Target Score Prediction in the game of Cricket

Vijay Ramakrishnan **Sethuraman K** **Parameswaran R**
{v.ramakrishnan, sethuraman.krishnan , params.raman}@gatech.edu

Abstract

There has been a recent increase in the use of technology in sports to increase the fairness of the results. The objective of this paper is to apply machine learning techniques to the game of cricket for target prediction in case of interruption to the game. We first evaluate and identify some of the shortcomings of the predominantly used Duckworth - Lewis method (D/L) for target prediction. The data for this was obtained by crawling websites which maintain statistics of cricket matches in the past and selected a set of features which are most likely to affect the outcome of the game. Then, we ran feature estimation algorithms to identify the important features that help in target prediction. Then we discuss how the some of the shortcomings of D/L method can be overcome with Machine Learning ideas. Then we discuss Regression type algorithms which perform as well as D/L method but also take into consideration some of the features that D/L overlooks. Then we provide a framework to compare the target prediction algorithms with one another. Finally we discuss the results of the Machine Learning algorithm against benchmarks and ways to enhance the prediction models.

1 Introduction

Cricket is a team sport which originated in England. The game is contested between two teams of eleven players each. One team bats trying to score as many runs as possible in the stipulated time without being dismissed, while the other team bowls and fields, trying to dismiss the other team's batsmen and limits the runs being scored. When the batting team has used all its available overs or has no remaining batsmen, the roles become reversed and it is now the fielding team's turn to bat and try to outscore the opposition.

In the game of cricket, it often happens that the second half of the match is interrupted due to natural causes like rain. Then it becomes necessary to revise the target to ensure that the team chasing has a fair target and the game produces a result due to play on the field. In other words, given a particular game and its current state, if it is interrupted, what should be the ideal target that should be set accounting for the loss of time due to the natural cause(s) so that the game can have a decisive result and at the same time provide the team chasing a fair chance to play. Currently Duckworth-Lewis (DL) method is popularly used to revise the target of the game in case of interruptions.

Though the Duckworth/Lewis method works a lot better than the "Rain rule", which was used prior to it, it is not perfect. We explore some of the features like dynamic usage of power play, venue and teams that are involved, which are overlooked by the Duckworth/Lewis method, but which we feel are important in making the prediction. We have incorporated these factors and created an efficient model that could represent the pattern of a significant number of matches and reflect all possible scenarios in real-time matches. This model would then be applied to a given current situation determined by a set of features, and used to predict the target.

We then provide a generic mode of evaluating any prediction model with Duckworth/Lewis method and finally propose a new attribute called 'momentum', which we feel would significantly affect prediction in any match.

The challenge faced in such a task is made harder by the fact that Duckworth/Lewis method is 'closed' source and only the run table of the method is available for the public and not the math formulae that make D/L a possibility.

Also very little research has been done so far in field of cricket, this made the task of obtaining data and finding the right algorithm to evaluate the game model tough.

2 Data Extraction

A HTML parser was constructed to extract data from a cricket website. This obtained the commentary of a game on a ball-by-ball basis and calculated the current score and wickets at hand. This was later formatted and written to a file and used later for running the algorithm. We restricted ourselves to the details – Runs, Wickets, Team, Venue, batsman, bowler, and nature of the over. The other details that are available are details pertaining to the batsmen like the batting average and team composition which help in making an accurate prediction of the match progression.

2.1 Dataset Formatting:

General Machine Learning Algorithms are designed and implemented with Classification in mind. Hence the Dataset had to be modeled as a Classification dataset even though the algorithms are used functionally as predictors. The data had to be tweaked in order to be used to make predictions. The total score at the end of the inning was used as a class metric to classify matches which we are trying to predict. Hence we added an additional column to indicate the total score and used it as a particular class. For example, all the matches which have the same total score belong to a particular class. The idea was that the matches which follow a similar pattern would likely lead to a similar result.

As of now scores for all the one day internationals that took place in the year 2009 have been crawled.

3 Algorithms

3.1 Duckworth/Lewis Method

The D/L method sets revised targets in rain-interrupted limited-overs matches in accordance with the relative run scoring resources which are at the disposal of the two sides. These are not in direct proportion to the number of overs available to be faced, as with the average run rate method of correction. Instead they depend on how many overs are to go and how many wickets are down when the interruptions occur. To calculate the revised targets, you need to know the resources available at the stage of the match when suspensions and resumption of play occur. All possible values of resources have been pre-calculated and these are listed in a table. The table covers each individual ball in a game of up to 50-overs per side. The figures given in the table are percentages of the resources available for a complete 50-over innings. For matches with less than 50-overs per innings before they start, the resource percentages available at the start of an innings will be less than 100%. But the same table and the same method of calculation are used irrespective of the number of overs per innings.

