Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning

2 authors:

Rabindra Lamsal
Jawaharlal Nehru University
5 PUBLICATIONS   **0** CITATIONS

SEE PROFILE

Ayesha Choudhary
Jawaharlal Nehru University
26 PUBLICATIONS   **65** CITATIONS

SEE PROFILE

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

Project   Disaster Response using Artificial Intelligence View project

Project   Help Nepal Network E-library Project View project

Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning

Rabindra Lamsal, Ayesha Choudhary

School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi 110067
rabindralamsal@outlook.com, ayeshac@mail.jnu.ac.in

ABSTRACT

Cricket, especially the Twenty20 format, has maximum uncertainty, where a single over can completely change the momentum of the game. With millions of people following the Indian Premier League (IPL), developing a model for predicting the outcome of its matches is a real-world problem. A cricket match depends upon various factors, and in this work, the factors which significantly influence the outcome of a Twenty20 cricket match are identified. Each players performance in the field is considered to find out the overall weight (relative strength) of the team. A multivariate regression based solution is proposed to calculate points of each player in the league and the overall weight of a team is computed based on the past performance of the players who have appeared most for the team. Finally, a dataset was modeled based on the identified seven factors which influence the outcome of an IPL match. Six machine learning models were trained and used for predicting the outcome of each 2018 IPL match, 15 minutes before the gameplay, immediately after the toss. The prediction results are impressive. The problems with the dataset and how the accuracy of the classifier can be improved further is discussed.

KEYWORDS

Cricket prediction; sports analytics; multivariate regression; neural networks

1. Introduction

With technology growing more and more advanced in the last few years, an in-depth acquisition of data has become relatively easy. As a result, Machine Learning is becoming quite a trend in sports analytics because of the availability of live as well as historical data [1–9]. Sports analytics is the process of collecting past matches data and analyzing them to extract the essential knowledge out of it, with a hope that it facilitates in effective decision making. Decision making may be anything including which player to buy during an auction, which player to set on the field for tomorrow's match, or something more strategic task like, building the tactics for forthcoming matches based on players' previous performances.

Machine Learning can be used effectively over various occasions in sports, both on-the-field and off-the-field. When it is about on-the-field, machine learning applies to the analysis of a players fitness level, design of offensive tactics, or decide shot selection. It is also used in predicting the performance of a player or a team, or the outcome of a match. On the other hand, the off-the-field scenario concerns the business perspective

of the sport [10,11], which includes understanding sales pattern (tickets, merchandise) and assigning prices accordingly. The main focus is the healthy growth in business and profitability of the team owners and other stakeholders. On-the-field analytics generally make use of supervised machine learning algorithms, example: (i) regression for calculating the fitness of a player, (ii) classification for predicting an outcome of a match; while off-the-field analytics concerns around performing sentiment analysis to understand peoples opinion about a player or a team or a sport league. At present, Twitter has become one of the primary sources of data for sentiment analysis.

Sport Lisboa e Benfica, one of Portugals most successful football clubs advancing in the use of data modeling techniques while making decisions [12] is one real-world example of the use of machine learning in sports science. The club monitors and analyzes almost every aspect of a player, including their sleeping, eating, training habits. Once raw player data is recorded, various models are designed to analyze the data for optimizing match readiness and defining personalized practice schedules. With the application of machine learning and predictive analysis, the facts coming out of the devised models enable players to improve their performance continually. On the other cards, with those facts at hand, manager/coach gets a better idea about which player to be replaced, which player to be kept in the playing list and which player to be kept in the bench.

Major League Baseball (MLB) has seen enormous growth in the arena of sports analytics in the last few years [13,14]. Professional MLB teams collect tremendous amount of ball-by-ball data and apply various machine learning approaches to get clear insights into the game, which is usually not visible through human analysis. Predicting the outcome of a match, classifying if a team will intentionally make a player walk at bat or classifying non-fastball pitches according to pitch type, etc. are some of the classification problems dealt using machine learning in baseball world [15].

