A Comparison of Batter Performance and League Quality Across Domestic T-20 Cricket Leagues

Matthew Stuart[1,2], Hassan Raffique[3], Leigha DeRango[1,2]
Gregory J. Matthews[1,2]

[1] Department of Mathematics and Statistics, Loyola University Chicago, Chicago, IL, U
[2] Center for Data Science and Consulting, Loyola University Chicago, Chicago, IL, U
[3] Syracuse University, Syracuse, NY, USA
[+] Corresponding: mstuart1@luc.edu

Abstract

Wicked Googly

Keywords: Cricket

Abstract:

This paper investigates batter performance across major domestic Twenty20 (T20) cricket leagues to understand how player effectiveness varies by competition context. Using a comprehensive dataset comprising individual match-level batting records from the Indian Premier League (IPL), Big Bash League (BBL), Caribbean Premier League (CPL), Pakistani Super League (PSL), and South Africa 20 (SAT), we fit a multinomial regression model for the number of runs scored per bowled ball accounting for batter and league conditions as well as in-match effects. A hierarchical modeling framework utilizing a Bayesian framework is employed to quantify both player-specific ability and league-specific correlations, allowing for quantifiable player comparisons across leagues and the estimation of league-adjusted performance ratings. Our findings will offer insights into talent evaluation and the transferability of performance across T20 competitions.

1 Introduction

openWAR and cricWAR.

In the game of cricket, the number of runs scored on a particular pitch typically ranges between 0 and 6 (though theoretically values larger than 6 are possible, they are rare and do not occur at all in our particular data set).

2 Data

year	BBL	IPL	CPL	PSL	SAT
2011	8	10	NA	NA	NA
2012	8	9	NA	NA	NA
2013	8	9	6	NA	NA
2014	8	8	6	NA	NA
2015	8	8	6	5	NA
2016	8	8	6	5	NA
2017	8	8	6	6	NA
2018	8	8	6	6	NA
2019	8	8	6	6	NA
2020	8	8	6	6	NA
2021	8	8	6	6	NA
2022	8	10	6	6	6
2023	8	10	6	6	6
2024	NA	10	6	NA	NA

We have data from the Indian Premier League (IPL), Big Bash League (BBL), Caribbean Premier League (CPL), Pakistani Super League (PSL), and South Africa 20 (SAT), consisting of 515486 balls thrown, providing $L = 5$ leagues worth of data. From 2015 - 2021, the IPL, the longest running T20 league, had 8 teams followed by 10 teams in the 2022 season. In addition, the SAT was first contested in the 2022-2023 season.

year	BBL	CPL	IPL	PSL	SAT
2011	6459	NA	17013	NA	NA
2012	6383	NA	17767	NA	NA
2013	7853	5385	18152	NA	NA
2014	7816	6314	14288	NA	NA

year	BBL	CPL	IPL	PSL	SAT
2015	7586	7268	13641	5413	NA
2016	8259	6785	14096	5418	NA
2017	10263	7708	13849	7728	NA
2018	13576	7890	14286	8077	NA
2019	13503	8064	14293	6144	NA
2020	14125	7308	14510	4289	NA
2021	14234	8042	14413	12978	NA
2022	14168	7349	17912	8209	7392
2023	9277	7233	17863	7806	7400
2024	NA	2598	17103	NA	NA

Runs in cricket are either scored by running back and forth between the wickets once the ball is put into play (generally resulting in 1 or 2 runs, but theoretically any value is possible). In addition, a ball that is hit in the air over the boundary (termed a "boundary") is worth 6 runs and if the ball rolls to the boundary or bounces in the field of play and then clears the boundary this is worth 4 runs (termed a "boundary 4"). As a result the distribution of runs scored on a a particular pitch has large peaks are 0 and 1 with a big drop off from 1 to 2. Values of 3 and 5 are extremely rare accounting for only 0.4% and 0.02% of values across all pitches in our data set. Values of 4 and 6 spike because of boundaries and boundary fours and together account for 15.82% of all values.

3 Models

Figure 1 displays a histogram of the number of runs scored per ball in IPL, BBL, CPL, PSL, and SAT matches from 2011-2024, consisting of $n = 515,486$ balls thrown. It is likely that a distribution used for modelling counts, such as the Poisson distribution, will violate the necessary assumptions. For this reason, we treat this as a classification problem and fit the number of runs scored per ball, Y_i, by a multinomial distribution. In addition, we exclude any balls that scored three or five runs because of their prementioned rarity of occuring. Thus, we are left with a dataset consisting of $n = 513,343$ balls thrown.

