Indian Premier League Match Prediction

Aagam Shah, Michael Neuman

Abstract

The Indian Premier League is the most thrilling T20 cricket format watched every year by over 200 million viewers around the world. The aim of our project is to predict the winner of an IPL match based on a team's performance in the previous seasons. Using data from the entire history of the IPL, we develop a model to predict the winner of a match given information from the first part of a match. Using a logistical regression machine learning prediction model, we are able to achieve prediction results with an accuracy of >83% for any given team.

1 Introduction

The Indian Premier League (IPL) is a professional Twenty20 Cricket league in India contested during April and May of every year by teams representing Indian cities. The IPL has recently become a mammoth, money-spinning cricket venture. According to BCCI, the 2015 IPL season alone contributed 182 million USD to the GDP of the Indian economy. International and domestic cricketers participate in the bid to be a part of a team. Each team plays another team twice during the tournament and four teams that score the highest number of points make it to the semifinals with two of them moving further to the final match for the IPL trophy.

Among over 200 million fans across the world, there has always been lot of curiosity and enthusiasm to predict during each game which team will win. Looking at the large number of fans to the format and huge market it resides in, we considered how a machine learning algorithmic approach might help in predicting the result of a match. This would not only become popular among fans but also become useful to the teams in order to strategically design their game plan in each match.

We set out to design a machine learning algorithm that could obtain a highly accurate prediction of the winner of a given match while the match was still going on.

2 Experiment

2.1 Dataset

Our dataset contains match results from the past 9 Indian Premier League Seasons. We had an initial data set containing ball-by-ball information of each match and another data set containing the match-level statistics. The complete first data set consisted of 21 attributes and 136,598 instances and the complete second dataset had 17 attributes and 577 instances. After pre-processing these datasets, we generated a dataset which contained the over-by-over (6 deliveries) match statistics of each of the 577 matches played so far. Our final experimental dataset had 16 attributes and 22,084 instances. Within each instance, there are 16 attributes with the following possible values: *inning, over, team1, team2, batting team, total runs, player dismissed, innings wickets, innings score, score target, remaining target, run rate, required run rate, run rate difference, is batting team, winner.*

2.2 Training and Testing

In order to run predictions for a particular match, we felt it would be more effective to consider statistics from the previous seasons for only those two teams playing in the match. Hence we generated separate datasets for each teams. In order to see how different machine learning algorithms performed on our dataset, we divided the over-by-over instances into 87.5% for training and 12.5% for validation. To be more specific, in this format of 20 overs played by each team, the final 4-5 overs usually tend to be the deciding overs. So we removed last 6 over (as some of the matches get finished before completion of 20 overs in the second innings) and checked accuracy based on