Similarly, cricket has also been making use of sports analytics to perform prediction of outcome of a match, while the gameplay is in progress or before the match has even begun [16–19]. Even problem like predicting runs or wickets of a player for a match, based on his/her past performance is an interesting problem to work on. Some real-world tools which have been implemented in cricket include WASP (Winning and Score Predictor) [20], a tool which predicts a score and possible outcome of a limited over cricket match, i.e., One-day or Twenty20. Sky Sports New Zealand first introduced this tool in 2012 during an ongoing Twenty20 match. Technology like Hawk-Eye [21–23] which tracks the trajectory of a ball and visually displays the most statistically significant path, has also been officially in use as the Umpire Decision Review System since 2009. Similarly, other sports like tennis, badminton, snooker also make use of this computer-assisted intelligent technology.

1.1. Machine Learning in Sports

Various machine learning algorithms have been applied and tested for their efficiency in solving the problems in sports. The relation between machine learning and games dates back to the initial days of artificial intelligence when Arthur Samuel, a pioneer in the field of gaming and artificial intelligence studied machine learning approaches using the game of checkers [24].

A study [25] was performed to predict the outcome (win, lose or draw) of football matches played by a professional English Premier League (EPL) team, Tottenham Hotspurs, based on matches that were played in the year between 1995 to 1997. It

was observed that Bayesian networks relatively outperformed other machine learning algorithms which included MC4 - a decision tree learner, Naive Bayesian learner, Data-driven Bayesian, and K-nearest neighbor. The prediction accuracy of the Bayesian nets model was 59.21%.

Match outcome prediction and game-play analysis are a prevalent problem that is tackled using machine learning. Another area where machine learning approaches are being used is extracting highlights from an on-going match. A study was performed to extract baseball match highlights on a set-top device [26]. The relative strength of classification algorithms, namely Support Vector Machine (SVM), Gaussian Fitting (GAU) and K-Nearest Neighbours (KNN) was considered for "excited speech" classification, and finally, SVM was applied. Six baseball matches covering 7 hours of game-play time was fed to the algorithm. 75% of the highlights extracted by the algorithm were common with the highlights extracted manually by a human.

Just like in football, supervised machine learning algorithms have also been used in predicting the outcome of baseball matches. A project [27] used two learning methods, i.e., logistic classification and Artificial Neural Network (ANN) to predict the result of the baseball post-season series. Although ANN came up with very poor accuracies, the accuracies out of the logistic model were satisfactory with training and test accuracies of 73.6% and 62.6% respectively. Another project applied four machine learning algorithms to understand career progression in Baseball [28]. The implemented algorithms were Linear Regression (Ridge Model), Multi-Layer Perceptron Regression (Neural Network), Random Forests Regression (Tree Bagging Model), Support Vector Regression (SVR). The dataset which was used to train these algorithms contained match data of the first six seasons of players' career. And the players' value were predicted. The prediction was near 60% for the batters, while for pitchers the accuracy was very poor, i.e., something around 30-40%.

1.2. Machine Learning in Cricket

In cricket, to predict an outcome of a match, the primary task is to extract out the essentials factors (features) which affect result of a match. Interesting works have been done in the field of predicting outcome in cricket. The literature survey concluded that the majority of the published works which predicted a result of a cricket match prior were for the test or one-day international cricket format.

Bandulasiri [16] has analyzed the factors like home field advantage, winning the toss, game plan (first batting or first fielding) and the effect of Duckworth Lewis method [29] for one-day cricket format. Furthermore, Bailey and Clarke mention in their work [17] that in one-day cricket format, home ground advantage, past performances, venue, performance against the specific opposition, current form are statistically significant in predicting total runs and predicting the outcome of a match. Similarly [18] discusses modeling home-runs and non-home runs prediction algorithms and considers taking runs, wickets, frequency of being all-out as historical features into their prediction model. But, they have not leveraged bowlers features and have given more emphasis to batsmen. Kaluarachchi and Aparna [19] have proposed a tool that predicts match apriori, but player performance has not been considered into their model.

