Analysis of Data in Network and Natural Language Formats

by

Michael John Davis

M.Sc., University of British Columbia - Okanagan, 2011
B.Sc., University of British Columbia - Okanagan, 2008

Dissertation Submitted in Partial Fulfillment
of the Requirements for the Degree of
Doctor of Philosophy

in the
Department of Statistics and Actuarial Science
Faculty of Science

© Michael John Davis 2016
SIMON FRASER UNIVERSITY
Summer 2016

All rights reserved.
However, in accordance with the Copyright Act of Canada, this work may be reproduced without authorization under the conditions for “Fair Dealing.” Therefore, limited reproduction of this work for the purposes of private study, research, criticism, review and news reporting is likely to be in accordance with the law, particularly if cited appropriately.
Approval

Name: Michael John Davis
Degree: Doctor of Philosophy (Statistics)
Title: Analysis of Data in Network and Natural Language Formats
Examing Committee: Chair: Dr. Jiguo Cao
Associate Professor
Chair

Dr. Steven Thompson
Professor
Senior Supervisor

Dr. Tim Swartz
Professor
Supervisor

Dr. Richard Lockhart
Professor
Internal Examiner

Dr. Shane Reese
Professor, Department of Statistics,
Brigham Young University
External Examiner

Date Defended: 6 June 2016
Abstract

The work herein describes a predictive model for cricket matches, a method of evaluating cricket players, and a method to infer properties of a network from a link-traced sample.

In Chapter 2, player characteristics are estimated using a frequency count of the outcomes that occur when that player is batting or bowling. These characteristics are weighted against the relative propensity of each outcome in each of 200 game situations (10 wickets times 20 overs), and incorporate prior information using a Metropolis-Hastings algorithm. The characteristics of players in selected team rosters are then fed into a system we developed to simulate outcomes of whole games. The winning probabilities of each team are shown to perform similarly to competitive betting lines during the 2014 Cricket World Cup.

In Chapter 3 the simulator is used to estimate the effect, in terms of expected number of runs, of each player. The effect of the player is reported as expected runs scored or allowed per innings above an average player in the same batting or bowling position.

Chapter 4 proposes a method based on approximate Bayesian computation (ABC) to make inferences on hidden parameters of a network graph. Network inference using ABC is a very new field. This is the first work, to the author’s knowledge, of an ABC based inference using only a sample of a network, rather than the either network. Summary statistics are taken from the sample of the network of interest, networks and samples are then simulated using hidden parameters from a prior distribution, and a posterior of the parameters is found by a kernel density estimate conditioned on the summary statistics.

Chapter 5 describes an application of the method proposed in Chapter 4 to real data. A network of precedence citations between legal documents, centered around cases overseen by the Supreme Court of Canada, is observed. The features of certain cases that lead to their frequent citation are inferred, and their effects estimated by ABC.

Future work and extensions are briefly discussed in Chapter 6.

Keywords: Simon Fraser University; approximate Bayesian computation; kernel density estimation; network inference; Cricket; Metropolis-Hastings; moneyball; player evaluation; simulated annealing; text analysis
# Table of Contents

Approval ii  
Abstract iii  
Table of Contents iv  
List of Tables vi  
List of Figures viii  

1 Introduction 1

2 A Simulator for Twenty20 Cricket 4  
2.1 INTRODUCTION ................................................. 4  
2.2 PRELIMINARIES ................................................. 6  
2.3 PARAMETER ESTIMATION ....................................... 9  
2.4 EXTENDING THE SIMULATOR ................................. 12  
\hspace{1em} 2.4.1 The Impact of the Bowler .......................... 12  
\hspace{1em} 2.4.2 The Home Team Advantage ......................... 13  
\hspace{1em} 2.4.3 Second Innings Simulation ......................... 14  
2.5 ADEQUACY OF THE SIMULATOR ................. 15  
\hspace{1em} 2.5.1 An Example Concerning the Practical Use of the Simulator 19  
2.6 DISCUSSION ................................................. 20  

3 Twenty20 Moneyball 22  
3.1 INTRODUCTION ................................................. 22  
3.2 OVERVIEW OF SIMULATION METHODOLOGY ............. 25  
3.3 PLAYER EVALUATION ........................................... 26  
\hspace{1em} 3.3.1 Pure Batsmen ........................................... 28  
\hspace{1em} 3.3.2 Bowlers ................................................. 29  
\hspace{1em} 3.3.3 All-Rounders .......................................... 30  
\hspace{1em} 3.3.4 Additional Analyses .................................. 31  
3.4 DISCUSSION ................................................. 34
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2.1</td>
<td>Summary statistics for the T20 dataset corresponding to matches from February 17, 2005 through November 13, 2013. The variables $\bar{R}^{(S)}$ and $\bar{R}^{(A)}$ denote the average number of first innings runs scored and runs allowed, respectively, with the number of matches in parentheses.</td>
<td>6</td>
</tr>
<tr>
<td>Table 2.2</td>
<td>Sample proportions corresponding to batting outcomes in two forms of cricket.</td>
<td>7</td>
</tr>
<tr>
<td>Table 2.3</td>
<td>Batting probabilities (characteristics) for an average batsman, Shane Watson and AB de Villiers at different stages of a match. The quantity $E(RR)$ is the expected run rate for the over where $2/3$’s are treated as $2$’s. Note that $3$’s occur very rarely ($&lt;1%$ of the time).</td>
<td>16</td>
</tr>
<tr>
<td>Table 2.4</td>
<td>Win probabilities for specified 2014 World Cup matches beyond the qualification stage.</td>
<td>20</td>
</tr>
<tr>
<td>Table 3.1</td>
<td>Performance metrics of pure batsmen with at least 250 balls faced. Wicketkeepers are marked with an asterisk.</td>
<td>35</td>
</tr>
<tr>
<td>Table 3.2</td>
<td>Performance metrics of bowlers with at least 250 balls bowled.</td>
<td>36</td>
</tr>
<tr>
<td>Table 3.3</td>
<td>Performance metrics of all-rounders with at least 250 balls faced and 250 balls bowled.</td>
<td>37</td>
</tr>
<tr>
<td>Table 3.4</td>
<td>Pearson correlation between $E(D)$ and four established performance metrics: batting average, strike rate (SR), bowling average and economy rate.</td>
<td>38</td>
</tr>
<tr>
<td>Table 3.5</td>
<td>Comparison of career average and current form characteristics for selected players where the final column denotes the expected number of runs per over.</td>
<td>38</td>
</tr>
<tr>
<td>Table 4.1</td>
<td>F-statistics of models of parameters of interest by summary statistics.</td>
<td>59</td>
</tr>
<tr>
<td>Table 4.2</td>
<td>$R^2$ of models of parameters of interest by summary statistics.</td>
<td>60</td>
</tr>
<tr>
<td>Table 4.3</td>
<td>True parameter values and weighted-posterior mean after two simulation runs.</td>
<td>65</td>
</tr>
<tr>
<td>Table 5.1</td>
<td>Summary statistic values for Supreme Court of Canada cases.</td>
<td>72</td>
</tr>
<tr>
<td>Table 5.2</td>
<td>Summary statistic values for Supreme Court of Canada cases, actual and simulated.</td>
<td>73</td>
</tr>
</tbody>
</table>
Table A.1  Summary statistics of a typical Twenty 20 International match. . . 87
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Proportion of batting outcomes stratified by over. The vertical line denotes the termination of the powerplay.</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Histogram of the number of first innings batting attempts by the 384 batsmen.</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Q-Q plots for Australia and Zimbabwe for first innings runs where the fits appear reasonable.</td>
<td>17</td>
</tr>
<tr>
<td>2.4</td>
<td>Average number of wickets lost versus overs completed for actual and simulated matches.</td>
<td>18</td>
</tr>
<tr>
<td>3.1</td>
<td>Scatterplot of $E(D)$ (current form) against the Reliance ICC rating for batsmen.</td>
<td>39</td>
</tr>
<tr>
<td>3.2</td>
<td>Scatterplot of $E(D)$ (current form) against the Reliance ICC rating for bowlers.</td>
<td>39</td>
</tr>
<tr>
<td>3.3</td>
<td>Scatterplot of $E(D)$ (current form) against the Reliance ICC rating for all-rounders.</td>
<td>40</td>
</tr>
<tr>
<td>3.4</td>
<td>Most recent IPL salary versus current form $E(D)$ for pure batsmen. Triangles represent 2012 salaries, plus signs (+) represent 2013 salaries, and cross signs represent 2014 salaries.</td>
<td>40</td>
</tr>
<tr>
<td>3.5</td>
<td>Most recent IPL salary versus current form $E(D)$ for bowlers. Triangles represent 2012 salaries, plus signs (+) represent 2013 salaries, and cross signs represent 2014 salaries.</td>
<td>41</td>
</tr>
<tr>
<td>3.6</td>
<td>Most recent IPL salary versus current form $E(D)$ for all-rounders. Triangles represent 2012 salaries, plus signs (+) represent 2013 salaries, and cross signs represent 2014 salaries.</td>
<td>41</td>
</tr>
<tr>
<td>3.7</td>
<td>Most recent IPL salary versus ICC Reliance rating for batsmen (top), bowlers (middle) and all-rounders (bottom).</td>
<td>42</td>
</tr>
<tr>
<td>3.8</td>
<td>Scatterplot of $E(D)$ using first and second innings data against $E(D)$ using only first innings data.</td>
<td>43</td>
</tr>
<tr>
<td>3.9</td>
<td>Scatterplot of Croucher’s (2000) statistic for pure batsmen versus $E(D)$.</td>
<td>43</td>
</tr>
<tr>
<td>4.1</td>
<td>Network motifs of order 3, and selected motifs of order 4.</td>
<td>50</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>$\Delta_{used}/\Delta_{Sample}$ statistic, as a function of the population average degree among 2500 generated networks in a simulation run</td>
<td>52</td>
</tr>
<tr>
<td>Figure 4.3</td>
<td>Sample of order 400 from the population of interest, with all the edges found in link-tracing in the left panel, and with only the recruiting edges retained in the right panel</td>
<td>57</td>
</tr>
<tr>
<td>Figure 4.4</td>
<td>$\Delta_{used}/\Delta_{sample}$ statistic, closeness preference parameter $\gamma$ and population average degree among 500 randomly generated networks in an initial simulation run</td>
<td>61</td>
</tr>
<tr>
<td>Figure 4.5</td>
<td>$\Delta Degree/\Delta_{sample}$ statistic, population network order parameter $N_V$, and population average degree among 500 randomly generated networks in an initial simulation run</td>
<td>62</td>
</tr>
<tr>
<td>Figure 4.6</td>
<td>Initial round weighted-posterior density of infective probability, number of population nodes, initial infection proportion, and preference for closeness in forming connections, shown as two-dimensional marginals</td>
<td>63</td>
</tr>
<tr>
<td>Figure 4.7</td>
<td>Final round weighted-posterior density of infective probability, number of population nodes, initial infection proportion, and preference for closeness in forming connections, shown as two-dimensional marginals</td>
<td>64</td>
</tr>
<tr>
<td>Figure 5.1</td>
<td>Induced subgraph of the citations between 1500 Supreme Court of Canada cases. Each point represents a case. The size of the point (above a minimum) represents the number of cases from all sources that cite this case. The vertical position of each point represents the year of that case (i.e. the ‘birth’ time). Lines between points represent a citation from the newer case to the older one. The horizontal position of points is arbitrary, and the points have been arranged to minimize the number of crossing lines while preserving vertical position.</td>
<td>71</td>
</tr>
<tr>
<td>Figure 5.2</td>
<td>Probability density maps of $\beta_{corp}$, $\beta_{crown}$, $\beta_{dis}$, and $\beta_{pa}$</td>
<td>76</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The perfect data set is a long-format matrix of 50,000 rows of 5 columns. It is a sensible ideal because it allows for the demonstration of many methodologies. A large set of complete data in long format is the statistical equivalent to a frictionless vacuum in physics, or a collection of rational, informed agents in economics. That is, such a data format is good for teaching principles and methods, but is rarely found in practical scenarios.

In my five years at Simon Fraser University, I’ve had the privilege of hearing from dozens of guest lecturers from academies and countries around the world. Several of these lecturers from industrial and government settings have shown estimated breakdowns of their work time; all of them said the majority of their time was spent formatting data. The recurring theme throughout this dissertation is unusual data types (specifically network and textual data). That is, formats of data far removed from the long-format ideal that so many methods depend upon.

Chapters 2 and 3 are copies of the papers by Davis, Perera, and Swartz (2015b and 2015a) respectively. Everything in these papers is derived from the archived text summaries started as an MSc project by Dr. Paramjit Gill and myself in the summer of 2011. This text analysis was picked up and continued in 2013 with Dr. Tim Swartz and fellow doctoral student Gamage "Harsha" Perera and resulted in a parsing system that can extract pertinent information about every ball bowled in a cricket match. The simulator we developed that uses this ball-by-ball data has been used in four papers as of September 2015, including the two included in this thesis and two more by Perera, Davis, and Swartz (2015a and 2015b). There are at least two more related papers in progress that are discussed in the last chapter. A large majority of the data gathering and programming work involved is in the text parser and in the simulator I developed.

More than 1400 games from One-Day International, Twenty20 International, and Indian Premier League cricket were processed. The resulting datasets are collectively more than 750,000 rows and 20 columns of detailed ball-by-ball data. Effectively, the thousands of lines of cricket text have been put in a frictionless vacuum.
The work in Chapters 2 and 3 focuses on two applications of the textual analysis, and not the text analysis itself. In Chapter 2, player characteristics are estimated using a frequency count of the outcomes that occur when that player is batting or bowling. These characteristics are weighted against the relative propensity of each outcome in each of 200 game situations, and incorporate prior information using a Metropolis-Hastings algorithm. The characteristics of players in selected team rosters are then fed into a system we developed to simulate outcomes of whole games. The winning probabilities of each team are shown to perform similarly to competitive betting lines during the 2014 Cricket World Cup.

The game simulator developed in Chapter 2 was used to estimate the number of runs for and runs against each proposed lineup. The work in Chapter 3 applies the simulator to estimate the effect, in terms of expected number of runs, of each player. The first part of this method is finding the average characteristics of players in each place in the batting order, against 20 overs of bowlers with similarly averaged characteristics, and simulating to get the expected runs for that theoretically average team’s games to get a baseline. The second part is to, for each player of interest, find their archetype (e.g. pure batsman, all-rounder, bowling specialist), replace the appropriate average characteristics with those of the player of interest, and simulate another batch of games. The effect of the player estimated by the difference in the average runs per innings in the simulations with the player and in the average runs per innings in the simulations using only theoretical average players. A similar application of the game simulator is found in Perera, Davis & Swartz (2015a).

Chapter 4 diverges from the previous two in that deals with network data rather than textual data, and does not have any obvious application to the sport of cricket. However, like textual data, network data is relatively poorly explored by statistical methodologists. In many other methodologies that deal with better explored data formats, the relationship between observations is assumed. For regression analysis, every observation is assumed to be independent. In a time series, every observation is modeled to have a set relationship, up to coefficient values, with the observations that came before it. In a network sample, some of the relationships between the observations, in essence the elements of the graph structure of the network, are unknown. Often these relationships are the parameters of interest. Furthermore, inference is still non-trivial when the the entire population (i.e. the entire network graph of interest) is known, as useful inference can still be done on the process that generated the network.

Chapter 4 proposes a method of estimating such network generating parameters. The method in this chapter was inspired by a network that was delivered in a way that systematically discarded some of the connections between the observations, thus presenting a triple-layered problem: Estimate the generating parameters of the network, for which there is only a sample, and only a portion of the sample data is retained. Because of these difficulties, we chose to adapt and employ a computationally intensive ‘last resort’ estimation method of simulation and rejection called approximate Bayesian computation (ABC).
To run an approximate Bayesian computation, a set of summary statistics are taken as a rejection criterion, and set of simulations are done to produce datasets (in this case network samples) for a joint prior of the parameters of interest. Network samples that have summary statistics close to the actual observed sample are kept in the posterior and the rest are rejected, where ‘close’ is determined by a distance function selected by the user.

Although there is a recent and growing body of literature on using ABC to solve otherwise intractable problems, and lots of research interest on network inference, very little work has been on using ABC for Network Inference, and I believe attempting to use ABC to infer generating parameters from a sample of a network is novel. Furthermore, it can take in a wide variety of summary statistics, and can be used for sampling protocols besides the one we chose to work from, making it a significant advancement rather than a special case.

Chapter 5 is an application of the approximate Bayesian computation-based method from Chapter 4, but to a naturally occurring database of citations between Supreme Court of Canada rulings. These cases (nodes) and the citations (edges) between them form a natural directed network with citations always pointing backwards in time. This network has the additional complications of being a sample of a much larger network of Canadian legal documents, and growing with new cases and citations regularly. Furthermore, the parameters being estimated are temporal in nature. Like the cricket data analyzed in Chapters 2 and 3, these Supreme Court of Canada cases are originally in a natural language format with some conventions that can be exploited.

Chapter 6 is a brief discussion of future work, and work not otherwise covered. This includes discussion of a recently completed paper in which the fielding abilities of cricket players is evaluated. A fielding player is one who is not currently bowling or batting. I also propose an expansion of the Cricket analytics work to estimate player characteristics across multiple leagues. Computational methods used in ABC that are potentially novel are also mentioned here. Also mentioned are intended future applications of ABC-based network inference to infectious disease and to deeper legal analysis.

Appendix A describe some of the text processing work that was necessary to collect and format the data for the cricket analysis in Chapters 2 and 3. Appendices B and C describe the estimation of situational effects which are key to finding and using the player characteristics that we researched in Chapters 2 and 3.
Chapter 2

A Simulator for Twenty20 Cricket

2.1 INTRODUCTION

The game of cricket has a long history dating back to the 16th century. The most recent form of cricket, known as Twenty20 cricket (or T20 cricket), began in 2003 involving matches between English and Welsh domestic sides. Since 2003, T20 cricket has exploded in popularity with five World Cups having been contested (2007, 2009, 2010, 2012 and 2014). The Indian Premier League (IPL) which had its inaugural season in 2008 is known as the showcase for T20 cricket. The IPL continues to grow in popularity with respect to the number of teams, television contracts, salaries, etc.

Except for some subtle differences (e.g. fielding restrictions, limits on the number of overs for bowlers, etc.), T20 cricket shares many of the features of one-day cricket. One-day cricket was introduced in the 1960s, and like T20 cricket, it is a version of cricket based on limited overs. The main difference between T20 cricket and one-day cricket is that each batting side in T20 is allotted 20 overs compared to 50 overs in one-day cricket. This difference allows T20 matches to finish in roughly three hours, a length of time comparable to the duration of matches in many other professional sports.

Simulation methodologies have been developed and proven useful for many types of complex systems. For example, the simulation of weather systems using mathematical models has a long history in both short-term weather forecasts and in the prediction of climate change (Lynch 2008). A match simulator for T20 cricket would likewise be useful. For example, the prediction of match outcomes is obviously of interest to cricket enthusiasts. A match simulator would also facilitate the investigation of various match characteristics for which there does not exist a sufficient number of actual matches. For example, suppose that a T20 team is considering a new batting lineup. They may be interested in the distribution of runs scored by the hypothetical lineup. Naturally, a good simulator for T20 cricket is one which is realistic and captures the complexity of the game. To our knowledge, there have not been any realistic simulators developed for T20 cricket. A difficulty in the development
of a realistic T20 match simulator involves gaining a detailed understanding how various interacting factors (e.g. overs, wickets, batsmen, bowlers, the target, the powerplay\(^1\), etc.) affect run progression.

Simulators have been investigated for other forms of cricket. The earliest “simulators” were proposed by Elderton (1945) and Wood (1945) who fitted simple geometric distributions for the number of runs scored in test cricket. Dyte (1998) also considered the simulation of test cricket matches where the only inputs were career batting and bowling averages. In one-day cricket, Bailey & Clarke (2006) introduced covariates related to run scoring and used the normal distribution for the generation of runs. In test cricket, Scarf et al. (2011) model the number of runs by fitting a zero-inflated negative binomial distribution to each of the 10 partnerships. More closely related to this paper, Swartz, Gill & Muthukumarana (2009) developed a Bayesian latent variable model which provided batting outcome probabilities in one-day cricket. A criticism of Swartz, Gill & Muthukumarana (2009) is that they use a coarse discretization of wickets lost and overs consumed based on 9 overall categories. In particular, their structure does not account for powerplays.

Section 2.2 is concerned with preliminaries related to the T20 simulator. We first introduce the extensive dataset which is used throughout the paper. Exploratory data analyses are carried out to motivate the subsequent modelling. The T20 simulator is then described in simple terms for first innings batting. Section 2.3 discusses the inputs to the simulator. Specifically, batting outcomes are enumerated and the corresponding probabilities are derived from multinomial distributions. Our model is highly parametrized and we use an amalgam of classical estimation techniques and a hierarchical Bayesian model to estimate the multinomial parameters. One of the key features of the approach is that the estimators from a given scenario borrow information from related scenarios to improve reliability. Another noteworthy aspect of the approach concerns the detail which is provided in ball-by-ball scoring. Simulators which simply generate the total number of runs for each over do not address the manner in which runs are scored. In Section 2.4, the simulator is extended in various ways. We consider the case of specific batsman/bowler matchups, the home team advantage and second innings simulation where the target score is taken into account. Clearly, higher target scores force the second innings batting team to be more aggressive. When they are more aggressive, they score more runs but are more likely to be dismissed. In Section 2.5, we demonstrate the realism of the simulator via some goodness-of-fit diagnostics. A notable consequence of the validation exercise is the suggestion that teams may not be batting optimally during the second innings. Specifically, teams that are falling behind in the second innings may not be increasing their aggressiveness in an incremental fashion. We also illustrate the utility of the simulator by addressing some problems of prediction. We conclude with a short discussion in Section 2.6.

\(^1\)the powerplay is defined later
2.2 PRELIMINARIES

For the analysis, we consider all T20 matches that took place from 2005 until the end of 2013 which involved full member nations of the International Cricket Council (ICC). Currently, the 10 full members of the ICC are Australia, Bangladesh, England, India, New Zealand, Pakistan, South Africa, Sri Lanka, West Indies and Zimbabwe. Details from these matches can be found in the Archive section of the CricInfo website (www.espncricinfo.com). A proprietary R-script was used to parse and extract ball-by-ball information from the Match Commentaries. In total, we obtained data from 250 matches. In Table 2.1, we provide summary statistics for the matches where we observe that Bangladesh and Zimbabwe are clearly the weakest T20 teams. Amongst the other 8 teams, the winning percentages do not vary greatly. When looking at the differences between runs scored versus runs allowed for individual teams, it appears that Sri Lanka's win percentage is lower than what might be expected.

<table>
<thead>
<tr>
<th>Team</th>
<th>Matches</th>
<th>Win %</th>
<th>$\bar{R}^{(S)}$</th>
<th>$\bar{R}^{(A)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australia</td>
<td>63</td>
<td>52%</td>
<td>161.7 (33)</td>
<td>157.8 (30)</td>
</tr>
<tr>
<td>Bangladesh</td>
<td>22</td>
<td>18%</td>
<td>134.0 (07)</td>
<td>169.5 (15)</td>
</tr>
<tr>
<td>England</td>
<td>58</td>
<td>55%</td>
<td>159.7 (23)</td>
<td>158.7 (36)</td>
</tr>
<tr>
<td>India</td>
<td>42</td>
<td>50%</td>
<td>159.4 (23)</td>
<td>164.7 (19)</td>
</tr>
<tr>
<td>New Zealand</td>
<td>64</td>
<td>44%</td>
<td>153.6 (32)</td>
<td>157.8 (32)</td>
</tr>
<tr>
<td>Pakistan</td>
<td>67</td>
<td>55%</td>
<td>152.3 (36)</td>
<td>144.9 (31)</td>
</tr>
<tr>
<td>South Africa</td>
<td>58</td>
<td>62%</td>
<td>147.6 (29)</td>
<td>148.9 (29)</td>
</tr>
<tr>
<td>Sri Lanka</td>
<td>52</td>
<td>48%</td>
<td>159.7 (26)</td>
<td>139.3 (26)</td>
</tr>
<tr>
<td>West Indies</td>
<td>49</td>
<td>41%</td>
<td>159.7 (28)</td>
<td>147.5 (21)</td>
</tr>
<tr>
<td>Zimbabwe</td>
<td>25</td>
<td>20%</td>
<td>129.9 (13)</td>
<td>171.8 (12)</td>
</tr>
</tbody>
</table>

Table 2.1: Summary statistics for the T20 dataset corresponding to matches from February 17, 2005 through November 13, 2013. The variables $\bar{R}^{(S)}$ and $\bar{R}^{(A)}$ denote the average number of first innings runs scored and runs allowed, respectively, with the number of matches in parentheses.

We now study various features related to batting. We temporarily ignore extras (sundries) that arise via wide-balls and no-balls, and note that there are only 8 broadly defined outcomes that can occur when a batsman faces a bowled ball. These batting outcomes are listed below:

\[
\begin{align*}
\text{outcome } j = 0 & \equiv 0 \text{ runs scored} \\
\text{outcome } j = 1 & \equiv 1 \text{ runs scored} \\
\text{outcome } j = 2 & \equiv 2 \text{ runs scored} \\
\text{outcome } j = 3 & \equiv 3 \text{ runs scored} \\
\text{outcome } j = 4 & \equiv 4 \text{ runs scored} \\
\text{outcome } j = 5 & \equiv 5 \text{ runs scored} \\
\text{outcome } j = 6 & \equiv 6 \text{ runs scored} \\
\text{outcome } j = 7 & \equiv \text{ dismissal}
\end{align*}
\] (2.1)
In the list (2.1) of possible batting outcomes, we include byes, leg byes and no balls where the resultant number of runs determines one of the outcomes $j = 0, \ldots, 7$. We note that the outcome $j = 5$ is rare but is retained to facilitate straightforward notation.