When a revised target has been calculated and the match has been played out to its completion, the result is described exactly as in the case of an uninterrupted match; if Team 2 achieve their revised target they win by the number of wickets they have in hand when they reach this score; if they fall short of their revised target by exactly one run the result is a tie, and if they make a lower score Team 1 win by the margin of runs by which Team 2 fall short of the score needed to achieve a tie.

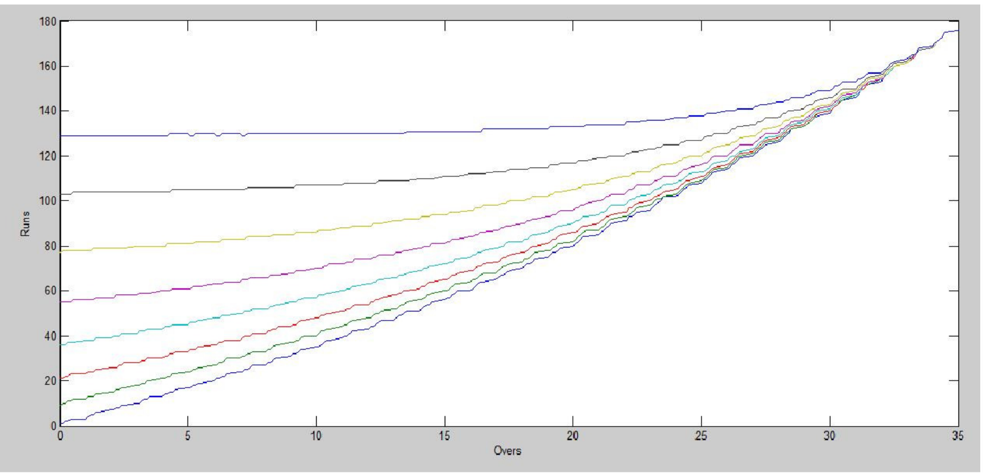

Fig 1. Duckworth Lewis chart for a team chasing 177 in 35 overs (Each graph correspond to a wicket)

3.2 Feature Extraction

We observed that Duckworth/Lewis method used only 2 features runs scored and overs remaining in their target prediction model. Though it is obvious that these are the two most important features for score prediction in the game of cricket, we explored the effect of adding more features. We added the features for venues, teams that are taking part in the game, data regarding the batsman and the bowler and the nature of the over. i.e if it is an over with fielding restrictions applied, also called as power play. Power play is duration of play in the game, usually 5 overs in which there is restriction in field placements. Power plays are usually accompanied by heavy scoring as the task of scoring runs is made easier due to restrictions imposed on the other team.

We ran a few feature estimation algorithms before finally settling on Correlation based subset feature selection with fast correlation based filter search strategy. Correlation based subset feature selection works on the principle that *A good feature subset is one that contains features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other.*

If the correlation between each of the components in a test and the outside variable is known, and the inter-correlation between each pair of components is given, then the correlation between a composite test consisting of the summed components and the outside variable can be predicted from

$$r_{zc} = k \ r_{zi} / \sqrt{(k + k(k-1))} \ r_{ii}$$

where r_{zc} is the correlation between the summed components and the outside variable, k is the number of components, r_{zi} is the average of the correlations between the components and the outside variable, and r_{ii} is the average inter-correlation between components.

This analysis showed that the most important attributes were runs scored, wickets, venue and power play. The same results were validated by other feature estimator like Principal component analysis and Independent Component Analysis with ranking strategy.

The case of power play is vital as they were not dynamic in the time Duckworh/Lewis. i.e now , a power play can be taken by the fielding captain or batsman at any point of time as long as they are eligible to take one.

The feature venue is the other important feature that we will store along with Duckworth/Lewis' features. Even Duckworth/Lewis method accepts the fact that, score prediction needs to take into account 'venue' feature. To this effect, Duckworth and analysis has a variable called G250 which can be used for venue weighted prediction.