1.3. Indian Premier League

1.3.1. Introduction

Indian Premier League (IPL) is a professional cricket league based on Twenty20 format and is governed by Board of Control for Cricket in India. The league happens every year with participating teams name representing various cities of India. There are many countries active in organizing Twenty20 cricket leagues. While most of the leagues are being overhyped and team franchises are routinely losing money, IPL has stood out as an exception [30]. As reported by espncricinfo, with Star Sports spending $2.5 billion for exclusive broadcasting rights, the latest season of IPL (2018, 11th) saw 29% increment in the number of viewers including both the digital streaming media and television. The 10th season had 130 million people streaming the league through their digital devices and 410 million people watching directly on the TV [31]. The numbers prove that IPL is a successful Twenty20 format based cricket league.

1.3.2. Machine Learning in Indian Premier League

Some interesting machine learning works have also been performed on data acquired from Indian Premier League matches. In a study [32], the Naive Bayesian classifier was used to classify the performance of all-rounder players (bowler plus batsman) into four various non-overlapping categories, viz., a performer, a batting all-rounder, a bowling all-rounder or an underperformer by being based on their strike rate and economy rate. Step-wise multinomial logistic regression (SMLR) was used to extract the essential predictors. When validated, the Naive Bayesian model was able to classify 66.7% of the all-rounders correctly. The same authors later published a work in which an Artificial Neural Network model was used to predict the performance of bowlers based on their performance in the first three seasons of IPL [33]. When the predicted results were validated with actual performance of the players in season four, the developed ANN model had an accuracy of 71.43%.

Although not related to IPL, a study performed at University College London in the area of predicting the outcome of a Twenty20 match [34] would be a healthy addition as the literature work in the Twenty20 domain. The study made use of Naive Bayes, Logistic Regression, Random Forests, Gradient Boosting algorithms to predict the outcome of English County cricket matches. Two models were developed, each was given input of two different sets of features. The team only related features were input to the first model, while team and players related features were input to the second model. The study was concluded with Naive Bayes outperforming all other algorithms with the first model giving out average prediction accuracy of 62.4% and second model giving average prediction accuracy of 63.9%, i.e., 64% average accuracy with 2009-2014 data and 63.8% average accuracy with 2010-2014 data.

1.4. Organization of Paper

The paper is organized as follows: The proposed work is discussed in detail in various subsections of section 2. Results are shown in section 3, and section 4 concludes the paper.

2. The Proposed Work

The literature survey concluded that there was a need for a machine learning model which could predict the outcome of an IPL match before the game begins. Among all formats of cricket, Twenty20 format sees a lot of turnarounds in the momentum of the game. An over can completely change a game. Hence, predicting an outcome for a Twenty20 game is quite a challenging task. Besides, developing a prediction model for a league which is wholly based on auction is another hurdle. IPL matches cannot be predicted simply by making use of statistics over historical data solely. Because of players going under auctions, the players are bound to change their teams; which is why the ongoing performance of every player must be taken into consideration while developing a prediction model.

In sports, most of the prediction job is done using regression or classification tasks, both of which come under supervised learning. In simple terms, y = f(x) is a prediction model which is learned by the learning algorithm from a set of dataset: D = ((X_1,y_1), (X_2,y_2), (X_3,y_3), ... (X_n,y_n)). Based on the type of output (y) supervised learning is divided further into two categories, viz., regression, and classification. In Regression, the output is a continuous value; however, classification deals with discrete kind of output. For predicting continuous values, Linear Regression appeared to be quite effective, and for classification problems like predicting the outcome of matches or classifying players, learning algorithms like Naive Bayes, Logistic Regression, Neural Networks, Random Forests were found being used in most of the previous studies.