We first calculate the proportions $\hat{p}_0, \ldots, \hat{p}_7$ corresponding to the first innings batting outcomes. Table 2.2 provides a comparison of these proportions based on the T20 dataset compared with the proportions for fourth innings batting in test cricket as reported by Perera, Gill & Swartz (2013). We observe that T20 batting is much more aggressive than batting in test cricket. For example, 6’s occur with a much greater frequency (by a factor of 14) in T20 cricket than in test cricket. Consequently, the modelling of runs is dependent on the particular form of cricket under consideration.

<table>
<thead>
<tr>
<th></th>
<th>$\hat{p}_0$</th>
<th>$\hat{p}_1$</th>
<th>$\hat{p}_2$</th>
<th>$\hat{p}_3$</th>
<th>$\hat{p}_4$</th>
<th>$\hat{p}_5$</th>
<th>$\hat{p}_6$</th>
<th>$\hat{p}_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T20 (1st innings)</td>
<td>0.301</td>
<td>0.411</td>
<td>0.078</td>
<td>0.006</td>
<td>0.103</td>
<td>0.002</td>
<td>0.042</td>
<td>0.057</td>
</tr>
<tr>
<td>Test (4th innings)</td>
<td>0.743</td>
<td>0.128</td>
<td>0.035</td>
<td>0.010</td>
<td>0.065</td>
<td>0.000</td>
<td>0.003</td>
<td>0.016</td>
</tr>
</tbody>
</table>

Table 2.2: Sample proportions corresponding to batting outcomes in two forms of cricket.

In one-day cricket, it is well-known that opening batsmen begin matches cautiously, attempting to avoid wickets and hoping to develop a batting rhythm. As the match proceeds, batting tends to become more aggressive. In T20 cricket, there are only 20 overs, and it is conceivable that batsmen behave uniformly throughout a match. In other words, T20 batsmen exhibit aggressiveness at the beginning of a match, and display the same level of aggressiveness as the match proceeds. The intuition is that it is beneficial to always be aggressive since the 10 allocated wickets are likely to suffice for 20 overs. For example, in our dataset, teams were made “all out” during the first innings in only 11% of the matches. If it were true that batsmen displayed constant aggressiveness in T20, this would facilitate modelling since batting characteristics would not change with respect to wickets lost nor overs consumed.

In Figure 2.1, we provide plots of the proportions of batting outcomes in the first innings of the T20 dataset stratified by over. We observe that the above intuition about constant aggressiveness in T20 batting is clearly false. For example, we observe that batsmen have very few 4’s during the first two overs which is a period of adjustment to the bowler, to the ball, to the pitch, to the weather, etc. This initial period is followed by a spike in 4’s which corresponds to the powerplay (i.e. the first 6 overs of a match where fielding restrictions are in place). Once the powerplay terminates, the proportion of 4’s plummets in the 7th over, and then there is a gradual (nearly monotonic) increase in 4’s until the completion of the innings. These observations are important for our subsequent modelling assumptions.

Note that it is also possible to produce a plot of the proportion of batting outcomes in the first innings stratified by wickets lost. Such a plot suggests that batting characteristics are also dependent on wickets lost. Furthermore, with respect to batting characteristics,
there is clearly an interaction between the number of overs consumed and the number of wickets lost. For example, a batsman is more aggressive in the 19th over with two wickets lost than in the 19th over with 9 wickets lost. The existence of the interaction is one of the guiding principles in the development of the Duckworth-Lewis resource table by Duckworth & Lewis (2004) used to reset targets in rain-interrupted matches.

![Figure 2.1: Proportion of batting outcomes stratified by over. The vertical line denotes the termination of the powerplay.](image)

According to the enumeration of the batting outcomes in (2.1), the preceding discussion suggests a statistical model for the number of runs scored by the \( i \)th batsman:

\[
(X_{iow0}, \ldots, X_{iow7}) \sim \text{multinomial}(m_{iow}; p_{iow0}, \ldots, p_{iow7})
\] (2.2)

where \( X_{iowj} \) is the number of occurrences of outcome \( j \) by the \( i \)th batsman during the \( o \)th over when \( w \) wickets have been taken. In (2.2), \( m_{iow} \) is the number of balls that batsman \( i \) has faced in the dataset corresponding to the \( o \)th over when \( w \) wickets have been taken. The multinomial distributions (2) define the likelihood which is used to estimate the characteristics \( p_{iowj} \). In Section 3, we address the difficulty of parameter estimation in a highly parametrized setting with sparse data; i.e. \( m_{iow} \approx 0 \) for many of the situations \((i, o, w)\). In Section 4, we address the problem where batsmen face bowlers of varying quality.

Assume temporarily that we are able to obtain (i.e. estimate) the multinomial parameters in (2.2). We would then be able to generate (from a multinomial distribution with
m = 1) the outcome of a single ball. A straightforward algorithm for simulating first innings runs against an average bowler would proceed as follows: The batting team begins with a fixed batting lineup where \(k = 1, w = 0\) and the 11 batsmen are described by their batting characteristics \(p_{iowj}\). **Start:** Suppose that ball \(k \leq 120\) of the match is being delivered which determines the corresponding over \(o\). Provided that \(w < 10\) wickets have been taken, and provided that \(o \leq 20\), a variate \(u_1 \sim \text{Uniform}(0, 1)\) is generated. Otherwise, the innings are complete. If \(u_1 \leq q_1\), then an extra has occurred, a single run is added to the run counter and we return to **Start.** Based on our extensive T20 dataset, we have determined that extras occur with probability \(q_1 = 0.033\). If \(u_1 > q_1\), then a second variate \(u_2 \sim \text{Uniform}(0, 1)\) is generated. Then, according to the specific probabilities \(p_{iow0}, \ldots, p_{iow7}\) with the \(i\)th batsman batting, one of the outcomes in (1) is determined. The run counter, the ball counter \(k\) and the wicket counter \(w\) are updated accordingly. If the batsman scores 1 run, 3 runs or 5 runs (essentially impossible), then his batting partner faces the next ball. If the batsman is dismissed, the batsman is replaced by the next batsman in the batting lineup. We then return to **Start.**

Therefore, given the multinomial parameters \(p_{iowj}\) in (2.2), it is a very simple coding exercise to develop a first innings simulator for T20 cricket. In Section 4.3, we introduce modifications for second innings simulation. With a match simulator, we are then able to investigate various situations of interest with respect to T20 cricket. In the following section, we discuss the fundamental problem of parameter estimation.

### 2.3 Parameter Estimation

Under model (2.2), we are concerned with the estimation of the multinomial parameters \(p_{iowj}\) subject to the constraints \(\sum_{j=0}^{7} p_{iowj} = 1\), for all \(i, o, w\). Whereas maximum likelihood estimation of the \(p_{iowj}\) is “easy”, it does not lead to reliable estimation due to the sparsity of the data in many of the situations \((iowj)\). For example, a bowler would never bat in the early overs of an innings. With so many parameters, we make some simplifying assumptions based on our observations in Section 2. Specifically we let

\[
p_{iowj} = \frac{\tau_{owj} p_{i70j}}{\sum_j \tau_{owj} p_{i70j}}. \tag{2.3}
\]

In (2.3), the parameter \(p_{i70j}\) represents the baseline characteristic for batsman \(i\) with respect to batting outcome \(j\). The characteristic \(p_{i70j}\) is the probability of outcome \(j\) associated with the \(i\)th batsman at the juncture of the match immediately following the powerplay (i.e. the 7th over) when no wickets have been taken. The multiplicative parameter \(\tau_{owj}\) scales the baseline performance characteristic \(p_{i70j}\) to the stage of the match corresponding to the \(o\)th over with \(w\) wickets taken. The denominator in (2.3) ensures that the relevant probabilities sum to unity.
There is an implicit assumption in (2.3) that the stage of the first innings (overs completed and wickets lost) affects all batsmen in the same fashion. We argue that the stage of the match determines the aggressiveness of batting in general. Although batsmen are unique, their batting characteristics change by the same multiplicative factor which is essentially an indicator of aggressiveness. For example, when aggressiveness increases relative to the baseline state, one would expect \( \tau_{ow4} > 1 \) and \( \tau_{ow6} > 1 \) since bolder batting leads to more 4’s and 6’s. In Section 2.5, we demonstrate that the batting characteristics modelled according to (2.3) lead to a realistic match simulator.

Given (2.3), we need to estimate the multiplicative parameters \( \tau_{owj} \) and the individual baseline probabilities \( p_{i70j} \). With \( N = 384 \) batsmen in the dataset, there are \((19)(9)(7) + (384)(7) = 3885\) unknown parameters. And with 30458 batting outcomes, the ratio of data to parameters is roughly 8:1. This is clearly a problem where parameter estimation is challenging.

We consider a Bayesian approach to estimation where model (2.2) describes the sampling distribution of the data. Since the \( p_{i70j} \) are probabilities defined on simplices, we make the prior assumption \( (p_{i700}, \ldots, p_{i707}) \sim \text{Dirichlet}(a_0, \ldots, a_7) \). Letting \( p \) denote the vector of all baseline parameters and letting \( [x \mid y] \) denote generic notation for the conditional density of \( x \) given \( y \), we obtain the posterior density

\[
[p \mid X] \propto [X \mid p] [p] \propto \left( \prod_{i,o,w} \left( \frac{\tau_{ow0} p_{i700}}{\sum_j \tau_{owj} p_{i70j}} \right) X_{iow0} \right) \cdots \left( \frac{\tau_{ow7} p_{i707}}{\sum_j \tau_{owj} p_{i70j}} \right) X_{iow7} \left( \prod_{i} p_{i700}^{a_0-1} \cdots p_{i707}^{a_7-1} \right)^{24}
\]

where independence is assumed across the batsmen, overs and wickets.

Ideally, we would like to estimate the parameters in (2.4) simultaneously. A common approach in Bayesian problems with high dimensionality involves the construction of a Markov chain whose equilibrium distribution is the posterior distribution. One would then obtain parameter estimates by averaging simulated parameters from the Markov chain. However, we were unable to do so because of our inability to obtain a convergent Markov chain using various Metropolis samplers. Instead, we opted for a hybrid method of estimation. We first estimated the multiplicative parameters \( \tau_{owj} \), and then given the \( \tau_{owj} \), we estimated the baseline probabilities \( p_{i70j} \) for individual batsmen. The spirit of the two-step estimation procedure is reminiscent of profile likelihood methodology (Davison 2003).

In Appendix C, we discuss the estimation of the multiplicative factors \( \tau_{owj} \). In the remainder of this section, we describe the hierarchical model and the methodology used to estimate the baseline probabilities \( p_{i70j} \) given the \( \tau_{owj} \).

In a Bayesian setting, it is standard practice to use posterior means as estimators. Given the complexity of the posterior density (2.4), we propose a sampling based approach based on Markov chain Monte Carlo (MCMC) methods to estimate the parameters \( p_{i70j} \). With
the \( \tau \)'s specified, we note that the posterior (2.4) is amenable in the sense that it factors according to each batsman \( i \). Specifically, we first consider a Gibbs sampling algorithm (Gilks, Richardson & Spiegelhalter 1996) where the full conditional densities take the form

\[
[p_{\tau 700}, \ldots, p_{\tau 707} \mid \cdot] \propto \prod_{o, w} (\sum_j X_{iowj}/\tau_{owj})^{a_{owj} + a_{owj} - 1}.
\] (2.5)

Unfortunately, the full conditional densities (2.5) are nonstandard in the sense that there does not appear to be a simple way to generate variates directly from the corresponding distributions. We therefore consider a Metropolis within Gibbs step where the proposal distributions are Dirichlet with parameters corresponding to the exponents in the numerator of (2.5).

The described procedure is fully Bayesian and only requires the subjective setting of the hyperparameters \( a_0, \ldots, a_7 \). Note that the default setting \( a_0 = \cdots = a_7 = 0 \) is an obvious choice although it does not take prior knowledge into account. As the \( a_i \)'s get larger, there is greater shrinkage of the individual estimates towards a common characteristic. In this application, we take an empirical Bayes approach where we use the data to specify the hyperparameters.

We begin by setting \( a_j = c \sum_{i, o, w} (X_{iowj}/\tau_{owj}) / (\sum_{i, o, w, k} X_{iowk}/\tau_{owk}) \) for some \( c > 0 \) and \( j = 0, \ldots, 7 \). As \( c \to \infty \), the posterior density (2.4) approaches a product Dirichlet density dominated by the \( a_j \) terms and where the posterior mean \( \hat{p}_{\tau 70j} \) is such that

\[
\hat{p}_{\tau 70j} \to \frac{\sum_{i, o, w} (X_{iowj}/\tau_{owj})}{\sum_{i, o, w, k} (X_{iowk}/\tau_{owk})}
\] (2.6)
as \( c \to \infty \). The estimate (2.6) may be thought of as the common mean (over all batsmen) in over 7 with zero wickets after transforming all situations to the case \( o = 7 \) and \( w = 0 \).

Therefore, our problem is the selection of \( c > 0 \) such that it is not too large (i.e. all batsmen have identical characteristics) but is also not too small (since small sample sizes may give rise to unrealistic characteristics). We note that the standard deviations of the prior parameters \( p_{\tau 70j} \) are proportional to \( 1/\sqrt{c+1} \). Therefore, the tuning parameter \( c > 0 \) may be thought of as the prior sample size. We have tinkered with various choices and have found that \( c = 60.0 \) provides realistic characteristics. Also, we have observed that the simulation results do not differ greatly for \( c \) in the interval \((50, 80)\).

Again we emphasize that there is a need to choose \( c \) substantially greater than zero since there are batsmen with limited batting histories. We do not want the sparsity of their batting attempts to result in unrealistic batting characteristics. Choosing \( c \) substantially larger than zero shrinks their observed proportions closer to the means for all batsmen. In Figure 2.2 we provide a histogram of the number of batting attempts by the batsmen in our dataset. We observe that more than 200 batsmen have faced fewer than 40 balls. Only a handful of batsmen have faced more than 500 balls.
We note that the player characteristics $p_{iowj}$ estimated in this section may be viewed as average characteristics taken over a player’s career. In some applications, it might be more meaningful to carry out simulations based on current form. The natural way to do this is to weight the data with more weight given to recent observations (i.e. matches). Operationally, this was achieved by replacing $X_{iowj}$ in (2.4) with $\sum_g r^g X_{iowjg}$ where $r$ is a decay ratio and the subscript $g$ is the match index that denotes the number of games before the last game was played. We determined $r = 0.88$ via maximization of the posterior density.

2.4 EXTENDING THE SIMULATOR

Although the proposed simulator is detailed and captures many of the essential features of T20 cricket, it may be extended in various ways to enhance realism.

2.4.1 The Impact of the Bowler

In model (2.1), the data $X_{iowj}$ correspond to batting outcomes. Noting the symmetry between batting and bowling, one can also specify a model from the point of view of bowlers. Specifically,

$$(Y_{iow0}, \ldots, Y_{iow7}) \sim \text{multinomial}(m_{iow}; q_{iow0}, \ldots, q_{iow7})$$
where \( Y_{iowj} \) is the number of occurrences of outcome \( j \) as defined in (2.1) experienced by the \( i \)th bowler during the \( o \)th over when \( w \) wickets have been taken. The parameters \( q_{iowj} \) therefore describe bowling characteristics with respect to an average batsman. The parameters \( q_{iowj} \) can be estimated using the same empirical Bayes approach described in Section 2.3.

In a given match simulation, rather than having batsmen face average bowlers, it is more realistic to have batsmen face specified bowlers. It is suggested that a modification can be made in the case of batsman \( i_1 \) facing bowler \( i_2 \) by using the characteristic

\[
p_{i1owj} + q_{i2owj} - \bar{p}_{owj} \tag{2.7}
\]

for outcome \( j \) where \( \bar{p}_{owj} \) is the average batting characteristic of outcome \( j \) taken over all batsmen. Note that \( \bar{p}_{owj} \) is also the average bowling characteristic of outcome \( j \) taken over all bowlers.

The batting characteristics given by (2.7) are sensible in the sense that if batsmen \( i_1 \) is average, then (2.7) reduces to \( q_{i2owj} \), and if bowler \( i_2 \) is average, then (2.7) reduces to \( p_{i1owj} \). The quantity (2.7) provides a synthesis of the individual characteristics of the batsman and the bowler.

### 2.4.2 The Home Team Advantage

Although the underlying causes of the home team advantage are difficult to pinpoint precisely, the effect of the home team advantage is real and the magnitude of the advantage varies according to the sport (Swartz & Arce 2014). One of the interesting findings is that there is not a strong case for differential home team advantages amongst teams that compete in the same league (hockey and basketball).

We note that de Silva, Pond & Swartz (2001) considered the effect of the home team advantage in one-day international cricket. They defined the home team advantage as the number of runs one would expect a home team to defeat the road team when both teams are of equal strength. The advantage was estimated to be worth approximately 16 runs (their Model D).

In our T20 simulator, we take a simple approach for implementing the effect of the home team advantage. From the data set, the average number of first innings runs scored by the home team was 158.4 and the average number of first innings runs scored by the away team was 149.4. Therefore, in the simulator, we scale the number of runs scored by the home team by the factor \( 2(158.4)/(158.4+149.4) = 1.03 \), and we scale the number of runs scored by the away team by the factor \( 2(149.4)/(158.4+149.4) = 0.97 \). This is done at the individual batsmen level which can lead to a non-integer number of runs scored. At the end, we round the total number of runs scored. In a match played at a neutral site, no adjustment is made for the home team advantage.
2.4.3 Second Innings Simulation

Up until now, we have focused on first innings simulation. We would like to extend the simulator to the second innings so that we can address questions such as “What is the probability that the team batting second wins the match given a specific target score obtained during the first innings?”

As we have seen, batting characteristics vary according to the aggressiveness of the batsmen, and aggressiveness is determined by the state of the match. And in the second innings, the target score forms a component of the state of the match. Naturally, batsmen need to be more aggressive when the target score is greater.

The idea for second innings simulation (borrowed from Swartz, Gill & Muthukumarana 2009) is that batting characteristics are tweaked according to changes in aggressiveness. For this, let \( r_1 \) denote the number of runs scored during the first innings (the target) and let \( r_2 \) be the number of runs scored in the second innings up to the current stage of the match (determined by the number of overs completed and the number of wickets taken). Then the team batting second requires \( r_1 - r_2 + 1 \) additional runs to win the match.

Next, we let \( d \) denote the number of resources remaining in the match from the given stage of the match. The value \( d \) is obtained from the Duckworth-Lewis table (Duckworth & Lewis 2004) modified for T20. Resources are a synthesis of the number of overs completed and the number of wickets taken. As the number of overs increases and the number of wickets increases, a team’s batting resources decrease. Therefore, in order for the team batting second to win the match, they need to bat with a “runs to resources ratio” of at least

\[
\frac{r_1 - r_2 + 1}{d}.
\] (2.8)

The quantity (2.8) is a measure of the required level of aggressiveness where larger values indicate increasing aggressiveness. In terms of aggressiveness, (2.8) is essentially a ratio of what is needed to what is typically available.

But according to the batting characteristics \( p_{iow} \) of the \( i \)th batsman, what is the actual level of batting aggressiveness? At the given stage \((o, w)\) of the match, the expected number of runs scored on the next ball bowled is

\[
E^{(1)}_{iow} = p_{iow1} + 2p_{iow2} + 3p_{iow3} + 4p_{iow4} + 6p_{iow6}.
\]

Similarly, the expected number of resources consumed on the next ball bowled is

\[
E^{(2)}_{iow} = xp_{iow0} + xp_{iow1} + xp_{iow2} + xp_{iow3} + xp_{iow4} + xp_{iow6} + (x + y)p_{iow7}
\]

where \( x \) are the resources lost due to the current ball and \( y \) are the resources lost due to a wicket. The values \( x \) and \( y \) are obtained from the Duckworth-Lewis table.
Putting this all together, the logic is that if the current level of batting aggressiveness $E^{(1)}_{iow}/E^{(2)}_{iow}$ is not sufficiently large to win the match, then batting aggressiveness should be increased. Using the above notation, batting aggressiveness should be increased if

$$\frac{E^{(1)}_{iow}}{E^{(2)}_{iow}} < \frac{r_1 - r_2 + 1}{d}. \quad (2.9)$$

Since batting aggressiveness increases as the overs increase for a fixed number of wickets lost, our approach is to find the minimum value $o^* > o$ such that

$$\frac{E^{(1)}_{iow}^*}{E^{(2)}_{iow}^*} \geq \frac{r_1 - r_2 + 1}{d}. \quad (2.10)$$

When we have solved for $o^*$ in (2.10), then even though batting is taking place at stage $(o, w)$ of the match, we will use the batsman’s characteristics $p_{iow^*j}$. The numerical determination of $o^*$ is straightforward. We begin with $o^* = o$, and then increment $o^*$ until (2.10) is satisfied. If $o^*$ cannot be determined, then the simulation proceeds with the batsman’s personal maximum aggressiveness as given by the characteristics $p_{iowj}$ where $o = 20$.

The modelling of the second innings was one of the more challenging aspects of the paper. Whereas everyone agrees that batting tendencies change as the match progresses, it is not clear exactly how batting characteristics are modified. As a player, you may know that your team needs more runs, but is it possible to increase your run output without sacrificing additional wickets? If a batsman is able to do this, then the obvious question is why don’t they enact this batting behaviour all of the time? A feature of our second innings modelling is that batsmen modify their batting characteristics in a manner that is within their capabilities. When they need to be more aggressive, they simply behave as though they were batting later in the innings, and for this we have good estimation procedures.

We also emphasize that the description of second innings batting is how we believe batsmen should modify their batting when trailing in a match. It is an optimal behaviour which we are suggesting. In Section 5, we observe that batsmen do not quite behave in this optimal fashion.

## 2.5 Adequacy of the Simulator

Our general approach in the development of the simulator has been to simulate matches and then critically examine the output. When the output is in conflict with empirical results or our intuition, we revisit the underlying theory and iterate towards a realistic simulator.

Our resulting simulator has been coded in the R programming language. The time required to simulate a single innings is virtually instantaneous. To simulate 1000 innings requires approximately 1.5 minutes of computation on an older laptop computer.
We first investigate individual player characteristics obtained through our hybrid estimation scheme based on the multinomial model (2.2) and the simplifying assumptions given by (2.3). In Table 2.3, we provide batting characteristics for two prominent batsmen, Shane Watson of Australia and AB de Villiers of South Africa. Watson is an all-rounder who is typically an opening batsman and has faced 673 balls in T20 cricket. AB de Villiers is a pure batsman who usually bats in the third or fourth position in the lineup and has faced 666 balls. In comparing batsmen, we see that de Villiers resembles an average batsman. In fact, his characteristics are a little on the cautious side, scoring slightly fewer sixes and getting out less often than an average batsman. On the other hand, Watson is a power hitter who scores sixes at a high rate but is also dismissed frequently. It is interesting to compare batting performances at different stages of a match. As expected, run rates increase in the 10th over with zero wickets compared to the period immediately after the powerplay (7th over with zero wickets). Run rates continue to increase at the stage of the 15th over with two wickets lost. This is sensible as there are many wickets remaining and batsmen can take chances by increasing their aggressiveness. Note that the probability of a wicket increases dramatically at this stage. We then observe a decrease in productivity in the 15th over with 8 wickets. This also corresponds to our intuition as batsmen need to protect their wickets as only two replacement batsmen remain.

<table>
<thead>
<tr>
<th>Player</th>
<th>Over</th>
<th>Wicket</th>
<th>0 runs</th>
<th>1 run</th>
<th>2/3 runs</th>
<th>4 runs</th>
<th>6 runs</th>
<th>Out</th>
<th>E(RR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>7</td>
<td>0</td>
<td>0.3421</td>
<td>0.3700</td>
<td>0.0956</td>
<td>0.1212</td>
<td>0.0341</td>
<td>0.0370</td>
<td>7.5</td>
</tr>
<tr>
<td>Watson</td>
<td>7</td>
<td>0</td>
<td>0.2804</td>
<td>0.3794</td>
<td>0.0868</td>
<td>0.1046</td>
<td>0.0935</td>
<td>0.0553</td>
<td>9.2</td>
</tr>
<tr>
<td>de Villiers</td>
<td>7</td>
<td>0</td>
<td>0.3087</td>
<td>0.4035</td>
<td>0.1153</td>
<td>0.1116</td>
<td>0.0254</td>
<td>0.0356</td>
<td>7.4</td>
</tr>
<tr>
<td>Average</td>
<td>10</td>
<td>0</td>
<td>0.2905</td>
<td>0.4000</td>
<td>0.0998</td>
<td>0.1074</td>
<td>0.0432</td>
<td>0.0591</td>
<td>7.7</td>
</tr>
<tr>
<td>Watson</td>
<td>10</td>
<td>0</td>
<td>0.2294</td>
<td>0.3950</td>
<td>0.0873</td>
<td>0.0893</td>
<td>0.1139</td>
<td>0.0851</td>
<td>9.7</td>
</tr>
<tr>
<td>de Villiers</td>
<td>10</td>
<td>0</td>
<td>0.2604</td>
<td>0.4333</td>
<td>0.1197</td>
<td>0.0982</td>
<td>0.0319</td>
<td>0.0565</td>
<td>7.4</td>
</tr>
<tr>
<td>Average</td>
<td>15</td>
<td>2</td>
<td>0.2056</td>
<td>0.4556</td>
<td>0.0944</td>
<td>0.0838</td>
<td>0.0533</td>
<td>0.1072</td>
<td>7.8</td>
</tr>
<tr>
<td>Watson</td>
<td>15</td>
<td>2</td>
<td>0.1532</td>
<td>0.4246</td>
<td>0.0779</td>
<td>0.0657</td>
<td>0.1329</td>
<td>0.1456</td>
<td>9.8</td>
</tr>
<tr>
<td>de Villiers</td>
<td>15</td>
<td>2</td>
<td>0.1826</td>
<td>0.4889</td>
<td>0.1120</td>
<td>0.0759</td>
<td>0.0390</td>
<td>0.1014</td>
<td>7.5</td>
</tr>
<tr>
<td>Average</td>
<td>15</td>
<td>8</td>
<td>0.2200</td>
<td>0.4921</td>
<td>0.0846</td>
<td>0.0836</td>
<td>0.0455</td>
<td>0.0742</td>
<td>7.6</td>
</tr>
<tr>
<td>Watson</td>
<td>15</td>
<td>8</td>
<td>0.1687</td>
<td>0.4718</td>
<td>0.0718</td>
<td>0.0675</td>
<td>0.1165</td>
<td>0.1037</td>
<td>9.5</td>
</tr>
<tr>
<td>de Villiers</td>
<td>15</td>
<td>8</td>
<td>0.1948</td>
<td>0.5264</td>
<td>0.1002</td>
<td>0.0755</td>
<td>0.0332</td>
<td>0.0700</td>
<td>7.4</td>
</tr>
</tbody>
</table>

Table 2.3: Batting probabilities (characteristics) for an average batsman, Shane Watson and AB de Villiers at different stages of a match. The quantity E(RR) is the expected run rate for the over where 2/3’s are treated as 2’s. Note that 3’s occur very rarely (< 1% of the time).