We utilize a mixed effects model, incorporating fixed effects for the general in-match situations as well as random effects for the variability of the bowler, batter, and runner. Denote Y_i as the number of runs scored on ball $i = 1, \ldots, n$ and $\boldsymbol{X}_i$ as the vector of covariates for the fixed effects of ball i. Table @ref(tab:covariates) provides a description of the covariates for the fixed effects of our model. The model is specified with four logit transformations

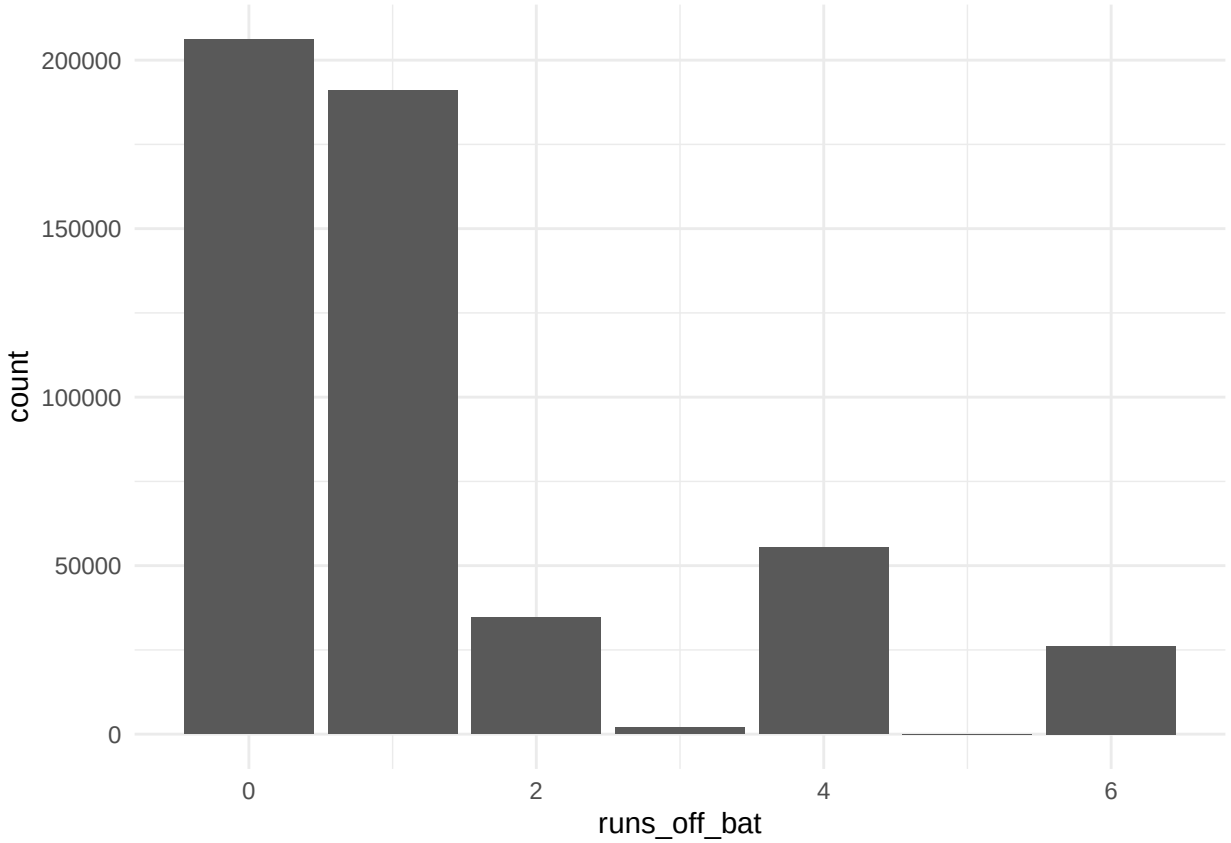

Figure 1: Bar Plot of the number of runs scored of a particular ball

relative to the event $Y_i = 0$, or, written explicitly

$$\log\left(\frac{P(Y_i = k|\boldsymbol{X}_i)}{P(Y_i = 0|\boldsymbol{X}_i)}\right) = \boldsymbol{X}_i\boldsymbol{\beta}_k + u_{k,b_i t_i l_i} \tag{1}$$

for $k \in \{1, 2, 4, 6\}$ where $\boldsymbol{\beta}_k$ is the fixed effect for scoring k runs and $u_{k,b_i t_i l_i}$, is the random effect for the batter(b_i), year(t_i), and league(l_i) for ball i, respectively. For the distribution of the random effects, we denote $\boldsymbol{u}_{k,bt} = [u_{K,bt1}, \cdots, u_{K,btL}]^T$ as the $L \times 1$ vector of the random effects for batter $b = 1, \cdots, B$ and $t = 1, \ldots, T$. To encapsulate the between-league correlation of random effects as well as year-to-year autocorrelation, we model the random effects for each b via a multivariate AR1 process. More specifically, for each $k \in \{1, 2, 4, 6\}$, we set