Other features like batsman and bowler were not as important. But if the earlier rule of having a super sub in the team was still prevalent, a strong case for including it might be made since it would affect the team composition and by extension would affect the runs/wicket lookup table on which Duckworth/Lewis works.

Feature	**Merit of the Feature**
Run	0.418
Wicket	0.418
Venue	0.095
Power play	0.076
Batsman	0.007

Fig 2. Feature Estimation through CBF Subset selector

3.3 Baseline benchmarking with naïve classifiers

Any target prediction algorithm for cricket that is currently used, does function-approximation on the rate of scoring in a match and used this run scoring model to predict the final score. Since any classifier could also do this we benchmarked the effort of these datasets with standard basic classifiers.

While interpreting the results it is pertinent to keep in mind that we have modeled a predictive model as a classification model. Hence a blind classification rate of 1/# of classes is the lower bound.

The Classification rate of the simple classifiers and the Duckworth/Lewis method are tabulated. Though the classification rates are not as high as the Duckworth/Lewis methods, they are way ahead of the random guess.

Even this classification rate can be improved by with additional data and as of now, we have data only 69 matches played in the year and hence the data sample space is a lot sparser than space over which Duckworth/ Lewis was trained. Hence a low classification rate can only be increased with more data samples.

The classification rates especially of kNN and regression are very good and justify the effort to find the best algorithm. And if we assume that more samples will increase the accuracy and relax the condition for correct target score as original target score +/- 5, then the classification rate of the algorithms go even higher.

Algorithm	Error Rate
Neural Network	51%
Linear Regression	19%
KNN	16%
REPTree	23%

Fig 3. Classification Error rate for different algorithm.

3.4 Quadratic Regression with Smoothing by Neighbour polling

The fact that Linear regression and kNN works well points towards an approach where we can use them together. In [4], the author lists that the curve fitting function that he uses to predict cricket scores is a cubic function. This led us to using quadratic and cubic regression models. On using, a quadratic regression model as the learner, we get a better classification than linear regression.

An analysis of the results suggests that, the absolute mean error can be lowered by smoothening the spikes in the prediction error curve. These spikes occur in the region which is surrounded by reasonably accurate predictions. These spikes might also be caused by incomplete or inadequate data. But for now they can be smoothened by using a neighbour polling. i.e when points corresponding to scores 101, 102,104 predict a score around 250, if the point 103 predicts a score around 250, if the 103 predicts a score of 325, it is highly likely that 103 is a singular value and it might be lowered with more data. But as of now, it can be smoothened by polling the predictions of neighbour, which is very similar to KNN algorithm. Hence the algorithm that we use is a hybrid of quadratic regression model and KNN. This model gives us a classification success rate as good as the Duckworth/Lewis model having seen just 1/1000 of the data that was used to construct the Duckworth/ Lewis method.

In addition to providing good prediction, it also uses the features for venue and power play, which makes the prediction by this algorithm, a more complete prediction.

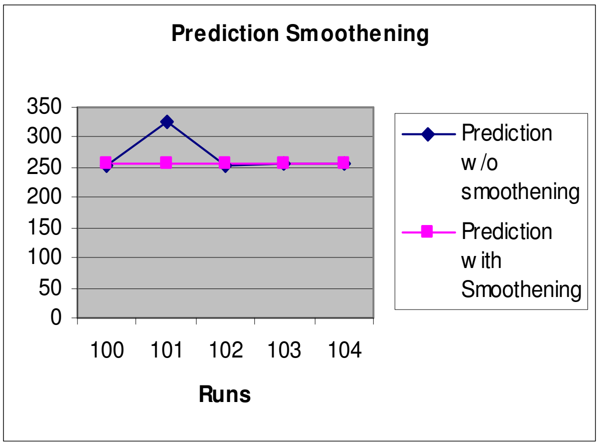

Fig 4 Prediction with smoothening

4 Evaluation metrics for Prediction algorithm:

Through the course of this paper, we realized that it is easy to find better algorithms than D/L and if there is a framework for evaluating the algorithms , then it is easy to compare and rank one algorithm over another. * As of now we have evaluated the algorithms based on variance analysis, where we take the scoring curve of an actual match and try to fit the prediction of the algorithm over the match proceedings and generate a predicted scoring curve. The variance over the whole match and the average variance over all the matches is one of the evaluations metric that we have used.

