetweenWickets = (Total runs scored - (4*Fours + 6*Sixes))/ Number of balls faced without boundary

The above five measures are typical T20 measures for a more detailed analysis of the performance of the batsmen. Similarly, in order to define Bowling Capability five indices are considered as follows.

1) Economy= Total number of runs conceded / Total number of overs bowled
2) WicketTaker = Total number of balls bowled / Total number of wickets taken
3) Consistent = Total number of runs conceded / Total number of wickets taken
4) BigWicketTaker = Number of times four wickets or five wickets taken / Number of innings played
5) ShortPerformance = (Number of wickets taken – 4* Number of times four wickets – 5* Number of times five wickets taken) / (Number of innings played - Number of times four wickets or five wickets taken)

Ten indices are computed as above for each batsman and each bowler considering their latest T20 performance statistics available and their IPL performance up to season VIII.

Recursive Feature Elimination algorithm with Random Forests is used in order to select the important features and their relative importance. Random Forests algorithm is one of the most important algorithm in data science for various classification and regression tasks. The Algorithm works as follows:

In Random Forest as the name suggests , several trees are constructed using a different bootstrap sample of data. At any particular node of any tree in the Random Forest, the node is split using the best among a subset of predictors randomly chosen at that node. Thus instead of learning the node split using all the features which are available , only a subset of those features are randomly selected. This strategy has been shown to perform well and is robust against overfitting since each node split has been done using randomly selected features.

Recursive Feature Elimination (RFE) is used for determining the relevant features and their corresponding importance for player ranking [13]. The algorithm fits the Random Forest model to all 10 features which are the indices in the current work. Each feature is ranked according to its importance to the model at a particular node split which is done using randomly selecting the features. At each iteration of feature selection process, the N top ranked features are retained, the Random Forest model is refit and its performance is re-assessed. The value of N with the best performance is determined and the top N features are used to fit the final Random Forest model. The feature rankings are recomputed using the final Random Forest model on the reduced feature set. While calculating performance of feature selection using Random Forest, resampling methods (e.g. cross-validation, the bootstrap) are used to reduce variability. 10-fold cross-validation is used to ensure better robustness of results and provide better estimates of performance. A consensus ranking is used to finally determine the best predictors to retain.

The features and weightages for batting and bowling obtained are shown in table 1.

Table 1: Batting and Bowling features and their relative importance

Batting Features	Weightage	Bowling Features	Weightage
T20_Consistency	0.4851	T20_Consistency	0.3011
IPL_Consistency	0.3574	IPL_ShortPerformance	0.3001
T20_FastScorer	0.1251	IPL_Consistency	0.2327
T20_FScore	0.0275	T20_WicketTaker	0.1429
IPL_FastScorer	0.0047	IPL_BigWicketTaking	0.0480

After selecting the features, they are normalized and composite ranking index is determined for evaluating batting performance and bowling performance. The formulae obtained are as follows.

Batting Index: T20_Consistency*0.4851+ IPL_Consistency*0.3574+ T20_FastScorer*0.1251+ T20_FScore*0.0275+ IPL_FastScorer*0.0047

Bowling Index: T20_Consistency*0.3011+ IPL_ShortPerformance*0.3001+IPL_Consistency*0.2327+ T20_WicketTaker*0.1429+ IPL_BigWicketTaking*0.0480

3. PREDICTION MODEL 1 (PM1)

In PM1, the training data is prepared by taking data up to IPL 8 for all the players. The 5 features for batting and bowling performance identified in section 2 are utilized. Using these features and the weights calculated for these features using the Random Forest Algorithm, batting and bowling ranks of these players are calculated. The differences of these ranks for individual players playing at the same position in the rival teams are computed and these are used as the training vectors. Thus there are 22 features (11 for the corresponding batting rank differences and 11 for the corresponding bowling rank differences) in the training set. The target vector is comprised of the results of the matches i.e. 1 if the first team wins and 0 if the second team wins. A Support Vector Machine [14] model is trained using this training data and is used for prediction based on data for IPL 9.