We also investigate the adequacy of the simulator by looking at the larger picture in terms of team performance. For each of the 10 ICC teams, we recorded their first innings runs for all T20 matches from the past two years. We then determined a team batting lineup that was representative during that period, and simulated 1000 matches against
average opposition. The actual runs and the corresponding simulated quantiles are given in Figure 2.3 for Australia and Zimbabwe. According to Table 2.1, Australia and Zimbabwe are the highest and the lowest scoring teams respectively. The Q-Q plots suggest that the simulator produces first inning runs that are in line with the actual number of runs scored. Similar plots were obtained for the other ICC teams.

![Figure 2.3: Q-Q plots for Australia and Zimbabwe for first innings runs where the fits appear reasonable.](image)

To investigate wicket estimation, Figure 2.4 provides a plot of the average number of wickets lost versus the number of overs completed for first innings batting. Figure 2.4 contains two lines; one based on average wickets lost from actual matches and the other based on averages wickets lost from simulated matches involving randomly chosen batsmen. We observe that the wicket rate increases as the match progresses. There appears to be reasonable agreement between the two lines. This is important because the occurrence of wickets greatly affects run scoring.

To investigate the second innings batting formulation, we considered the lineups used in the 2014 World Cup final between Sri Lanka and India held on April 6. Our simulations give strikingly different probabilities of winning depending on which team bats first. We obtained Prob(SL wins | SL bats first) = 0.46 and Prob(SL wins | India bats first) = 0.61. In contrast, various studies including de Silva & Swartz (1997) and Saikia & Bhattacharjee (2010) have suggested that batting second confers at most a minor advantage.

How do we reconcile these observations? It seems to us intuitive that batting second should provide a competitive advantage as the team batting second has knowledge of the target and can adjust their batting strategy accordingly. This appears to be the case in Major League Baseball where home teams (which bat in the “bottom” innings) win roughly
54% of their games (Stefani 2008). In second innings simulation, we emphasize that our modified batting characteristics are not unattainable batting characteristics. In fact, they are the characteristics that batsmen display at various stages of a match. It is within their capabilities to modify their characteristics in the manner which we have prescribed. What we posit is that batsmen do not behave in this “optimal” manner. Instead, we believe that batsmen delay increasing their aggressiveness when their team begins falling behind in the second innings. To investigate this, we modify the condition (2.9) which stipulates an increase in aggressiveness. We adjust the condition for increased aggressiveness by multiplying the right hand side of (2.9) by the factor 0.8. This states that the team batting second must fall behind an additional 20% before they begin altering their style. In a match with 150 runs, this is essentially saying that a team increases its aggressiveness when it perceives that it is on track to lose by 30 runs. When we introduce the factor 0.8, we obtain $\text{Prob}(\text{SL wins} \mid \text{SL bats first}) = 0.51$ and $\text{Prob}(\text{SL wins} \mid \text{India bats first}) = 0.55$, and now, the benefit of batting second is much reduced.

The preceding discussion has implications for batting strategy in the second innings. We believe that teams would be better served by increasing their aggressiveness incrementally when they begin falling behind rather than react in a panicky fashion at some later stage when it becomes obvious that they are on the verge of losing.
2.5.1 An Example Concerning the Practical Use of the Simulator

The 2014 World Cup that took place in Bangladesh from March 16 through April 6 provided an interesting application for our methodology. We considered matches beyond the qualification stage that involved the teams from our dataset. We excluded matches involving Bangladesh since the data collected on Bangladesh (see Table 2.1) was not as comprehensive. Bangladesh had several “new” players for whom we had minimal data and we did not want to introduce a home team effect for Bangladesh. We note that the Netherlands were the “surprise” team of the tournament as they advanced to the qualification stage at the expense of Zimbabwe. We also did not consider matches involving the Netherlands since we had no data on their past performances.

For a match between Team A and Team B, we simulated 10,000 first innings for each team and calculated the proportion of time that Team A had more runs than Team B. We used this as a proxy for the probability that Team A defeats Team B. Note that sportsbook odds do not take into account which team bats first since this is determined by the coin flip at the beginning of a match. The batting and bowling lineups that we selected in the simulations were the lineups used in the actual matches.

In Table 2.4, we present the win probabilities from the simulations and the win probabilities implied by sportsbook odds. We see fairly strong agreement between the two sets of probabilities. This is a further endorsement of the realism of the simulator since sportsbooks are thought to be “efficient markets” in the sense that sportsbook odds capture all of the available information. One of our observations from the exercise is that the inclusion/exclusion of key players in the lineup can have a meaningful impact on the probabilities. We also note that relative to the sportsbook, our winning probabilities for Pakistan were considerably higher. We believe that this was partly due to the inclusion of Zulfiqar Babar and Biliawal Bhatti into the lineups as relatively new bowlers. Whereas the sportbook discounted their abilities, our model provided them with performance characteristics that were in line with average performance. Pakistan also did badly in some of their T20 matches leading up to the World Cup, matches for which we did not collect data. We also note that sportsbook odds are dynamic and sometimes the odds can change by several percent in the hours leading up to a match.

To investigate the simulator further, and possibly assess whether its output is superior to the sportsbook odds, we wagered a hypothetical $100 on each of the 15 matches from Table 2.4. The team that we wagered on was the team whose simulated probabilities exceeded the implied sportsbook probabilities. The $100 was wagered at the odds corresponding to the sportsbook. The net result of this exercise was a hypothetical profit of $399 where 9 of the 15 winning teams were chosen correctly. Of course, this is too small a sample of matches to guarantee long run profitability.
<table>
<thead>
<tr>
<th>Date</th>
<th>Team A</th>
<th>Team B</th>
<th>Winner</th>
<th>Prob(Team A wins)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Simulator</td>
<td>Sportsbook</td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 21</td>
<td>India</td>
<td>Pakistan</td>
<td>India</td>
<td>0.54</td>
</tr>
<tr>
<td>March 22</td>
<td>Sri Lanka</td>
<td>South Africa</td>
<td>Sri Lanka</td>
<td>0.44</td>
</tr>
<tr>
<td>March 22</td>
<td>England</td>
<td>New Zealand</td>
<td>New Zealand</td>
<td>0.47</td>
</tr>
<tr>
<td>March 23</td>
<td>Pakistan</td>
<td>Australia</td>
<td>Pakistan</td>
<td>0.51</td>
</tr>
<tr>
<td>March 23</td>
<td>West Indies</td>
<td>India</td>
<td>India</td>
<td>0.39</td>
</tr>
<tr>
<td>March 24</td>
<td>New Zealand</td>
<td>South Africa</td>
<td>South Africa</td>
<td>0.41</td>
</tr>
<tr>
<td>March 27</td>
<td>England</td>
<td>Sri Lanka</td>
<td>England</td>
<td>0.36</td>
</tr>
<tr>
<td>March 28</td>
<td>West Indies</td>
<td>Australia</td>
<td>West Indies</td>
<td>0.40</td>
</tr>
<tr>
<td>March 29</td>
<td>England</td>
<td>South Africa</td>
<td>South Africa</td>
<td>0.42</td>
</tr>
<tr>
<td>March 30</td>
<td>India</td>
<td>Australia</td>
<td>India</td>
<td>0.52</td>
</tr>
<tr>
<td>March 31</td>
<td>Sri Lanka</td>
<td>New Zealand</td>
<td>Sri Lanka</td>
<td>0.69</td>
</tr>
<tr>
<td>April 1</td>
<td>West Indies</td>
<td>Pakistan</td>
<td>West Indies</td>
<td>0.33</td>
</tr>
<tr>
<td>April 3 (Semifinal)</td>
<td>Sri Lanka</td>
<td>West Indies</td>
<td>Sri Lanka</td>
<td>0.64</td>
</tr>
<tr>
<td>April 4 (Semifinal)</td>
<td>India</td>
<td>South Africa</td>
<td>India</td>
<td>0.55</td>
</tr>
<tr>
<td>April 6 (Final)</td>
<td>Sri Lanka</td>
<td>India</td>
<td>Sri Lanka</td>
<td>0.53</td>
</tr>
</tbody>
</table>

Table 2.4: Win probabilities for specified 2014 World Cup matches beyond the qualification stage.

### 2.6 DISCUSSION

In the development of our simulator, batting outcome probabilities are dependent on the batsman, the bowler, the number of overs consumed, the number of wickets lost, the home team advantage and the target score (in the case of the second innings). The proposed model is complex and captures the essential features of T20 cricket, there is no doubt that there are other variables that may influence batting performance. For example, the fielding quality of the opposing team affects run scoring. Also, if various players are in particularly good or poor form, one may consider tinkering with their characteristics. As discussed at the end of Section 2.3, one way to accomplish this may involve a weighted estimation scheme where more weight is given to recent performances. The implementation of these sorts of ideas is something that may be considered in future research.

One of the interesting by-products of our work is that we have posited that teams are not batting optimally in the second innings. We suggest that teams are not incrementally increasing their aggressiveness when they begin falling behind. Instead, we believe that they wait until the situation becomes dire, and only then, increase their aggressiveness. Although it may be difficult to train batsmen to increase their aggressiveness incrementally in the prescribed fashion, we see an opportunity to move somewhat in this direction. This change of strategy could provide a significant benefit to teams.

We believe that the modelling of batting behaviour and the subsequent development of the simulator are important steps in gaining a deeper understanding of strategic aspects related to T20 cricket. For example, with a realistic simulator, it may be possible to determine
player worth and to investigate optimal team selection and optimal batting orders. These are topics which we plan to pursue in future work. We also understand that in-game cricket forecasting is a difficult problem which has applications to wagering. The methodology of Section 2.4.3 may be useful in this regard. We therefore see this paper as seminal work in the advancement of T20 cricket analytics.
Chapter 3

Twenty20 Moneyball

3.1 INTRODUCTION

Player evaluation is the Holy Grail of analytics in professional team sports. Teams are constantly attempting to improve their lineups through player selection, trades and drafts taking into account relevant financial constraints. A salary cap is one financial constraint that is present in many professional sports leagues. If a team spends excessively on one player, then there is less money remaining for his teammates.

In sports of a “continuous” nature (e.g. basketball, hockey, soccer), player evaluation is a challenging problem due to player interactions and the subtleties of “off-the-ball” movements. Nevertheless, a wealth of simple statistics are available for comparing players in these sports. For example, points scored, rebounds, assists and steals are common statistics that provide insight on aspects of player performance in basketball. More complex statistics are also available, and we refer the reader to Oliver (2004) for basketball, Gramacy, Taddy & Jensen (2013) for hockey and McHale, Scarf & Folker (2012) for soccer.

In sports of a “discrete” nature (e.g. baseball) where there are short bursts of activity and players have well-defined and measurable tasks that do not depend greatly on interactions with other players, there is more hope for accurate and comprehensive player evaluation. There has been much written about baseball analytics where Bill James is recognized as a pioneer in the subject area of sabermetrics. A biography of James and his ideas is given by Gray (2006). James was given due credit in the book Moneyball (Lewis 2003) which was later developed into the popular Hollywood movie starring Brad Pitt. Moneyball chronicled the 2002 season of the Oakland Athletics, a small-market Major League Baseball team who through advanced analytics recognized and acquired undervalued baseball players. Moneyball may be the inspiration of many of the advances and the interest in sports analytics today. In particular, the discipline of sabermetrics continues to flourish. For example, Albert & Marchi (2013) provide baseball enthusiasts with the skills to explore baseball data using computational tools.
Cricket is another sport which may be characterized as a discrete game and it shares many similarities with baseball. Both sports have innings where runs are scored, and whereas baseball has batters and pitchers, cricket has batsmen and bowlers. Although analytics papers have been written on cricket, the literature is far less extensive than what exists in baseball. A somewhat dated overview of statistical research in cricket is given by Clarke (1998).

There are various formats of cricket where the governing authority for the sport is the International Cricket Council (ICC). This paper is concerned with player evaluation in the version of cricket known as Twenty20 cricket (or T20 cricket). Twenty20 is a recent form of limited overs cricket which has gained popularity worldwide. Twenty20 cricket was showcased in 2003 and involved matches between English and Welsh domestic sides. The rationale behind Twenty20 was to provide an exciting version of cricket where matches conclude in roughly three hours duration. There are now various professional Twenty20 competitions where the Indian Premier League (IPL) is regarded as the most prestigious. The IPL has been bolstered by the support of Bollywood stars, extensive television contracts, attempts at competitive balance, short but intense seasons, lucrative sponsorships, etc.

In Twenty20 cricket, there are two common statistics that are used for the evaluation of batting performance. However, before defining the statistics it is important to remind the reader that there are two ways in which batting ceases during the first innings. Batting is terminated when the batting team has lost 10 wickets. That is, there have been 10 dismissals (“outs” in baseball parlance). Batting is also terminated when a team has used up its 20 overs. This means that the batting team has faced 120 bowled balls (i.e. six balls per over) not including extras. With this background, the first popular batting statistic is the batting average which is the total number of runs scored by a batsman divided by the number of innings in which he was dismissed. A logical problem with this statistic can be seen from the pathological case where over the course of a career, a batsman has scored a total of 100 runs during 100 innings but has been dismissed only once. Such a batsman has an incredibly high batting average of 100.0 yet he would be viewed as a detriment to his team since he scores so few runs per innings. The second popular batting statistic is the batting strike rate which is calculated as the number of runs scored by a batsman per 100 balls bowled. A logical problem with this statistic can be seen from the pathological case where a batsman always bats according to the pattern of scoring six runs on the first ball and then is dismissed on the second ball. Such a batsman has an incredibly high batting strike rate of 300.0 yet he would be viewed as a detriment to his team since he uses up wickets so quickly. We remark that similar logical flaws exist for the two main bowling statistics referred to as the bowling average and the bowling economy rate.

Although various authors have attempted to introduce more sophisticated cricket statistics (e.g. Croucher 2000, Beaudoin & Swartz 2003 and van Staden 2009), it is fair to say
that these approaches have not gained traction. We also mention the Reliance ICC Player
Rankings (www.relianceiccrankings.com) which are a compilation of measurements based
on a moving average and whose interpretation is not straightforward. Despite the preva-
lence and the official nature of the rankings, the precise details of the calculations may be
proprietary as they do not appear to be available.

In this paper, we propose a method of player evaluation in Twenty20 cricket from the
point of view of relative value statistics. Relative value statistics have become prominent
in the sporting literature as they attempt to quantify what is really important in terms of
winning and losing matches. For example, in Major League Baseball (MLB), the VORP
(value over replacement player) statistic has been developed to measure the impact of player
performance. For a batter, VORP measures how much a player contributes offensively in
comparison to a replacement-level player (Woolner 2002). A replacement-level player is a
player who can be readily enlisted from the minor leagues. Baseball also has the related
WAR (wins above replacement) statistic which is gaining a foothold in advanced analytics
(http://bleacherreport.com/articles/1642919). In the National Hockey League (NHL), the
plus-minus statistic is prevalent. The statistic is calculated as the goals scored by a player’s
team minus the goals scored against the player’s team while the player is on the ice. More
sophisticated versions of the plus-minus statistic have been developed by Schuckers et al.
(2011) and Gramacy, Taddy & Jensen (2013).

In Twenty20 cricket, a team wins a match when the runs scored while batting exceed
the runs conceded while bowling. Therefore, it is run differential that is the key measure
of team performance. It follows that an individual player can be evaluated by considering
his team’s run differential based on his inclusion and exclusion in the lineup. Clearly, run
differential cannot be calculated from match results in a meaningful way since conditions
change from match to match. For example, in comparing two matches (one with a specified
player present and the other when he is absent), other players may also change as well
as the opposition. Our approach to player evaluation is based on simulation methodology
where matches are replicated. Through simulation, we can obtain long run properties (i.e.
expectations) involving run differential. By concentrating on what is really important (i.e.
expected run differential), we believe that our approach addresses the essential problem of
interest in player evaluation.

In Section 3.2, we provide an overview of the simulator developed by Davis, Perera
& Swartz (2015b) which is the backbone of our analysis and is used in the estimation of
expected run differential.

In Section 3.3, we analyze player performance where players are divided into the follow-
ing broad categories: pure batsmen, bowlers and all-rounders. Our analyses lead to ratings,
and the ratings have a clear interpretation. For example, if one player has an expected
run differential that is two runs greater than another player, we know exactly what this
means. We observe that some of our results are in conflict with the Reliance ICC ratings.
In cases like these, it provides opportunities for teams to implement positive changes that are in opposition to commonly held beliefs. This is the “moneyball” aspect of our paper. We extend our analyses further by looking at salary data in the IPL where we indicate the possibility of players being both overpaid or underpaid. We conclude with a short discussion in Section 3.4.

3.2 OVERVIEW OF SIMULATION METHODOLOGY

We now provide an overview of the simulator developed by Davis, Perera & Swartz (2015b) which we use for the estimation of expected run differential. There are 8 broadly defined outcomes that can occur when a batsman faces a bowled ball. These batting outcomes are listed below:

\[
\begin{align*}
\text{outcome } j &= 0 \equiv 0 \text{ runs scored} \\
\text{outcome } j &= 1 \equiv 1 \text{ runs scored} \\
\text{outcome } j &= 2 \equiv 2 \text{ runs scored} \\
\text{outcome } j &= 3 \equiv 3 \text{ runs scored} \\
\text{outcome } j &= 4 \equiv 4 \text{ runs scored} \\
\text{outcome } j &= 5 \equiv 5 \text{ runs scored} \\
\text{outcome } j &= 6 \equiv 6 \text{ runs scored} \\
\text{outcome } j &= 7 \equiv \text{ dismissal}
\end{align*}
\] (3.1)

In the list (3.1) of possible batting outcomes, we exclude *extras* such as *byes*, *leg byes*, *wide-balls* and *no balls*. We later account for extras in the simulation by generating them at the appropriate rates. Extras occur at the rate of 5.1% in Twenty20 cricket. We note that the outcome $j = 5$ is rare but is retained to facilitate straightforward notation.

According to the enumeration of the batting outcomes in (3.1), Davis, Perera & Swartz (2015b) suggested the statistical model:

\[
(X_{iow0}, \ldots, X_{iow7}) \sim \text{multinomial}(m_{iow}; p_{iow0}, \ldots, p_{iow7})
\] (3.2)

where $X_{iowj}$ is the number of occurrences of outcome $j$ by the $i$th batsman during the $o$th over when $w$ wickets have been taken. In (3.2), $m_{iow}$ is the number of balls that batsman $i$ has faced in the dataset corresponding to the $o$th over when $w$ wickets have been taken. The dataset is “special” in the sense that it consists of detailed ball-by-ball data. The data were obtained using a proprietary parser which was applied to the commentary logs of matches listed on the CricInfo website (www.espncricinfo.com). We considered Twenty20 matches involving full member nations of the ICC. Currently, the 10 full members of the ICC are Australia, Bangladesh, England, India, New Zealand, Pakistan, South Africa, Sri Lanka, West Indies and Zimbabwe. In total, we obtained data from 282 matches that spanned the period from January 2005 to August 2014. The first innings data consist of 35356 balls bowled where $j = 1$ in (1) is the most common occurrence with 14542 (41.1%) cases. The
outcomes $j = 5$ and $j = 3$ are the least frequent occurrences with 67 (0.2%) and 214 (0.6%) cases respectively.

The estimation of the multinomial parameters in (2) is a high-dimensional and complex problem. The complexity is partly due to the sparsity of the data; there are many match situations (i.e. combinations of overs and wickets) where batsmen do not have batting outcomes. For example, bowlers typically bat near the end of the batting order and do not face situations when zero wickets have been taken.

To facilitate the estimation of the multinomial parameters $p_{iowj}$, Davis, Perera & Swartz (2015) introduced the simplification

$$
p_{iowj} = \tau_{owj} p_{i70j} / \sum_j \tau_{owj} p_{i70j}.
$$

(3.3)

In (3.3), the parameter $p_{i70j}$ represents the baseline characteristic for batsman $i$ with respect to batting outcome $j$. The characteristic $p_{i70j}$ is the probability of outcome $j$ associated with the $i$th batsman at the juncture of the match immediately following the powerplay (i.e. the 7th over) when no wickets have been taken. The multiplicative parameter $\tau_{owj}$ scales the baseline performance characteristic $p_{i70j}$ to the stage of the match corresponding to the $o$th over with $w$ wickets taken. The denominator in (3.3) ensures that the relevant probabilities sum to unity. There is an implicit assumption in (3.3) that although batsmen are unique, their batting characteristics change with respect to overs and wickets by the same multiplicative factor which is essentially an indicator of aggression. For example, when aggressiveness increases relative to the baseline state, one would expect $\tau_{ow4} > 1$ and $\tau_{ow6} > 1$ since bolder batting leads to more 4’s and 6’s.

Given the estimation of the parameters in (3.3) (see Davis, Perera & Swartz 2015b), an algorithm for simulating first innings runs against an average bowler is available. One simply generates multinomial batting outcomes in (3.1) according to the laws of cricket. For example, when either 10 wickets are accumulated or the number of overs reaches 20, the first innings is terminated. Davis, Perera & Swartz (2015) also provide modifications for batsmen facing specific bowlers (instead of average bowlers), they account for the home field advantage and they provide adjustments for second innings simulation. In summary, with such a simulator, we are able to replicate matches, and estimate the expected runs scored when Team A (lineup specified) plays against Team B (lineup specified). Davis, Perera & Swartz (2015) demonstrate that the simulator generates realistic Twenty20 matches.

### 3.3 PLAYER EVALUATION

Recall that our objective in player evaluation is the development of a metric that measures player contribution in terms of run differential relative to baseline players. We restrict our attention to first innings performances since the second innings involves a target score whereby players alter their standard strategies. Accordingly, we define $R_s(l)$ as the number
of runs scored in the first innings with a batting lineup $l$. Letting $t_{bat}$ denote a typical batting lineup, the quantity $R_s(t_{bat})$ is therefore the standard of comparison and

$$E(D_s(l)) = E(R_s(l) - R_s(t_{bat}))$$

$$= E(R_s(l)) - E(R_s(t_{bat}))$$

(3.4)

is the expected differential for runs scored due to batting lineup $l$. An above average batting lineup $l$ is one which produces a positive value of $E(D_s(l))$ and a below average batting lineup $l$ is one which produces a negative value of $E(D_s(l))$. Operationally, the calculations of $E(R_s(l))$ and $E(R_s(t_{bat}))$ are obtained via match simulation. The simulation follows the description provided in Section 3.2 where the batting characteristics of player $i$ who forms part of a batting lineup are given by $p_{iowj}$.

Since success in cricket depends on both scoring runs and preventing runs, we introduce analogous measures for bowling. Accordingly, we define $R_c(l)$ as the number of runs conceded by the bowling lineup $l$ in the first innings. Letting $t_{bowl}$ denote a typical bowling lineup, the quantity $R_c(t_{bowl})$ is therefore the standard of comparison and

$$E(D_c(l)) = E(R_c(l) - R_c(t_{bowl}))$$

$$= E(R_c(l)) - E(R_c(t_{bowl}))$$

(3.5)

is the expected differential for runs conceded due to bowling lineup $l$. An above average bowling lineup $l$ is one which produces a negative value of $E(D_c(l))$ and a below average bowling lineup $l$ is one which produces a positive value of $E(D_c(l))$. The simulations used in the calculation of (3.5) require bowling characteristics for each of the bowlers in the bowling lineups. The bowling characteristics $q_{iowj}$ for player $i$ are analogous to the batting characteristics.

