International Journal of Forecasting 32 (2016) 34–43

Contents lists available at ScienceDirect

International Journal of Forecasting

journal homepage: www.elsevier.com/locate/ijforecast

In-play forecasting of win probability in One-Day International cricket: A dynamic logistic regression model

Muhammad Asif[a,b,*], Ian G. McHale[c]

[a] Centre for Sports Business, Salford Business School, University of Salford, UK
[b] Department of Statistics, University of Malakand, KP, Pakistan
[c] School of Mathematics, University of Manchester, UK

ARTICLE INFO

Keywords:
Binary
Dynamic
Regression
Sports
Cricket

ABSTRACT

The paper presents a model for forecasting the outcomes of One-Day International cricket matches whilst the game is in progress. Our 'in-play' model is dynamic, in the sense that the parameters of the underlying logistic regression model are allowed to evolve smoothly as the match progresses. The use of this dynamic logistic regression approach reduces the number of parameters required dramatically, produces stable and intuitive forecast probabilities, and has a minimal effect on the explanatory power. Cross-validation techniques are used to identify the variables to be included in the model. We demonstrate the use of our model using two matches as examples, and compare the match result probabilities generated using our model with those from the betting market. The forecasts are similar quantitatively, a result that we take to be evidence that our modelling approach is appropriate.

1. Introduction

Unlike soccer, American football and tennis, relatively little work has been published on forecasting in cricket. This seems especially strange given that there are known to be huge betting markets for cricket. The work that has been done on forecasting in cricket has largely been concerned with pre-match forecasting. However, in recent times, the growth in the popularity of in-play betting in all sports, where punters place bets during a game (or match), has meant that models that enable forecasts to be made as the game progresses are in high demand. Cricket is a sport that lends itself particularly to in-play betting: unlike soccer, for example, the discrete nature of the game means that bookmakers and punters alike have ample opportunities to be active in markets during the game, and as such, cricket attracts extremely large in-play betting volumes. For example, the total amount bet during a typical major One-Day International (ODI) involving Pakistan or India is in the order of $1bn (according to a personal communication from a betting industry insider in 2013). In this paper, we present an in-play forecasting model for One-Day International cricket, and use the model to estimate the probability of victory for a team at any moment during a game.

Of course, betting is not the only use of a forecasting model. A forecasting model like that presented here could be used for several purposes. Team coaches may use in-play forecasting probabilities to assess the merits of various different strategies or to analyse player and team performances. In addition, the media could use the model to identify key moments in a match and enhance the television coverage further.

Previous work in One-day International (ODI) cricket has focussed largely on the problem of resetting targets in limited overs cricket following interruptions to play.

[*] Corresponding author at: Centre for Sports Business, Salford Business School, University of Salford, UK.
E-mail addresses: m.asif609@gmail.com, m.asif@uom.edu.pk (M. Asif).

http://dx.doi.org/10.1016/j.ijforecast.2015.02.005

The most well-known work in this area is of course that of Duckworth and Lewis (1998, 2004). In fact, the Duckworth–Lewis method, as their work has been named, can be used as a forecasting tool itself, as it essentially predicts the number of runs that are still to be scored in an innings, given the number of balls remaining and the number of wickets lost so far. Several other authors have developed alternatives to the Duckworth–Lewis method, and these can also be used for forecasting. For example, Preston and Thomas (2002) use dynamic programming to estimate the probabilities of different match outcomes at any stage of the innings, and use this forecasting model to revise targets in interrupted ODI cricket matches. Similarly, Carter and Guthrie (2004) investigate the distribution of the runs remaining to be scored in an innings at any given stage of the innings. They then use this distribution to estimate match outcome probabilities, and go on to use these probabilities to revise targets in interrupted limited overs cricket matches.

Some work has been done focussing directly on the problem of forecasting. Brooks, Faff, and Sokulsky (2002) estimate test match outcome probabilities using an ordered response model. Similarly, for test cricket, Scarf and Shi (2005) forecast match outcome probabilities using a multinomial logistic regression model, with the specific aim of helping team management to decide on the most appropriate time to declare in an innings. Following on from Scarf and Shi (2005), Akhtar and Scarf (2012) present a multinomial logistic regression based model for forecasting the results of Test matches in which predictions are made after each session of play. They estimate 15 separate multinomial logistic models that could be used at 15 particular stages of the match (at the end/start of each session). Such an approach allows for the covariates and their estimated coefficients to vary, session by session. The work that is related most closely to ours is that of Bailey and Clarke (2006), who develop a forecasting model for predicting the margin of victory in limited overs cricket before the match begins, then, with the help of the Duckworth and Lewis (1998) method, update these predictions whilst the game is in progress. However, our methodology differs fundamentally from theirs, in that the effect of a covariate on the match outcome is allowed to evolve as the match progresses.

In this paper, we present an in-play forecasting model for ODI cricket. The model produces forecasts of the probabilities of different match outcomes (win or lose), both once the match has begun, and at each stage through the match as it progresses. The model that we adopt is a dynamic logistic regression (DLR) model, in that the parameters (the coefficients of covariates) are allowed to evolve smoothly as the match progresses. To the best of our knowledge, no such approach to the production of forecasts of the outcome while the game is in progress exists in the literature.

Before presenting our model, we first describe the data that we obtained and the possible covariates that will be experimented with in the modelling. Transformations of the independent variables and the motivations for these transformations are also given. In Section 3, we describe the procedure for developing our dynamic logistic regression model. In Section 4, we present the final model specification and results. In Section 5, we present model fit diagnostics and use two example matches to compare our predicted probabilities with those of the betting market. Section 6 concludes with some closing remarks and discusses potential future work.

2. Data and covariates

We obtained ball-by-ball data for ODI matches played between January 2004 and February 2010. The data were collected from the commentary logs on the Espncricinfo website. We did not include matches for which the data were incomplete, or in which one of the teams had played fewer than five matches prior to the time of play, or in which play was interrupted. In total, we fit our model to data from 606 ODI matches.

The data set includes several variables that could potentially be used as covariates. We divide these variables into two categories: *pre-match* covariates, which are measured prior to the start of the match, and *in-play* covariates, which are measured only during play. In the next two subsections, we explain how and why we experiment with certain variables and functions of variables as covariates in our models.

