IPL Match Analysis and Prediction

Avishi Gupta
IIIT Delhi
avishi19155@iiitd.ac.in

Jahnvi Kumari
IIIT Delhi
jahnvi19469@iiitd.ac.in

Manvi Goel
IIIT Delhi
amanvi19472@iiitd.ac.in

Abstract

Cricket is a globally recognized sport and has a worldwide appeal. It is especially sensational in India to the extent that we established our own T20 tournament called Indian Premier League (IPL) in 2007. IPL is the richest league with the BCCI earning over Rs 2500 crore in FY 2019-20 and is celebrated in many households nationally and internationally. While there have been various statistical studies on ODI and T20 formats of cricket as well as on cricket as a game, we found that IPL also has great potential for research. Additionally, IPL launches various new players, with no past cricket record, hence studies on other high revenue-generating formats like ODIs can not be generalized to it. In this paper, we analyse the datasets to gain insights, apply pre-processing techniques, and compare different models.

1. Introduction

Cricket is a complex sport; there are many parameters for a given match like venue, toss winner, players, etc. These different covariates can be used to study a match and predict victory. The problem statement of our project is as follows:

> Predict winning team for a match after toss using various performance and non-performance metrics.

In this report, we have included a literature survey, the dataset description and pre-processing techniques. After exploring various techniques and understanding the data, we have created models and analyzed the results. We conclude with our learning and individual member contribution.

In this paper, we analyse the two IPL datasets (2008-2020): ball-by-ball data and match summary. The ball-by-ball data has information on various deliveries in different matches, including data on player performance and match summary includes non-performance data like toss winner, venue etc. We use ball-by-ball data to calculate player performance scores and then predict the performance of competing teams using a regression model. We use this performance along with the match summary data to predict the winning team of a match. Our objective is to predict a winning team for a given match after toss decision is declared.

2. Literature Survey

We studied past papers to understand the effect of various parameters of a match on winning. Sasank et al. [8] used the relative strength of the team - found using batsman and bowler ratings to predict the IPL match outcomes for the second innings. Vysali Kand PriyaIyer [4] used data mining algorithms for the prediction of the winner of IPL 2020. Singh and Sharma [2] used batting strike rate, and the bowling run rate in addition to other parameters and predicted the winner using three machine learning models. All three algorithms gave high accuracy results. Lamsal and Chaudhary (2018) [1] selected seven features using Recursive Feature Elimination and predicted the outcome of IPL matches using multiple classifiers. Pithadia (2020) [7] studied the effect of toss, opt choices, and other various factors in predicting the outcome of IPL matches. The author explored various algorithms and found that SVM gave the best results.

Apart from T20 matches, ODI matches have also been studied in great detail in the field of machine learning. Wadhwa and Pathak [6] studied the outcomes and performances of different models and developed a tool that outputs the winning probability of an ODI match. Kaluarachchi and Varde [5] used Bayesian classifiers to predict victory in an ODI cricket match. They studied various factors like a home game advantage, day/night effect, winning the toss, and batting first. Ananda Bandulasiri [3] studied home ground advantage and found a correlation between toss winner and match outcome. The author also analyzed Duckworth Lewis Method.

3. Dataset

We found a rich dataset on Kaggle which contains the match summary of each game played in the IPL from 2008-2020. The dataset has a CSV file with 17 features (columns)

and 816 values (rows). The features are as follows: id, city, date, player_of_match, venue, neutral_venue, team1, team2, toss_winner, toss_decision, winner, result, result_margin, eliminator, method, umpire1, umpire2. We also have a supporting dataset that provides the score and player for each ball in each match. This dataset has 193468 rows and 18 columns. The dataset contains the following features: id, inning, over, ball, batsman, non_striker, bowler, batsman_runs, extra_runs, total_runs, non_boundary, is_wicket, dismissal_kind, player_dismissed, fielder, extras_type, batting_team, and bowling_team.