Summarizing, (3.4) measures the batting contribution of a batting lineup $l$. Similarly, (3.5) measures the bowling contribution of a bowling lineup $l$. We now wish to synthesize these two components to evaluate the overall contribution of an individual player. For player $i$, let $l_{bat,i} = t_{bat}$ except that player $i$ is inserted into the batting lineup. Similarly, let $l_{bowl,i} = t_{bowl}$ except that player $i$ is inserted into the bowling lineup. If player $i$ is a pure batsman, then he is not inserted into the bowling lineup and $l_{bowl,i} = t_{bowl}$. It follows that

$$E(D(i)) = E(D_s(l_{bat,i})) - E(D_c(l_{bowl,i}))$$

(3.6)

is the overall expected run differential due to player $i$. The quantity (3.6) is interpreted as the average number of runs that player $i$ contributes to his team over a baseline player. Since runs is the currency of winning matches, $E(D(i))$ provides a direct evaluation of player worth.
There are two remaining details required in the evaluation of (3.4) and (3.5). We need to define the typical batting lineup $t_{bat}$ and the typical bowling lineup $t_{bowl}$. For $t_{bat}$, we consider all 448 players in our dataset, and for each player, we determine their mean batting position $(1, \ldots, 11)$ based on their individual match histories. For all batsmen $i$ who are classified according to batting position $k$, we average their batting characteristics $p_{iowj}$ to obtain batting characteristics for the typical batsman who bats in position $k$. We note that there is not a lot of data available for batting performances in batting positions 10 and 11. In these two positions, we use a pooled average over the two positions. For $t_{bowl}$, we average bowling characteristics over all 306 bowlers. We then set $t_{bowl}$ to consist of five identical bowlers with the average bowling characteristics, each who bowl four overs. In the above discussion, all averages refer to weighted averages where the weights reflect the number of matches played by individual players.

We note that there is considerable flexibility in the proposed approach. Whereas (3.6) provides the number of runs that player $i$ contributes over a baseline player, the lineups $t_{bat}$ and $t_{bowl}$ do not need to be typical lineups. For instance, these baseline lineups could correspond to a player’s team, and then (3.6) quantifies the contribution of the player to his specific team. Also, the development of (3.4) and (3.5) suggest that not only can we compare individual players but subsets of players. For example, a team may be interested in knowing how the substitution of three players from their standard roster affects expected run differential.

### 3.3.1 Pure Batsmen

Pure batsmen do not bowl. It follows that their overall performance is based entirely on batting and the metric of interest (3.6) for a pure batsman $i$ reduces to

$$E(D(i)) = E(D_{s}(h_{bat,i})) .$$

(3.7)

When assessing pure batsmen, it is important to compare apples with apples. Therefore, in the calculation of (3.7), we always insert a pure batsman $i$ into batting position 3 when simulating matches. The third batting position is the average batting position for pure batsmen.

Table 3.1 provides the performance metric (3.7) for the 50 batsmen in our dataset who have faced at least 250 balls. These are primarily well-established batsmen with a long history in Twenty20 cricket. Wicketkeepers in Table 3.1 are marked with an asterisk; it may be reasonable to assess them separately from the other pure batsmen since wicketkeepers contribute in a meaningful way that goes beyond batting.

Ahmed Shehzad is the best pure batsman with $E(D) = 7.83$. This means that if an average pure batsman is replaced by Shehzad, a team’s scoring would increase by 7.83 runs on average. There are some surprises in Table 3.1. For example, AB de Villiers does not
have an exceptional expected run differential \( (E(D) = 1.66) \) yet he is regarded as one of the best Twenty20 batsmen. On the other hand, MDKJ Perera is rated as the best Sri Lankan pure batsman, and is ranked above the Sri Lankan legends Jayawardene and Sangakkara.

There are no pure batsmen who are much worse than \( E(D) = 0 \), likely because their poor performances prevented them from playing long enough to face 250 balls. We also observe that there are few wicketkeepers at the top of the list (only BB McCullum and K Sangakkara). This might be anticipated because the specialized skills of a wicketkeeper may be sufficient for their continued selection.

The \( E(D) \) measure can also be used to estimate the effect of specific player replacements. For example, although they did not play during the same time period, it is interesting to compare the South African wicketkeepers Mark Boucher (now retired) and Quinton de Kock. With de Kock \( (E(D) = -1.85) \) in the batting lineup instead of Boucher \( (E(D) = -4.04) \), South Africa could expect to score \(-1.85 - (-4.04) = 2.19\) additional runs.

### 3.3.2 Bowlers

Surprisingly, the term “bowler” is not well-defined. The intention is that a player designated as a bowler is one who specializes in bowling and is not “good” at batting. We are going to make the term precise and define a bowler as a player who bowls and whose average batting position is 8, 9, 10 or 11. Since a bowler bats late in the lineup, he does not bat often and his expected differential for runs scored \( E(D_s(l_{bat}, i)) \) is negligible. Therefore the metric of interest (3.6) for bowler \( i \) reduces to

\[
E(D(i)) \approx E(D_c(l_{bowl}, i))
\]

The calculation of (3.6) is obtained by simulation where four bowling overs are uniformly selected from the innings and these are the overs that are assigned to bowler \( i \). The simulation is based on estimated bowling characteristics where \( q_{iowj} \) denotes the probability of outcome \( j \) by bowler \( i \) in the \( o \)th over when \( w \) wickets have been taken.

As any cricket fan knows, the taking of wickets is something that distinguishes bowlers and is highly valued. We wish to emphasize that wicket taking is an important component of our metric (3.6). A bowler \( i \) who takes wickets regularly has larger bowling characteristics \( q_{iow7} \) than a typical bowler. Therefore, in the simulation procedure, such bowlers take wickets more often, runs conceded are reduced and wicket taking ability is recognized.

Table 3.2 provides the performance metric (3.6) for the 60 bowlers in our dataset who have bowled at least 250 balls. These are primarily well-established bowlers with a long history in Twenty20 cricket. When comparing Table 3.2 to Table 3.1, we observe that the bowlers at the top of the list contribute more to their team than do the top batsmen. This may be relevant to the IPL auctions where teams should perhaps spend more money on top bowlers than on top batsmen. We also note that Chris Mpofu has a very poor expected run
differential $E(D) = -11.45$. The natural question is how can he continue to play? Perhaps this is due to the fact that he plays for Zimbabwe, a weak ICC team that has little depth in its bowling selection pools.

Interestingly, among the top five bowlers according to the October 2014 ICC rankings, only Sachithra Senanayake and Samuel Badree place highly in terms of $E(D)$. The other three bowlers, Sunil Narine, Saeed Ajmal and Mitchell Starc are found near the top quartile of the $E(D)$ rankings. Coincidently, Senanayake, Ajmal, and Narine have been recently banned by the ICC for illegal bowling actions.

Table 3.2 also suggests that there is little difference between fast and spin bowlers in terms of $E(D)$. In cricket commentary and tactics, much is made about the distinction between fast and spin bowlers. For example, it is customary for teams to begin innings with fast bowlers and to impose a particular composition of both fast and spin bowlers in the bowling lineup. We believe that teams should consider bowler selection with a greater emphasis on actual performance. The $E(D)$ statistics in Table 3.2 tell us precisely about bowling contributions in terms of runs. If a team, for example, has a preponderance of quality fast bowlers, they should perhaps think twice about substituting one of these exceptionally fast bowlers for a mediocre spin bowler.

### 3.3.3 All-Rounders

As with bowlers, the term “all-rounder” is not well-defined although it is intended to convey that a player excels at both batting and bowling. We define an all-rounder as a player who bowls and whose average career batting position is 7 or earlier in the lineup. The calculation of (3.6) involves simulations where the all-rounder of interest is inserted into position 5 of the batting order. For bowling, four overs are uniformly selected from the 20 overs in the innings and these are the overs that are bowled the all-rounder.

Table 3.3 provides the performance metric (3.6) for the 25 all-rounders in our dataset who have faced at least 250 balls and who have bowled at least 250 balls. These are primarily well-established all-rounders with a long history in Twenty20 cricket.

Among the all-rounders, there are some players who have exceptionally good batting components of their $E(D)$. For example, Thisara Perera is considered one of the best all-rounders in our data, owing entirely to his outstanding batting performance, and in spite of his poor bowling performance. Perera would take the top spot in Table 3.1, had he been a pure batsman during his career, which has now ended. Strategically, it may have been preferable for Sri Lanka to utilize Perera as a pure batsman rather than an all-rounder. The same might be said of Kieron Pollard of the West Indies. And by a similar logic, Pakistan might be better served to use Abdul Razzaq as a bowler rather than an all-rounder. These are strategies that may be of considerable benefit to teams.
3.3.4 Additional Analyses

In Tables 3.1, 3.2 and 3.3, we calculated the expected run differential metric (3.6) for pure batsmen, bowlers and all-rounders, respectively. It is interesting to see how the new measures for batting (3.4) and for bowling (3.5) compare to standard performance measures.

In Table 3.4, we provide correlations involving the new measures against the traditional batting average, strike rate, bowling average and economy rate. The correlations are stratified over the three classes of players. We observe that all metrics have similar correlations, neither strong nor weak. If we take E(D) as the gold standard for performance evaluation, then strike rate should be slightly preferred to batting average as a batting measure in Twenty20. Similarly, economy rate should be slightly preferred to bowling average as a bowling measure in Twenty20. These findings are in keeping with the view that wickets are less important in Twenty20 due to the shorter nature of the game when compared to one-day cricket. We note that both bowling average and batting average express runs relative to wickets.

Up until now, our analyses have focused on career performances. However, in some situations such as team selection, it is current form which is of greater importance. Davis, Perera & Swartz (2015) provide methodology for determining current form. The approach is implemented by providing more weight to recent match performances. To see that the distinction between career performance and current form is meaningful, Table 5 reports the baseline characteristics for AB de Villiers, Mohammad Hafeez and Umar Gul based on both career performance and current form (up to August 2014). AB de Villiers, a pure batsman, has better recent form than his average career performance where he is now scoring roughly one more run per over than his career average. Much of de Villiers improvement may be attributed to added power as he is now scoring 4’s and 6’s with more regularity. On the other hand, Umar Gul, a bowler, is experiencing a decline in performance in recent matches compared to his career values, allowing 1.66 additional runs per over. We observe that the current form of Mohammad Hafeez is in keeping with his average career performance in both batting and bowling.

More generally, it is interesting to investigate how current form compares with career performances across all players. We look at the correlation between E(D) in (3.6) with respect to current form and career for the players available in our dataset. The correlations are 0.77 for pure batsmen, 0.91 for bowlers and 0.68 for all-rounders. This suggests that although performances change over time, the changes are not typically great. The cases of AB de Villiers and Umar Gul (discussed above) are two of the most dramatic in our dataset.

With the availability of batting and bowling characteristics representing current form as in Table 3.5, we carry out further simulations to obtain the expected run differential metric (3.6) based on current form. It is interesting to compare our metric (3.6) with the
Reliance ICC ratings which also reflect current form. The Reliance ICC ratings are taken from October 5, 2014.

In Figure 3.1, we provide a scatterplot of our metric (3.6) based on current form against the Reliance ICC rating for the 50 pure batsmen in our dataset who have faced at least 250 balls. There is a moderate correlation \( r = 0.56 \) between the Reliance ICC batting ratings and the \( E(D) \) for pure batsmen. We observe that Younis Khan is valued highly using expected run differential \( E(D) = 5.01 \) yet his Reliance ICC rating (309) is mediocre for a pure batsman.

In Figure 3.2, we provide a scatterplot of our metric (3.6) based on current form against the Reliance ICC rating for the 60 bowlers in our dataset who have bowled at least 250 balls. As in Figure 3.1, we obtained a moderate correlation \( r = 0.61 \) between the Reliance ICC bowling ratings and the \( E(D) \) for bowlers. We note that Samuel Badree (ICC = 831), Sunil Narine (ICC = 808), Graeme Swann (ICC = 750) and Sachithra Senanayake (ICC = 712) are each identified as outstanding bowlers using both measures. However, there are interesting discrepancies between our metric and the Reliance ICC ratings for bowlers. For example, Brett Lee is valued highly using expected run differential \( E(D) = 7.88 \) yet his Reliance ICC rating (501) is only average for a bowler. On the other hand, Chris Mpofu has an extremely poor expected run differential \( E(D) = -11.45 \) yet his Reliance ICC rating (418) is only a little below average.

In Figure 3.3, we provide a scatterplot of our metric (3.6) based on current form against the Reliance ICC rating for the 25 all-rounders in our dataset who have faced at least 250 balls and have bowled at least 250 balls. In this case, the correlation between our metric and the Reliance ICC all-rounder ratings was \( r = -0.04 \). If we believe in the metric \( E(D) \) as the gold standard for player evaluation, then there is little value in the Reliance ICC all-rounder rating. We note that the Reliance ICC all-rounder rating is proportional to the product of the Reliance ICC bowling and batting ratings. Taking a product is not a recommended approach for combining ratings.

Another investigation with “moneyball” in mind concerns salary. We are interested in how the expected run differential measure (which measures true contribution) compares against perceived worth expressed as salary. To make this investigation, we have collected salary data from the 2012-2014 IPL seasons.

Figures 3.4, 3.5, and 3.6 provide scatterplots of most recent IPL salaries against our metric (3.6) based on current form for the 21 pure batsmen, 26 bowlers, and 18 all-rounders from our dataset who played in the IPL during the period. In each case, there is no detectible correlation between a player’s performance by the \( E(D) \) metric and their salaries. The year of a player’s most recent IPL salary, denoted by the shape of the plotted points in Figures 3.4, 3.5, and 3.6, explains more of the variation in salaries than our metric. We take this as a sign that the IPL is increasing in popularity and that the players’ compensation is not reflective of their impact on a team. Player salaries may be confounded by the auction
system where players are assigned to teams and salaries are determined. Problems with the auction system including the limited information that teams have while bidding, are discussed in Swartz (2011).

For comparison purposes, Figure 3.7 provides scatterplots of the most recent IPL salaries against the Reliance ICC ratings. The three plots correspond to batsmen, bowlers and all-rounders. The correlations here seem a little stronger than in Figures 3.4, 3.5 and 3.6. If we believe that expected run differential $E(D)$ is the definitive measure of performance, then Figure 3.7 suggests that there may be mispricings in the IPL marketplace which are predicated on the ICC ratings.

We extend our analyses in two further directions. First, we ask whether it is a good idea to use only first innings data for the estimation of batting characteristics $p_{lowj}$ and bowling characteristics $q_{lowj}$. The rationale is that players are more directly comparable based on their first innings performances. In the second innings, batting behavior depends greatly on the target. For example, a second innings batsman behaves much differently with 3 overs remaining and 7 wickets taken when his team is behind 10 runs (he is very cautious) compared to the situation when he is behind 35 runs (he is very aggressive).

We therefore repeat our analysis of career performance by including second innings data. Perhaps it is the case that second innings conditions average out in terms of cautious and aggressive situations. In Figure 3.8, we provide a scatterplot of the $E(D)$ statistic based on both innings versus the $E(D)$ statistic based on the first innings. The correlation $r = 0.73$ indicates some agreement between the two approaches although there are cases where the differences are considerable. The natural question is which of the measures should be more trusted for player evaluation? We take the view that there is value in considering both measures. When there are large discrepancies between the two measures, it indicates a difference in performance between the two innings. We believe in such cases it would be useful to look at the circumstances associated with the second innings. For example, it is conceivable that some players may be well-suited or ill-suited for the pressure of a chase during the second innings.

Our final analysis compares our expected run differential metric $E(D)$ against another proposed performance metric. We have pointed out in the Introduction that there are logical flaws with the commonly used statistics batting average, strike rate, bowling average and economy rate. Croucher (2000) also recognized these limitations and consequently proposed the batting index

$$C_1 = \text{batting average} \times \text{strike rate} \quad (3.8)$$

and the bowling index

$$C_2 = \text{bowling average} \times \text{economy rate} \quad (3.9)$$
as alternative measures. The appeal of (3.8) and (3.9) is that both statistics take into account the three important components of cricket, namely runs, overs and wickets. The statistics are also easily computable. A drawback of $C_1$ and $C_2$ is that they are not directly interpretable. For example, what is meant by a batting index $C_1 = 2000$? Also, $C_1$ and $C_2$ compare batting and bowling contributions separately whereas $E(D)$ addresses a player’s overall contribution in terms of expected run differential. In Figure 3.9, we compare Croucher’s statistic (3.8) for batsmen with our metric using career Twenty20 performances. We observe some agreement between the two approaches ($r = 0.60$).

3.4 DISCUSSION

Traditional performance measures in Twenty20 cricket may not be seen as “fair”. For example, it is easier to score runs for an opening batsman than a batsman who bats in position 7. This paper overcomes these types of difficulties and develops performance measures that focus on expected run scoring differential relative to baseline players. Although there is no gold standard for measuring performance statistics, we take it as axiomatic that expected run differential is the correct metric in Twenty20 cricket. The reason is that the rules of the game are such that a team defeats its opponent if they score more runs. With an emphasis on what is really important in winning matches, the metrics introduce a “moneyball” philosophy to Twenty20 cricket. The metrics are also flexible in the sense that baseline players can be modified and subsets of players can be simultaneously evaluated.

We have observed that the magnitude of $E(D)$ values for pure batsmen, bowlers and all-rounders are comparable. The differences between the best and worst pure batsmen, bowlers, and all-rounders are approximately 13, 21, and 13 runs, respectively. This suggests that it is possible for all players to make meaningful contributions to the game regardless of position.

Whereas our performance analysis takes both batting and bowling into account, there exists the possibility for future refinements. For example, fielding is an important component of cricket and it would be useful to quantify fielding contributions in terms of expected run differential. Also, how can one measure a wicketkeeper’s contribution beyond his batting performances?

Another avenue for future research involves data collection. Currently, we use only Twenty20 international matches in forming player characteristics. Is there a way of combining information that comes from other competitions such as the IPL and the Big Bash?

3.5 ACKNOWLEDGEMENTS

Tim Swartz has been partially supported by grants from the Natural Sciences and Engineering Research Council of Canada. The authors thank the two Editors Philip Maymin
and Eugene Shen, and three anonymous reviewers whose comments have helped improve the manuscript.

<table>
<thead>
<tr>
<th>Name</th>
<th>Team</th>
<th>E(D)</th>
<th>Bat Avg</th>
<th>Name</th>
<th>Team</th>
<th>E(D)</th>
<th>Bat Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Shehzad</td>
<td>Pak</td>
<td>7.83</td>
<td>25.42</td>
<td>H Masakadza</td>
<td>Zim</td>
<td>0.82</td>
<td>33.98</td>
</tr>
<tr>
<td>BB McCullum*</td>
<td>NZ</td>
<td>6.93</td>
<td>38.25</td>
<td>C Kapugedera</td>
<td>SL</td>
<td>0.80</td>
<td>20.25</td>
</tr>
<tr>
<td>MDKJ Perera</td>
<td>SL</td>
<td>6.77</td>
<td>35.02</td>
<td>C White</td>
<td>Aus</td>
<td>0.69</td>
<td>29.26</td>
</tr>
<tr>
<td>K Pietersen</td>
<td>Eng</td>
<td>6.40</td>
<td>40.78</td>
<td>B Taylor</td>
<td>Zim</td>
<td>0.46</td>
<td>25.64</td>
</tr>
<tr>
<td>R Ponting</td>
<td>Aus</td>
<td>5.85</td>
<td>29.13</td>
<td>L Thirimanne</td>
<td>SL</td>
<td>0.29</td>
<td>17.86</td>
</tr>
<tr>
<td>A Hales</td>
<td>Eng</td>
<td>5.50</td>
<td>42.90</td>
<td>I Nazir</td>
<td>Pak</td>
<td>0.27</td>
<td>29.23</td>
</tr>
<tr>
<td>M Jayawardene</td>
<td>SL</td>
<td>5.04</td>
<td>33.18</td>
<td>C Kieswetter</td>
<td>Eng</td>
<td>0.23</td>
<td>23.80</td>
</tr>
<tr>
<td>Y Khan</td>
<td>Pak</td>
<td>5.01</td>
<td>23.91</td>
<td>O Shah</td>
<td>Eng</td>
<td>-0.10</td>
<td>27.84</td>
</tr>
<tr>
<td>E Morgan</td>
<td>Eng</td>
<td>4.98</td>
<td>32.08</td>
<td>K Akmal</td>
<td>Pak</td>
<td>-0.20</td>
<td>19.54</td>
</tr>
<tr>
<td>U Akmal</td>
<td>Pak</td>
<td>4.76</td>
<td>30.68</td>
<td>L Simmons</td>
<td>WI</td>
<td>-0.31</td>
<td>29.94</td>
</tr>
<tr>
<td>MEK Hussey</td>
<td>Aus</td>
<td>4.07</td>
<td>39.77</td>
<td>S Butt</td>
<td>Pak</td>
<td>-0.41</td>
<td>27.21</td>
</tr>
<tr>
<td>D Miller</td>
<td>SA</td>
<td>3.88</td>
<td>24.91</td>
<td>MS Dhoni*</td>
<td>Ind</td>
<td>-0.78</td>
<td>39.03</td>
</tr>
<tr>
<td>D Warner</td>
<td>Aus</td>
<td>3.49</td>
<td>29.08</td>
<td>A Lumb</td>
<td>Eng</td>
<td>-0.92</td>
<td>23.48</td>
</tr>
<tr>
<td>K Sangakkara*</td>
<td>SL</td>
<td>3.25</td>
<td>35.21</td>
<td>D Ramdin*</td>
<td>WI</td>
<td>-1.06</td>
<td>18.08</td>
</tr>
<tr>
<td>G Smith</td>
<td>SA</td>
<td>3.04</td>
<td>32.19</td>
<td>G Bailey</td>
<td>Aus</td>
<td>-1.31</td>
<td>26.31</td>
</tr>
<tr>
<td>F du Plessis</td>
<td>SA</td>
<td>2.58</td>
<td>41.64</td>
<td>Misbah-ul-Haq</td>
<td>Pak</td>
<td>-1.32</td>
<td>40.19</td>
</tr>
<tr>
<td>N Jamshed</td>
<td>Pak</td>
<td>1.99</td>
<td>21.44</td>
<td>B Haddin</td>
<td>Aus</td>
<td>-1.58</td>
<td>17.92</td>
</tr>
<tr>
<td>AB de Villiers</td>
<td>SA</td>
<td>1.66</td>
<td>20.66</td>
<td>D Chandimal</td>
<td>SL</td>
<td>-1.77</td>
<td>12.71</td>
</tr>
<tr>
<td>G Gambhir</td>
<td>Ind</td>
<td>1.53</td>
<td>34.33</td>
<td>Q de Kock*</td>
<td>SA</td>
<td>-1.85</td>
<td>35.15</td>
</tr>
<tr>
<td>J Buttler</td>
<td>Eng</td>
<td>1.52</td>
<td>23.83</td>
<td>S Chanderpaul</td>
<td>WI</td>
<td>-2.23</td>
<td>20.08</td>
</tr>
<tr>
<td>H Gibbs</td>
<td>SA</td>
<td>1.19</td>
<td>18.59</td>
<td>T Iqbal</td>
<td>Ban</td>
<td>-2.28</td>
<td>20.29</td>
</tr>
<tr>
<td>H Amla</td>
<td>SA</td>
<td>1.15</td>
<td>26.18</td>
<td>J Charles</td>
<td>WI</td>
<td>-2.37</td>
<td>21.25</td>
</tr>
<tr>
<td>R Taylor</td>
<td>NZ</td>
<td>1.08</td>
<td>26.73</td>
<td>M Boucher*</td>
<td>SA</td>
<td>-4.04</td>
<td>19.40</td>
</tr>
<tr>
<td>M Guptill</td>
<td>NZ</td>
<td>1.06</td>
<td>32.72</td>
<td>R Sarwan</td>
<td>WI</td>
<td>-4.33</td>
<td>21.83</td>
</tr>
<tr>
<td>V Sehwag</td>
<td>Ind</td>
<td>1.01</td>
<td>27.43</td>
<td>M Rahim*</td>
<td>Ban</td>
<td>-4.64</td>
<td>24.29</td>
</tr>
</tbody>
</table>