2.1. Pre-match covariates

Pre-match covariates are variables that can be measured prior to the start of the game. There are a number of factors that might affect the probability of a match outcome before the play has commenced, for example, home advantage, winning the toss to decide whether to bat first or second, a day–night effect, a team's quality, and a team's current form.

In any format of cricket, it is commonly believed that teams who are playing at home experience some sort of advantage. Amongst other possible explanations, in cricket, this is most likely to be because the home team will typically have played many matches at the venue, and therefore will be more familiar with the conditions.

Similarly, winning the toss in cricket is also considered to be an advantage to a team. However, in the literature, the effect of a binary covariate *toss* on the match outcome has not previously attracted statistical significance (see for example Akhtar & Scarf, 2012, and Bailey & Clarke, 2006). Nonetheless, we experimented with including a *toss* variable. Our results on a *toss* variable agreed with previous findings. However, an interaction term with the binary variable day–night (*dn*) was found to be statistically significant. In addition to experimenting with an interaction effect between the variables *toss* and *dn*, denoted by *dnt*, we also experimented with including all other two-factor interaction effects between categorical variables, but none of these were found to be statistically significant.

In regard to the past performances of the teams, we use the difference between the ICC official ODI ratings (*rd*) for the two teams at the time of the match. The ICC official ratings reflect a team's performance based on the matches they have played over the last three years. These ratings are calculated as the total number of points that a team has earned divided by the total number of matches that they have played over the last three years. A team earns points at the end of each match. Broadly speaking, these points depend on the result of the match and the strength of the opposition. For example, a team can get more points if they win against a higher ranked opposition team (for more details, see the ICC official website).

The ICC official ratings go some way toward measuring a team's quality, but do not indicate a team's current form explicitly. For example, in spite of having a low ICC rating of 62 at the time of the ICC Asia Cup 2012 tournament, Bangladesh beat India (with a rating of 117) and Sri Lanka (with a rating of 113), and narrowly lost in the final against Pakistan. To accommodate such streaks of 'good-form', we calculate a team's current form as a weighted mean of match outcomes over their last five games. Specifically, let $y_t = 1$ if a team won the match played t matches ago, and 0 otherwise. We then define the team's current form as

$$form = \frac{\sum_{t=1}^{5} w(t,\theta) y_t}{\sum_{t=1}^{5} w(t,\theta)}, \tag{1}$$

where $w(t,\theta) = (1-\theta)^{t-1}$ and $0 < \theta < 1$.

The current form of a team, as defined above, ranges between zero and one. A team will have $form = 1$ if they have won their most recent five matches and $form = 0$ if none of these matches were won. The function $w(t,\theta)$ is a discounting factor, so that the most recent match receives the highest weight. This implies that, for a given θ, two teams with the same numbers of wins in the last five matches could have different values of the form, depending on the order of their wins. The covariate in the model is the difference in form between the two teams, *fd*. The parameter θ is estimated when fitting the model, as described later, and is found to be 0.2041.

2.2. In-play covariates

The in-play covariates describe the changing state of play (or the position of a team) with respect to the progression of an innings. Fundamentally, the current 'state' of a cricket match can be summarised by three pieces of information: first, the number of runs scored (or the number of runs required to win in the second innings), second, the number of wickets lost, and third, the number of balls, denoted by k (or overs, denoted by $u = k/6$) remaining. The state of play changes with each ball of the game, and the relationship between runs, wickets and balls remaining is not a simple one to describe.

We incorporate runs into our model using two variables, one for each innings. In the first innings, runs are described by the run rate (runs per over), which we denote by *rpo*. In the second innings, on the other hand, what matters is the number of runs per over required for the batting team to win the match. We denote this variable by *rrpo*.

To incorporate information on the number of wickets that a team has lost, we transform the 'number of wickets lost' into wicket resources lost (wrl). One could simply use the 'number of wickets lost', but this does not acknowledge the unequal values of wickets in cricket. Indeed, this non-uniformity is the subject of much of the literature on cricket when targets have to be reset. In accordance with this literature, we believe that the value of losing a wicket should depend on which wicket has been lost and when in the match the wicket was lost. This is partly due to teams putting higher quality batsmen at the top of the order, and partly because the relative importance of each wicket partnership changes as an innings progresses. For example, in a case where there are five overs left, losing the next wicket should have a larger impact on the team's expected remaining runs (and therefore on its win probability) if the batting team has already lost eight wickets than if it has lost only one wicket. In such situations, we believe that it is clear that the resource value of losing the next wicket will have a different value depending upon which wicket is lost and at what stage of the innings. We define wicket resources lost, wrl, as the proportion of the expected runs value lost in the remainder of the innings for the loss of w wickets, as compared to the expected remaining runs with no wickets lost in the remainder of the innings of u overs. This can be written as

$$wrl = \frac{Z(u,0) - Z(u,w)}{Z(u,0)}, \tag{2}$$

where $Z(u,w)$ is the expected runs in the remainder of the innings with u overs to go and w wickets lost. Duckworth and Lewis (1998) model the expected remaining runs, Z, as a function of u and w, as a part of their method for resetting targets in interrupted matches. McHale and Asif (2013) extend this work and propose an alternative model for the Duckworth–Lewis method. We use the McHale–Asif version of the model for the expected remaining runs as a function of u and w.

As defined above, wrl is a continuous variable ranging from zero to one. We multiply it by 10 to give it an intuitive meaning, as there are ten wickets available to each team in cricket. Further, we note that the relationship between wrl and w is dynamic with respect to the progression of the innings. Fig. 1 demonstrates how the relationship between wrl and w evolves (from top to bottom) as an innings progresses. It can be seen that the relationship between wrl and w is more linear in the early stages of an innings (for example, 50 overs remaining) than for the later stages of the innings. This implies that the top order wicket partnerships have smaller wicket resources values than lower order partnerships during the later stages of the innings. This is somewhat intuitive, as a common strategy in limited overs ODI cricket is to play defensively in the early stages in order to save wickets in preparation for more aggressive play in the later stages of the innings.