In this work, the various factors that affect the outcome of a cricket match were analyzed, and it was observed that home team, away team, venue, toss winner, toss decision, home team weight, away team weight, influence the win probability of a team. The proposed prediction model makes use of multivariate Regression to calculate points of each player in the league and compute the overall strength of each team based on the past performance of the players who have appeared most for the team.

2.1. The Prediction Model

2.1.1. Dataset

The official website of Indian Premier League [35] was the primary source of data for this study. The data was scraped from the site and maintained in a Comma Separated Values (CSV) format. The initial dataset had many features including date, season, home team, away team, toss winner, man of the match, venue, umpires, referee, home team score, away team score, powerplay score, overs details when team reached milestone of multiple of 50 (i.e., 50 runs, 100 runs, 150runs), playing 11 players, winner and won by details. In a single season, a team has to play with other teams in two occasions, i.e., once as a home team and next time as an away team. For example, once KKR plays with CSK in its home stadium (Eden Gardens) next time they play against CSK in their home stadium (M Chinnaswamy Stadium). So, while making the dataset, the concept of home team and away team was considered to prevent the redundancy.

Indian Premier League has just been 11 years old, which is why only 634 matches data were available after the pre-processing. This number is considerably less with comparison to the data available relating to the test or ODI formats. Due to certain difficulties with some ongoing team franchises, in some seasons the league has seen the participation of new teams, and some teams have discontinued. Presence of those

inactive teams in the dataset was not really necessary, but if the matches data were omitted where the inactive teams appeared, the chances were that the valuable knowledge about the teams which were still active in the league would deteriorate. For better understanding and to make the dataset look somehow cluttered-free, acronyms were used for the teams. Table 1 lists the acronyms used in the dataset.

Table 1.   Team names and their acronym.

Team Name	Acronym
Chennai Super Kings	CSK
Delhi Daredevils	DD
Kings XI Punjab	KXIP
Kolkata Knight Riders	KKR
Mumbai Indians	MI
Rajasthan Royals	RR
Royal Challenger Bangalore	RCB
Sunrisers Hyderabad	SRH
Rising Pune Supergiant	RPS*
Deccan Chargers	DC*
Pune Warriors India	PWI*
Gujrat Lions	GL*
Kochi Tuskers Kerala	KTK*

*Inactive teams as of 2018 AD.

2.1.2. Calculating points of a Player

There are various ways a player can be awarded points for their performance in the field. The official website of IPL has a *Player Points* section where every player is awarded points based on these 6 features: (i) number of wickets taken, (ii) number of dot balls given, (iii) number of fours, (iv) number of sixes, (v) number of catches, and (vi) number of stumpings. To find out how IPL management was assigning points to each player based on these 6 features, a multivariate regression was used on the players' points data. Freedman [36] has beautifully explained the mathematics behind the Regression models. For this problem with six independent variables, the multivariate regression model takes the following form:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 \tag{1}$$

Where, y is points awarded to a player, β_0 is Intercept term, β_1 is per wicket weight, β_2 is per dot ball weight, β_3 is per four weight, β_4 is per six weight, β_5 is per catch weight, and β_6 is per stumping weight.

When regression analysis was done on the player's point data, the following values were obtained for the weights β_n in equation 1:

$$\beta_0 = 0, \beta_1 = 3.5, \beta_2 = 1, \beta_3 = 2.5, \beta_4 = 3.5, \beta_5 = 2.5, \beta_6 = 2.5$$

2.1.3. Calculating Team weight

For a team, there can be as many as 25 players. This is a limit put on by IPL governing council to the franchises. To find the average strength of a team, every player of the team is first sorted in the descending order according to their number of appearances in previous matches of the same season. Once players have been sorted, the top 11 players

are considered for calculating the weight of the team because these players have played more games for the team and their performance influence the overall team strength.

$$weight \ of \ a \ team = \frac{\sum_{i=1}^{11} i^{\text{th}} player's \ points}{total \ appearance \ of \ the \ team \ in \ the \ ongoing \ season} \quad (2)$$

Now two more features, viz., home-team-weight and away-team-weight were also added to the previously designed dataset for all matches. Equation 2 was used recursively to calculate the team weight based on the players who appeared the most for the team. Figuring team weight for all 634 matches was a tedious task. So, for example purpose, the final results of each season were considered, and the team weight for each team was calculated accordingly, and the same score was used for all the matches in that particular season. For better performance of the classifier, the team weight must be calculated immediately after the end of each match. This way, the real-time performance of each team and the newly computed weight can be used in predicting upcoming games.