Table 3.1: Performance metrics of pure batsmen with at least 250 balls faced. Wicketkeepers are marked with an asterisk.
<table>
<thead>
<tr>
<th>Name</th>
<th>Team</th>
<th>E(D)</th>
<th>Econ</th>
<th>Style</th>
<th>Name</th>
<th>Team</th>
<th>E(D)</th>
<th>Econ</th>
<th>Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>S Senanayake</td>
<td>SL</td>
<td>9.94</td>
<td>5.72</td>
<td>Spin</td>
<td>J Botha</td>
<td>SA</td>
<td>1.84</td>
<td>6.51</td>
<td>Spin</td>
</tr>
<tr>
<td>H Singh</td>
<td>Ind</td>
<td>9.43</td>
<td>6.50</td>
<td>Spin</td>
<td>S Tait</td>
<td>Aus</td>
<td>1.69</td>
<td>6.66</td>
<td>Fast</td>
</tr>
<tr>
<td>D Vettori</td>
<td>NZ</td>
<td>9.34</td>
<td>5.63</td>
<td>Spin</td>
<td>J Butler</td>
<td>NZ</td>
<td>1.68</td>
<td>8.12</td>
<td>Fast</td>
</tr>
<tr>
<td>I Tahir</td>
<td>SA</td>
<td>8.44</td>
<td>5.57</td>
<td>Spin</td>
<td>R Price</td>
<td>Zim</td>
<td>1.38</td>
<td>6.47</td>
<td>Spin</td>
</tr>
<tr>
<td>B Lee</td>
<td>Aus</td>
<td>7.88</td>
<td>7.46</td>
<td>Fast</td>
<td>F Edwards</td>
<td>WI</td>
<td>1.33</td>
<td>7.82</td>
<td>Fast</td>
</tr>
<tr>
<td>S Badree</td>
<td>WI</td>
<td>6.66</td>
<td>5.72</td>
<td>Spin</td>
<td>W Parnell</td>
<td>SA</td>
<td>0.87</td>
<td>7.67</td>
<td>Fast</td>
</tr>
<tr>
<td>BAW Mendis</td>
<td>SL</td>
<td>6.34</td>
<td>6.65</td>
<td>Spin</td>
<td>J Anderson</td>
<td>Eng</td>
<td>0.82</td>
<td>7.55</td>
<td>Fast</td>
</tr>
<tr>
<td>R Peterson</td>
<td>SA</td>
<td>5.97</td>
<td>7.54</td>
<td>Spin</td>
<td>R Ashwin</td>
<td>Ind</td>
<td>0.75</td>
<td>7.16</td>
<td>Spin</td>
</tr>
<tr>
<td>D Steyn</td>
<td>SA</td>
<td>5.57</td>
<td>6.37</td>
<td>Fast</td>
<td>R McLaren</td>
<td>SA</td>
<td>0.74</td>
<td>7.46</td>
<td>Fast</td>
</tr>
<tr>
<td>G Swann</td>
<td>Eng</td>
<td>5.34</td>
<td>6.45</td>
<td>Spin</td>
<td>A Razzak</td>
<td>Ban</td>
<td>0.74</td>
<td>7.24</td>
<td>Spin</td>
</tr>
<tr>
<td>M Amir</td>
<td>Pak</td>
<td>5.33</td>
<td>7.19</td>
<td>Fast</td>
<td>D Nannes</td>
<td>Aus</td>
<td>0.70</td>
<td>7.50</td>
<td>Fast</td>
</tr>
<tr>
<td>J Taylor</td>
<td>WI</td>
<td>4.79</td>
<td>7.59</td>
<td>Fast</td>
<td>D Fernando</td>
<td>SL</td>
<td>0.58</td>
<td>7.43</td>
<td>Fast</td>
</tr>
<tr>
<td>S Bond</td>
<td>NZ</td>
<td>4.60</td>
<td>7.11</td>
<td>Fast</td>
<td>S Benn</td>
<td>WI</td>
<td>0.36</td>
<td>7.00</td>
<td>Spin</td>
</tr>
<tr>
<td>S Narine</td>
<td>WI</td>
<td>4.51</td>
<td>5.93</td>
<td>Spin</td>
<td>R Jadeja</td>
<td>Ind</td>
<td>-0.11</td>
<td>7.34</td>
<td>Spin</td>
</tr>
<tr>
<td>U Gul</td>
<td>Pak</td>
<td>4.31</td>
<td>7.21</td>
<td>Fast</td>
<td>R Hira</td>
<td>NZ</td>
<td>-0.18</td>
<td>7.73</td>
<td>Spin</td>
</tr>
<tr>
<td>M Morkel</td>
<td>SA</td>
<td>4.24</td>
<td>7.10</td>
<td>Fast</td>
<td>Z Khan</td>
<td>Ind</td>
<td>-0.41</td>
<td>7.24</td>
<td>Fast</td>
</tr>
<tr>
<td>S Ajmal</td>
<td>Pak</td>
<td>4.17</td>
<td>6.41</td>
<td>Spin</td>
<td>T Southee</td>
<td>NZ</td>
<td>-0.62</td>
<td>8.52</td>
<td>Fast</td>
</tr>
<tr>
<td>M Starc</td>
<td>Aus</td>
<td>4.15</td>
<td>6.89</td>
<td>Fast</td>
<td>S Broad</td>
<td>Eng</td>
<td>-1.09</td>
<td>7.69</td>
<td>Fast</td>
</tr>
<tr>
<td>M Yardy</td>
<td>Eng</td>
<td>4.11</td>
<td>6.38</td>
<td>Spin</td>
<td>S Akhtar</td>
<td>Pak</td>
<td>-1.14</td>
<td>8.36</td>
<td>Fast</td>
</tr>
<tr>
<td>N Kulasekara</td>
<td>SL</td>
<td>4.10</td>
<td>7.12</td>
<td>Fast</td>
<td>M Muralitharan</td>
<td>SL</td>
<td>-1.24</td>
<td>6.60</td>
<td>Spin</td>
</tr>
<tr>
<td>S Finn</td>
<td>Eng</td>
<td>3.60</td>
<td>7.58</td>
<td>Fast</td>
<td>T Bresnan</td>
<td>Eng</td>
<td>-1.44</td>
<td>7.88</td>
<td>Fast</td>
</tr>
<tr>
<td>M McClanaghan</td>
<td>NZ</td>
<td>3.57</td>
<td>8.29</td>
<td>Fast</td>
<td>IK Pathan</td>
<td>Ind</td>
<td>-2.70</td>
<td>7.84</td>
<td>Fast</td>
</tr>
<tr>
<td>N Bracken</td>
<td>Aus</td>
<td>3.43</td>
<td>6.93</td>
<td>Fast</td>
<td>J Dernbach</td>
<td>Eng</td>
<td>-3.64</td>
<td>8.35</td>
<td>Fast</td>
</tr>
<tr>
<td>S Tanvir</td>
<td>Pak</td>
<td>3.28</td>
<td>7.06</td>
<td>Fast</td>
<td>K Mills</td>
<td>NZ</td>
<td>-3.77</td>
<td>8.22</td>
<td>Fast</td>
</tr>
<tr>
<td>NL McCullum</td>
<td>NZ</td>
<td>2.83</td>
<td>6.86</td>
<td>Spin</td>
<td>J Tredwell</td>
<td>Eng</td>
<td>-4.12</td>
<td>8.21</td>
<td>Spin</td>
</tr>
<tr>
<td>R Sidebottom</td>
<td>Eng</td>
<td>2.66</td>
<td>7.22</td>
<td>Fast</td>
<td>B Hogg</td>
<td>Aus</td>
<td>-6.66</td>
<td>7.86</td>
<td>Spin</td>
</tr>
<tr>
<td>L Malinga</td>
<td>SL</td>
<td>2.45</td>
<td>7.14</td>
<td>Fast</td>
<td>I Sharma</td>
<td>Ind</td>
<td>-7.38</td>
<td>8.69</td>
<td>Fast</td>
</tr>
<tr>
<td>M Johnson</td>
<td>Aus</td>
<td>2.04</td>
<td>6.98</td>
<td>Fast</td>
<td>M Mortaza</td>
<td>Ban</td>
<td>-7.83</td>
<td>9.09</td>
<td>Fast</td>
</tr>
<tr>
<td>S Pollock</td>
<td>SA</td>
<td>1.98</td>
<td>7.35</td>
<td>Fast</td>
<td>R Rampaul</td>
<td>WI</td>
<td>-8.37</td>
<td>8.45</td>
<td>Fast</td>
</tr>
<tr>
<td>P Utseya</td>
<td>Zim</td>
<td>1.93</td>
<td>6.66</td>
<td>Spin</td>
<td>C Mpofu</td>
<td>Zim</td>
<td>-11.45</td>
<td>8.84</td>
<td>Fast</td>
</tr>
</tbody>
</table>

Table 3.2: Performance metrics of bowlers with at least 250 balls bowled.
<table>
<thead>
<tr>
<th>Name</th>
<th>Team</th>
<th>Style</th>
<th>$E(D)_{Bat}$</th>
<th>$E(D)_{Bowl}$</th>
<th>$E(D)$</th>
<th>BowlEcon</th>
<th>BatAvg</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Razzaq</td>
<td>Pak</td>
<td>Fast</td>
<td>0.30</td>
<td>8.25</td>
<td>8.55</td>
<td>7.39</td>
<td>19.75</td>
</tr>
<tr>
<td>T Perera</td>
<td>SL</td>
<td>Fast</td>
<td>7.95</td>
<td>0.56</td>
<td>8.50</td>
<td>8.31</td>
<td>34.90</td>
</tr>
<tr>
<td>D Sammy</td>
<td>WI</td>
<td>Fast</td>
<td>2.38</td>
<td>5.62</td>
<td>7.99</td>
<td>7.24</td>
<td>18.97</td>
</tr>
<tr>
<td>A Mathews</td>
<td>SL</td>
<td>Fast</td>
<td>2.96</td>
<td>3.76</td>
<td>6.72</td>
<td>6.98</td>
<td>26.76</td>
</tr>
<tr>
<td>Y Singh</td>
<td>Ind</td>
<td></td>
<td>4.98</td>
<td>0.39</td>
<td>5.37</td>
<td>7.46</td>
<td>31.39</td>
</tr>
<tr>
<td>M Samuels</td>
<td>WI</td>
<td>Spin</td>
<td>3.43</td>
<td>1.19</td>
<td>4.62</td>
<td>7.78</td>
<td>28.59</td>
</tr>
<tr>
<td>K Pollard</td>
<td>WI</td>
<td>Spin</td>
<td>7.00</td>
<td>-2.50</td>
<td>4.50</td>
<td>8.11</td>
<td>25.47</td>
</tr>
<tr>
<td>S Afridi</td>
<td>SL</td>
<td>Spin</td>
<td>1.90</td>
<td>2.35</td>
<td>4.25</td>
<td>6.66</td>
<td>18.99</td>
</tr>
<tr>
<td>C Gayle</td>
<td>WI</td>
<td>Spin</td>
<td>5.14</td>
<td>-1.35</td>
<td>3.79</td>
<td>7.23</td>
<td>36.70</td>
</tr>
<tr>
<td>S Styris</td>
<td>NZ</td>
<td>Spin</td>
<td>0.56</td>
<td>3.13</td>
<td>3.69</td>
<td>6.69</td>
<td>20.26</td>
</tr>
<tr>
<td>J Kallis</td>
<td>SA</td>
<td>Spin</td>
<td>1.80</td>
<td>1.83</td>
<td>3.63</td>
<td>7.34</td>
<td>36.12</td>
</tr>
<tr>
<td>DJ Hussey</td>
<td>Aus</td>
<td>Spin</td>
<td>1.70</td>
<td>1.92</td>
<td>3.62</td>
<td>6.57</td>
<td>21.69</td>
</tr>
<tr>
<td>JA Morkel</td>
<td>SA</td>
<td>Fast</td>
<td>2.07</td>
<td>1.07</td>
<td>3.15</td>
<td>7.99</td>
<td>23.09</td>
</tr>
<tr>
<td>J Franklin</td>
<td>NZ</td>
<td>Fast</td>
<td>-0.37</td>
<td>3.26</td>
<td>2.88</td>
<td>7.46</td>
<td>23.06</td>
</tr>
<tr>
<td>S Al Hasan</td>
<td>Ban</td>
<td>Spin</td>
<td>-0.85</td>
<td>2.91</td>
<td>2.06</td>
<td>6.57</td>
<td>17.58</td>
</tr>
<tr>
<td>M Hafeez</td>
<td>Pak</td>
<td>Spin</td>
<td>-0.66</td>
<td>2.16</td>
<td>1.50</td>
<td>6.61</td>
<td>21.84</td>
</tr>
<tr>
<td>S Watson</td>
<td>Aus</td>
<td>Fast</td>
<td>1.51</td>
<td>-0.13</td>
<td>1.37</td>
<td>7.69</td>
<td>26.20</td>
</tr>
<tr>
<td>S Malik</td>
<td>Pak</td>
<td>Spin</td>
<td>-1.04</td>
<td>2.30</td>
<td>1.27</td>
<td>6.69</td>
<td>23.10</td>
</tr>
<tr>
<td>JP Duminy</td>
<td>SA</td>
<td>Spin</td>
<td>3.58</td>
<td>-3.28</td>
<td>0.29</td>
<td>7.62</td>
<td>41.87</td>
</tr>
<tr>
<td>R Bopara</td>
<td>Eng</td>
<td>Spin</td>
<td>-2.13</td>
<td>1.04</td>
<td>-1.09</td>
<td>7.05</td>
<td>23.34</td>
</tr>
<tr>
<td>J Oram</td>
<td>NZ</td>
<td>Fast</td>
<td>-0.85</td>
<td>-0.43</td>
<td>-1.28</td>
<td>8.53</td>
<td>22.23</td>
</tr>
<tr>
<td>DJ Bravo</td>
<td>WI</td>
<td>Fast</td>
<td>1.79</td>
<td>-4.10</td>
<td>-2.31</td>
<td>8.49</td>
<td>30.68</td>
</tr>
<tr>
<td>L Wright</td>
<td>Eng</td>
<td>Spin</td>
<td>-1.04</td>
<td>-1.88</td>
<td>-2.93</td>
<td>8.21</td>
<td>14.51</td>
</tr>
<tr>
<td>M Mahmudullah</td>
<td>Ban</td>
<td>Fast</td>
<td>-1.05</td>
<td>-2.21</td>
<td>-3.26</td>
<td>7.78</td>
<td>24.81</td>
</tr>
<tr>
<td>S Jayasuriya</td>
<td>SL</td>
<td>Spin</td>
<td>-1.28</td>
<td>-3.06</td>
<td>-4.34</td>
<td>7.56</td>
<td>20.32</td>
</tr>
</tbody>
</table>

Table 3.3: Performance metrics of all-rounders with at least 250 balls faced and 250 balls bowled.
### Table 3.4: Pearson correlation between $E(D)$ and four established performance metrics: batting average, strike rate (SR), bowling average and economy rate.

<table>
<thead>
<tr>
<th>Role</th>
<th>Name</th>
<th>Form</th>
<th>$p_{700}$</th>
<th>$p_{701}$</th>
<th>$p_{702-3}$</th>
<th>$p_{704}$</th>
<th>$p_{706}$</th>
<th>$p_{70w}$</th>
<th>E(R)/Over</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batting</td>
<td>AB de Villiers</td>
<td>Career</td>
<td>0.305</td>
<td>0.389</td>
<td>0.109</td>
<td>0.136</td>
<td>0.028</td>
<td>0.033</td>
<td>7.97</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current</td>
<td>0.294</td>
<td>0.367</td>
<td>0.099</td>
<td>0.176</td>
<td>0.036</td>
<td>0.029</td>
<td>8.96</td>
</tr>
<tr>
<td></td>
<td>M Hafeez</td>
<td>Career</td>
<td>0.368</td>
<td>0.365</td>
<td>0.091</td>
<td>0.113</td>
<td>0.027</td>
<td>0.037</td>
<td>7.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current</td>
<td>0.343</td>
<td>0.373</td>
<td>0.109</td>
<td>0.107</td>
<td>0.032</td>
<td>0.037</td>
<td>7.31</td>
</tr>
<tr>
<td>Bowling</td>
<td>U Gul</td>
<td>Career</td>
<td>0.400</td>
<td>0.322</td>
<td>0.098</td>
<td>0.117</td>
<td>0.022</td>
<td>0.041</td>
<td>6.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current</td>
<td>0.340</td>
<td>0.306</td>
<td>0.127</td>
<td>0.153</td>
<td>0.037</td>
<td>0.037</td>
<td>8.44</td>
</tr>
<tr>
<td></td>
<td>M Hafeez</td>
<td>Career</td>
<td>0.349</td>
<td>0.422</td>
<td>0.084</td>
<td>0.090</td>
<td>0.029</td>
<td>0.026</td>
<td>6.86</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current</td>
<td>0.351</td>
<td>0.422</td>
<td>0.081</td>
<td>0.094</td>
<td>0.029</td>
<td>0.023</td>
<td>6.78</td>
</tr>
</tbody>
</table>

Table 3.5: Comparison of career average and current form characteristics for selected players where the final column denotes the expected number of runs per over.
Figure 3.1: Scatterplot of $E(D)$ (current form) against the Reliance ICC rating for batsmen.

Figure 3.2: Scatterplot of $E(D)$ (current form) against the Reliance ICC rating for bowlers.
Figure 3.3: Scatterplot of $E(D)$ (current form) against the Reliance ICC rating for all-rounders.

Figure 3.4: Most recent IPL salary versus current form $E(D)$ for pure batsmen. Triangles represent 2012 salaries, plus signs (+) represent 2013 salaries, and cross signs represent 2014 salaries.
Figure 3.5: Most recent IPL salary versus current form $E(D)$ for bowlers. Triangles represent 2012 salaries, plus signs (+) represent 2013 salaries, and cross signs represent 2014 salaries.

Figure 3.6: Most recent IPL salary versus current form $E(D)$ for all-rounders. Triangles represent 2012 salaries, plus signs (+) represent 2013 salaries, and cross signs represent 2014 salaries.
Figure 3.7: Most recent IPL salary versus ICC Reliance rating for batsmen (top), bowlers (middle) and all-rounders (bottom).
Figure 3.8: Scatterplot of $\text{E}(D)$ using first and second innings data against $\text{E}(D)$ using only first innings data.

Figure 3.9: Scatterplot of Croucher’s (2000) statistic for pure batsmen versus $\text{E}(D)$. 
Chapter 4

Network Inference from a Link-Traced Sample using Approximate Bayesian Computation

In this chapter, we present a new inference method based on approximate Bayesian computation for estimating parameters governing an entire network based on link-traced samples of that network. To do this, we first take summary statistics from an observed link-traced network sample, such as a recruitment network of subjects in a hard-to-reach population. Then we assume prior distributions, such as multivariate uniform, for the distribution of some parameters governing the structure of the network and behaviour of its nodes. Then, we draw many independent and identically distributed values for these parameters. For each set of values, we simulate a population network, take a link-traced sample from that network, and find the summary statistics for that sample. The statistics from the sample, and the parameters that eventually led to that sample, are collectively treated as a single point. We take a Kernel Density estimate of the points from many simulations, and observe the density across the hyperplane coinciding with the statistic values of the originally observed sample. This density function is treat as a posterior estimate of the parameters of the network that provided the observed sample.

4.1 INTRODUCTION

Link-tracing samples (sometimes called respondent-driven samples or snowball samples in social science contexts), or adaptive web sampling (Thompson 2006) are used to sample hard-to-reach networked populations, such as endangered species, injection drug users, or other at-risk people. They are commonly used in the social sciences, and are a large source of non internet-based social network data.
In a link-tracing sample, part of the target population is selected, ideally by simple random sampling, but often by convenience or directed sampling. This set of initial contacts is sometimes called the ‘seeds’ of the sample. The variables of interest are measured from these seeds, and some mechanism is used to find additional subjects that are connected to at least one of the seeds in some predefined way. Common mechanisms are to ask seeds for contact information of their connections, and/or to give recruitment coupons to seeds that can be given to connected population members to entice these population members to join the sample and be measured as well.

The first wave of connected population members that are sampled are then asked for contacts and/or given coupons as the seeds were in order to bring a second wave of members into the sample. Sampling continues in this fashion until no new members of the population are recruited into the sample, or a predetermined sample size has been reached. Additional seeds may be selected if recruitment is exhausted before the sample size is reached.

As a proportion of published research using link-traced samples, there is relatively little focus on the network structure of the respondents. There are many barriers to inference of the network of a population from a link-traced sample. Stopping after a fixed sample size, non-response, non-recruitment, and selection of the initial seeds to recruit from all affect the sample network that is observed. For example, in some cases, connection information that leads to subjects already in the sample is discarded because it cannot bring in new respondents.

Use of link-traced samples in research is well established (Goodman 1961), including in the context of sampling from hard-to-find populations (Kaplan, Korf, and Sterk 1987), and with the respondents themselves determining recruitment (Heckathorn 1997). Advances in finding the network structure of the respondents is relatively recent. Crawford (2014) describes how partial information about a network structure can be inferred from a respondent-driven sample. Hancock, Gile, and Mar (2015) applies a respondent-driven sample to a population at risk of HIV infection, and infers features of the network including the number of nodes (i.e. the order of the network).

There are many barriers to inference of the network of a population from a link-traced sample. Stopping after a fixed sample size, non-response, non-recruitment, and selection of the initial seeds to recruit from all affect the sample network that is observed. For example, in some cases, connection information that leads to subjects already in the sample is discarded because it cannot bring in new respondents.

Inference of the specific structure of a population network may be infeasible from a snowball sample. However, it is possible to infer general features of such a network, such as the distribution of the number of connections between respondents, and parameters that dictate which connections form. This chapter describes a method, based on approximate Bayesian computation, for inferring such network parameters.
Section 4.2 of this chapter describes the approximate Bayesian computation (ABC) method in a general context. This section also contains a literature review of some recent advances in ABC that are closely related to the proposed method.

Section 4.3 includes a short glossary of network terms being used. It follows with some example network structure-based statistics established in previous work, and some sampling order-based statistics that are established specifically for this method.

Section 4.4 describes the method being proposed. This includes an expansion upon the description of ABC from Section 4.2 to apply it to a snowball sample. Also included are the steps taken to randomly generate the population network necessary for this method, the protocol used for taking a snowball sample, and the kernel density estimator used to estimate the probability density of the parameters given the sample's statistics.

Section 4.5 gives a demonstration of the proposed method on snowball samples taken from some populations with known parameters. In the demonstration, the ideal sample statistics are unknown, so two rounds of the network inference methods are performed. The initial round uses a small simulation run to determine which summary statistics of the sample would be useful, and to update the parameter priors. The final round uses a large simulation run and an updated prior to give a high-resolution estimate of the posterior of the parameters. Posterior means are taken from the final round estimate. Results for both rounds are given as topographic maps of the conditional probability density.

Section 4.6, the conclusion, describes how this method can be adapted to applications with additional parameterization. It also provides some priorities for future development of the method and its software to address wider application and computational challenges.

4.2 RECENT ADVANCES IN APPROXIMATE BAYESIAN COMPUTATION

Approximate Bayesian computation (ABC) is a Monte Carlo-based method of estimating the distribution of parameters from which inference would otherwise be intractable. It is used extensively in describing complex systems in evolutionary biology (Csilléry et al. 2010), non-linear regression models in statistical genetics (Blum and François 2010), and agent-based models in oncology (Sottoriva, and Tavaré 2010).

To perform a classic ABC on a sample, as described in Diggle (1984), first calculate statistics \( s' \) from the data, define a distance function \( \rho(s, s') \) such as Euclidean distance, and define a tolerance parameter \( \epsilon \). In a classic ABC, parameter values are randomly generated from a prior, and datasets are generated according to a given model and these parameters.

For each of the randomly generated datasets, the same statistics \( s \) are taken as those that were taken from the original data. If the distance between \( s \) and \( s' \), \( \rho(s, s') \) is greater
than $\epsilon$, the parameter set used to generate this dataset is rejected. Otherwise, the parameter set is accepted.

The sets of parameter values that are accepted are values that were used to produce simulated samples similar to the observed sample. We take this distribution (or some smoothing of it) to be the posterior distribution of parameter values.

Two common modifications to approximate Bayesian computation are partial rejection and adaptive parameter selections. When partial rejection is used, datasets are either accepted with probability or with weights, rather than accepted or rejected outright. The acceptance probability or weight is based on their statistics’ distance to the target statistics. The weights of the simulations used in Beaumont et al. (2002) are determined by an Epanechnikov kernel, with density $\frac{3}{4}(1 - d^2)$ for distances $0 \leq d \leq 1$, and density 0 otherwise. When adaptive parameter selection is used, the parameters that are used for each simulation are informed in some way by the results of the previous simulations, usually in order to produce simulations that have summary statistics close to the observed statistics. Parameters in the ABC implementation in Del Moral et al. (2012) are adaptively improved with sequential Monte Carlo, and the criterion for rejection is made stricter as more simulations are used.

The proposed method in this chapter employs partial rejection, but unlike the application of kernel density in Beaumont et al. (2002), the probability mass across a hyperplane is computed, rather than at a single point. Some possibilities for adaptive parameter selection are discussed in Section 4.6.

Inference on network structure is a new and sparsely explored avenue of research. Mukherjee and Speed (2008) describes a Monte Carlo Markov-chain method of inferring the existence of specific edges in directed network graphs, but only for graphs of order (i.e node count) $\approx 10$ and size (i.e edge count) $\approx 20$. Related methods that address larger networks, like those described and proposed in Küffner et al. (2012) produce summary statistics with large amounts of uncertainty and noise, as demonstrated in Petri et al. (2015).

Applications of approximate Bayesian computation to network data are also rare. Toni et al. (2009) briefly mentions the possibility of extending the ABC approach to dynamical systems therein to networks of chemical signals. The approach to find the size of a hidden population in Félix-Medina and Thompson (2004) uses a link-tracing design and a simulation of many networks, but does not employ ABC. Phillips et. al. (2013) use inference methods similar to ABC to model an HIV epidemic, but do not use the words ‘approximate’ or ‘Bayesian’ to describe their method.

The proposed method of this chapter is very similar to Fay et al. (2014) in that it uses ABC, an observed network, and an assumed network model to make inferences about the parameters behind the generation of that network. However, in Fay et al. (2014), the observed network is assumed to be generated directly from the given model and parameters. For the method proposed herein, only a sample of the generated network is observed, rather
than the entire network. Inferences are to be made on both the parameters that generated the population network as well as the parameters such as response-to-recruitment chance that dictate the sample network.

4.3 NETWORK TERMINOLOGY AND METRICS

4.3.1 Quick glossary of network terms

Terminology in network analysis has not yet been standardized, so we chose to adhere to the terms used in Kolaczyk (2009) and Kolaczyk and Csárdi (2014). That is, a network graph \( < V, E > \) is an abstract structure composed of a set of \( n \) nodes \( v_i \) for \( i = 1, \ldots, n \), and a set of edges \( E \). Each edge in \( E \) an ordered pair \((v_i, v_j)\), \( i,j \) members of \( 1, \ldots, n \), representing a one-way connection from \( v_i \) to \( v_j \). If \( i = j \), then the edge is a connection from \( v_i \) to itself and the edge is called a self-loop, or ‘loop’ for short.

The size of a network graph refers to the number of edges, contained in \( E \). The order of a network graph refers to the number of nodes in \( V \). The size and order will be referred to as \( N_E \) and \( N_V \) throughout this chapter.

If for a given edge \((v_i, v_j)\), there is an edge \((v_j, v_i)\), the connection between nodes \( v_i \) and \( v_j \) is considered to be bi-directional. If for all edges \((v_i, v_j)\), there exists an edge \((v_j, v_i)\), the network graph is called undirected; otherwise it is called a directed graph. If there exists more than one edge \((v_i, v_j)\) for any pair \( i \) and \( j \), the network graph is called a multigraph.