3.1. Exploratory Data Analysis

We started by exploring the dimensions of our data. Fig 1 shows the number of matches played every year. The most matches were played in 2012 and 2013. We show the percentage of matches won by each team in Fig 2. We can see that Mumbai Indians won the most matches closely followed by Chennai Super Kings. The least the number of matches are won by teams that did not participate in throughout the years and have thus played in very few matches compared to the others.

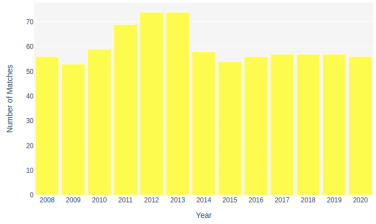

Figure 1. Matches in each season.

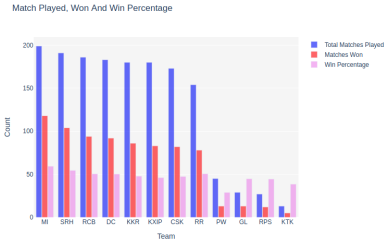

Figure 2. Win percentage for each team.

Fig 4 shows the type of win for each match that is either wickets or by runs. Since both the results are comparable, we conclude that both dismissals and average runs scored can be important features. Fig 3 analyzes the average score per over of the teams as a means to study the performance of players. Fig 5 shows the percentage of times teams chose batting and fielding. It shows teams prefer fielding over batting. Fig 6 shows the percentage of times toss winner wins

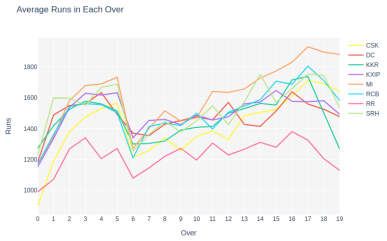

Figure 3. Average runs in each over by each team.

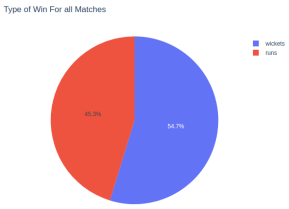

Figure 4. Type of win.

the match and the when they lose the match. It looks like winning the toss does not give much advantage to a team.

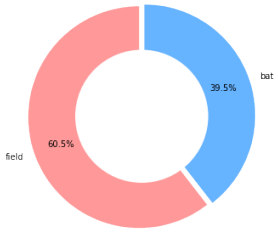

Figure 5. Toss decision percentage.

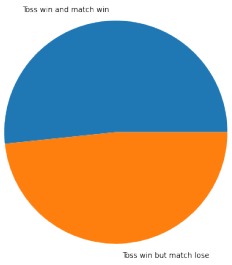

Figure 6. Match winner and toss winner.

Fig 7 shows the percentage of times the toss winner team won the match for different decisions. It is clear that toss winner teams that choose fielding have a more winning percentage than when they chose batting. With Fig 8 we attempt to analyse the home ground advantage to the teams.

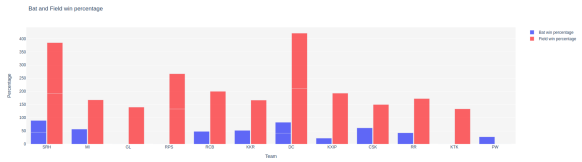

Figure 7. Percentage of win for different decisions of toss winning team.

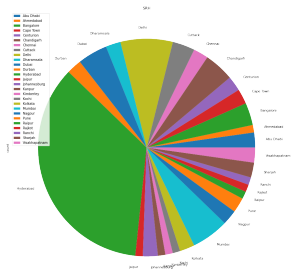

Figure 8. Home ground advantage.

In this case, we analyse the winning percentage of Sunrisers Hyderabad in different cities and we see that the percentage of winning is much higher in Hyderabad.

3.2. Data Preprocessing

For each player, we build a list containing their scores for 2008 to 2019 calculated via the player value formula given on the IPL website. We used various regression models on these values to obtain a prediction for a player's performance in 2020 to analyse the model's performance.