2.1.4. Feature Selection

In this study, Recursive Feature Elimination (RFE) algorithm was used as a feature selection method.

Algorithm 1: Recursive Feature Elimination

---

1 **for** *for each sample iteration* **do**
2   Split data into train and test set.
3   Train the model on training set using all features.
4   Predict on the test set.
5   Calculate features ranking.
6   **for** *for each subset of features S_i, where i = 1 to S* **do**
7   Keep the S_i most important features.
8   Train the model on training set using S_i features.
9   Predict on the test set.
10   **end**
11 **end**
12 Calculate performance of the model over S_i using held-back samples.
13 Identify appropriate number of features.
14 Identify final list of features to keep in the model.
15 Train the model using the optimal set of features on original training set.

As the name suggests, RFE recursively removes an unessential feature from a set of features, re-builds the model using the remaining features and recalculates the accuracy of the model. The process goes on for all the features in the dataset. Once completed, RFE comes up with top k number of features which influence the target variable (independent variable) at a level of extent. Sometimes, ranking the features and using the top k features for building a model might result in wrong conclusions [37]. To prevent this from happening, the dataset was resampled, and RFE was operated in the subsets. The results were the same set of features obtained initially; hence, the initial set of features obtained from RFE did not seem to be biased. Using the RFE

model, the number of features was reduced to 7. Thus obtained features which highly influenced the target variable were the home team, the away team, the venue, the toss winner, toss decision, and the respective teams' weight. Table 2 shows a portion of dataset with seven features.

Table 2. Final set of features considered for designing the prediction model.

home team	away team	toss winner	toss decision	stadium	w1	w2
CSK	RR	RR	field	Dr DY Patil Sports Academy	101.75	123.65625
KKR	DC	DC	bat	Eden Gardens	95.76923077	97.03571429
DD	KXIP	DD	bat	Feroz Shah Kotla	105.9285714	114.9333333
RCB	MI	MI	field	M Chinnaswamy Stadium	90.89285714	102.2142857

w1 represents home-team-weight, and w2 represents away-team-weight

2.1.5. Use of Dummy Variables

There were categorical variables in the dataset. The categorical variable should be converted to dummy variables. Dummy variables are a simple way of introducing information contained in a variable, which is not measured in a continuous fashion [38], e.g., gender, marital status, etc. There are some constraints to be considered while using dummy variables. Any one of the columns (exactly one) should be removed from a set of dummy variables to prevent falling into the dummy variable trap [38]. Since there were five categorical columns in the dataset, each one of them was converted to a set of k dummy variables and the k-1 set of the newly generated dummy variables were used as a representative of the respective categorical variable.

3. Results

A study carried out by Kohavi [39] indicates that for model selection (selecting a good classifier from a set of classifiers), the best method is 10-fold stratified Cross-Validation (CV). This CV approach splits the whole dataset into k=10 equal partitions (folds) and uses a single fold as a testing set and union of other folds as a training set. The creation of folds is random. This process repeats for every fold. That means each fold will be testing set for once. Finally, the average accuracy is calculated out of the sample accuracy from each iteration.