A subgraph \( < V^*, E^* > \) of \( < V, E > \) is a network graph in which \( V^* \) is a subset of \( V \), or \( E^* \) is a subset of \( E \), or both. For all edges \((v_i, v_j)\) in \( E^* \), at least one of \( v_i \) or \( v_j \) must belong to \( V^* \). If both \( v_i \) and \( v_j \) are in \( V^* \) for all edges in \( E^* \), and all such edges in \( E \) are in \( E^* \), then the subgraph is called an induced subgraph.

A path from nodes \( v_k \) to \( v_l \) is a set of edges \{ \((v_k, v_i), (v_i, v_m), \ldots, (v_n, v_j), (v_j, v_l)\) \}, \( i, j \) in \( 1 \ldots n \). This number of edges in the set is called the length, and it is possible for a path to be of length 1. If \( k = l \), then the path is called a cycle.

A connected component, or ‘component’ for short, is a subgraph of a network graph in which, for each node in the component, there is a path either to or from each node in the component. If there is path both to and from each node, the component is strongly connected. Every component in an undirected network graph is strongly connected, and in the context of undirected network graphs, components are simply referred to as connected. A connected component that does not contain any non-trivial cycles (cycles involving 3 or more nodes) is called a tree. Also, if every node from the network graph that could be in a component already is, that component is considered maximally connected.

The method described in this chapter pertains to populations described by undirected graphs that are not multigraphs and do not have self-loops. The population may contain cycles, but the samples do not. Furthermore, since most samples considered do not include
every node in the population, components of the network observed from the sample may not be maximally connected.

When the desired number of nodes have been sampled, any remaining recruitment links are ignored. This makes the snowball sample taken by this protocol a subgraph of an induced subgraph of the population network graph. That is, the nodes of the sample are a subset of the population nodes, and only links strictly between nodes with those subset are included in the subgraph. There are no links ‘leading out’ of the sample.

### 4.3.2 Limitations of structure-based network statistics of link-traced samples

Completely observed network graphs can be summarized using statistics that are based solely on the structure of the graph, such as the distribution of edges per node, the distribution of component sizes, and more complex measures based on cliques and centrality. Such is the case with the method proposed in (Fay et al. 2014). However, when sampling is introduced such as the link-tracing protocols that are used in our method, many of these statistics lose their inferential utility.

For the link-tracing sampling protocol of interest, only edges that lead to recruitment are retained in the dataset. This means that no connections from respondents to other respondents already in the sample are retained. Thus, no non-trivial cycles that are in the sample are observed. The observed data shows only a subtree of each sampled network component.

In cases where only a subtree or subtrees of the network graph are available, many common measures of network structure fail to be useful. Consider that the size (number of links) of a tree-shaped component is one less than the order (number of nodes). The network graphs observed from a subtree sample of \( n \) nodes can only have average degree on \([0, 2(n-1)/n]\). A graph with average degree 0 would be one with no links between any of the nodes. A graph with the maximal average degree of \( 2(n-1)/n \) would have all \( n \) nodes in a single giant component. To see this, consider the construction of a recruitment network with one component from an arbitrary starting point. To add a node to the network, we need to add an edge to an existing node to maintain the one-component property. Edges are only ever added when nodes are added, therefore the number of edges added is always equal to the number of nodes added. Now consider that a single node is a (trivial) recruitment network of one component. By induction, a recruitment network has one more node than edges. Since any edge contributes two degrees to the nodes, a one-component recruitment network of \( n \) nodes has \( 2(n-1) \) total degrees. Obviously, recruitment networks with more components and \( n \) nodes have fewer than \( n-1 \) edges.

More detailed methods of describing of network structure have similar problems. Consider a motif census. A motif is a small induced subgraph of larger network graph. The four possible non-isomorphic motif arrangements of order 3 and some of the 11 such order
4 arrangements are shown in Figure 4.1. A census of the order k motifs of an order $n_{samp}$ graph counts the number of motifs that appear in all $\binom{n_{samp}}{3}$ induced subgraphs of the order $n_{samp}$ graph.

Figure 4.1: Network motifs of order 3, and selected motifs of order 4

The counts from a motif census can be compared to the expected counts from networks of known random processes. The actual and expected motif counts can be compared with goodness-of-fit tests to infer if a given network could have been generated by a given process.

When the graph is a single tree, as it is in cases where average degree fails to be useful, only a thin range of motif distributions are possible. Only considering motifs in a vacuum, only 3 of 4 order 3 motifs, and 6 of 11 order 4 motifs are possible without creating a cycle.

In addition to the above issues, other problems relating to non-response can arise. For example, each sample unit reports its own degree within the population, which may be greater than the degree observed in the sample network for one of three reasons: A node never responds to a given recruitment attempt, sampling is completed before a potentially recruited node responds, or a node is already in the sample, and does not respond after the first recruitment. In the face of these difficulties from relying solely on network structure to inform our summary statistics of choice, we also incorporate sampling order.

### 4.3.3 Sampling-order based metrics

We assume that for any sample, the chronological order in which nodes are added in a sample is available. Sampling order could be obtained directly by incrementing identity number, or inferred from time stamps.

The sampling order of any node affects the reasons that its edges may fail to be included in a sample. Edges from a node that are added near the end of a sample are less likely, all else being equal, to be followed because the desired number of nodes will have been sampled before these edges are followed. Similarly, as sampling continues, an increasing number of
nodes are included, and, all else being equal again, a given link from a newly added node is more likely to lead to a node already in the sample.

The trend of increasing loss-to-redundancy is of special interest because it depends on the structure of the population network. Specifically, this trend occurs in samples taken from populations where there are more links than nodes, and is stronger in populations with higher link density, up to a given point. This is intuitive when one considers that in a sparsely connected graph, connected components will be small and will be explored completely in a relatively small part of a sample; leaps to new, unconnected units by simple random sampling are common, and each leap brings fresh connections. However, in network graphs with average degree between 2 and 6, a sample will typically stay in one component during its entire run, and such leaps to new components are not made. In densely-connected graphs where average degree is close to the order (i.e. where there are nearly $N_E = N_V(N_V - 1)/2$ edges, in a graph with $N_V$ nodes) of the graph, links are so abundant that recruitment sampling begins to resemble simple random sampling in that the most recently sampled node gives little information about the next node to be sampled.

We compute $Pr_{LinkUsed}$, the proportion of a unit’s reported links that are used for recruitment in a sample are defined by.

$$Pr_{LinkUsed} = \frac{\text{Recruitment links included in the sample}}{\text{Recruitment links reported}}.$$  

If the node was recruited into the sample by another node, then

$$Pr_{LinkUsed} = \frac{|\text{Edges leading to sample}| - 1}{|\text{Edges}| - 1}.$$  

If the node was selected from the population by simple random sampling, then

$$Pr_{LinkUsed} = \frac{|\text{Edges leading to sample}| - 1}{|\text{Edges}|}.$$  

We define the measure $\frac{\Delta used}{\Delta Sample}$ to be the rate of change of the proportion of potential links used for recruitment over the time the sample is taken. We compute $\frac{\Delta used}{\Delta Sample}$ as the slope-coefficient of the linear regression model of $Pr_{LinkUsed}$, weighted by degree and as a function of $t$, where $t = 0$ for the first unit sampled, and $t = 1$ for the last unit. In other terms, $\frac{\Delta used}{\Delta Sample}$ is the estimated difference in proportion of recruitment attempts that are successful from the beginning to the end of a sample. This interpretation of the time variable $t$ is just one of several viable ones. For example, $t$ could refer to the number of nodes explored in each nodes’ component, which would account for ‘leaps’ in the link-tracing design to new components, but potentially lend undue weight to results from large components. Similarly, a binomial generalized linear model with a logit link could be used to find the ratio of the log-odds of recruitment at the beginning or the end of sampling. However, results based on the log-odds ratio becomes unstable for samples
that use most or all of the nodes in the population because the chance of recruitment drops to near zero. A more philosophical limitation of \( \Delta_{\text{used}}/\Delta_{\text{Sample}} \) is its blindness to the reason a given edge was used for recruitment, so whatever it reveals could be a result of non-response or edges being exhausted, and it isn’t a panacea to the difficulties described in Section 4.2.

The panels of Figure 4.2 show an experimental example of how sample link density and how \( \Delta_{\text{used}}/\Delta_{\text{Sample}} \), respectively, change as the average responding degree in a population increases. The metric \( \Delta_{\text{used}}/\Delta_{\text{Sample}} \) changes rapidly as average population degree changes from 1 to 3 edges per node. At approximately 2.3 edges per node, the loess-smoothed average of \( \Delta_{\text{used}}/\Delta_{\text{Sample}} \) reaches its minimum of \(-0.4\), indicating that the recruitment chance at the end of a sample of one of these populations is 40 percentage points lower. This behaviour indicates that \( \Delta_{\text{used}}/\Delta_{\text{Sample}} \) is useful in making inferences about networked populations in situations where metrics that ignore sampling order are not useful.

![Figure 4.2: \( \Delta_{\text{used}}/\Delta_{\text{Sample}} \) statistic, as a function of the population average degree among 2500 generated networks in a simulation run](image)

Three other sampling order-based statistics are used in this study - all but one computed from the slope of a linear model of some node information. The statistic \( \Delta_{\text{degree}}/\Delta_{\text{sample}} \) is the rate that the average degree of sampled nodes changes throughout a sample. For a range of edge density, \( \Delta_{\text{degree}}/\Delta_{\text{sample}} \) is likely to be negative because nodes with high degree are more likely to be reached early in a sample.

In the simulation study in Section 4.5, an infection mechanic is introduced to show the utility of this network inference method. Consequently, the statistic \( \Delta_{\text{infect}}/\Delta_{\text{sample}} \) is
used. This statistic represents the difference in the proportion of nodes that are infected from the beginning to the end of the sample.

For more work on incorporating sampling order information into a sample using network structure see (Crawford 2014) and (Crawford, Wu, & Heimer 2015).

### 4.4 INFEERENCE WITH LINK-TRACED SAMPLES USING APPROXIMATE BAYESIAN COMPUTATION

We are interested in inferring useful parameters of the network, such as degree distribution, rather than the specific structure of the network. We estimate these parameters with a strategy that involves a prior specification step, iterations of population generation steps and sampling steps, and a kernel density estimation step.

#### 4.4.1 Overview

From a sample of interest, we compute a set of network statistics \( s \). We have a set of parameters about the population whose values we wish to infer from the observed sample. For those parameters, we define a joint prior distribution with support \( \Omega \).

For each of many simulations, we randomly generate a parameter set \( p^* \) from the specified prior, and generate a population according to that parameter set. From that population, and possibly further guided by the parameter set \( p^* \), we take a sample of the population by link-tracing. We compute summary statistics \( s^* \) from the sample.

After the simulations are completed and we have a collection of parameter sets \( p^* \) and their consummate statistic sets \( s^* \), we employ kernel density estimation over the Cartesian product of \( \Omega \) and \( S \), where \( S \) is the space of possible statistic sets. From that density function, we factor out the prior, and condition on the observed statistics \( s \) to obtain a model of the conditional density across \( \Omega \).

#### 4.4.2 Prior specification step

Parameters used in the simulation could determine the links per node, the response rates of recruits in general, or the response rates of important subsets of recruits. Other parameters determine the level of preferential attachment in link selection, or determine the propensity of nodes of form links with those similar to themselves. Similarity is defined here by relative proximity in some social or physical space, much like the network social space being inferred in Hoff et al. (2002).

These parameters are to be estimates from three components:

A set of statistics that describe the sample of samples being observed, \( s \). Let the set of statistics be \( s \) and let \( S \) be the space of possible \( s \) values.

Let \( \Omega \) be the Cartesian product of the assumed ranges \((P_{min1}, \ldots, P_{minNP}) \) to \((P_{max1}, \ldots, P_{maxNP}) \) respectively of the parameters.
A set of points in the Cartesian product of $S$ and $\Omega$, $<S, \Omega>$, where points $p^*$ in $\Omega$ are selected from a joint random probability distribution, and $s^*$ is the result of a link-traced sample from a population described by $p^*$.

Possible probability distributions for the values of $p^*$ include uniform, discrete uniform, and geometric and beta with location-scale transformations. Using non-uniform distributions requires a weighting adjustment, as described in the kernel density estimation step in Section 4.4.5.

Specification of the priors is a balancing task between result reliability and computational efficiency. If the global maximum of the density is outside of $\Omega$ or simply at the periphery of $\Omega$, then it may be missed. To avoid this, boundaries for $\Omega$ that include all plausible areas, such as $0 < p < 1$ for response-to-recruitment probability, and $0 < p < 8$ for average link density, may be used.

For each simulation, a parameter set $p^*$ is taken with probability according to the prior distribution and within $\Omega$, and a population is randomly generated, according to $p^*$. However, if $\Omega$ is very inclusive, then many generated parameter sets will lead to samples with statistics far removed from the observed statistics, and thus will be of little use in estimating the true parameter values. This problem can be mitigated by generating many populations. Section 4.6 proposes some strategies for improving the selection of $\Omega$.

4.4.3 Population generation step

To generate a population, first generate the set of parameter values $p^*$ according to the given prior, including $N_V$ and $N_E$, the number of nodes and edges in this population respectively. For each node, randomly assign pertinent characteristics according to $p^*$, such as the location of the node in the unit square of space, link propensity, and any application-specific variables that are desired.

To form an edge, select a node $v_i$, either by simple random sampling, or by a weighted sample if a distribution of edge propensity is selected. Given $v_i$, select node $v_j$ with probabilities proportional to $D^{-\gamma}$, where $D$ is a distance measure such as Euclidean distance in physical or social space, and $\gamma$ is a parameter in $p^*$ defining a level of tendency for nodes to form links with spatially close neighbours. The assignment of edges between nodes may be subject to application-specific variables, such as sex, sexual preference, the number of edges already attached to nodes, and dynamic markers like infection state.
4.4.4 Link-tracing step

We are given $n_{samp}$, the number of nodes to be sampled from the population by link-tracing. These are selected from the population according the following algorithm:

Step 1. If this is the first node, or the sampling queue has been exhausted, select a node by simple random sampling from those not yet selected. (After the first node this is known as a leap). Add the selected node, $X$, to the sample and skip to Step 4.

Step 2. If there are nodes in the queue, remove any nodes that already included in the sample. For each node $X$ that is removed from the queue for already being in the sample, look up $source(X)$, which is the node that led to $X$, and increment the count of redundant links for $source(X)$.

Step 3. If there are no remaining nodes in the queue, return to Step 1.

Step 4. Query for the list of nodes in the population that are connected to $X$. For each connected node $sink(X)$, mark the node as responding with probability $Pr(response)$ determined from $p^*$. Increment the number of links reported for node $X$.

Step 5. Append the nodes marked as responding to the end of the sampling queue, and increment the number of links responding to $X$ for each such node. If the number of nodes in the sample is now equal to $n_{amp}$ or the number of the nodes in the population, stop. Else, return to Step 1.

**Algorithm 1**: Example protocol used for sampling nodes by link-tracing

4.4.5 Kernel density estimation step

The values for given parameter for the set of simulations is i.i.d. realizations of some specified prior density $f(p)$. Each parameter is generated independently, so the joint prior distribution is the product of individual priors, $f(\bar{p}) = \prod_i^{NP} f(p_i)$, where $NP$ is the number of parameters.

Each simulation produces a link-traced sample of its population. Statistics $s_i, \ldots$ for each sample are taken, producing an $N_{runs} \times NS$ matrix. For each simulation, we have $NP$ and $NS$ parameter and statistic values, respectively. These values are collectively interpreted as a point in $\mathbb{R}^{(NP+NS)}$ space. The $N_{runs}$ spatial points are scaled linearly to fit into a unit hypercube $[0, 1]^{(NP+NS)}$. Specifically, let $x_i^*$ be the $i^{th}$ unscaled parameter or statistic value of $NP$ and $NS$, such values respectively, and let $x_i$ be the scaled value. For $i = 1, \ldots, NP$, use the assumed parameter value bounds to scale, such that

$$x_i = (x_i^* - P_{min_i})/(P_{max_i} - P_{min_i}).$$  \hspace{2cm} (4.1)
For \( i = NP + 1, \ldots, NP + NS \), use the uniform method-of-moments estimates to scale, that is
\[
x_i = (x^*_i - Q_{\text{min}i})/(Q_{\text{max}i} - Q_{\text{min}i}),
\]
where \( Q_{\text{min}i} = (\min(x^*_i) - (1/(N_{\text{runs}} + 1)) \ast (\max(x_i) - \min(x_i)) \), and \( Q_{\text{max}i} = (\max(x^*_i) + (1/(N_{\text{runs}} + 1)) \ast (\max(x_i) - \min(x_i)) \).

After rescaling, a posterior density is estimated by kernel-smoothing \( N_{\text{runs}} \) point masses with an independent multivariate Gaussian-kernel.

Posterior density \((x_1, \ldots, x_{NP}, x_{NP+1}, \ldots, x_{NP+NS}) =
\[
G(x,y) = (2\pi)^{(NP+NS)/2} \sum_{i=1}^{N_{\text{runs}}} \exp \left( \frac{\sum_{j=1}^{N_p+Ns} (y_j - x_{ij})^2}{\det(\Sigma)} \right),
\]
where \( \det(\Sigma) \) is the determinant of the variance-covariance matrix of the multivariate-normal, which is \( \prod_j^{NP+NS} \sigma_j \) by the independence constraint placed on the kernel. However, since it is the likelihood, not the posterior, of ultimate interest, consider the result of kernel-smoothing points with masses inversely proportional to their prior densities:
\[
H(x,y) = (2\pi)^{(NP+NS)/2} \sum_{i=1}^{N_{\text{runs}}} w_i \exp \left( \frac{\sum_{j=1}^{N_p+Ns} (y_j - x_{ij})^2}{\det(\Sigma)} \right),
\]
where \( w_i = 1/f(\bar{p}) \).

Finally, to obtain the likelihood information \( h(X|y) \), we condition \( H(X,Y) \) on \( y^* \), the vector of statistic values from the actual sample,
\[
h(X|Y \approx y^*) = h(x|y^*) = \frac{H(x,y^*)}{\int_{x \in X} H(x,y^*)dx}.
\]

\( h(X|y^*) \) is not a true conditional in that it incorporates information from all values of \( y \). However, controlling for prior density, point masses with \( y \) values near \( y^* \) weigh more heavily on the modeled distribution of \( X \). That is, the contribution of a simulated sample increases with its proximity of its statistics to \( y^* \). To plot the probability density over a range in order to observe the posterior maximum, mean, and the credible interval spatial smoothing methods such as kriging are employed.
4.5 SIMULATION STUDY

4.5.1 Overview

Consider the network graph shown in the two panels of Figure 4.3. This is a network of order 400 taken from a population of unknown size by a link-tracing design. Each point represents a node, and each line segment represents a undirected link. Both panels show the same sample. The left panel includes the links that were reported, but not used for recruitment. In the right panel, only links that led to a recruitment into the sample are included. Notice that in the left panel, some of the only have a node at one end; the missing nodes were not recruited and therefore not observed directly, so they do not appear in the figure.

The location of the points in Figure 4.3 represent the locations of each node (e.g. a person, animal, or computer) in a social or physical space. The shade and shape of each point represents an infection status; filled squares represent infected nodes and empty circles represent uninfected nodes. Many infection mechanics are possible, but for this study, we assume that a proportion of nodes have been infected by an outside source, and that this happened before any of the connections between nodes were formed. Let this initial infection proportion be $\phi$. After the initial infection step, we establish connections in a random order. For each connection made, a node $v_i$ is selected with equal probability to any other node, and node $v_i$ makes a connection with another node with probability proportional to $D^{-\gamma}$, where $D$ is the Euclidean distance between $v_i$ and the other node.

![Figure 4.3: Sample of order 400 from the population of interest, with all the edges found in link-tracing in the left panel, and with only the recruiting edges retained in the right panel](image)

We wish to infer five parameters:
1) The average degree of the population,
2) the number of nodes in the population, \( N_V \)
3) the proportion of nodes that initially have the ‘infected’ status, \( \phi \),
4) the transmission chance, \( \alpha \), and
5) the ‘nearness preference’ parameter, \( \gamma \).

We will assume for simplicity that selection into the sample, by either simple random selection or by link-tracing, never fails. In short a perfect response rate is assumed. We also assume the only factor determining edge formation is distance, and that there are no latent effects such as unequal edge propensity or preferential attachment based on degree or other node variable (e.g. sex, social status, known infected status).

Recall from Section 4.4 that to use the proposed method, a prior and a set of summary statistics are required. Having neither prior nor statistics given, we conduct two rounds of ABC Network Inference: one to determine a reasonable prior and set of statistics, and a more computationally expensive round using said prior and statistics.

In the initial round, we simulate 500 populations using parameter sets from a non-informative joint prior distribution. This initial set is used to inform the selection of statistics to condition on to obtain a function for the conditional probability density \( h(X|y^*) \). We then use kernel density estimation to compute the conditional density throughout points in the parameter support \( \Omega \) to decide if the prior needs to be changed to better find the global maximum, or if a large portion of the support isn’t simulating over.

In the final round, we proceed similarly, but simulate 2500 populations using an improved prior, and condition the posterior from the KDE on the statistics identified from the first round rather than going through a statistic selection process.

### 4.5.2 Initial Round

For the five parameters of interest, we use a joint prior of the product of population average degree \( \sim unif(0, 7) \), \( N_V \sim 200 + \text{geom}(\text{mean} = 1000) \), \( \alpha \sim unif(0, 0.5) \), \( \phi \sim unif(0, 0.3) \), and \( \gamma \sim unif(-2, 10) \).

Available are the following summary statistics from the sample:

1) Mean degree recruited = 1.995
2) mean degree reported (among all nodes) = 3.908
3) mean degree reported (among only infected nodes) = 4.104
4) mean degree recruited difference (infected - not infected) = 3.999
5) sample infection proportion = 0.560
6) \( \Delta \text{degree}/\Delta \text{sample} = -0.0798 \)
7) \( \Delta \text{depth}/\Delta \text{sample} = 0.602 \)
8) \( \Delta \text{used}/\Delta \text{sample} = -0.139 \)
9) \( \Delta \text{infect}/\Delta \text{sample} = -0.086 \)
The statistic $\Delta degree/\Delta sample$ is defined as the average rate of change in reported degree among sample units from the beginning of sampling to the end of sampling. As a simplified example, if the first few units sampled had mean degree 5, and the mean degree decreased linearly as the network were explored until the mean degree was 2 at the end of the sample, then $\Delta degree/\Delta sample \approx -3$ for that sample.

The statistic $\Delta depth/\Delta sample$ is defined as the average rate of change of depth from the beginning to the end of sampling. The depth of a node, in this instance, means the geodesic distance to any other connected node, averaged over those connected nodes. Well-connected and central nodes have low depth, and nodes on the periphery of a network have high depth. The hypothesis behind using this statistic was that in networks where the sample was a large proportion of the population that the low-depth nodes would be exhausted before the sample had completed, and that the average depth of the nodes being sampled would increase.

The statistics $\Delta used/\Delta sample$, and $\Delta infect/\Delta sample$ are computed similarly. The term ‘used’ refers to the proportion of a node’s connections that are used for recruitment. The term ‘infect’ refers to a binary variable of infected status. In the case of $\Delta infect/\Delta sample$, the log-odds ratio was used, instead of the linear regression slope.

We take a link-traced sample from each of 500 simulated populations guided by the parameter sets from this prior. We compute all the candidate summary statistics from each sample. For each of the five parameters $Y$ and the each of the nine candidate statistics $X$, we fit a cubic regression, $Y \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3$. Statistics that are strongly related to the parameters of interest are likely to fit a cubic regression better than a null model. The coefficients of determination, $R^2$, and the F-statistics for each model comparison are shown in Tables 4.1 and 4.2 respectively.

<table>
<thead>
<tr>
<th>Sample Statistic</th>
<th>Avg.Degree</th>
<th>NV</th>
<th>Init. Infection</th>
<th>Pr(Infect)</th>
<th>Closeness</th>
<th>Param</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Deg. Recruited</td>
<td>173.02</td>
<td>3.61</td>
<td>8.54</td>
<td>1.11</td>
<td>1.61</td>
<td></td>
</tr>
<tr>
<td>Mean Deg. Reported</td>
<td>6051.57</td>
<td>1.40</td>
<td>11.37</td>
<td>0.51</td>
<td>1.84</td>
<td></td>
</tr>
<tr>
<td>Mean Deg. Infected</td>
<td>4794.38</td>
<td>1.22</td>
<td>11.80</td>
<td>1.13</td>
<td>1.86</td>
<td></td>
</tr>
<tr>
<td>Mean Deg. Difference</td>
<td>22.87</td>
<td>1.27</td>
<td>12.27</td>
<td>13.89</td>
<td>3.00</td>
<td></td>
</tr>
<tr>
<td>Infection Prop.</td>
<td>16.31</td>
<td>0.27</td>
<td>11674.92</td>
<td>17.38</td>
<td>0.40</td>
<td></td>
</tr>
<tr>
<td>$\Delta degree/\Delta sample$</td>
<td>3.82</td>
<td>39.46</td>
<td>1.29</td>
<td>0.46</td>
<td>13.61</td>
<td></td>
</tr>
<tr>
<td>$\Delta depth/\Delta sample$</td>
<td>48.68</td>
<td>11.79</td>
<td>3.65</td>
<td>0.76</td>
<td>24.45</td>
<td></td>
</tr>
<tr>
<td>$\Delta used/\Delta sample$</td>
<td>186.58</td>
<td>8.28</td>
<td>9.67</td>
<td>1.00</td>
<td>120.54</td>
<td></td>
</tr>
<tr>
<td>$\Delta infect/\Delta sample$</td>
<td>2.91</td>
<td>2.05</td>
<td>9.77</td>
<td>3.18</td>
<td>2.50</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1: F-statistics of models of parameters of interest by summary statistics

Many different summary statistics can be used, but since each one increases the dimensionality of the kernel density estimation step, extraneous ones should be avoided.

