Department of Mathematics and Statistics

A Framework for Near Real-Time AFL Match Outcome Prediction

Casey Josman

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University

December 2022

Declaration

To the best of my knowledge and belief, this thesis contains no material previously published by any other person except where due acknowledgment has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

....

Casey Josman

23rd December 2022

Abstract

The research herein concerns itself with the real-time prediction and forecasting of Australian Football League (AFL) match outcomes, and consequently aims to remedy the lack of real-time analysis within the sport. To this effect data has been acquired as follows: data on past performances (static data), in-game statistics (dynamic data); after which statistical modelling methods were used in order to develop of a robust yet multifaceted analysis methodology.

This research has been conducted in two major phases; firstly, the assessment and application of static data for the prediction of match outcomes and their relevant applications with respect to match, fixture, and team performance analysis by means of regression and machine learning algorithms. Secondly, the utilisation of both the static data from the previous phase and in-game play-by-play metrics in order to develop a real-time prediction methodology.

Phase 1 considers four candidate models as such to account for the breadth of methodologies found in the available literature. These models in order of increasing complexity are as follows: multinomial logistic regression (MLogR), logistic model tree (LMT), random forest (RF), and support vector machine (SVM). Whereas phase 2 utilises a continuous time inhomogeneous Markov model to account for the sporadic nature of the real-time data observed as well as the computational optimisations afforded by said model.

The results for both static and dynamic data models are significant, the static MLogR model yielded comparative results to those found in the literature with an accuracy of 69.60% while the dynamic Markov model achieved impressive results with an average epoch prediction accuracy in excess of 80% and an average match outcome prediction accuracy in excess of 90%.

The outcome of this research are promising and will aid coaches in making informed strategic decisions during matches as well as assist them in retrospective analysis of previous matches. In addition, it is a goal of this research that the methodological framework developed here be easily transferred across other sports.

List of publications

The following papers were published or accepted for publication during the PhD candidature:

- C. Josman, R. Gupta, and S. Robertson. 2016a. "Fixture Difficulty and Team Performance Models for use in the Australian Football League." In *Proceedings of the 13th Australasian Conference on Mathematics and Computers in Sport,* 15–20. ANZIAM MathSport. ISBN: 978-0-646-95741-8

- C. Josman, R. Gupta, and S. Robertson. 2020a. "Markov Chain Models for the Near Real-Time Forecasting of Australian Football League Match Outcomes." In *Soft Computing for Problem Solving 2019,* 111–25. Springer. https://doi.org/10.1007/978-981-15-3287-0_9

Authorship Attribution Statement

The following is a description of the contribution of the main and co-authors for each of the published manuscripts supporting this thesis:

Paper 1 - C. Josman, R. Gupta, and S. Robertson. 2016b. "Fixture Difficulty and Team Performance Models for use in the Australian Football League." In *Proceedings of the 13th Australasian Conference on Mathematics and Computers in Sport,* 15–20. ANZIAM MathSport. ISBN: 978-0-646-95741-8

	Concept and Design	Acquisition of Data and Method	Data Conditioning and Manipulation	Analysis and Statistical Method	Interpretation and Discussion
Casey Josman	✓	✓	✓	✓	✓

I acknowledge that these represent my contribution to the above research output and I have approved the final version.

Signature:

	Concept and Design	Acquisition of Data and Method	Data Conditioning and Manipulation	Analysis and Statistical Method	Interpretation and Discussion
Ritu Gupta	✓	✓	✓	✓	✓

I acknowledge that these represent my contribution to the above research output and I have approved the final version.

Signature:

	Concept and Design	Acquisition of Data and Method	Data Conditioning and Manipulation	Analysis and Statistical Method	Interpretation and Discussion
Sam Robertson	✓	✓			✓

I acknowledge that these represent my contribution to the above research output and I have approved the final version.

Signature:

Paper 2 - C. Josman, R. Gupta, and S. Robertson. 2020b. "Markov Chain Models for the Near Real-Time Forecasting of Australian Football League Match Outcomes." In *Soft Computing for Problem Solving 2019*, 111–25. Springer. https://doi.org/10.1007/978-981-15-3287-0_9

	Concept and Design	Acquisition of Data and Method	Data Conditioning and Manipulation	Analysis and Statistical Method	Interpretation and Discussion
Casey Josman	✓	✓	✓	✓	✓

I acknowledge that these represent my contribution to the above research output and I have approved the final version.