$$\begin{aligned} \boldsymbol{u}_{k,bt} &\sim \mathcal{N}(\Phi_k \boldsymbol{u}_{k,b(t-1)}, Q_k), \\ \boldsymbol{u}_{k,b1} &\sim \mathcal{N}(0, \Sigma_k) \qquad (2) \\ labelranefs \qquad (3) \\ (4) \end{aligned}$$

for $t = 2, \cdots, T$ where $\Phi_k = [\phi_{k,1}, \cdots, \phi_{k,L}]$ and $\Sigma_k = \Phi_k^{-1} Q_k \Phi_k^{-1}$, ensuring that the model for $\boldsymbol{u}_{k,bt}$ is second-order stationary.

Given the size of the random effects and that we have 116 fixed effects in our dataset, the size of our free, unknown parameter vector $\boldsymbol{\Theta} = \{\boldsymbol{\beta}_k, \boldsymbol{u}_k, Q_k, \Phi_k : k \in \{1, 2, 4, 6\}\}$ is 24588. To handle such a large parameter vector as well as the complicated structure of our model, we perform a Bayesian analysis on the data with prior distributions and sampling procedure outlined in the supplemental file.

Table 3: Description of covariates for fixed effects of model

Variable	Variable.Description
First Innings	Indicator for the 1st innings of the match
Balls Remaining	Number of balls remaining in the innings
Runs to Win	Number of runs remaining to score to win the match (2nd innings)
Runs Scored	Number of runs scored in the innings up to current ball
Wickets Lost	Number of wickets lost in the innings up to current ball
Venue	Grounds in which the match is played

4 Results

5 Discussion, Future work and conclusions

Acknowledgements

Supplementary Material

All code for reproducing the analyses in this paper is publicly available at https://github.com/gjm112/cricketIPL

5.1 Bayesian priors and posterior sampling

In the data analysis outlined in Section 3 of the main manuscript, we set the following prior distributions for the fixed effects $\boldsymbol{\beta}_k$ for $y \in \{1, 2, 4, 6\}$ as well as the variance components of the random effects: Φ_k and Q_k. For ease of use, we decompose $Q_k = \boldsymbol{\tau}_k R_k \boldsymbol{\tau}_k$ where $\boldsymbol{\tau}_k =$

$diag([\tau_{k1}, \cdots, \tau_{kl}]^T)$, the vector of standard deviations and R_k is the associated correlation matrix.

$$\beta_{k,p} \overset{iid}{\sim} \mathcal{N}(0,5), \tau_{k,l} \overset{iid}{\sim} \mathcal{N}(0,1)I(\tau_{k,l} > 0), \phi_{k,l} \overset{iid}{\sim} \mathcal{N}(1,1)I(-1 < \phi_{k,l} < 1), R_k \overset{iid}{\sim} LKJ(2), \tag{5}$$

for $p = 1, \ldots, 116$, where LKJ is the Lewandowski-Kurowicka-Joe distribution for correlation matrices (Lewandowski, Kurowicka, and Joe (2009)). We update the model parameters $\{\boldsymbol{\beta}_k, \boldsymbol{\tau}_k, R_k, \Phi_k : k \in \{1, 2, 4, 6\}\}$ individually using a Metropolis-adjusted Langevin algorithm (MALA). MALA is a version of a Metropolis Hastings algorithm where the new states are proposed using overdamped Langevin dynamics. More specifically, at step r of the algorithm, for each parameter $\boldsymbol{\Theta}$, we sample a proposal

$$\boldsymbol{\Theta}^* \sim \mathcal{N}\left(\boldsymbol{\Theta}_r + a\nabla \log \pi(\boldsymbol{\Theta}_r | \boldsymbol{y}, \boldsymbol{X}), \sqrt{2a}\right)$$

where π is the functional form of the posterior distribution for $\boldsymbol{\Theta}$ and a is a tuning parameter for the proposal distribution. The tuning paramter a is chosen via an adaptation of the primal-dual algorithm from Nesterov (2009), which was also utilized in Homan and Gelman (2014), to obtain an acceptance probability of 0.574.

References

Homan, Matthew D., and Andrew Gelman. 2014. "The No-u-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo." *J. Mach. Learn. Res.* 15 (1): 1593–623.

Lewandowski, Daniel, Dorota Kurowicka, and Harry Joe. 2009. "Generating Random Correlation Matrices Based on Vines and Extended Onion Method." *Journal of Multivariate Analysis* 100 (9): 1989–2001. https://doi.org/https://doi.org/10.1016/j.jmva.2009.04.008.

Nesterov, Yurii. 2009. "Primal-Dual Subgradient Methods for Convex Problems." *Mathematical Programming* 120: 221–59. https://doi.org/10.1007/s10107-007-0149-x.