In this simulation, average population degree is very well predicted from the average reported degree of respondents. Assuming that we have the number of edges of each node
Table 4.2: $R^2$ of models of parameters of interest by summary statistics

<table>
<thead>
<tr>
<th>Sample Statistic</th>
<th>Avg.Deg.</th>
<th>NV</th>
<th>Init. Infection</th>
<th>Pr(Infect)</th>
<th>Closeness Param</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Deg. Recruited</td>
<td>0.58</td>
<td>0.03</td>
<td>0.06</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>Mean Deg. Reported</td>
<td>0.98</td>
<td>0.01</td>
<td>0.08</td>
<td>0.00</td>
<td>0.01</td>
</tr>
<tr>
<td>Mean Deg. Infected</td>
<td>0.98</td>
<td>0.01</td>
<td>0.09</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>Mean Deg. Difference</td>
<td>0.16</td>
<td>0.01</td>
<td>0.09</td>
<td>0.10</td>
<td>0.02</td>
</tr>
<tr>
<td>Infection Prop.</td>
<td>0.12</td>
<td>0.00</td>
<td>0.99</td>
<td>0.12</td>
<td>0.00</td>
</tr>
<tr>
<td>$\Delta degree/\Delta sample$</td>
<td>0.03</td>
<td>0.24</td>
<td>0.01</td>
<td>0.00</td>
<td>0.10</td>
</tr>
<tr>
<td>$\Delta depth/\Delta sample$</td>
<td>0.28</td>
<td>0.09</td>
<td>0.03</td>
<td>0.01</td>
<td>0.16</td>
</tr>
<tr>
<td>$\Delta used/\Delta sample$</td>
<td>0.60</td>
<td>0.06</td>
<td>0.07</td>
<td>0.01</td>
<td>0.49</td>
</tr>
<tr>
<td>$\Delta infect/\Delta sample$</td>
<td>0.02</td>
<td>0.02</td>
<td>0.07</td>
<td>0.03</td>
<td>0.02</td>
</tr>
</tbody>
</table>

(even if we don’t know what the edges are), then estimating population average degree by approximate Bayesian computation is unnecessary. From the cubic regression model using sample average degree, the 99% prediction interval for population average degree is 2.772 to 4.290. The initial prevalence of the infection is predicted well from the sample infection prevalence, but still has some variation left to explain.

The parameter of closeness preference, $\gamma$, is partially predicted by $\Delta used/\Delta sample$. Figure 4.4 is a scatterplot of values from the 500 simulations showing the interaction between $\gamma$, $\Delta used/\Delta sample$, and the average degree of the population. For networks with an average of 2 or more edges attached to each node, there is an increasing drop-off in reported edges that lead to recruitment. When the closeness preference is large ($6 \leq \gamma \leq 10$), the change in proportion of links used is small ($0 \leq \Delta used/\Delta sample \leq 0.2$).

For networks with average degree between 1 and 2, the relationship is less pronounced. For networks with average degree less than 1, no relationship between $\gamma$ and $\Delta used/\Delta sample$ is apparent. This is unsurprising as the samples from these networks are much more likely than their denser counterparts to fully explore a network component and move on to another.

Population size is only marginally predicted by one of the chosen summary statistics - the rate that node degree decreases through a sample, $\Delta degree/\Delta sample$. Figure 4.5 shows how $\Delta depth/\Delta sample$ decreases dramatically when the sample of the network includes all or most of the nodes in the population. The dashed line is at 400 nodes; any points below the line represent samples that include all the nodes in the population. Any interaction involving the average degree in the population is tenuous at best.

None of the summary statistics show a substantial relationship with the probability of infection, $\phi$. The statistic with the strongest relationship is the observed infection proportion, which shows a relationship with $\phi$ in part as an artifact of the prior. The statistic with the second strongest relationship to infection probability, infection-degree differential, has no obvious pattern.
Figure 4.4: $\Delta used/\Delta sample$ statistic, closeness preference parameter $\gamma$ and population average degree among 500 randomly generated networks in an initial simulation run.

For the four parameters with no singularly determining factor, there are four potentially useful summary statistics: the infection-degree difference, the sample infection prevalence, $\Delta degree/\Delta sample$, and $\Delta used/\Delta sample$; thus we have eight dimensions along which to do kernel density estimation. The six panels of Figure 4.6 show spatially-smoothed two-dimensional cross-sections of the results from the KDE step, conditioning on the observed summary statistics. They show large regions of the space of population order and initial infection prevalence with almost no posterior density. Also, the boundaries of the support have a non-trivial share of the density. Informed by the initial kernel density estimation and linear predictions, the prior is updated to better include possible parameter values and to reduce the proportion of non-productive simulations.
4.5.3 Final Round

Using the information from the first round we now use a joint prior of the product population average degree $\sim \text{unif}(2.772, 4.290)$, $N_V \sim 200 + \text{geom}(\text{mean} = 1000)$, $\alpha \sim \text{unif}(0.2, 0.6)$, $\phi \sim \text{unif}(0, 0.4)$, and $\gamma \sim \text{unif}(2, 12)$.

In this round take a link-traced sample from each of 2500 simulated populations, normalize, and use kernel density estimation on the joint space of $\Omega_1$, which is $\Omega$ with updated bounds a dimension for population average degree included, and $S_1$, which is $S$ with updated bounds. The panels in Figure 4.7 show the two-dimensional cross-sections of the conditional density from the improved estimates. The centroid of the conditional probability and the real parameter values are labelled as triangles and squares in each panel, respectively. The marginal posterior means, as well as the true values of the parameters are listed in Table 4.3.
Figure 4.6: Initial round weighted-posterior density of infective probability, number of population nodes, initial infection proportion, and preference for closeness in forming connections, shown as two-dimensional marginals.
Figure 4.7: Final 1 round weighted-posterior density of infective probability, number of population nodes, initial infection proportion, and preference for closeness in forming connections, shown as two-dimensional marginals.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>True</th>
<th>Mean (Initial Round)</th>
<th>Mean (Final Round)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>1412</td>
<td>2049</td>
<td>2094</td>
</tr>
<tr>
<td>Initial Infection $\alpha$</td>
<td>0.467</td>
<td>0.399</td>
<td>0.374</td>
</tr>
<tr>
<td>Transmission Chance $\phi$</td>
<td>0.093</td>
<td>0.175</td>
<td>0.201</td>
</tr>
<tr>
<td>Nearness Preference $\gamma$</td>
<td>7.298</td>
<td>6.753</td>
<td>9.352</td>
</tr>
</tbody>
</table>

Table 4.3: True parameter values and weighted-posterior mean after two simulation runs

### 4.6 DISCUSSION

The proposed method applies the extreme flexibility of approximate Bayesian computation to provide answers to otherwise intractible network inference problems. It is the first attempt, to the author’s knowledge, at estimating parameters that govern the generation of a network by using a sample of the network. Furthermore, it is the first known attempt to apply a kernel density estimation method to estimate a parameter distribution for an approximate Bayesian computation.

The network simulator that was created for this study is adaptable to a wide set of possibilities, including the law citation networks simulated in Chapter 5. The output of the simulation program is a detailed data set of each sample node in the population. It provides, in long format for each node, the sampling order, location in social space, the ID of the node, if any, that recruited this one, a number of connections in total, responding connections, and outward connections that were eventually recruited into the sample.

Each line in the output also has application-specific parameters which may be used in the simulation. Consider a case in which the network of interest is a sexual network, such as the one observed in Colorado Springs, USA (Potterat et al. 2002) or on Likoma Island, Malawi (Helleringer and Kohler 2007). For cases like this, each node has a sex and a sexual preference. Rather than a single response-to-recruitment probability across the whole population, two rates - one for heterosexual and one for homosexual connections, are generated. Also the connection propensity assigned to each node can be interpreted as a level of sexual promiscuity. The distribution of connection propensity can be made a mixture distribution to account for the subpopulations of sex workers or clients.

As mentioned in Section 4.4, simulations ideally produce statistics close to the observed sample’s statistics. Immediate future work will focus on methods of parameter value generation that optimize the value of a given simulation.

Simulations from a small pilot run across an inclusive parameter space, such as the one used in Section 4.4, could be combined with a larger main run by weighting simulations by inverse-density. Rather than limiting the parameter space adjustment to a single, manual step, an adaptive system which changes the distributions that parameters are drawn from could be implemented. Such a system could make cursory estimates of the conditional density and change the shape parameters of the generating distribution to drive future simulations towards the regions of maximal density. This system could also change the
scale and location parameters of the generating distribution if a region of non-negligible density was found at the extremes of the parameter space, which may indicate that the global maximum density is not being included in the parameter space yet.

Other avenues of development include using parameter dependency, and computational parallelization. Dependency structures such as a copulas could also be introduced, although the additional value may not be worth the added complexity and effort. Parallelization, however, shows immediate potential because the bulk of the computational cost comes from repeating a complex simulation many times over. Parallelization of ABC has gone as far as using graphics processing hardware by Liepe et al. (2010), which allows a single desktop computer to run thousands of operations at once. This could be further extended with adaptive rounds by allowing each graphics processing unit (GPU) to communicate interim results to the more general-purpose central processing unit (CPU) acting as a ‘queen’ node.

The proposed method, and the software developed to explore it, have addressed only a handful of protocols of respondent-driven sampling. These protocols have fixed sample sizes, and only edges that lead to recruitment are retained. However, with greater protocol flexibility, practitioners could conduct what-if analyses. That is, before any real sampling is performed, practitioners could estimate from assumed statistics and observed data how the conditional density function and the parameter estimates from it differ under different protocols. For example, one could see on a spatial graph of the conditional density, how much additional uncertainty is introduced by not including edges that leap back into the sample already taken.

Another limitation to be addressed in future work is the sequential nature of the sampling protocol in the code. A ‘seed’ member of the population is only selected by simple random sampling when all sampling-by-recruitment is exhausted. In a practical setting, a wave of multiple subjects is selected by simple random sampling before any recruitment links are followed; additional waves, rather than single subjects, are selected when recruitment links fail to bring in the requisite sample size. The size of these initial cohorts may impact the network graph that arises from the sample, as well as the sampling order. This could be especially problematic if some members of the initial population share a network component and recruitment paths interfere with other.

Uncertainty relating to sampling order is currently addressed in part by timing options in the simulator; users can choose to have recruitment links followed according to a breadth-first search algorithm, or in a fashion where each recruitment link is assigned a random response delay and the links are explored in time-order.

There are some mechanical limitations that need to be addressed before development of the software for this method continues much further. Foremost among these limitations, the network population is stored in a sparse vector-and-matrix system. The edges of a network graph of order \( N \) requires \( O(N^2) \) units of memory to store. In relation simulation work in Thompson (2013), network information is stored as a linked list of node objects. For a
maximum number of links per node $M$, storage of the same network graph’s edges requires $O(MN)$ memory, which is effectively $O(N)$ because $M$ is small compared to $N$ for sparse networks. A linked-list system, although more challenging to program initially, also allows for dynamic network graphs in which nodes undergo birth and death processes.
Chapter 5

Inference on the CanLII Database using Approximate Bayesian Computation

In this chapter, we demonstrate the application potential of the method described in Chapter 4 to a naturally occurring, rather than a synthetic, dataset. The dataset examined in this chapter is a collection of cases and citations of Canadian laws, centering around cases involving the Supreme Court of Canada (SCC). In many cases and other legal decisions, previous cases are cited as precedents to provide context to the situation being examined, and to ensure consistency.

By treating each case as a node, and each citation as a directed edge from the citing case to the cited case, it becomes apparent that this collection is a network. This network is fully dynamic in time - new cases occurring represent a birth process of nodes, and the citations that new cases make to existing work represent a birth process of edges. We are interested in a small number of parameters governing which cases receive many citations and which receive few. Since the number of interesting parameters is small, and the dataset is a large, complex network, with complications such as a birth process, the extreme flexibility of Approximate Bayesian Computation makes it uniquely well suited to this problem.

5.1 MOTIVATION

We wish to do some preliminary work to identify 'dead' laws, which have not been used in a very long time. The Canadian Law Information Institute (CanLII) database contains more than 1 million publicly available documents of decisions from the Supreme Court of Canada and of each Canadian Province and Territory. It also includes similar documents from lower courts like the Federal Court of Canada, specific focus courts like (xx), and legislation and regulations from Parliament from each legislative assembly. Legislation is available as a set of aggregated documents, but old versions of the aggregate aren’t available.
The database includes all decisions from the Supreme Court of Canada’s (SCC) decisions, which amounts to more than 10,600 documents over 140 years. The database also includes all recent (i.e. the last 10-20 years) documents of lower courts, and some of the earlier documents. Years with incomplete coverage for a given court are identified in the database. For the sake of feasibility, our sample is the text of the SCC decisions, and the list of documents that cite each of these decisions.

The CanLII database represent a sample of the network of public legal documents in Canada. In this network, documents are represented by nodes and citations are represented as directed edges. In our sample of the network (the 10,600 Supreme Court decisions), each document only cites older documents and is cited by future documents, therefore the sample network is acyclic. This acyclic nature may not necessarily be the case for a wider range of documents, if multiple revisions of the document are considered a single node.

Most documents cite, and are cited by, other documents. For clarity, we refer to the previous documents that a given document cites as its 'parents', and any future documents that cite the document in question as its 'children'. If a document’s parent appears in the CanLII database, a web hyperlink appears in the document to that parent. Likewise, if a document’s child appears in the database, then a reference to that child appears in the document’s ‘cited by’ list. A document may have parents or children that do not appear in the database. Out-of-database parents can be uniquely identified by scanning the text of a document for citations, which allows us to know the number of parents a document has and makes identification of co-citations possible, but not guaranteed. There is no information available about the identity or number of out-of-database children, however we can infer that these missing children are limited to periods and courts of partial or no database coverage.

Some pertinent features of the network appear in Figure 5.1. First, no citation arrows are shown, because citations are always from one case to a previous case. Second, multiple observed citations may lead to the same out-of-database document. By design, citations include identifying information, allowing us to match up documents by co-citation even without a complete database. Citations sometimes include a year, allowing us to directly observe the distribution of age of cited laws. Finally, missingness becomes more prevalent with age; in reality this coincides closely with the start of digital record-keeping by most courts in the 1990’s.

The citation network of laws has several dynamic elements. It includes a birth process of new documents being written and a death process of them being repealed or expiring by design. The topology between nodes may also change when a document is revised and new citations are added. A document’s revision may also affect the probability that the document is cited in the future as well, so revisions can change the underlying model even without changing the topology.
The method of inference proposed in Chapter 4 is flexible enough to account for both the dynamic elements of legal documentation, as well as the sampling structure of the CanLII database.

5.2 METHOD

We wish to estimate a confidence region for parameters that determine the relative attractiveness of cases to receiving citations.

We do this by creating many simulations of the Supreme Court of Canada (SCC) cases and the citations they receive using the competitive attractiveness model. Each simulation will use randomly generated parameter values from a multivariate uniform prior. Summary statistics will be taken from the results of each simulation. The values of the $N$ parameters of interest and the $M$ statistics from each simulation will be used to create a point in $\mathbb{R}^{N+M}$ space.

A Kernel Density Estimator (KDE) is used to create a density function in this space. We take the summary statistics from the actual set of SCC cases investigate the density along the $N$ dimensional hyperplane that matches the observed statistics.

Using the Canadian Law Information Institute’s (CanLII) database, we collected 10623 documents produced by the Supreme Court of Canada (SCC). These documents span from the beginning of the SCC in 1867 to August 31st, 2015. The text processor we designed using the stringr package in R identified 9847 of these documents to be cases.

A case is defined to be a dispute between two parties. One of these parties may be a surrogate for the public, such as “The Queen”, “The King”, or simply “R.”. One or both parties may be a corporation, as identified by the inclusion of “Company”, “Co.”, “Ltd.”, or “Inc.” in the name of the party.

In some cases, the opinions of the members of the Supreme Court were not unanimous. In such cases, the judges that were in the minority are listed in a special ‘dissenting’ section of the case document.

Most cases include citations to previous cases as precedents. These citations are identifiable in most or all situations thanks to strict citation naming conventions. However, not every cited case has been recorded digitally or made available by CanLII. Citations to available cases include hyperlinks, making it possible to explore part of the network backwards in time.

Each case also includes a ‘cited by’ list of the available documents that are known this case. Hyperlinks are included to available reverse citation, but un-digitized work is not listed in these ‘reverse citations.

Figure 5.1 shows a sample, specifically an induced subgraph of the citations between Supreme Court of Canada cases. The vertical axis represents time; recent cases are at the top, and early cases are at the bottom. There are 1500 cases represented in this figure,
which is less than 15 percent of all SCC cases. As such, the density of the citations between SCC cases is understated, as only links between cases in the figure are shown. If all 10623 such cases were represented, we would expect approximately 50 times as many lines in the figure.

Figure 5.1: Induced subgraph of the citations between 1500 Supreme Court of Canada cases. Each point represents a case. The size of the point (above a minimum) represents the number of cases from all sources that cite this case. The vertical position of each point represents the year of that case (i.e. the ‘birth’ time). Lines between points represent a citation from the newer case to the older one. The horizontal position of points is arbitrary, and the points have been arranged to minimize the number of crossing lines while preserving vertical position.

For the sake of computation speed, time is quantized into steps of 5 years. Table 5.1 shows the number of Supreme Court cases in each time step, as well as the number of digitized documents citing these cases. The table also includes the proportion of these cases that involve at least one corporation, a dissenting judge, or the crown respectively.
<table>
<thead>
<tr>
<th>Period</th>
<th>Cases</th>
<th>Cites Received</th>
<th>Corporate</th>
<th>Crown</th>
<th>Dissent</th>
</tr>
</thead>
<tbody>
<tr>
<td>1950-4</td>
<td>220</td>
<td>1185</td>
<td>0.32</td>
<td>0.25</td>
<td>0.52</td>
</tr>
<tr>
<td>1955-9</td>
<td>287</td>
<td>1087</td>
<td>0.40</td>
<td>0.17</td>
<td>0.39</td>
</tr>
<tr>
<td>1960-4</td>
<td>384</td>
<td>1363</td>
<td>0.44</td>
<td>0.19</td>
<td>0.31</td>
</tr>
<tr>
<td>1965-9</td>
<td>398</td>
<td>1675</td>
<td>0.43</td>
<td>0.21</td>
<td>0.30</td>
</tr>
<tr>
<td>1970-4</td>
<td>425</td>
<td>2971</td>
<td>0.39</td>
<td>0.19</td>
<td>0.35</td>
</tr>
<tr>
<td>1975-9</td>
<td>510</td>
<td>4482</td>
<td>0.36</td>
<td>0.31</td>
<td>0.33</td>
</tr>
<tr>
<td>1980-4</td>
<td>590</td>
<td>5672</td>
<td>0.29</td>
<td>0.38</td>
<td>0.20</td>
</tr>
<tr>
<td>1985-9</td>
<td>462</td>
<td>6870</td>
<td>0.24</td>
<td>0.49</td>
<td>0.27</td>
</tr>
<tr>
<td>1990-4</td>
<td>536</td>
<td>9433</td>
<td>0.16</td>
<td>0.53</td>
<td>0.33</td>
</tr>
<tr>
<td>1995-9</td>
<td>564</td>
<td>12373</td>
<td>0.24</td>
<td>0.52</td>
<td>0.36</td>
</tr>
<tr>
<td>2000-4</td>
<td>460</td>
<td>18445</td>
<td>0.27</td>
<td>0.37</td>
<td>0.31</td>
</tr>
<tr>
<td>2005-9</td>
<td>416</td>
<td>21076</td>
<td>0.29</td>
<td>0.38</td>
<td>0.32</td>
</tr>
<tr>
<td>2010-4</td>
<td>313</td>
<td>24947</td>
<td>0.26</td>
<td>0.44</td>
<td>0.26</td>
</tr>
</tbody>
</table>

Table 5.1: Summary statistic values for Supreme Court of Canada cases.

Bell (1975) explains the proportion of total sales of a product, or market share, by assigning an attractiveness value $a(s_i)$ to each vendor $i$. In this model, each buyer individually chooses a vendor with probability proportional to this attractiveness. We adapt this attractiveness framework by interpreting each a citation as a purchase, and each case that could be cited as a vendor. According to Table XX, in the 1970-1974 time step, for instance, there were 2971 citations received by SCC cases. The 2971 citations from this time step can refer to any cases, including those that happened long before the year 1970.

In a given time step, the citation attractiveness of case $i$ is modeled as

$$a(s_i) = \exp [\beta_{\text{irrel}} x_{\text{irrel},i} + \beta_{\text{pa}} x_{\text{pa},i} + \beta_{\text{corp}} x_{\text{corp},i} + \beta_{\text{crown}} x_{\text{crown},i} + \beta_{\text{dis}} x_{\text{dis},i}] \quad (5.1)$$

where $x_{\text{irrel},i}$ is the number of time steps since case $i$ was either created or cited, where $x_{\text{corp},i}$, $x_{\text{crown},i}$, and $x_{\text{dis},i}$ are indicator functions of whether case $i$ involved a corporation, the crown, or dissent between judges respectively, and where $x_{\text{pa}}$ is the square root of the number of citations that case received in the previous time step.

The parameter $\beta_{\text{irrel}}$ represents the tendency of a case to fall into irrelevance over time. The parameter $\beta_{\text{pa}}$ represents the tendency towards preferential attachment, whereby a case that receives many citations is likely to be seminal or famous, and therefore is used as a reference more often.

In each of many simulations, data is generated in two ways. Cases and citations before 1950 are generated deterministically from the observed data. For each such case, $x_{\text{irrel},i}$ is computed from the citation history of each case up to 1950, and $x_{\text{corp},i}$, $x_{\text{crown},i}$, and $x_{\text{dis},i}$ are taken directly from observation. Cases and citations happening from 1950 onwards are randomly generated.
Each simulation runs through 13 time steps consisting of the years 1950-4, 1955-9, ..., 2010-4. In each time step, cases are created, then citations are assigned, then \( x_{irrel,i} \) and \( x_{irrel,i} \) are updated accordingly. In each simulation, the number of cases created and number of citations assigned in each time step match the true number of Supreme Court cases and citations for that time step. For these cases, \( x_{irrel,i} = x_{pa,i} = 0 \) upon inception, and \( x_{corp,i} \), \( x_{crown,i} \), and \( x_{dis,i} \) are independently set to 1 with probabilities equal to relevant proportions found in Table 5.1.

After case creation, each citation is assigned to an existing case \( i \), including those that were created this time step, with probability \( a(s_i)/\sum_i a(s_i) \) Finally, \( x_{irrel,i} \) and \( x_{pa,i} \) are updated according to the number of citations they received.

We have selected five statistics to summarize the simulation results. The includes the standard deviation of the distribution of values \( \sum_t C_{t,i} \), the total number of citations, where \( C_{t,i} \) is the number of citations that case \( i \) receives in time step \( t \).

We use the estimated probability of ‘going cold’. That is, the chance in any given time step that a citation will receive no new citations, given that it received as least one citation in the time step immediately previous. This is computed by

\[
p_{cold,i} = \frac{\sum_{t>1} 1(C_{t-1,i} \geq 1)1(C_{t,i} = 0)}{1(C_{t-1,i} \geq 1)} \tag{5.2}
\]

The other three statistics we use are the estimates of \( \gamma_{corp,i}, \gamma_{crown,i}, \gamma_{dis,i} \) in the Poisson family generalized linear model,

\[
\log(\sum_i C_{t,i}) = \gamma_{corp,i}x_{corp,i} + \gamma_{crown,i}x_{corp,i} + \gamma_{dis,i}x_{dis,i} + \epsilon, \epsilon \sim Gaussian. \tag{5.3}
\]

These summary statistics are also calculated for the actual set of cases and their citation history. Table 5.2 has the statistic values, before normalization, of the actual cases, as well as the mean and standard deviation for 2000 simulated cases. The prior for the parameters used was the product of independent uniforms \( \beta_{irrel} \sim Unif(-0.5, 0), \beta_{corp} \sim Unif(-2.0, 3.0), \beta_{crown} \sim Unif(-0.5, 5.0), \beta_{dis} \sim Unif(0.5, 2.0) \)

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Real Data Value</th>
<th>Mean(SD) of values from simulations</th>
</tr>
</thead>
<tbody>
<tr>
<td>SD(Total Citations)</td>
<td>16.6</td>
<td>65.3 (49.2)</td>
</tr>
<tr>
<td>“Gone Cold” probability</td>
<td>0.743</td>
<td>0.440 (0.091)</td>
</tr>
<tr>
<td>( \gamma_{corp,i} )</td>
<td>-0.181</td>
<td>-0.101 (0.235)</td>
</tr>
<tr>
<td>( \gamma_{crown,i} )</td>
<td>0.553</td>
<td>0.288 (0.336)</td>
</tr>
<tr>
<td>( \gamma_{dis,i} )</td>
<td>0.373</td>
<td>-0.131 (0.305)</td>
</tr>
</tbody>
</table>

Table 5.2: Summary statistic values for Supreme Court of Canada cases, actual and simulated.
These summary statistics are compared to the statistics of the actual dataset of Supreme Court decisions, a weighted Euclidean distance will be computed between each simulation and the real data, and a weight assigned to each simulation’s parameter values based on this distance and on the multivariate normal joint probability distribution. The weighted mean of the simulated parameter values is the Approximate Bayesian Computation estimate of those parameters.