Signature:

	Concept and Design	Acquisition of Data and Method	Data Conditioning and Manipulation	Analysis and Statistical Method	Interpretation and Discussion
Ritu Gupta	✓	✓	✓	✓	✓

I acknowledge that these represent my contribution to the above research output and I have approved the final version.

Signature:

	Concept and Design	Acquisition of Data and Method	Data Conditioning and Manipulation	Analysis and Statistical Method	Interpretation and Discussion
Sam Robertson	✓	✓			✓

I acknowledge that these represent my contribution to the above research output and I have approved the final version.

Signature:

Acknowledgements

This may be cliché, but if you asked me how I reached this point in my life – I could not tell you. As a child I dreamt of becoming a 'scientist'; spurred on by episodes of *Bill Nye the Science Guy* and *Dexter's Lab*. However, this desire waned over the years with my aspirations shifting to computers and electronics, then literature, and finally settling on mathematics and statistics.

I immigrated to Australia in 2008 to pursue a degree in actuarial science, however, transitioned to actuarial and applied statistics in my final year. As my final semester came to a close; and with myself unsure of what to do next, my lecturer at the time (Dr Ritu Gupta) asked if I had considered undertaking an honours degree. I replied that I had not – but the seeds had already been planted, and had begun sprouting.

And so, from honours to PhD, Ritu has guided me down the path of academia, with my love for the field growing ever stronger. Thusly, I would first like to thank my supervisor Dr Ritu Gupta for her help, guidance, and most of all patience – and for that I am truly grateful. I would also like to thank my co-supervisors Dr Sam Robertson of the University of Victoria and Western Bulldogs, and Dr Aloke Phatak of Curtin University. Sam for providing insight into the world of Australian sports and acting as a liaison between myself and Champion Data for the purposes of data procurement; and Aloke whom I probably did not bother enough, for his insight into statistical methods, and acting as an occasional sounding board.

Thank you to Champion Data and Western Bulldogs for the use of their real-time match data, and Paul from AFL Tables for providing me with historical data for the VFL and AFL since their inception.

Most importantly I would like to thank my family for all their love and support over this difficult time. Ma and Gramps for ensuring my continued existence and for putting up with me and my idiosyncrasies; Dad and Lisa for urging me to follow my dreams, being indisposable voices of encouragement, and for putting up with me not coming home for so long; Chad for being a continual source of humour and encouragement; and all of those not mentioned and those lost along the way.

Finally, Dr Julia Charkey-Papp for helping me through the thick of it, listening to my worries, reassuring me all through my existential crises, and helping me battle the ennui and despair that come with living in a world over which I have no control.

In Memory of my Mother

"Terpsichore is a jealous goddess,
and those who seek fame among her votaries
must sacrifice at her alter years of patient study
and hours of physical labour."

(Cyril W. Beaumont)

Contents

List of Figures

List of Tables

Nomenclature

$\mathcal{H}$	Set of home teams
$\mathcal{A}$	Set of away teams
i	Home team index
j	Away team index
t	Current time in match
T	Time at end of match
$R_{(\mathcal{H},\mathcal{A},m)}$	Match result indicator
S	General set of static features
$S^{\mathcal{H}}$	Set of home team static features
$S^{\mathcal{A}}$	Set of away team static features
F	Full set of static features
D_t	General set of dynamic features at time t
$D^{\mathcal{H}}(t)$	Set of home team dynamic features at time t
$D^{\mathcal{A}}(t)$	Set of away team dynamic features at time t
F_t	Full set of dynamic features
$C(\cdot)$	Match outcome probabilities
$C_t(\cdot)$	Match outcome probabilities at time t
k	Head to head match look back factor
l	Past match look back factor
$\mathfrak{q}$	Lower probability threshold
$\mathfrak{p}$	Upper probability threshold
$\mathfrak{p}_1$	Lower point threshold
$\mathfrak{p}_2$	Upper point threshold
$\mathcal{D}_{\mathbb{T},\mathcal{R}}$	Fixture difficulty
$\mathbb{S}$	State Space
$Q\left(F_{t_j}\right)$	Transition intensity matrix
$\mathbf{u}_*$	Initial model probabilities
π_0	Initial transition probability matrix
$\pi_{t,t+1}$	Transition probability matrix from time t to $t+1$