3.2.1 Data Cleaning

Some IPL teams were renamed. e replaced the old names with the latest ones to avoid repetition. We analysed column variance of variance features. Zero-variance predictor may or may not effectively contribute to the model, but it is important to identify such features. We found that neutral_venue, toss_decision, and result have less than 1 percentage of unique values. This is because they have binary values.

3.2.2 Data Integration

We calculated year-wise performance score of each player by using ball-by-ball data. We then used these performance scores to calculate team's total performance score. These team performance scores are appended in the match summary data.

3.2.3 Data Reduction

We are dropping columns like index, venue, result_margin, umpire1, umpire2, match_id, date, and player_of_the_match because they are bad features.

3.2.4 Data Transformation

We are using label encoder to convert the string data from the columns to numeric data. After engineering a multinomial model for the problem, we realized that the model can be improved by introducing a feature to remodel the problem as binary classification. We added a column for the first-team win which is 1 if the first team wins the match and 0 otherwise. A similar feature was introduced for the toss winner.

4. Methodology

Fig 9 shows a schematic representation of our model's pipeline. We are using ball-by-ball data to calculate performance score of each player in different years. We are training our models on 2008-2019 dataset and predicting outcomes for the year 2020. We use the performance scores data to train our regression model and predict the performance scores of players in the year 2020. After predicting the performance measure of each player, we calculate the team performance by averaging the performance scores of all the players playing for the team in the given year. We then use this metrics along with match summary data to predict the winner of the match.

Firstly, we attempted to model the problem as multi-class classification and used Bernoulli Naive Bayes to predict the winning team. We considered this as baseline and other models as improvement on the same. We found the performance of the model to be unsatisfactory. On further analysis, we figured that predicting the winning team from the set of all teams without considering the two participating teams was the reason for the poor performance. To circumvent this problem, we remodeled the problem as binary classification. We trained a logistic regression model on the matches from 2006 to 2019 keeping the default parameters. We tested our model on the matches played in 2020 and found that this model outperformed the previous model by almost 70 percent.

In the next step, we tried different regression and classification models. We achieve the best performance by using Lasso Regression for predicting the player performance and Decision Tree for the final winning team classification.

5. Results and analysis

We explored various model combinations in the search of the best combination. We have tabulated results of our

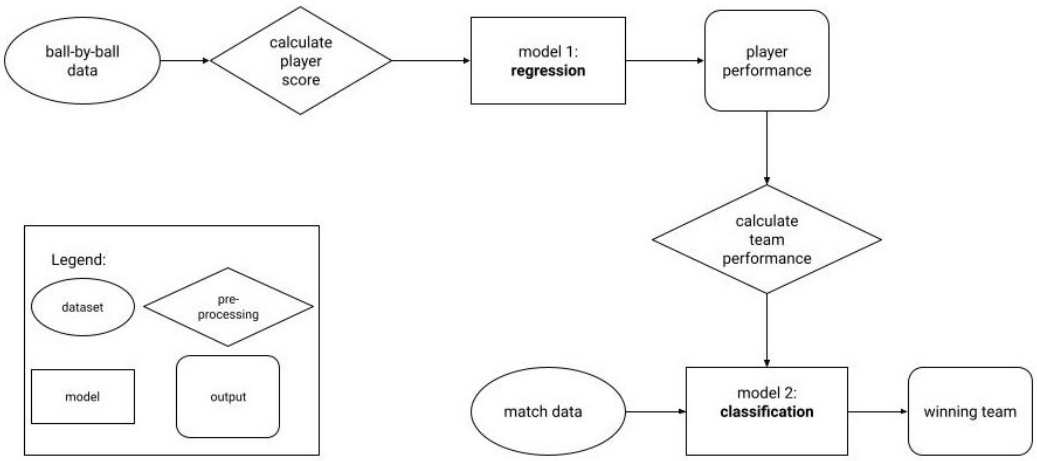

Figure 9. Schematic representation of model pipeline.