The Duckworth–Lewis model for Z can be used to estimate the expected remaining runs, for any $u > 0$ and $w = 0, 1, \ldots, 9$. One interesting example of this is to suppose that there is an infinite number of overs remaining, thus approximating a test match innings. The asymptotic behaviour of the Duckworth–Lewis model suggests that the expected number of runs in this case, $Z(\infty, 0)$, is around 290. The modified Duckworth–Lewis model of McHale and Asif (2013) suggests this is around 340.

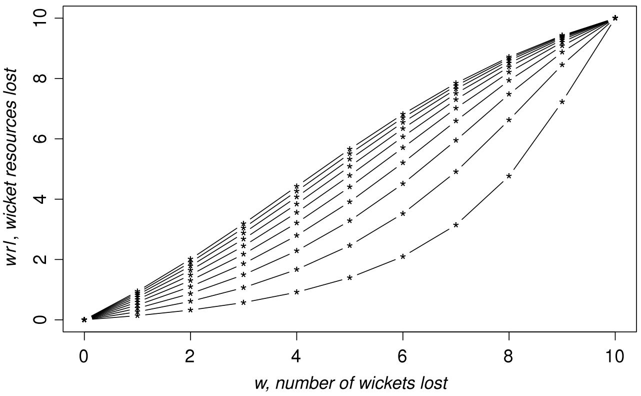

Fig. 1. Plot of the relationship between wickets lost (w) and wicket resources lost (wrl) for each $u = 50$ (top line), 40, ..., 10, 5 (bottom line) overs remaining.

Table 1
Variable names and definitions.

Variable	Description
home	A binary variable taking the value 1 if the reference team is playing at home, otherwise 0.
away	A binary variable taking the value 1 if the reference team is playing at an away venue, otherwise 0.
dn	A binary variable taking the value 1 if the match is a day–night game, otherwise 0.
dnt	*dn* × *toss*, an interaction term between the day–night and toss variables.
fd	A continuous variable, ranging from −100% to 100%, representing the percentage difference between the forms of the reference and opposition teams.
rd	The difference in the ICC official ratings of the reference and opposition teams, immediately prior to the match.
toss	A binary variable taking the value 1 if the batting team wins the toss.
w	The number of wickets lost.
wrl	Wicket resources lost: a continuous positive increasing function, ranging from 0 to 10, of two variables: w, wickets lost, and u, overs remaining.
rpo	Runs per over scored by the reference team.
rrpo	Runs per over required in order for the reference team to win the match.

Table 1 gives the variable names and definitions used in the modelling process.

2.3. Data structure for modelling

To facilitate the modelling procedure, we organize the complete ball-by-ball data into a series of data matrices (one for each ball of each innings, first or second). Letting k represent the number of balls remaining in the innings and $n(k)$ the number of observations for which there were k balls remaining, each data matrix contains the observed $n(k) \times 1$ vector of response variables, $\boldsymbol{y}_k$, and a matrix of independent variables, denoted by $\boldsymbol{X}_k$. For ODI cricket, there are $K = 300$ balls in each innings, so that the data to be used in our dynamic logistic regression model, for one inning, are organized into 300 data matrices. Table 2 is an extract of the data set for $k = 150$ balls remaining (or $u = 25$ overs left) in the first innings of ODI.

In our data, we note that matrices may not all have the same number of rows (sample size, $n(k)$), because not all innings necessarily end with all of the pre-allotted overs having been played. Fig. 2 shows a plot of how the number of data points, $n(k)$, varies with overs remaining. Note that the x-axes for each of the plots in Fig. 2, and in all figures in which the x-axis represents the stage of the innings, are reversed (so that the plot view shows a match progressing from left to right). Furthermore, cricket analysts and fans traditionally think in terms of numbers of 'overs', and therefore the x-axis is shown in units of overs remaining.

It is clear from Fig. 2 that the sample sizes decrease more rapidly for the team batting second than for that batting first. This is because the second innings can end in more ways than the first innings: either the batting team uses all of its wickets or overs (as in the first innings), or it achieves the runs target. Modelling the distribution of the number of balls played in an innings might be an interesting problem to address in future work.

3. A dynamic logistic regression model

We adopt a logistic regression model for estimating the probability of the batting team winning the match. However, the model is dynamic, in the sense that the parameters are allowed to vary as the match progresses. We develop two different forecasting models, one for each innings of ODI cricket. The reason why we use a separate model for each innings is twofold: first, the batting team (reference team) in each innings plays with a different strategy. For example, Preston and Thomas (2002) argue that the team batting in the first innings plays with the aim of scoring as many runs as possible, in order to maximise

Table 2
Extract of the data matrix for the first innings, given $k = 150$ balls remaining. The 'win' column forms the $\boldsymbol{y}_k$ response vector, and the remaining columns (with an added column of ones for the intercept term) comprise the $\boldsymbol{X}_k$ data matrix. Each row represents a different match.

Win	*toss*	*home*	*away*	*dn*	*fd*	*rd*	*w*	*wrl*	*rpo*
0	0	0	1	0	100	14	4	3.320	3.16
1	1	1	0	0	80.96	−1	5	4.402	3.44
1	0	0	1	0	100	5	2	1.454	5.28
1	1	1	0	1	3.05	14	2	1.454	5.12
0	1	0	0	1	13.09	−69	5	4.402	4.40
0	0	0	0	0	−29.7 5	0	2	1.454	3.96
1	1	1	0	0	−72.58	−21	1	0.676	4.48
1	1	0	0	1	−29.75	−3	1	0.676	4.16
1	1	0	1	1	−9.28	−7	1	0.676	5.16
1	1	1	0	1	16.71	−6	4	3.320	2.72
1	1	1	0	1	100	57	2	1.454	5.56