Six commonly used classification-based machine learning algorithms [40], viz., Naive Bayes[41], Extreme Gradient Boosting [42], Support Vector Machine [43], Logistic Regression [44,45], Random Forests [46], and Multilayer perceptron (MLP) [47] were trained on the IPL dataset. The dataset contained all the match data since the beginning of Indian Premier League till 2017. The trained models were used to predict the outcome of each 2018 IPL match, 15 minutes before the gameplay, immediately after the toss. Table 3 shows the performance of all classifiers. Among the six classification models, the MLP classifier outperformed all other classifiers by a notable margin in terms of prediction accuracy and weighted mean of precision-recall (F1 Score). The MLP classifier correctly predicted outcome of 43 matches of 2018 season, with classification accuracy of 71.66% and F1 Score of 0.72. The precision, recall and F1 Score metrics for the MLP classifier is shown in Table 4. Based on the classification accuracy, the MLP classifier was followed by Logistic Regression, Random Forests and SVM classifiers. However, Naive Bayes and Extreme Gradient Boosting classifiers performed poorly in predicting the outcomes of 2018 IPL matches.

Table 5 lists the hyper-parameters of the MLP classifier which were considered experimentally. The MLP classifier was a three-hidden-layered artificial neural network with ten hidden units in each layer. The selection for the number of layers and the number of hidden units in each layer was made experimentally. The activation function in the hidden layer was Rectified Linear Unit (ReLU). Predicting the winner of a cricket match between a home team and an away team is a binary classification problem; hence, a sigmoid function was used as the activation function in the output layer.

Table 3. Performance of the classifiers on 2018 IPL matches

Classifier	Correct Predictions (out of 60 matches)	Accuracy
Naive Bayes	30 matches	50%
Extreme Gradient Boosting	33 matches	55%
Support Vector Machine	38 matches	63.33%
Logistic Regression	41 matches	68.33%
Random Forests	41 matches	68.33%
Multilayer perceptron	43 matches	71.66%

Table 4. Precision, Recall and F1 Score of the Multilayer perceptron classifier

	Precision	Recall	F1 Score
0	0.62	0.59	0.60
1	0.77	0.79	0.78
average	0.71	0.72	0.72

Table 5. Hyper-parameters of the Multilayer perceptron

Hyper-parameter	number/value/function/type
Number of hidden layers	3
Number of hidden units in Layer 1	10
Number of hidden units in Layer 2	10
Number of hidden units in Layer 3	10
Number of units in output layer	1
Activation for hidden layers	ReLU function
Activation for the output layer	Sigmoid function
Optimizer	Adam [48]
Regularization	L2
Initial learning rate	0.001

4. Conclusion

In this study, the various factors that influence the outcome of an Indian Premier League matches were identified. The seven factors which significantly influence the result of an IPL match include the home team, the away team, the toss winner, toss decision, the stadium, and the respective teams' weight. A multivariate regression based model was formulated to calculate the points earned by each player based on their past performances which include (i) number of wickets taken, (ii) number of dot balls given, (iii) number of fours hit, (iv) number of sixes hit, (v) number of catches, and (vi) number of stumpings. The points awarded to each player was used to compute the relative strength of each team. Various classification-based machine learning algorithms were trained on the IPL dataset designed for this study. The dataset contained all the match data since the beginning of Indian Premier League

till 2017. The trained models were used to predict the outcome of each 2018 IPL match, 15 minutes before the game-play, immediately after the toss. The Multilayer perceptron classifier outperformed other classifiers with correctly predicting 43 out of 60, 2018 Indian Premier League matches. The accuracy of the MLP classifier would have improved further if the team weight was calculated immediately after the end of each match. Because this is the only way, the classifier gets fed with real-time performance of the participating teams. The Twenty20 format of cricket carries a lot of randomness, because a single over can completely change the ongoing pace of the game. Indian Premier League is still at infantry stage, it is just a decade old league and has way less number of matches compared to test and one-day international formats. Hence, designing a machine learning model for predicting the match outcome of an auction-based Twenty20 format premier league with an accuracy of 72.66% and F1 score of 0.72 is highly satisfactory at this stage.