The prediction problem has been converted into classification problem. In a classification problem a new set of observation is categorized in one of the classes based on the training set data observations. In this work the predictor does the classification in 1 or 0 based on whether the team batting first will win or not. The classification is done using Support Vector Machine (SVM) Model. Based on the previous season matches the SVM builds a model that assigns new matches one of the two categories. The result of the match can be inferred. The basic principle behind the working of SVM is that it maps the data points into space in such a way that the gap between the two classes (1 and 0) are as wide as possible. When the evaluation of new data points has to be done then their mapping into the space is done and based on which side of the classification gap they fall on, their value is predicted according to that.

In this work the training set data is as follows:

(x_1,y_1), (x_2,y_2),.................(x_n,y_n). x_i are features for different matches and y_i are the corresponding classification results (1 or 0). The goal of SVM is to map the features x_i and try to find a hyperplane that divides matches for which, team batting first won ($y_i=1$) and matches for which, team batting second

won ($y_i=0$). SVM tries to find a hyperplane which maximizes the distance between the nearest point of either of the class and the hyperplane.

In mathematical way the model can be expressed as follows.

Let $a^{->}$ be a vector normal to the hyperplane, $x^{->}$ be the set of points then a hyperplane can mathematically be written as :

$$a^{->}.x^{->} + b^{->} = 0$$

Here $b^{->}/ \|a^{->}\|$ is the offset of the above mentioned hyperplane from the origin.

For each match i there are constraints in order to construct the hyperplane such that the two classes are correctly classified.

$a^{->}.x^{->} + b^{->} >= 1$ when the team batting first wins ($y_i=1$)

or

$a^{->}.x^{->} + b^{->} <= 0$ when the team batting first loses ($y_i=0$)

The goal of SVM is to minimize $a^{->}$ satisfying the above mentioned two equations and to find the parameters of the hyperplane ($a^{->}$ and $b^{->}$) which can classify the new matches into one of the two classes (1 or 0).

Pseudo code for Predictive Model 1 is as follows:

(i) For each team Calculate the overall feature values for batting and bowling for the actual playing eleven for a particular match

(ii) For every player in the playing eleven find his corresponding batting and bowling rank.

(iii) Take the differences between the Ranks of players playing at the same position, but from opponent teams, for batting and bowling

(iv) Prepare training set with these selected 22 features for IPL season 8 with corresponding results of the match and train the SVM.

(v) Prepare test set with these selected 22 features for season 9

(vi) Use the SVM to predict the results of matches using the training set.

4. PREDICTION MODEL 2 (PM2)

This model is similar to previous model. The difference is that in order to incorporate the current form of the players the training set is updated with each new match that is being played in season 9 and thus the data is part of training set for the subsequent matches.

Pseudo code for Predictive Model 2 is as follows:

(i) For every team:
Calculate the overall feature values for batting and bowling .
Use the actual playing eleven for a particular match

(ii) For every player in the playing eleven find his corresponding batting and bowling rank.

(iii) For a particular match:

Take the differences in Ranks of players playing at the same position, but from opponent teams, for his batting and bowling.

(iv) Prepare training set with these selected 22 features for IPL season 8 with corresponding results of the match .

Keep updating the training set with the matches which have been played in season 9 so far before a particular match for which prediction is going to be made.

Thus for every match the training set is going to be different.

(v) Prepare test set with these selected 22 features for season 9

(vi) Use Support Vector Machine to predict the result of matches of season 9 using the training set.

5. PREDICTION MODEL 3 (PM3)

In PM3, the training data is prepared by taking the data up to IPL 8 for all the players. The features considered are the 5 features for batting and bowling performances each which have been identified earlier. These feature values are averaged for each of the 11 players in the playing XI. Thus a total of 10 features are obtained for each team. The differences of these are computed for the rival teams and these are used as the training vectors. The target vector is comprised of the results of the matches i.e. 1 if the first team wins and 0 if the second team wins. A Support Vector Machine model is trained using this training data and is used for prediction based on data for IPL 9.

Pseudo code for Predictive Model 3 is as follows:

(i) For every team:Calculate the overall feature values for batting and bowling.Use the actual playing eleven for a particular match.

(ii) For every player in the playing eleven find his corresponding batting and bowling features.