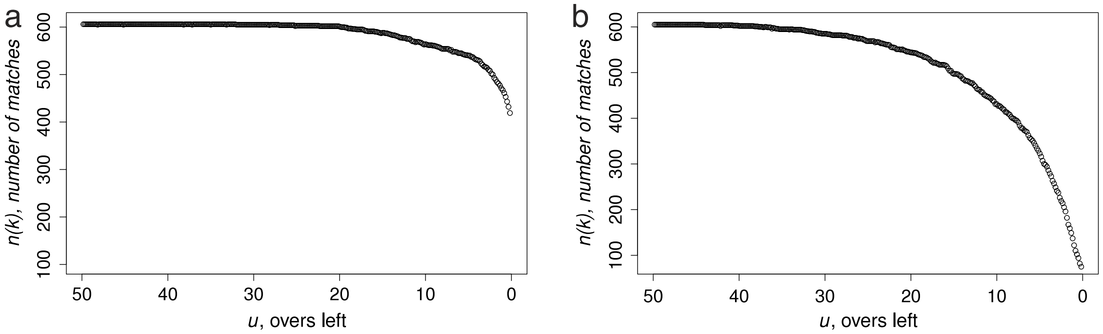

Fig. 2. Sample size (the number of matches, $n(k)$) for each number of overs remaining for the team batting (a) first, and (b) second.

its chances of winning. The team batting in the second innings, on the other hand, plays with the aim of achieving its target before either all of its wickets have been lost or all of its pre-allotted overs have been played. Second, some covariates, for example the number of runs required for each of the remaining overs (*rrpo*) in order to win, exist only in the second innings.

The dynamic nature of our model arises out of an automated iterative process which merges separate and independent logistic regression models.

Let p_k be the probability that the team batting, with k balls remaining ($k = K, K-1, \ldots, 1$), will win the match. For ODI cricket, $K = 300$. For k balls remaining, we use a logit link function, so that

$$\text{logit}(p_k) = \boldsymbol{X}_k^T \boldsymbol{\beta}_k, \tag{3}$$

where $\boldsymbol{X}_k$ is a matrix of $M + 1$ columns (M covariates plus a vector of ones) and $n(k)$ rows, and $\boldsymbol{\beta}_k$ is a vector of regression parameters.

Suppose that we fit K logistic models for each innings; then, the estimated coefficient on the m^{th} covariate, $\hat{\beta}_m(k)$, would be different for each value of k. Indeed, one might expect the values of the estimated coefficients to vary deterministically with k. For example, we will see that the effect of the rating difference covariate, *rd*, on the outcome of the match decreases (in terms of the magnitude of the estimated coefficient) as the end of the match approaches. It is this 'time-varying' nature of the regression coefficients that we wish to mimic. To do this, we add another level of estimation to the logistic model, so that the series of estimated coefficients, $\beta_m(k)$, varies smoothly with k. In the next section, we look at the approach that we use to do this.

The seemingly obvious alternative to our dynamic logistic regression model would be to fit a single model with an additional covariate representing the number of balls remaining. However, the value of the response variable (match outcome) with respect to the ball-by-ball data in a specific innings of a specific match remains unchanged, whilst the in-play covariates change after each ball, meaning that the independence assumption of observations is violated. Fitting a series of K independent models is appealing, in that the sample of matches over which the regression coefficients are estimated are played independently.

3.1. Smoothing the series of estimated coefficients on covariates

After fitting a series of independent logistic models, the next step in building our dynamic logistic regression model is to smooth the estimated coefficients on each covariate.

Let the coefficient on the m^{th} ($m = 0, 1, \ldots, M$) covariate, estimated when there are k balls remaining, be denoted by $\hat{\beta}_m(k)$. We approximate the series $\hat{\beta}_m(k)$ by a smooth function, so that

$$\hat{\boldsymbol{\beta}}_m(k) = f_m(k, \boldsymbol{\alpha}_m),$$

where $\boldsymbol{\alpha}_m$ is a vector of parameters and f_m is some function that must be identified for each of the M covariates.

We note that the identification of the functional form of f_m is somewhat subjective, as it depends on a visual inspection of the scatterplot of the series of parameter estimates $\hat{\beta}_m(k)$ for each m, and a subsequent testing of the statistical significance of the series of estimates. For example, polynomials of varying degrees proved to be appropriate in many cases, however, one can also use splines or other functions.

Once f_m has been chosen, we estimate the parameters $\boldsymbol{\alpha}_m$ by minimising a weighted sum of squares, to account for the heterogeneity present in the $\hat{\beta}_m(k)$. The weighted sum of squares that we minimise is given by

$$WSSE(m) = \sum_{k=1}^{K-1} \left(\frac{\hat{\beta}_m(k) - f_m(k, \boldsymbol{\alpha}_m)}{s.e.(\hat{\beta}_m(k))} \right)^2, \qquad (4)$$

where $s.e.(\hat{\beta}_m(k))$ is the standard error of the estimated coefficient.

After fitting a series of 299 logistic models, each with M covariates and an intercept term, we fit the model $f_1(k, \boldsymbol{\alpha}_1)$ to the series of estimates: $\hat{\beta}_1(299), \hat{\beta}_1(298), \dots \hat{\beta}_1(1)$, by minimising the *WSSE* in Eq. (4) for the first covariate.

We note that the $M + 1$ parameters are not estimated independently for any k-balls-remaining logistic model. Therefore, the remaining M estimated parameters, for each k, must be updated prior to smoothing the next series of estimated coefficients. Therefore, after fitting the smoothing function for the first coefficient on covariate, we refit the series of independent logistic models, but under the parameter constraint $\beta_1(k) = f_1(k, \hat{\boldsymbol{\alpha}}_1)$. After updating the estimates of the remaining M parameters, for each k, we then smooth the next series of estimates, $\hat{\beta}_2(299), \hat{\beta}_2(298), \dots, \hat{\beta}_2(1)$, in a similar way. This process is continued until we have smoothed the estimated intercept terms, $\hat{\beta}_0(299), \hat{\beta}_0(298), \dots, \hat{\beta}_0(1)$. We then follow the same procedure for the second innings model.