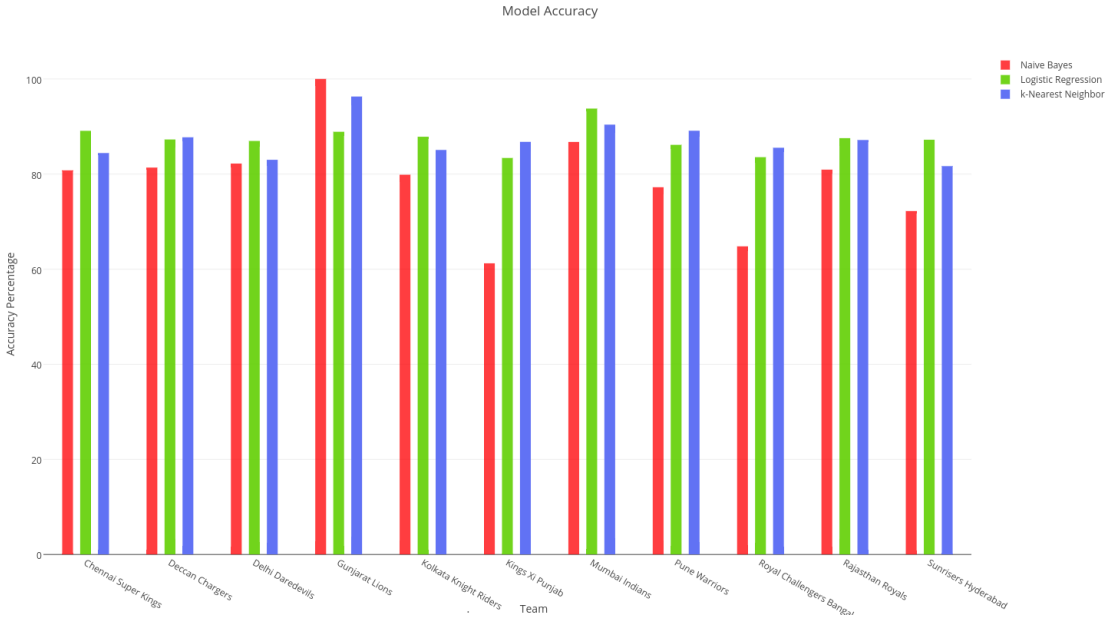

Figure 1: Model Accuracy Predictions By Team

Algorithm	Accuracy
Naive Bayes	78.85%
Logistic Regression	87.42%
k-Nearest Neighbor	87.01%

Table 1: Average accuracy among the different team specific datasets by top three algorithms.

the training set of entire first inning and the first 14 overs of the second inning. Since there is only one winner for match, we replace the winner (classification) column by '?' and declare the winner only on the 14th over of the second inning of a match and we try to run the model for accuracy on the remaining overs of the match.

We tested our dataset on Weka for various machine learning algorithms and calculated their accuracy for different teams manipulating different features (like 'k' in KNN). These algorithms include:

- ZeroR
- J48 Decision tree
- Naive Bayes
- k-nearest neighbor
- Logistic Regression

3 Results

After running each of our models, we obtained the following results:

3.1 Logistic Model Findings

- Chennai Super Kings did not have a strong advantage or disadvantage against any single given team (all team weights were between -1 and 1)
- Kolkata Knight Riders had a strong disadvantage against the Rajasthan Royals (coefficient of -375.5865)

- Large variance in data due to the small number of unique matches (577). The league has only been around for 10 seasons and there has been volatility in team participation. As the league continues, more consistent data can be collected and used to better improve the model. Our hypothesis is that as the teams play each other more often, the coefficients of team variables will decrease in importance and move closer to zero with variations depending on records between a given pair of teams
- Similarly the Gurajat Lions and Delhi Daredevils have only played in two matches which the Delhi Daredevils both won. This explains the large negative coefficient assigned (-34.061)
- Overall, we can see that the logistic model will perform with an accuracy >83% for any given team, meaning we can confidently predict the winner of a match if we have information from the early overs

3.2 Naive Bayes Model Findings

- The 100% accuracy obtained on the Gujarat Lions can be attributed to the small size of the match dataset (231 instances) compared to all other datasets (on average, 2132 instances)
- With the exception of the Gujarat Lions dataset, all other NaiveBayes models were outperformed by the Logistic Regression model
- A NaiveBayes model doesn't completely make sense for our given context as certain parameters are not independent of each other (run rate difference)

3.3 Nearest Neighbor Model Findings

- The Rajasthan Royals were the only team to achieve maximum accuracy with $k \neq 1$ (accuracy was obtained at $k = 5$)
- The Gujarat Lions model's higher accuracy can again be attributed to small dataset size
- Results were comparable to the Logistic Regression model with 5 out of 11 teams' datasets outperforming

4 Future Work

Our model only offers predictions based off previous knowledge of statistics from a given match. One possible area for exploration would be to develop a prediction model to take in two specific teams and give the probability that each team could win prior to a match. Looking even deeper, it would be great if we could develop a probabilistic model that could be used in real-time to see a team's probability during a given match. Currently, our model does not include individual player's statistics as attributes. This is due to lack of a dataset that provides such statistics. One area for future work could be to also include those attributes in our model. This model could then be used by the teams themselves to help them decide which players should be included in a particular match. As the Indian Premier League continues to grow in size and continues through more and more seasons, the right time for developing a great prediction model is now rather than later.

5 Contributions

Since we are a small team of just two members, we performed most of the pre-processing tasks and Weka algorithm analysis together during our weekly group meetings. Michael developed the final website and Aagam worked on this report.