(iii) For a particular match:Take the average value of each feature for a particular team's batting and bowling.Take the difference of average values of batting and bowling features of players from opposite teams and obtain 10 features for a particular match

(iv) Prepare training set with these selected 10 features for IPL season 8 with corresponding results of the match

(v) Prepare test set with these selected 10 features for season 9

(vi) Use Support Vector Machine to predict the result of matches of season 9 using the training set

6. RESULTS

The results obtained from PM1, PM2 and PM3 individually and by the composite Mayo model that simply reports the majority vote among the three other models are shown in Table 2. Mayo model is able to predict the outcomes with high accuracy getting 39 out of 56 matches. This is the first effort using machine learning techniques for predicting outcomes of IPL cricket matches.

Table 2: IPL-9 Match Results Prediction

Match	PM1	PM 2	PM 3	Mayo	Result
MI vs RPS	**RPS**	**RPS**	**RPS**	**RPS**	RPS
KKR vs DD	DD	DD	DD	DD	KKR
KXIP vs	KXIP	**GLR**	**GLR**	**GLR**	GLR
GLR					
RCB vs SRH	SRH	SRH	SRH	SRH	RCB
KKR vs MI	**MI**	**MI**	**MI**	**MI**	MI
GLR vs RPS	**GLR**	**GLR**	RPS	**GLR**	GLR
DD vs KXIP	**DD**	KXIP	**DD**	**DD**	DD
SRH vs KKR	**KKR**	SRH	**KKR**	**KKR**	KKR
MI vs GLR	**GLR**	**GLR**	**GLR**	**GLR**	GLR
KXIP vs RPS	RPS	RPS	RPS	RPS	KXIP
RCB vs DD	**DD**	**DD**	**DD**	**DD**	DD
SRH vs MI	**SRH**	MI	MI	MI	SRH
KXIP vs KKR	KXIP	**KKR**	**KKR**	**KKR**	KKR
MI vs RCB	RCB	RCB	**MI**	RCB	MI
GLR vs SRH	**SRH**	**SRH**	GLR	**SRH**	SRH
RPS vs RCB	**RCB**	**RCB**	**RCB**	**RCB**	RCB
DD vs MI	MI	MI	**DD**	MI	DD
SRH vs KXIP	**SRH**	**SRH**	KXIP	**SRH**	SRH
GLR vs RCB	**GLR**	**GLR**	**GLR**	**GLR**	GLR
RPS vs KKR	**KKR**	**KKR**	RPS	**KKR**	KKR
KXIP vs MI	**MI**	KXIP	**MI**	**MI**	MI
SRH vs RPS	SRH	**RPS**	**RPS**	**RPS**	RPS
DD vs GLR	DD	DD	DD	DD	GLR
MI vs KKR	**MI**	KKR	**MI**	**MI**	MI
RPS vs GLR	RPS	RPS	RPS	RPS	GLR
DD vs KKR	**DD**	**DD**	KKR	**DD**	DD
SRH vs RCB	**SRH**	**SRH**	**SRH**	**SRH**	SRH
GLR vs KXIP	GLR	GLR	GLR	GLR	KXIP
RPS vs MI	RPS	RPS	RPS	RPS	MI
RCB vs KKR	**KKR**	**KKR**	**KKR**	**KKR**	KKR
GLR vs DD	GLR	GLR	**DD**	GLR	DD
KKR vs KXIP	**KKR**	**KKR**	**KKR**	**KKR**	KKR
DD vs RPS	**RPS**	**RPS**	DD	**RPS**	RPS
SRH vs GLR	**SRH**	**SRH**	**SRH**	**SRH**	SRH
RCB vs RPS	**RCB**	**RCB**	RPS	**RCB**	RCB
KXIP vs DD	DD	DD	DD	DD	KXIP
MI vs SRH	**SRH**	**SRH**	**SRH**	**SRH**	SRH
KKR vs GLR	**GLR**	**GLR**	KKR	**GLR**	GLR
KXIP vs RCB	**RCB**	**RCB**	KXIP	**RCB**	RCB
RPS vs	RPS	**SRH**	RPS	RPS	SRH