The method for choosing which covariates should be included in the model remains to be discussed. To do this, prior to fitting the series of K independent logistic models for an innings, we identify a best subset of candidate covariates that might be included in our DLR model. There are various methods available for these purposes, such as the Akaike information criterion, AIC (Sakamoto, Ishiguro, & Kitagawa, 1986); the Bayesian information criterion, BIC (Akaike, 1977, 1978; Schwarz, 1978); Delete-d Cross-Validation (CVd) with random subsamples of size $d = n(1 - 1/(\ln(n) - 1))$ (Shao, 1997); and K-fold Cross Validation, CVhtm (Hastie, Tibshirani, & Friedman, 2009). Each of these methods has both advantages and disadvantages. Since our aim is to develop a model with the maximum forecasting power, we use the Delete-d Cross-Validation method. However, if the aim of the exercise had been to examine how the effects of the different covariates vary with the progression of the innings, then using the AIC would have resulted in a model with more covariates. In Section 4.1, we explain how the best subset of covariates can be obtained using the CVd method; first, though, we discuss some of the advantages of our DLR modelling approach.

3.2. *Advantages of the proposed dynamic logistic regression model*

To allow the effect of each covariate to depend on the stage of the innings, one could simply use the series of independent separate logistic regression models (one for each ball of an innings), and forecast the probabilities in a standard way. However, one consequence of fitting separate models is that there is an inherent instability in the predicted probabilities of match outcomes that result from the more volatile series of parameter estimates of each covariate. Our model does not have this problem.

A second advantage is that the number of parameters required for forecasting the match outcome in-play is reduced dramatically. For example, if $M + 1$ parameters are to be estimated for each k, then we need $(M + 1) \times 299$ estimates to forecast the match outcome at any stage of the first innings. Admittedly, this large number of parameters is not really an issue during the first innings or in the early part of the second innings, because these estimates are being produced using large sample sizes. However, it becomes problematic in the later stages of the second innings.

The final advantage that we highlight is that, by using a fitted functional value, $f_m(\boldsymbol{\alpha}_m)$, the resulting effect of the covariate on the predicted probability is more stable, and possibly also more precise, as it depends not only on the data matrix associated with k balls remaining, but also on all of the data for the innings. Again, this is a particular advantage in the final stages of the second innings, where the estimates might have a higher variance due to the small sample sizes.

4. Using the DLR to forecast the results of ODI cricket matches

We develop two dynamic logistic regression models, one for each innings. When building the model, the first step is to decide on the best subset of covariates for each innings. The second step is to fit a series of independent logistic models. In Section 4.2, we explain how the series of independent logistic models is reduced to a single DLR model. Finally, model diagnostic measures are used to validate the model. These steps are demonstrated in the subsequent sections.

4.1. *Choosing which covariates to include in the model*

To obtain a series of 'best' independent logistic models during the first innings, we run the bestglm() function in R (R Core Team, 2012) for each $k = 299, \dots, 1$, using all possible covariates. The list of candidate covariates for inclusion in our DLR model is: *home*, *away*, *toss*, *dn* (day–night), *fd* (form difference), *rd* (rating difference), *wrl* (wicket resources lost), *rpo* (runs per over), and all possible two-factor interactions between the categorical variables, for example *dnt* (*dn* × *toss*), or an interaction term between *wrl* and *rpo*. For the second innings, the covariate *rpo* is replaced with *rrpo*, required runs per over. Table 3 shows the frequency of appearance of each covariate in the best logistic model for the first and second innings. Based

Table 3
Number of times (or balls) that each covariate appeared in the best logistic model in each innings. Cross-Validation Delete-d (CVd) model selection was used to decide on the inclusion of the covariates.

	home	*dnt*	*fd*	*rd*	*wrl*	*rpo*	*rrpo*
First innings	0	60	165	299	286	294	NA
Second innings	0	0	0	53	257	NA	299

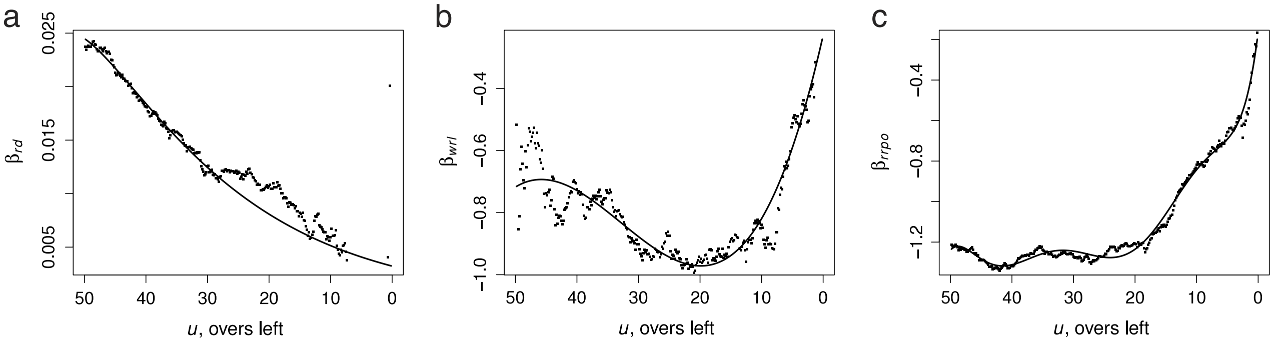

Fig. 3. The estimated coefficients (points) for the series of 299 second innings logistic regression models with covariates *rd*, *wrl*, and *rrpo*, together with the fitted curves (lines).

on the Cross-Validation Delete-d method, only five of the covariates identified appear at least once in the series of the 299 best logistic models. These covariates are *dnt*, *fd*, *rd*, *wrl*, and *rpo*. It is interesting, and somewhat intuitive, that the statistical significance of the pre-match covariates generally decreases as the innings progresses. For example, *dnt* is significant only for the first ten overs of the first innings.

Perhaps the most surprising result that emerges from Table 3 is that the *home* variable is not included in any of the best logistic models using the CVd method. Even when using the AIC for covariate selection, it appears only in the early stages of the first innings. We speculate that the effect of the home advantage may be absorbed into the in-play covariates very early on in a game, as the away players adjust quickly to their unfamiliar surroundings. Further research regarding this point is beyond the scope of this paper.

Upon inspection of Table 3, we chose to include five variables in the DLR model for the first innings, namely *dnt*, *fd*, *rd*, *wrl* and *rpo*, and three variables in the DLR model for the second innings, namely *rd*, *wrl* and *rrpo*.

4.2. Smoothing the series of estimated coefficients

Once the best subset of covariates has been finalized, we use an iterative process to develop a dynamic logistic regression model for each innings. To present the process, we will concentrate on the second innings, as it has fewer covariates.