Acknowledgements

References

[1] P. Halvorsen, S. Sægrov, A. Mortensen, D. K. Kristensen, A. Eichhorn, M. Stenhaug, S. Dahl, H. K. Stensland, V. R. Gaddam, C. Griwodz, *et al.*, "Bagadus: an integrated system for arena sports analytics: a soccer case study," in *Proceedings of the 4th ACM Multimedia Systems Conference*, pp. 48–59, ACM, 2013.

[2] A. S. Forouhar, M. M. Kellogg, K. Ohiomoba, and E. Akhmetgaliyev, "Methods, systems and software programs for enhanced sports analytics and applications," May 14 2015. US Patent App. 14/398,942.

[3] K. Goldsberry, "Courtvision: New visual and spatial analytics for the nba," in *2012 MIT Sloan sports analytics conference*, vol. 9, pp. 12–15, 2012.

[4] M. Gowda, A. Dhekne, S. Shen, R. R. Choudhury, L. Yang, S. Golwalkar, and A. Essanian, "Bringing iot to sports analytics," in *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*, pp. 499–513, 2017.

[5] R. M. Rodenberg and E. D. Feustel, "Forensic sports analytics: Detecting and predicting match-fixing in tennis.," *Journal of prediction markets*, vol. 8, no. 1, 2014.

[6] H. Pileggi, C. D. Stolper, J. M. Boyle, and J. T. Stasko, "Snapshot: Visualization to propel ice hockey analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2819–2828, 2012.

[7] D. Sacha, M. Stein, T. Schreck, D. A. Keim, O. Deussen, *et al.*, "Feature-driven visual analytics of soccer data," in *2014 IEEE conference on visual analytics science and technology (VAST)*, pp. 13–22, IEEE, 2014.

[8] L. Passfield and J. G. Hopker, "A mine of information: can sports analytics provide wisdom from your data?," *International journal of sports physiology and performance*, vol. 12, no. 7, pp. 851–855, 2017.

[9] R. Rein and D. Memmert, "Big data and tactical analysis in elite soccer: future challenges and opportunities for sports science," *SpringerPlus*, vol. 5, no. 1, p. 1410, 2016.

[10] T. H. Davenport, "What businesses can learn from sports analytics," *MIT Sloan Management Review*, vol. 55, no. 4, p. 10, 2014.

[11] G. Fried and C. Mumcu, *Sport analytics: A data-driven approach to sport business and management.* Taylor & Francis, 2016.

[12] Wired, "The unlikely secret behind benfica's fourth consecutive primeira liga title," May 2017.

[13] T. A. Severini, *Analytic methods in sports: Using mathematics and statistics to understand data from baseball, football, basketball, and other sports.* Chapman and Hall/CRC, 2014.

[14] H. Ghasemzadeh and R. Jafari, "Coordination analysis of human movements with body sensor networks: A signal processing model to evaluate baseball swings," *IEEE Sensors Journal*, vol. 11, no. 3, pp. 603–610, 2010.

[15] K. Koseler and M. Stephan, "Machine learning applications in baseball: A systematic literature review," *Applied Artificial Intelligence*, vol. 31, no. 9-10, pp. 745–763, 2017.

[16] A. Bandulasiri, "Predicting the winner in one day international cricket," *Journal of Mathematical Sciences & Mathematics Education*, vol. 3, no. 1, pp. 6–17, 2008.

[17] M. Bailey and S. R. Clarke, "Predicting the match outcome in one day international cricket matches, while the game is in progress," *Journal of sports science & medicine*, vol. 5, no. 4, p. 480, 2006.

[18] V. V. Sankaranarayanan, J. Sattar, and L. V. Lakshmanan, "Auto-play: A data mining approach to odi cricket simulation and prediction," in *Proceedings of the 2014 SIAM International Conference on Data Mining*, pp. 1064–1072, SIAM, 2014.

[19] A. Kaluarachchi and S. V. Aparna, "Cricai: A classification based tool to predict the outcome in odi cricket," in *2010 Fifth International Conference on Information and Automation for Sustainability*, pp. 250–255, IEEE, 2010.