5.3 DISCUSSION

In future analysis of the law database, we wish to establish criteria for predicting when a legal document is unlikely to be cited ever again. We will do this by modelling the probability that a document will be cited within the next time step (e.g. five years) as a function of the document’s intrinsic and network properties.

There are other intrinsic properties year of document, word length of document, whether the document has been revised from its original version, and indication (by key words) that a document involves a criminal offense. Network properties include the presence of a ‘child’ citation during the previous time step, the presence of a ‘child’ citation in any of the most recent 2, 3, or 4 time steps, the presence of a previous citation in a Supreme Court case.

The model given here is simple, but is ripe for expansion. It treats the documents making the citations as an unconnected pool of identical cases. Citations are obviously made to cases that are relevant to the citing case. A multiplicative factor could be applied to the attractiveness of cases based on the similarity between the citing document and case potentially being cited.

Similarity could be further established with the key terms attached to most cases. These key terms span from wide categories like “Criminal law”, “Constitutional law”, and “Contract” to terms used only once like “Operation of tramway” and “Automatic sprinklers”. Employing Approximate Bayesian Computation to a model incorporating key terms could be extremely powerful, but would also involve a realistic simulation of the key terms generated in cases, as well as a text processor that accounted to English and French terms and linguistic changes.

Furthermore, one case may supersede or effectively overwrite another earlier case. Much of the attractiveness of the old case should be transferred to the new case this situation, and this phenomenon could be detected with further network analysis.

Better text mining will be able to better characterize the contents of the document for better prediction. For example, family disputes are not always between family members with the same last name. Cases between parties of very common surnames like Singh and Nguyen can be detected as being family disputes, even when they are not. We were unable to use family disputes as an indicator function for cases because fewer than 2 percent of cases are between parties with the same last name.

74
Future network analyses may incorporate a wider array of courts and help identify laws with a high probability of irrelevance, and flag them for review and potential repeal. This has civil liberty implications, as well as being a source of political capital for individuals and groups looking to fine easy causes to rally behind and gain popularity.

Alternatively, this work could identify documents that have high citation potential and flag them as seminal for future law students and researchers. Combining a deeper text analysis with the results of the inference method demonstrated in this chapter may allow for identification of documents with long-term impact from the text alone.

Finally, new cases are frequently examined by the Supreme Court of Canada, and the set of 4 million legal documents in the CanLII database is ever-growing. Even the same analysis could be performed at different times and produce new and interesting results.
Figure 5.2: Probability density maps of $\beta_{\text{corp}}$, $\beta_{\text{crown}}$, $\beta_{\text{dis}}$, and $\beta_{\text{pa}}$.
Chapter 6

Discussion and Future Work

6.1 Cricket Analytics

Unlike North American sports in which each sport has an undisputed pinnacle league and format, such as the NHL for Ice Hockey and the NBA for basketball, professional cricket is played in three different formats and in multiple leagues. Professional players are not dedicated to a single league or format - any given player could represent their country in a One Day International match and represent a city in a Twenty20 Indian Premier League match within weeks of each other.

To fully characterize a player’s performance, all such matches should be taken into account. However, because of possible differences in the strength of the opponents between leagues, and because of game length differences between formats, information is only considered for one league at a time. This separation is true for the characterization methods developed and discussed throughout this dissertation, and also for established player ranking systems like Reliance ICC (International Cricket Council).

Ignoring cross-league data has caused sparsity problems and unrealistic estimates in cases like the 2014 World Cup of Cricket, in which players that were well-established in Test and One-Day matches were playing their first professional Twenty20 matches during the World Cup. In the immediate future, I intend to explore first cross-league performance, and later cross-format performance, by expanding the event-weighting system to consider not only overs and wickets remaining, but format and league as well.

The strategic aspects of the second Innings in Cricket are also under-explored. In Twenty20 and One-Day formats, the goal of the first innings is simply to maximize the number of runs scored to produce a difficult-to-attain target for the opponent. In the second innings, the goal is to score more runs than the target, which is subtly different from simply scoring the greatest possible number of runs.

For example, if the batting team in the second innings observes that it is currently on track to score fewer runs than the target, it can change to a riskier batting strategy, effectively sacrificing a high expected number of runs for a greater chance at attaining the
target. Previously Harsha Perera, Tim Swartz, and I have explored simulations using the Duckworth-Lewis table of resources, where ‘resource’ refers to an composite function of overs and wickets remaining to determine when a team should or will take an aggressive strategy, but more work is needed. As with all good research, working with cricket analytics has been a figurative hydra - solving problems has only led to more problems - and literature.

### 6.2 Network Inference

Several mechanical improvements to the approximate Bayesian computation - based method of estimating network parameters in Section 4.6, but little was said about dynamic network possibilities. Future work in the near-and-mid term will include adding the forming and breaking of edges between nodes over time, the drift of nodes in space, and the time-based spreading of infection. For other possibilities, see the network simulation work done in Thompson (2013) which will continue to guide these advancements.

Future work on ABC applications to networks will focus on two related areas: application and distribution.

Before wider dissemination of the ABCN analysis programs, ease of use and speed need to be greatly improved. Currently, simulation is done in a C program and the output, a large csv file, is fed into an R script to analyze results. Ideally, this would all be done as a seamless function, such as one that calls a C-code simulation from R with the help of a dynamic link library (i.e a DLL file).

Computationally, there are two bottlenecks: Simulation, and Kernel Density Estimation. The cost of simulation changes between applications, but every intended application involves a network, and there are common techniques that can be used to improve the speed and memory footprint of networks. These techniques are mentioned in Thompson (2013) and are still under development. The speed of kernel density estimation can be improved by building the analysis around the method outlined in O’Brien et al. (2016), and programmed in the fastKDE package for R. The memory cost of fastKDE is similar to that of traditional kernel density estimation, that is to say, exponentially increasing with the number of distinct summary statistics.

Real network disease datasets like that of Helleringer (2007) are difficult to acquire because of the cost of sampling, and because of concerns for subject privacy. Having a portable software package that is usable by health researchers will allow deeper analyses of these rare and valuable datasets without having to release confidential information beyond its intended range.

The application of our approximate Bayesian computation-based methods in citation analysis have only started. In the example in Chapter 5, we only estimated the relative importance of a few features of Supreme Court of Canada cases in determining the ability of these cases to attract citations. Although we considered citations from lesser courts, we did
not investigate the citations that these cases from other courts received. We also treated citations as coming from identical cases, when relevance is highly important. In future work, we intend to examine, for example, the attractiveness of corporation-involved cases to future corporation-involved cases. To do so would increase the dimensionality of the kernel density estimation problem, which continues to have the memory cost issue mentioned previously.
Bibliography


Appendix A

Text Processing of Cricket Data

The primary content of the raw data comes from the ESPN Cricinfo website. A typical (but interrupted) set of five lines of a raw text of interest from the commentary log of a twenty-over match from ESPN (2013) are:

0.1
Finn to Warner, no run, full length on middle and off, Warner blocks it on the crease, that disturbed the top of the surface, pretty dry wicket

0.2
Finn to Warner, 2 leg byes, back of a length, fended off down the leg side, it runs fine of long leg and they get back for second leg bye

0.5
Finn to Finch, SIX, just back of a length on off stump and Finch is onto it straight away to slam it over deep square leg, absolutely nailed, was probably a good length ball actually, that was creamed just rocking back.

These lines were parsed with regular expressions and functions from the stringr package in a manner similar to Algorithm 1,
while not at end of this document do
    read current line;
    if line is like "number1.number2" then
        over ← number1
        ball ← number2
        scan_for_event ← TRUE
    end
    if scan_for_event AND line is like "string1 to string2, string3, string4" then
        bowler ← string1
        batsman ← string2
        outcome ← string3
        fielding ← string4
        scan_for_event ← FALSE
        ready_to_process ← TRUE
    end
end

**Algorithm 2**: Core of algorithm for extracting pertinent data from cricket text commentary

After Algorithm 1 is performed, the variables ‘bowler’ and ‘batsman’ are checked against a list of players and some secondary data about the match to assign identity numbers. Sometimes additional disambiguation is required, such as when one of teammates and brothers Nathan McCullum and Brendan McCullum are playing, but not both. In these cases the play-by-play transcript will then refer to either player as ‘McCullum’.

The ‘outcome’ string is processed to extract all the pertinent game state information, such as the number of runs, wickets, and extras that occurred.

The ‘fielding’ string is first checked for an event called a run out, which affects the game state in the same way a wicket does, but may not be mentioned in the ‘outcome’ string. The ‘fielding’ string is also scanned for mentions of players who were in the field at the time to track notable fielding interventions. These fielding interventions are the basis of the paper Perera, Davis & Swartz (2015b), which also appears in Harsha Perera’s dissertation.

Identification of the players mentioned in the fielding portion of commentary poses additional challenges over identification of the bowler and batsman. Aside from the occasional alternate spelling of a name and the issue with last name ambiguity, bowling and batting name uses are consistent. When fielding, players are sometimes referred to by only their first names, by a shortened name, or by a well known nickname. Also, more than player’s name can appear in a single commentary line, including the bowler, the batsman’s partner, players that are not participating in the match (e.g. for comparison or colour commentary), or the captain of either team (e.g. reacting to a play, or appealing to an official).

Some of these ambiguities cannot be resolved with simple parsing or by consulting a table of player names. In Perera, Davis & Swartz (2015b), where fielding effects are addressed, Harsha and I manually curated 3000 of the approximately 15,500 commentary lines that my text filter had identified as possibly notable fielding plays. Of these candidates, about a quarter only mentioned a non-fielding player, and in another quarter the mentioned fielder
had no surprising effect on the result (e.g. by succeeding at a trivial play, failing to do something truly outstanding, or being the nearest player to a ball flying high above).

Other state based information is extracted from the commentary at the beginning of each match. For example, the match number and format are extracted from the text line containing “ODI No. (xx)” or “T20I No. (xx)” where ‘(xx)’ is the identifying number of the match. In One-Day International and Twenty20 International matches, the match number has been incrementing since the first international match in each format, respectively.

India Premier League matches have a similar commentary structure, but their matches are identified using the line containing “India Premier League = (xx)(st/nd/rd/th) match”, or in end-of-season cases by a line containing a phrase like “semifinal qualifier” in the place of “(xx)th match”. This information is combined with the year of play, which is similarly extracted, to construct a unique identifier of each recorded game.

The ID of each game is used to attach other pertinent information to that game that we parse from summaries of the games. These summaries are used in addition to the commentary logs that were used to obtain ball-by-ball data. Table 1.1 contains the game summary for the first innings of Twenty20 International match. From this summary, we extract that this was the 453th match of T20I format, that England batted first, the order that English players batted (starting with JJ Roy and AD Hales, and ending with BA Stokes), and the intended order of the remaining players that did not bat.

The player names in these summaries are always of the same format. That is, all names are included as initials before the last name. This consistency allows for the disambiguation of names in the commentary for all batting and bowling cases, and some fielding cases.

Other information, such as the numbers in each column representing the number of runs scored, minutes batted, balls faced, fours scored, sixes scored, and strike rate respectively are compiled from the ball-by-ball information instead. Similarly, none of the bowling summary information is needed.

Some fielding information is included in the dismissal column of the batting summary. For example, “c Coulter-Nile b Cummins” next to JJ Roy’s name indicates that Roy left the match when a ball he hit was caught before landing by NM Coulter-Nile, and that PJ Cummings had bowled that ball. This information is also found in the fielding portion of the ball-by-ball data, but this column in the summary allows for further disambiguation of the fielder of note.

The stringr package for R (Wickham 2012) was essential for constructing every text parser that was used. I have posted a tutorial on using stringr on cricket summary data, including some of the parsing code, at factotumjack.blogspot.ca/2015/01/package-spotlight-stringr.html. I have also written lecture notes and exercises on more general principles of text processing in R. Material on regular expressions, which are used to define generalized text patterns, are found at factotumjack.blogspot.ca/2015/07/lesson-prototype-regular-expressions-in.html. Material on edit distance, an instrumental concept in fuzzy matching, is found at factotumjack.blogspot.ca/2015/08/lesson-notes-edit-distance-text.html.
England v Australia  
England won by 5 runs  
T20I no. 453 | 2015 season  
Played at Sophia Gardens, Cardiff  
31 August 2015 (20-over match)  

<table>
<thead>
<tr>
<th>England innings (20 overs maximum)</th>
<th>R</th>
<th>M</th>
<th>B</th>
<th>4s</th>
<th>6s</th>
<th>SR</th>
</tr>
</thead>
<tbody>
<tr>
<td>JJ Roy c Coulter-Nile b Cummins</td>
<td>11</td>
<td>19</td>
<td>16</td>
<td>1</td>
<td>0</td>
<td>68.75</td>
</tr>
<tr>
<td>AD Hales b Cummins</td>
<td>3</td>
<td>16</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>42.85</td>
</tr>
<tr>
<td>MM Ali not out</td>
<td>72</td>
<td>70</td>
<td>46</td>
<td>6</td>
<td>3</td>
<td>156.52</td>
</tr>
<tr>
<td>EJG Morgan c Watson b Coulter-Nile</td>
<td>74</td>
<td>50</td>
<td>39</td>
<td>3</td>
<td>7</td>
<td>189.74</td>
</tr>
<tr>
<td>JC Buttler c Cummins b Starc</td>
<td>11</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>275.00</td>
</tr>
<tr>
<td>SW Billings run out (Smith/Starc)</td>
<td>2</td>
<td>11</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>28.57</td>
</tr>
<tr>
<td>BA Stokes not out</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Did not bat: AU Rashid, DJ Willey, ST Finn, RJW Topley

<table>
<thead>
<tr>
<th>Bowling</th>
<th>O</th>
<th>R</th>
<th>W</th>
<th>Econ</th>
<th>0s</th>
<th>4s</th>
<th>6s</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA Starc</td>
<td>4</td>
<td>32</td>
<td>1</td>
<td>8.00</td>
<td>12</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>NM Coulter-Nile</td>
<td>4</td>
<td>24</td>
<td>1</td>
<td>6.00</td>
<td>10</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>PJ Cummins</td>
<td>4</td>
<td>25</td>
<td>2</td>
<td>6.25</td>
<td>12</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>MR Marsh</td>
<td>2</td>
<td>21</td>
<td>0</td>
<td>10.50</td>
<td>5</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>GJ Maxwell</td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>6.00</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MP Stoinis</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>13.00</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>SR Watson</td>
<td>2</td>
<td>33</td>
<td>0</td>
<td>16.50</td>
<td>3</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>CJ Boyce</td>
<td>1</td>
<td>19</td>
<td>0</td>
<td>19.00</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Table A.1: Summary statistics of a typical Twenty 20 International match.
Appendix B

Estimation of Cricket Situation Effects

The work in Chapters 2 and 3, as well as the two analysis chapters to appear in Harsha Perera’s thesis, rely on a system that simulates cricket matches with a higher degree of realism than any other publicly available system. Much of this system is described in Chapter 2, but the estimation of the parameters $\tau_{owj}$, which govern the relative propensity towards various outcomes $j$ during the $d^{th}$ over and when $w$ players have been dismissed, was presented in a much-abridged appendix for the sake of brevity. In order for the reader to appreciate the cricket work as a whole, and to guide future extensions, the method of estimation of the parameters $\tau_{owj}$ is described here.

Assume that the outcome of any single ball thrown in a cricket match can be partitioned into one of eight possibilities $j \in 0, 1, 2, \ldots, 6, 7$, where $j = 7$ represents a dismissal of a batsman, and any $0 \leq j \leq 6$ represents the number of runs scored when a player is not dismissed. Special cases, such as no-balls and leg-byes, also fit into these outcomes. The proportion of each outcome that includes each of these special cases is recorded, and are assumed to be the conditional probabilities of each case given $j$. In simulation, the outcome is generated first, and the relevant conditional probabilities are used to determine if the simulated ball is a special case.

We wish to be able to estimate the probability distribution of the outcomes for a ball thrown by any bowler to any batsman in any possible game situation. This includes game situations that either the bowler or batter may never have been in, such as an opening batsman batting near the end of the match.

To achieve this flexibility, we need to estimate two things. First, we estimate the effect that each game situation has on the probability of each outcome $j$. Second, we estimate a player’s intrinsic ability to produce each outcome and some baseline game situation. The latter estimation procedure is covered in Chapter 2, but it requires the estimation of game situation to be completed first.

Game situation and player ability are entangled - reliable batsmen placed early the batting order, and therefore bat in early-game situations more often than other batsmen. Weaker batsmen, such as bowling specialists, only bat in end-game situations. Game situations
affect the probability of outcomes because the relative value of resources, that is, overs and wickets remaining, changes by situation. The estimation of bowler and batsman effects are done with the same method, so for simplicity, the method is only shown for batsmen.

From the ball-by-ball data we tabulate $X_{iowj}$, which is the number of times that player $i$ had outcome $j$ while batting when $o$ overs and $w$ wickets have been taken (i.e. when $w$ batsmen have been dismissed). With this tabulation, we can separate the effects of the situation $(o$ and $w$) from the random effects of each batsman $i$. We do this by assuming that situational and batsman effects are additive, and treating the observations of a single player as a block.

For each outcome $j$, the following process is identical. For a pair of situations $(o, w)$ and $(o+1, w)$ that are adjacent along the 'over' axis, we consider every player $i$ that has batted in both situations. That is, we consider any $i$ such that $\sum_j X_{iowj} > 0$ and $\sum_j X_{i(o+1)wj} > 0$. For each such player, we estimate the ratio of outcome probability between the situations. Let this estimated ratio be $\hat{\alpha}_{iowj}$ where

$$\hat{\alpha}_{iowj} = \frac{X_i(o+1)wj / \sum_j X_{i(o+1)wj}}{X_{iowj} / \sum_j X_{iowj}}. \tag{B.1}$$

The value of $\hat{\alpha}_{iowj}$ for any given $i$ and $j$ represents and estimates of the relative propensity of outcome $j$ in situation $(o+1, w)$ compared to $(o, w)$. If $\hat{\alpha}_{iowj} > 1$, then outcome $j$ is estimated to be more likely during situation $(o+1, w)$ than during situation $(o, w)$. However, each estimate only uses a single batsman’s data. We combine these estimates as a weighted sum, $\alpha^*_{owj}$, where weights are inversely proportional to the standard error of each estimate,

$$\alpha^*_{owj} = \sum_i \sqrt{\frac{1}{v_{xiowj}}} 1(0 < v_{xiowj} < +\infty). \tag{B.2}$$

The square of these standard errors, $v_{xiowj}$, are estimated as

$$v_{xiowj} = \hat{\alpha}_{iowj}^2 \times \left[ \frac{(\hat{p}_{iowj})(1 - \hat{p}_{iowj})}{\sum_j X_{iowj}} + \frac{(\hat{p}_{i(o+1)wj})(1 - \hat{p}_{i(o+1)wj})}{\sum_j X_{i(o+1)wj}} \right], \tag{B.3}$$

and where $\hat{p}_{iowj} = \frac{X_{iowj}}{\sum_j X_{iowj}}$ and $\hat{p}_{i(o+1)wj} = \frac{X_{i(o+1)wj}}{\sum_j X_{i(o+1)wj}}$.

We compute a composite estimate of the relative propensity across all pairs in adjacent situations $(o, w)$ and $(o, w+1)$ along the wicket axis, $\beta^*_{owj}$, similarly.

From the adjacency coefficients $\beta^*_{owj}$ and $\beta^*_{owj}$, for $o = 1, 2, \ldots, 18, 19; w = 0, 1, \ldots, 7, 8$, we estimate the situational effects $\tau_{woj}$, where $\tau_{woj}$ represents the factor by which outcome $j$ is more likely (before normalization) in situation $(o, w)$ than in some arbitrary baseline situation.

89
We compute the estimates of $\tau_{owj}$ by taking the product of the adjacency coefficients on one of the paths from the ‘corner’ situation of $o = 1, w = 0$ to the situation $(o, w)$ of interest that most closely followed a straight line from $(1, 0)$ to $(o, w)$. For example, such a path from $(1,0)$ to $(5,3)$ would be $(1,0) \rightarrow (2,0) \rightarrow (2,1) \rightarrow (3,1) \rightarrow (3,2) \rightarrow (4,2) \rightarrow (4,3) \rightarrow (5,3)$ the situational effect $\tau_{5,3,j}$ for any $j$ would be

$$\hat{\tau}_{5,3,j} = \tau_{1,0,j} \times \left[ \alpha_{1,0,j}^* \times \beta_{2,0,j}^* \times \alpha_{2,1,j}^* \times \beta_{3,1,j}^* \times \alpha_{3,2,j}^* \times \beta_{4,2,j}^* \times \alpha_{4,3,j}^* \times \beta_{4,3,j}^* \right].$$ (B.4)

When all possible $\tau_{owj}$ factors for a given outcome $j$ are estimated (some absurd situations are never observed, thus no estimation is possible), we smooth the estimates by taking the natural log of each $\tau_{owj}$ for that outcome and taking a weighted fit a third-order polynomial response surface to the values of $\ln(\tau_{owj})$ across the over-wicket plane. A step function is also included along the $o = 6.5$ line to allow for a jump discontinuity between the $6^{th}$ and $7^{th}$ overs. This is to reflect the effect of the ‘powerplay’, which is a restriction to fielding that is in force for the first 6 overs of each innings in a Twenty20 match. The weights used for fitting are equal to the number of observations across all players observed for that situation, so the response surface fits common situations very closely and neglects the absurd situations. The fitted values at each point $o = 1, 2, \ldots, 18, 19; w = 0, 1, \ldots, 7, 8$ are exponentiated to produce the smoothed estimates $\tau_{owj}^*$. Finally, we choose a ‘baseline’ situation of $o = 7, w = 0$, which is the situation of maximum batting resources outside of the powerplay. This choice of baseline and the use of the situational effects are discussed in Chapter 2. Some completed computational adjustments to this method and potential improvements are discussed in Chapter 5.
Appendix C

Computation of Situational Effects
(Chapter 2 appendix)

Recall that the multinomial model (2.2) is highly parametrized where the data are sparse and even nonexistent over regions of the parameter space. The simplifying assumption (2.3) leads to a more tractable model where the parameters $p_{i70j}$ and $\tau_{owj}$ are estimated in two steps. In Section 3, we described a hierarchical model where a Bayesian approach was taken to estimate the $p_{i70j}$. A key component of the approach was the recognition of similar batting characteristics amongst players. Here, in the Appendix, we describe the estimation of $\tau_{owj}$, the parameters used to describe the modification of batting characteristics with respect to the stage of the match (i.e. overs consumed and wickets taken).

Let $x_{iowj}$ denote the number of occurrences of outcome $j$ by batsman $i$ for all batting attempts in the $o$th over with $w$ wickets taken. The corresponding empirical probability is $\hat{p}_{iowj} = x_{iowj}/n_{iow}$ where $n_{iow} = \sum_j x_{iowj}$.

Next, we define the transition factor $\hat{\alpha}_{iowj} = \hat{p}_{iowj}/\hat{p}_{iowj}$ which represents the change in empirical probabilities for batsman $i$ when going from the stage of the match $(o, w)$ to the adjacent stage $(o', w) = (o + 1, w)$ corresponding to the next over. We then average the transition factors over all batsmen giving

$$\hat{\alpha}_{owj} = \frac{\sum_i v_{iowj}^{-1/2} \hat{\alpha}_{iowj}}{\sum_i v_{iowj}^{-1/2}} \tag{C.1}$$

The Delta Theorem is then used to obtain the variance expressions for ratios,

$$v_{iowj} = \hat{\alpha}_{iowj}^2 \left( \frac{1 - \hat{p}_{iowj}}{n_{iow} \hat{p}_{iowj}} + \frac{1 - \hat{p}_{iowj}}{n_{iow} \hat{p}_{iowj}} \right).$$

We can therefore view the estimates $\hat{\alpha}_{owj}$ as forming a matrix with the rows corresponding to overs ($o = 1, \ldots, 20$) and the columns corresponding to wickets ($w = 0, \ldots, 9$). For any stage $(o, w)$ of a match, the matrix entry $\hat{\alpha}_{owj}$ is the transition factor for changing the probability $p_{iowj}$ to the probability $p_{iowj'}$ for any batsman $i$. With respect to the matrix of
transition factors, the movement corresponds to going down column $w$ from row $o$ to row $o' = o + 1$. We smooth the matrix to improve the estimates.

Analogous to (2.11), transition factors $\hat{\beta}_{owj}$ can be defined when going from the stage of the match $(o, w)$ to the adjacent stage $(o, w') = (o, w + 1)$ corresponding to the next wicket. We then have a second matrix where $\hat{\beta}_{owj}$ describes the movement along row $o$ from column $w$ to column $w' = w + 1$.

Finally, to obtain the parameter $\tau_{owj}$, we recall that $\tau_{owj}$ is the multiplier that is used to modify the baseline probability $p_{i70j}$ in (3) to the probability $p_{iowj}$. We obtain $\tau_{owj}$ by taking the straight line from the matrix position corresponding to the start of the innings $(o = 1, w = 0)$ to $(o, w)$ and use the nearest transition factors $\hat{\alpha}$ and $\hat{\beta}$ as multipliers.

We remark that the proposed estimation procedure for $\tau_{owj}$ is based on incremental changes to overs and wickets. It is not possible to estimate directly from the baseline state $(o = 7, w = 0)$ to a distant stage $(o, w)$ since there are very few (if any) batsmen who have batted in both stages. However, by approaching the estimation incrementally, there exist common batsmen who bat in adjacent stages.