Regression Model	MAE	MSE	RMSE
Linear Regression (R)	27.71213303	2508.594006	50.085866
Lasso Regression (LR)	27.47154659	2500.697648	50.006976
Ridge Regression (RR)	27.61724167	2497.929879	49.979294

Table 1. Regression Model Evaluation Metrics

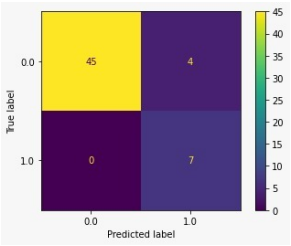

Figure 10. Decision Tree: Confusion Matrix.

best regression models in table 1. We also compare our results with previously evaluated binary classification model without performance metrics (Fig 11) which highlights the importance of performance as a feature. We are not including results of Ridge Regression as they are similar to Lasso Regression for brevity. After analysing the data, we noticed the general trends in the winning teams for IPL matches. We see that the winning teams can be predicted to a certain confidence using the selected features. We explored various models for both regression and classification and found that Lasso Regression with Decision Tree outperforms others.

6. Conclusion

In this paper, we have analyzed and addressed the problem of predicting the chances of victory in a match of the Indian Premier League. The main contributions of our work are:

- Comparison of machine learning techniques revealed that binary classification is the best approach to solve the problem.
- Evaluation of various classifiers over real data proves that the Decision Tree works best over the concerned datasets.
- Introduction of performance measures that improve the prediction model.

As future work, we are planning to expand our analysis using more attributes. It is also possible to apply the machine learning techniques we used in our research to predict the outcome in other outdoor sports such as basketball. The specific approach used may depend on the nature of the given datasets and applications.

References

[1] [1809.09813] Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning.

[2] Shilpi Agrawal, Suraj Pal Singh, and Jayash Kumar Sharma. Predicting Results of Indian Premier League T-20 Matches using Machine Learning. In *2018 8th International Conference on Communication Systems and Network Technologies (CSNT)*, pages 67–71, Nov. 2018. ISSN: 2329-7182.

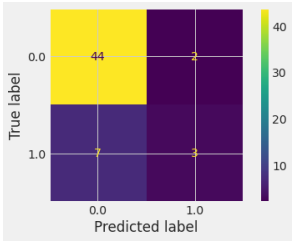

Figure 11. Confusion Matrix: Logistic Regression.

Model Combination	Accuracy	Precision	Recall	F1 Score
Logistic Regression + R	50	0.407	0.44	0.478
Decision Tree + R	62.5	0.963	0.712	0.712
Logistic Regression + LR	83	0.4	0.4706	0.571
Decision Tree + LR	92.86	1	0.777	0.636

Table 2. Model Combination Evaluation Metrics

[3] Ananda Bandulasiri. Predicting the Winner in One Day International Cricket. *Journal of Mathematical Sciences*, 3(1):12.

[4] Harshit Barot, Arya Kothari, Pramod Bide, Bhavya Ahir, and Romit Kankaria. Analysis and Prediction for the Indian Premier League. In *2020 International Conference for Emerging Technology (INCET)*, pages 1–7, June 2020.

[5] Amal Kaluarachchi and S. Varde Aparna. CricAI: A classification based tool to predict the outcome in ODI cricket. In *2010 Fifth International Conference on Information and Automation for Sustainability*, pages 250–255, Dec. 2010. ISSN: 2151-1810.

[6] Neeraj Pathak and Hardik Wadhwa. Applications of Modern Classification Techniques to Predict the Outcome of ODI Cricket. *Procedia Computer Science*, 87:55–60, Jan. 2016.

[7] Anurag Sinha. Application of Machine Learning in Cricket and Predictive Analytics of IPL 2020. Oct. 2020. Publisher: Preprints.

[8] Sasank Viswanadha, Kaustubh Sivalenka, Madan Gopal Jhawar, and Vikram Pudi. Dynamic Winner Prediction in Twenty20 Cricket: Based on Relative Team Strengths. page 10.