SRH					
RCB vs MI	**MI**	**MI**	**MI**	**MI**	MI
SRH vs DD	SRH	SRH	**DD**	SRH	DD
MI vs KXIP	MI	MI	MI	MI	KXIP
RCB vs GLR	**RCB**	**RCB**	GLR	**RCB**	RCB
KKR vs RPS	RPS	RPS	RPS	RPS	KKR
MI vs DD	**MI**	DD	DD	DD	MI
KXIP vs SRH	**SRH**	KXIP	**SRH**	**SRH**	SRH
KKR vs RCB	**RCB**	RCB	KKR	**RCB**	RCB
RPS vs DD	**RPS**	**RPS**	DD	**RPS**	RPS
RCB vs KXIP	**RCB**	RCB	KXIP	**RCB**	RCB
GLR vs KKR	**GLR**	**GLR**	KKR	**GLR**	GLR
DD vs SRH	**DD**	SRH	**DD**	**DD**	DD
RPS vs KXIP	**RPS**	**RPS**	**RPS**	**RPS**	RPS
GLR vs MI	**GLR**	**GLR**	**GLR**	**GLR**	GLR
KKR vs SRH	**KKR**	**KKR**	SRH	**KKR**	KKR
DD vs RCB	**RCB**	**RCB**	DD	**RCB**	RCB

Figure 5. Shows the Mayo Predictor's team -wise prediction accuracy. True Prediction includes those matches in which the predictor has predicted that the team will win and it won the match or the matches in which the predictor has predicted that team will lose the match and it lost the game. False prediction includes those matches in which the predictor has predicted that the team will win but it lost or the matches in which the predictor has predicted that the team will lose but they won the match.

7. CONCLUSIONS AND FUTURE WORK

The results show that the problem of match result prediction can be solved and a mathematical model can be created for prediction of the results of the matches prior to the match based on the knowledge of past matches, playing eleven and the toss result. In this work 5 features of IPL career and 5 features of International T20 Career have been taken into consideration for both batsmen and bowlers but in future work more features can be created and taken into account. The training data can be made larger in the future work for better model learning and classification. Various other data analytics techniques can also be used for improving the accuracy of the model in future work.

8. REFERENCES

[1] Indian Premier League, https://en.wikipedia.org/wiki/Indian_Premier_League

[2] Parker, David, Phil Burns, and Harish Natarajan. "Player valuations in the indian premier league." *Frontier Economics* 116 (2008).

[3] Singh, Sanjeet. "Measuring the performance of teams in the Indian Premier League." *American Journal of Operations Research* 1.03 (2011): 180.

[4] Saikia, Hemanta, and Dibyojyoti Bhattacharjee. "On classification of all-rounders of the Indian premier league (IPL): a Bayesian approach." *Vikalpa*36.4 (2011): 25-40.

[5] Lenten, Liam JA, Wayne Geerling, and László Kónya. "A hedonic model of player wage determination from the Indian Premier League auction: Further evidence." *Sport Management Review* 15.1 (2012): 60-71.

[6] Rastogi, Siddhartha K., and Satish Y. Deodhar. "Player pricing and valuation of cricketing attributes: exploring the IPL Twenty20 vision." *Vikalpa* 34.2 (2009): 15-23.

[7] Petersen, C., et al. "Analysis of Twenty/20 cricket performance during the 2008 Indian Premier League." *International Journal of Performance Analysis in Sport* 8.3 (2008): 63-69.

[8] http://www.espncricinfo.com/india/content/player/28081.html, T20 statistics of each player

[9] http://www.iplt20.com/teams/royal-challengers-bangalore/squad/236/chris-gayle , IPL statistics of each player

[10] http://www.rediff.com/cricket/report/icc-world-cup-de-villiers-maintains-big-lead-shami-rises-to-7th-in-most-valuable-player-table/20150320.htm

[11] C. Deep Prakash, C.Patvardhan and Sushobhit Singh, "A new Machine Learning based Deep Performance Index for Ranking IPL T20 Cricketers", International Journal of Computer Applications (0975 – 8887) Volume 137 – No.10, March 2016

[12] C. Deep Prakash, C.Patvardhan and Sushobhit Singh," A new Category based Deep Performance Index using Machine Learning for ranking IPL Cricketers", Int. Jl. of Electronics, Electrical and Computational System IJEECS ISSN 2348-117X Volume 5, Issue 2 February 2016