[20] E. Crampton and S. Hogan, "Cricket and the wasp: Shameless self promotion (wonkish).."

[21] P. McIlroy, "Hawk-eye: Augmented reality in sports broadcasting and officiating," in *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. xiv–xiv, IEEE, 2008.

[22] N. Owens, C. Harris, and C. Stennett, "Hawk-eye tennis system," in *2003 International Conference on Visual Information Engineering VIE 2003*, pp. 182–185, IET, 2003.

[23] B. Bal and G. Dureja, "Hawk eye: a logical innovative technology use in sports for effective decision making," *Sport Science Review*, vol. 21, no. 1-2, pp. 107–119, 2012.

[24] A. L. Samuel, "Some studies in machine learning using the game of checkers. iirecent progress," in *Computer Games I*, pp. 366–400, Springer, 1988.

[25] A. Joseph, N. E. Fenton, and M. Neil, "Predicting football results using bayesian nets and other machine learning techniques," *Knowledge-Based Systems*, vol. 19, no. 7, pp. 544–553, 2006.

[26] Y. Rui, A. Gupta, and A. Acero, "Automatically extracting highlights for tv baseball programs," in *Proceedings of the eighth ACM international conference on Multimedia*, pp. 105–115, ACM, 2000.

[27] R. Chen, A. Hobbs, and W. Maier, "Predicting baseball postseason results from regular season data," *CS229 Projects*, 2017.

[28] B. Bierig, J. Hollenbeck, and A. Stroud, "Understanding career progression in baseball through machine learning," *CS229 Projects*, 2017.

[29] F. C. Duckworth and A. J. Lewis, "A fair method for resetting the target in interrupted one-day cricket matches," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 220–227, 1998.

[30] ESPNcricinfo, "How can the ipl become a global sports giant?," Jun 2018.

[31] Livemint, "Star india eyes 700 million viewers during ipl 2018," Dec 2017.

[32] H. Saikia and D. Bhattacharjee, "On classification of all-rounders of the indian premier league (ipl): a bayesian approach," *Vikalpa*, vol. 36, no. 4, pp. 51–66, 2011.

[33] H. Saikia, D. Bhattacharjee, and H. H. Lemmer, "Predicting the performance of bowlers in ipl: an application of artificial neural network," *International Journal of Performance Analysis in Sport*, vol. 12, no. 1, pp. 75–89, 2012.

[34] S. Kampakis and W. Thomas, "Using machine learning to predict the outcome of english county twenty over cricket matches," *arXiv preprint arXiv:1511.05837*, 2015.

[35] IPL, "Indian premier league official website," 2018.

[36] D. A. Freedman, *Statistical models: theory and practice.* cambridge university press, 2009.

[37] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine learning*, vol. 46, no. 1-3, pp. 389–422, 2002.

[38] D. B. Suits, "Use of dummy variables in regression equations," *Journal of the American Statistical Association*, vol. 52, no. 280, pp. 548–551, 1957.

[39] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International Joint Conference on Articial Intelligenc*, vol. 14, pp. 1137–1145, Montreal, Canada, 1995.

[40] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[41] P. Langley, W. Iba, K. Thompson, *et al.*, "An analysis of bayesian classifiers," in *Aaai*, vol. 90, pp. 223–228, 1992.

[42] T. Chen, T. He, M. Benesty, V. Khotilovich, and Y. Tang, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, pp. 1–4, 2015.

[43] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[44] D. R. Cox, "The regression analysis of binary sequences," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 20, no. 2, pp. 215–232, 1958.

[45] S. H. Walker and D. B. Duncan, "Estimation of the probability of an event as a function of several independent variables," *Biometrika*, vol. 54, no. 1-2, pp. 167–179, 1967.

[46] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1, pp. 278–282, IEEE, 1995.

[47] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.

[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.