We start by fitting a series of 299 independent logistic models, each with the covariates *rd*, *wrl* and *rrpo*, to the associated series of 299 data matrices related to the second innings. We then smooth the series of estimated coefficients on *rd* using an appropriate smoothing curve based on the weighted least squares method described in Eq. (4). For example, Fig. 3(a) shows the series of estimated coefficients on *rd* and the fitted smoothing function for the 299 independent logistic models. It can be seen clearly that there is a strong deterministic evolution of the parameter value associated with the *rd* covariate.

One complication in the case of the *rd* covariate is that it is not statistically significant towards the end of the innings. In this case, we adopt a curve that decays towards zero, mimicking the behaviour of the series of parameter estimates. Following a visual inspection of the scatterplot of the series of estimated coefficients on *rd* (Fig. 3(a)), we chose to fit a gamma-type function that is a positive non-decreasing function of u, given by

$$g(u) = c(u_0 - u)^{a-1}e^{-b(u_0-u)}, \tag{5}$$

where $u_0 > 50$, $a > 1$, $b > 0$ and $c > 0$ are the parameters to be estimated.

After smoothing the series of estimated coefficients on *rd*, we update the remaining estimates (the coefficients on *rrpo* and *wrl* and the intercept term) by re-fitting the logistic models for each k under the single parameter constraint related to *rd*. The parameter constraint is to set the value equal to that from the fitted gamma-type function. After updating the estimates, we smooth the estimated coefficients on the next covariate *wrl*. For this covariate, after a visual inspection, we fit a weighted polynomial on the updated estimates, and again fit a series of 299 logistic models under the two-parameter constraint (on *rd* and *wrl*), to obtain updated estimates for the coefficients on *rrpo* and the intercept terms. We continue the process of refitting and smoothing until all of the covariates (including the intercept term) have been smoothed. Fig. 3 provides a graphical representation of smoothing the estimates using our proposed sequential procedure.

In our sequential process of smoothing the estimates, we note that smoothing the estimated coefficients on a covariate has approximately no effect on the remaining

Table 4
Model predicted win probabilities and the proportions of matches resulting in a win. The figures in parentheses show the numbers of matches in each category. For example, there were 56 matches with a predicted probability of victory of between 0 and 0.1 when there were 45 overs remaining of the first innings. Of these 56 matches, 0.09 × 56 = 5 were won.

Predicted	Overs remaining					
probability	First innings			Second innings		
	45	25	5	45	25	5
0–0.1	0.09 (56)	0.04 (69)	0.04 (48)	0.06 (89)	0.04 (161)	0.03 (126)
0.1–0.2	0.06 (48)	0.19 (67)	0.23 (43)	0.12 (51)	0.07 (43)	0.07 (14)
0.2–0.3	0.31 (67)	0.18 (60)	0.16 (49)	0.27 (51)	0.20 (25)	0.33 (9)
0.3–0.4	0.30 (77)	0.35 (62)	0.28 (57)	0.26 (54)	0.37 (27)	0.71 (7)
0.4–0.5	0.48 (84)	0.46 (59)	0.47 (45)	0.53 (36)	0.59 (22)	0.30 (10)
0.5–0.6	0.67 (82)	0.59 (59)	0.57 (44)	0.59 (51)	0.61 (23)	0.46 (13)
0.6–0.7	0.61 (62)	0.58 (66)	0.69 (75)	0.63 (46)	0.66 (32)	0.75 (8)
0.7–0.8	0.64 (58)	0.81 (69)	0.73 (66)	0.74 (43)	0.75 (24)	0.46 (13)
0.8–0.9	0.89 (37)	0.85 (47)	0.88 (57)	0.83 (60)	0.86 (35)	0.83 (24)
0.9–0.10	0.94 (35)	0.96 (45)	0.95 (56)	0.98 (124)	0.96 (177)	0.98 (99)

estimated coefficients. For example, we compare the estimated coefficients on *rrpo* before and after smoothing the estimated coefficients on *rd*, and find that such smoothing has no statistically significant effect on the estimated coefficients on *rrpo*. The same was found for the covariate wrl. One consequence of this property is that the order in which the covariates are smoothed has little effect on the final model predictions. However, we note that if the estimated intercept terms are smoothed before the estimated coefficients then there is an adverse ‘knock-on’ effect on the estimated coefficients on the covariates, and hence on the model predictions. Thus, we recommend smoothing the intercept terms once all of the estimated coefficients have been smoothed. An alternative strategy would be to smooth the estimated parameters in all possible orders and choose the one with the maximum forecasting accuracy. We have developed purpose-written R code to automate this process.

The number of parameters in our final DLR model for the second innings is dramatically lower than that for the independent models. Here, we reduce the 299 models with a total of $4 \times 299 = 1196$ parameters to a single dynamic logistic regression model with just 25 parameters. It is also worth noting the goodness-of-fit of the smoothing curves. The R^2 values are 0.987, 0.8711, 0.990, and 0.999 for the *rd*, wrl, *rrpo*, and intercept terms, respectively. Clearly, this DLR model is more attractive than the less parsimonious alternative.

For the first innings, we follow the same procedure as was used for the second innings, and develop a DLR model with five covariates: *dnt*, *fd*, *rd*, wrl, and *rpo*.

5. Testing the model

A first step in assessing our model validity is to compare the predicted probabilities with those observed. For different categories of the predicted probability of winning from our model, Table 4 shows the observed proportion of matches that finish in a victory. In general, the model-predicted probabilities and the corresponding empirical probabilities are aligned well, in that, reading down the columns, there is a monotonic increase in the observed proportion of wins for each increase in predicted probability band. There are some anomalies, but these are small, or occur when the sample of matches is (very) small.

We present two further tests of our model. First, we use leave-one-out cross-validation to examine the proportions of match results that were predicted correctly by our model as the matches progress, and second, we compare the model predictions to those of the betting market.