[13] Leo Breiman. Random forests. Machine Learning, 45(1): 5–32, 2001

[14] https://en.wikipedia.org/wiki/Support_vector_machine

9. APPENDIX

Player	Team	T20_Consistency	IPL_Consistency	T20_FastScorer	IPL_FastScorer	IPL_HHScore	DPI
Chris Gayle	6	1.00	0.99	0.92	0.98	0.98	0.98
David Miller	4	0.95	0.98	0.76	0.94	0.85	0.93
Shaun Marsh	4	0.99	0.96	0.59	0.76	0.79	0.91
MS Dhoni	7	0.94	0.94	0.72	0.84	0.64	0.88

Fig 1. Top 4 Batsmen and their corresponding DPI

Player	Team	IPL_ShortPerformance	T20_BigWicketTaking	IPL_Economy	T20_Consistent	IPL_Consistent	DPI
Lasith Malinga	5	0.98	0.94	0.98	0.92	0.99	0.96
Shakib al Hasan	3	0.95	0.88	0.95	0.88	0.92	0.92
Sunil Narine	3	0.75	0.99	1.00	0.96	1.00	0.90
Nathan Coulter-Nile	1	0.89	0.79	0.81	0.72	0.96	0.84

Fig 2. Top 4 Bowlers and their corresponding DPI

Bat1	Bowl1	Bat2	Bowl2	Bat3	Bowl3	Bat4	Bowl4	Bat5	Bowl5	Bat6	Bowl6	Bat7	Bowl7	Bat8	Bowl8	Bat9	Bowl9	Bat10	Bowl10	Bat11	Bowl11	Result
0.078	0.405	-0.26	0	0.007	0	0.345	0	-0.15	-0.236	0.338	-0.33	0.058	0.238	-0.193	0.328	0	-0.273	0	-0.183	0	-0.008	0
0.193	0.448	0.753	0	0.254	-0.256	0.645	0.474	-0.076	0.544	0.238	-0.522	-0.213	0.357	0	0.076	0	0.0274	0	-0.517	0	0.578	1
-0.137	-0.389	-0.07	0	0.077	0.529	-0.109	-0.05	0.331	-0.154	-0.95	0	-0.093	0.827	0.255	0.088	0	0.25	0	0.15	0	-0.152	1
-0.187	0.448	0.071	0	0.643	0.315	0.533	0	-0.179	0.25	0.282	0.736	0.5	-0.079	0	0.555	0	0.256	0	-0.153	0	0.6	1

Fig 3. Training Set for PM 1 and PM 2

| Bat1	Bat2	Bat3	Bat4	Bat5	Bowl1	Bowl2	Bowl3	Bowl4	Bowl5	Result
0.043685	0.009454	0.12805	0.052011	-0.01088	-0.02197	-0.09394	0.024242	0.044697	0.025758	0
0.169561	0.228805	0.130746	0.129724	0.085802	0.142424	0.125758	0.14697	0.026515	0.031818	1
-0.05822	-0.03677	-0.06844	-0.0858	-0.10521	0.05	0.184091	0.081818	0.09697	0.049242	1
0.110317	0.187947	0.229826	0.161389	0.168539	0.272727	0.305303	0.293182	0.22803	0.148485	1

Fig 4.Training Set for PM 3
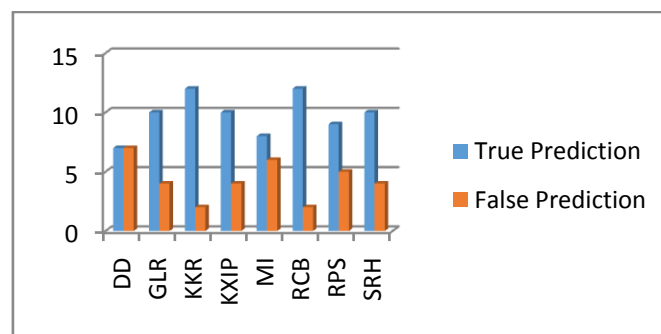
Fig.5 Teams and the Prediction Accuracy