This metric is similar to absolute mean error, except that this is a per- match (and hence more specific) metric and captures the accuracy locally over a continuous period. This would converge with the absolute mean error and least square error, if each match had the same number of samples.

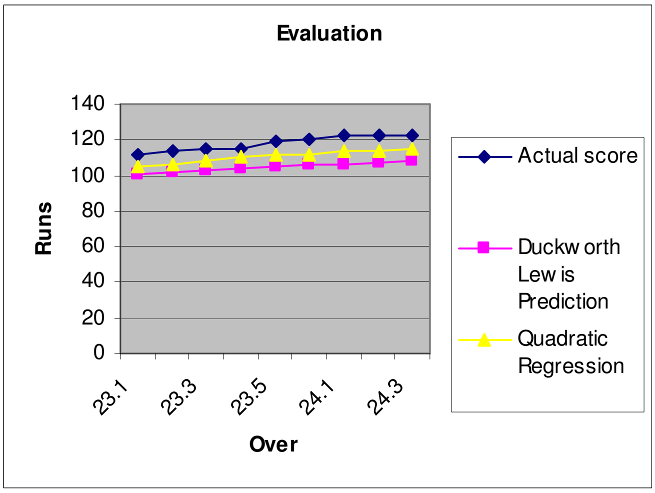

Fig 5 . Evaluation Metrics

5 Prediction of Scores with momentum:

One of the drawbacks of prediction algorithms seen so far is that they try to fit the historical data into a function curve and use this to predict the future match states. This approach though generic and scales well, looses the specificity of the match. For example, say in two instances a match is interrupted at 100/3 at the 25th over. The prediction/extrapolation for both the matches will be the same. But if one of the teams was 90/0 in 15 overs and the other team was 40/3 in 15 overs, then it is highly probable that the second team scores more than the first.

Each of the algorithms seen so far fails to capture this momentum of the innings while predicting the score. If this term is built into the prediction algorithm, it would also answer another accusation aimed at Duckworth/Lewis method , that team chasing do not get to play in the same conditions as the first team. And hence has to reassess and rebuild the innings effectively nullifying any momentum that they might have built before the rain break.

The implementation has a weighted look back variable which carries the weight of the last few overs with it. If the batsmen had scored heavily in the last few overs then, it would make this variable high and hence the predictor would have to take into consideration the high momentum that was lost due to the rain break. The case is vice versa for the reverse situation.

But the downside to this implementation is that when two teams scored the same runs before the rain break and then the second team begins to chase, each of them will have different scores to chase, even though the team batting first has scored the same number of runs. In short the prediction system might lose uniformity.

6 Conclusion:

In this paper, we analyzed the Duckworth / Lewis method of target prediction in the game of cricket and explained the pitfalls in the method. Then we used Correlation based subset evaluation method to do feature evaluation. Contrary to the belief of the Duckworth/Lewis method, the venue of the game and power play overs seems to affect the prediction. Then we benchmarked the predictions of naïve classifiers with a modified dataset. Inspite of the sparse nature of the dataset, regression algorithms and nearest neighbour algorithm did well. Hence a hybrid approach of using quadratic regression model with KNN as a smoothening function was used as a predictor and it did as well as the Duckworth/Lewis method inspite of having 1/1000 of the data to train. Finally we introduced the concept of prediction with momentum of the game as a feature. No other algorithm discussed in the paper has this feature.

References

[1] Duckworth, FC & Lewis, AJ "Your Comprehensive Guide to The Duckworth Lewis Method for Resetting Targets in One-day Cricket", Acumen Books, 2004.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer-Verlag.

[3] Correlation Based feature subset evaluation - Ph D Thesis , Mark A Hall, University of waikato, http://www.cs.waikato.ac.nz/~mhall/thesis.pdf

[4] Jayadevan, V. "A New Method for the Computation of Target Scores in Interrupted, Limited-Over. Cricket Matches." *Current Science* **83**, no. 5 (2002): 577–586.

[5] ICC's explanation of Duckworth Lewis Method: http://in.yimg.com/icccricket/pdfs/d-l_method.pdf

[6]Duckworth / Lewis FAQ, explained in Cricinfo:
http://www.cricinfo.com/link_to_database/ABOUT_CRICKET/RAIN_RULES/DL_FAQ.htm

[7] Multiple Regression Models: Quadratic Regression:
http://www.stat.uiowa.edu/~jcryer/22s008/Lecture%20Notes/chap8.PDF