5.1. Out-of-sample cross-validation

We now examine the proportions of correct out-of-sample predictions made by the two dynamic logistic regression models (one for the first innings and one for the second innings) using the leave-one-out cross-validation (LOOCV) method. We adopt an admittedly simple prediction rule which predicts the winning team to be the one with the highest probability of winning the match. To reduce the computation time for LOOCV, we consider over-by-over data rather than ball-by-ball data. That is, we fit a series of just 49 independent logistic models (rather than 299 logistic models), smooth the estimates, and finally produce over-by-over estimated outcomes for the out-of-sample match data. Fig. 4 shows the proportion of correct predictions for each over of both innings.

The model's predictive power is high from the start of the game. For example, just ten overs into the first innings, the model predicts the result correctly in over 72% of matches (see Fig. 4(a)). Interestingly, the model's predictive power increases from the start of the innings (as the in-play covariates gather information) until around 18 overs remain. We believe that this is because the matches that are easiest to predict will be ones in which the fifty overs allocated are not completed by the batting side because the team was bowled out (lost all of its wickets). In these matches, the team batting second will tend to have a big advantage, and therefore the prediction is easier to make. For matches when the team batting first did use its allocated fifty overs, predicting the winner is undoubtedly more difficult. Hence, the predictive power decreases over about the last 18 overs.

The over-by-over forecasting accuracy of the model during the second innings is even greater, with the proportion of results predicted correctly rising to over 82%

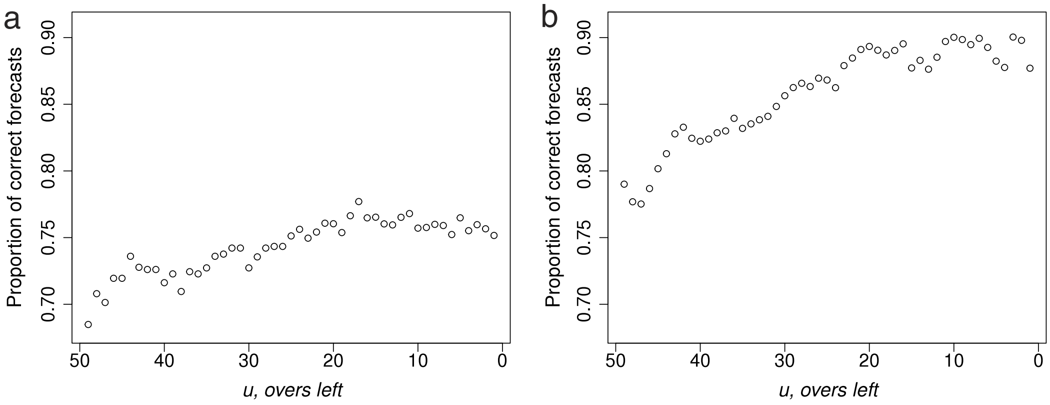

Fig. 4. Leave-one-out cross-validation: proportion of 'correct' forecasts made using DLR models for (a) the first innings, and (b) the second innings.

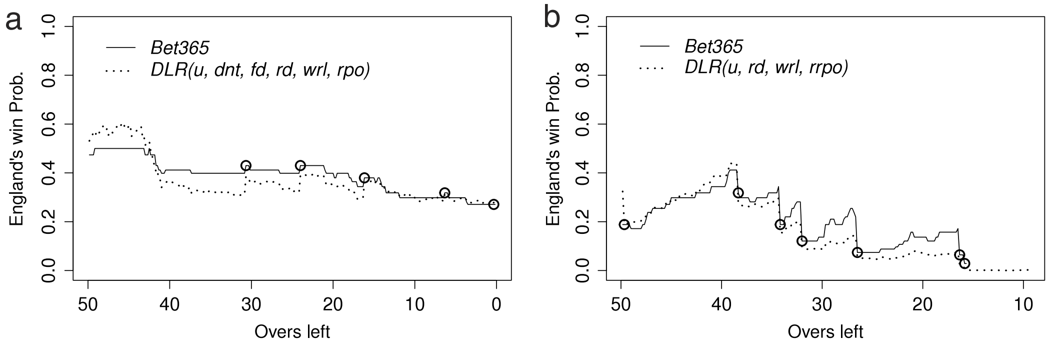

Fig. 5. Forecast probabilities of England winning versus South Africa for each ball of the game in the (a) first innings and (b) second innings. The solid lines represent the implied bookmaker probabilities, whilst the dotted lines represent the forecast probabilities from our DLR model. Circles indicate the loss of a wicket.

after ten overs. Fig. 4(b) shows a pattern similar to that of the first innings model, whereby the predictive power increases throughout the innings, only to fall during the last few balls of the match. Our explanation for this is that the matches that reach the final few balls are the ones in which the outcome is particularly uncertain.

5.2. Comparison with the betting market

Perhaps the sternest test of a forecasting model in sport is to compare it to the betting market. Numerous studies have shown that betting markets are generally efficient, in that it is not possible to beat the market systematically; see for example Sauer (1998). Here, we compare the probability forecasts generated using our dynamic logistic regression models to those implied by the in-play odds from the betting market (http://www.bet365.com) for two ODI matches: the second ODI match of the NatWest series between England and South Africa, played at the Rose Bowl ground in Southampton on August 28th 2012, and the second ODI of the series between Pakistan and Australia in UAE on August 31st 2012. This exercise serves as a good out-of-sample test of our model, as the data on these two matches were not used either when fitting the model or during the cross validation of model selection.

For our first example, South Africa won the toss and elected to bat first; they set England a target of 287, and went on to win the match. Fig. 5 shows the predicted probabilities of England winning the match during the first and second innings.

In general, our model forecasts follow a path similar to that of the bookmaker's forecasts, indicating that our model is performing as one would hope. From around the time when there were 40 overs remaining in the second innings, our model appears to have been 'outperforming' the betting market, in that it is on the correct side of the odds (predicting a defeat for England). Of course, to test the model properly one would need to have a large sample of games and test the model using some betting strategy.

Fig. 6 shows the estimated probabilities for our second example, Pakistan versus Australia. Australia won the toss and decided to bat first, setting a target of 249 for Pakistan to win. Pakistan then went on to win the game by seven wickets. As for the forecast probabilities for our first example, it is a testament to our model that the two predictions follow similar trajectories. In fact, it is noticeable in this example that the model suggests Pakistan's win probability to be higher than that implied by the bookmaker's odds from around midway through the first innings.

Although we only look at two matches, we believe that there is sufficient evidence to suggest that our model is performing well, and that events that occur during a match (like a wicket, or a period of high scoring by the batting team) are incorporated into the model appropriately. In

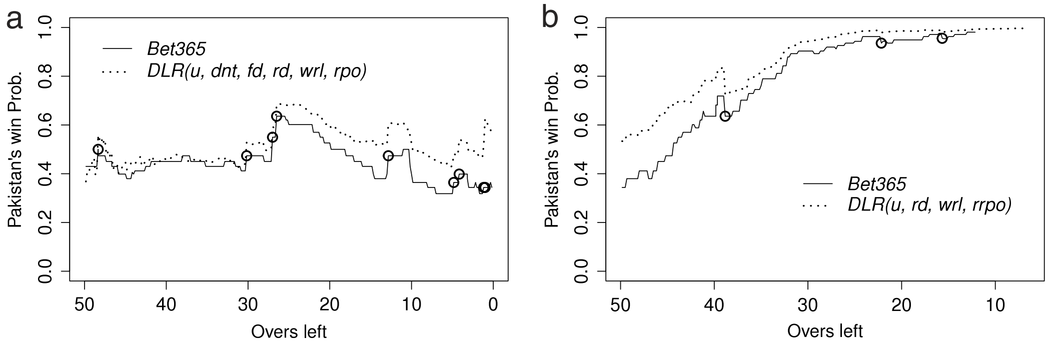

Fig. 6. Forecast probabilities of Pakistan winning against Australia for the (a) first innings and (b) second innings. The solid lines represent the forecast probabilities for an implied bookmaker, whilst the dotted lines represent the probabilities obtained using our DLR models. Circles indicate the loss of a wicket.

future, it would be interesting to experiment with our model as a tool for betting on large numbers of games, in order to form the basis for a more complete test of market efficiency.

6. Conclusions

In this paper, we present an in-play model for forecasting the winner of One-Day International cricket matches at any point through the game. The modelling approach that we take is one in which the estimated coefficients on covariates are allowed to evolve smoothly as a match progresses. We call this model a dynamic logistic regression (DLR) model. Cross-Validation techniques are used for model identification and the assessment of the forecast accuracy. Furthermore, in two examples, our model produces forecasts similar to those of the betting market.

Future work could concentrate on placing our dynamic logistic regression model in a more theoretical framework. In seeking improvements to the model presented here, one may wish to adopt an alternative measure of team strength to the ICC ratings at the time of the game. One may also wish to incorporate a 'pitch effect' to account for certain pitches that may favour high-scoring or low-scoring games.

In addition to forecasting, models like the one presented here could also be used to help inform strategy during a game, or could be used as a part of probability preservation methods for resetting targets in interrupted cricket matches. Finally, our modelling approach could also be used to develop a ranking system for teams and/or players in ODI cricket.

Acknowledgments

We would like to thank the referees and the associate editor for valuable comments on an earlier version of the paper.

References

Akaike, H. (1977). On entropy maximization principle. In *Paper presented at the applications of statistics (proceedings of symposium, Wright State University, Dayton, Ohio, 1976)*. Amsterdam: North-Holland.

Akaike, H. (1978). A new look at the Bayes procedure. *Biometrika*, *65*, 53–59.

Akhtar, S., & Scarf, P. A. (2012). Forecasting test cricket match outcomes in play. *International Journal of Forecasting*, *28*, 632–643.

Bailey, M., & Clarke, S. R. (2006). Predicting the match outcome in one day international cricket, while the game is in progress. *Journal of Sports Science and Medicine*, *5*, 480–487.

Brooks, R. D., Faff, R. W., & Sokulsky, D. (2002). An ordered response model of test cricket performance. *Applied Economics*, *34*, 2353–2365.

Carter, M., & Guthrie, G. (2004). Cricket interruptions: fairness and incentive in limited overs cricket matches. *Journal of the Operational Research Society*, *55*, 822–829.

Duckworth, F. C., & Lewis, A. J. (1998). A fair method for resetting the target in interrupted one-day cricket matches. *Journal of the Operational Research Society*, *49*, 220–227.

Duckworth, F. C., & Lewis, A. J. (2004). A successful operational research intervention in one-day cricket. *Journal of the Operational Research Society*, *55*, 749–759.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *Data mining, inference and prediction* (2nd ed.). New York: Springer-Verlag.

McHale, I. G., & Asif, M. (2013). A modified Duckworth–Lewis method for adjusting targets in interrupted limited overs cricket. *European Journal of Operational Research*, *225*(2), 353–362.

Preston, I., & Thomas, J. (2002). Rain rules for limited overs cricket and probabilities of victory. *Journal of the Royal Statistical Society, Series D*, *51*, 189–202.

R Core Team (2012). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing, ISBN 3-900051-07-0, URL http://www.R-project.org/.

Sakamoto, Y., Ishiguro, M., & Kitagawa, G. (1986). *Akaike information criterion statistics*. Tokyo: KTK Publishing House.

Sauer, R. D. (1998). The economics of wagering markets. *Journal of Economic Literature*, *36*, 2021–2064.

Scarf, P. A., & Shi, X. (2005). Modelling match outcomes and decision support for setting a final innings target in test cricket. *Journal of Management Mathematics*, *16*, 161–178.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461–464.

Shao, J. (1997). An asymptotic theory for linear model selection. *Statistica Sinica*, *7*, 221–264.

Muhammad Asif is Assistant Professor of Statistics at the Unversity of Malakand, Pakistan. In July 2013, Muhammad gained his Ph.D. by studying Statistical Modelling in Cricket at Center for Sports Business, University of Salford, UK. Muhammad has also worked in the sports betting industry as a Sports Statistician, modelling and forecasting cricket, in the United Kingdom.

Ian G. McHale is Reader in Statistics at the University of Manchester, UK. Ian studied extreme value statistics at the University of Manchester to gain his Ph.D., whilst his current research interests include the analysis of gambling markets and statistics in sport. Ian was co-creator of the EA Sports Player Performance Index, the official player ratings system of the English Premier League, and is Chair of the Statistics in Sports Section of the Royal Statistical Society.