CHAPTER 1

Introduction

The Australian Football League henceforth referred to as the AFL (the sport itself is also colloquially called AFL) is one of the most popular sports and leagues in Australia having estimated club memberships of 1.1 million for the 2021 season, yielding an approximate growth of 12% over the 2019 season. A testament to this growth also saw all 18 clubs fielding a team in the 2021 Australian Football League Women's (AFLW) premiership season, as well as a total increase in revenue of 9%.

1.1 Sports Analysis and Outcome Prediction

In the ever-present quest to outperform one's competitors, athletes are evermore pushing the limits of human physiology — but at what point does raw physiological supremacy cease being the stopgap by which victory is predicted? Beginning with the turn of the twenty-first century, both training methodologies and team related strategies have evolved in such a way that optimality is desired not only in team composition and training but also in injury management and financial return. The precursor to this paradigm shift would appear to be the work of Billy Beane (Baumer and Zimbalist 2014), who throughout his tenure at The Oakland Athletics popularised the burgeoning field of sports analytics and in turn changed the inner workings of competitive sports forever.

Traditionally, a team's performance was measured as the number of matches won and where appropriate augmented by the margin by which each match was won. However, commentators and analysts have often posited questions such as; does the home team have an inherent advantage? How does travel and time off affect a team's next match? What is the optimal match schedule for a given season? As well as many others, with the majority of answers having either anecdotal or non-empirical slants. And whilst researchers have tried to answer many of these questions, they are most often looked at separately with no regard for confounding factors. For example, the concept of home team advantage is often observed such that the relative strength of the away team is ignored.

Henceforth, in order to facilitate an all encompassing research methodology towards the goal of real-time AFL match outcome prediction — a multifaceted approach was adopted, with analysis integrating current literature as well as novel methodologies.

1.2 Research Objectives

The main objectives of this research can be summarised as follows:

- To screen for and extract relevant match features which are appropriate for the prediction of AFL match outcome probabilities with respect to the home team drawing, losing, or winning the match. For this purpose, features will be gathered from data published prior to each match as well as collected whilst matches are in progress.
- To investigate alternate metrics for team performance and fixture difficulty.
- To investigate various statistical and machine learning techniques for producing near real-time match outcome predictions. With 'near' referring to the time lag between recording an on-field transaction and supplying it to the model for prediction.
- To develop accurate prediction models which incorporate both classical and novel approaches to data screening, feature extraction, and model usage.

1.3 Significance of this Research

- This research seeks to remedy the lack of real-time analysis in the realm of Australian Rules Football and similar fast-paced sports. It has become apparent through review of current literature and consultation with industry professionals that this is due to the cost prohibitive and proprietary nature of real-time data collection as well as its applications and implications.
- As sporting clubs are becoming far more proactive in their own data management and in-house analysis, the need for far more sophisticated approaches in terms of performance analysis and outcome prediction is on the rise. This research provides a multifaceted framework for said real-time prediction and by extension performance and fixture analysis.
- Throughout the process of screening for and extracting key features, novel metrics for team performance and fixture difficulty were developed. It is posited that these metrics yield far more balanced representations than their currently used counterparts as they take into account both forecast match outcomes and perceived opponent difficulty based on past performances.

- The results obtained from this research validate the effectiveness of the proposed framework and methodologies contained within. In addition to this, the framework and methodologies have been designed in such a way that it should be easily transferable to other sports.

1.4 Structure of this Thesis

This thesis is organised as follows:

Chapter 2 provides a concise review of the available literature in the fields of Australian Rules Football analytics, ex-ante and real-time AFL match outcome prediction; as well as summarises the statistical and machine learning methods to be utilised within. Chapter 3 provides detailed insight into the acquisition and processing of data, and the methodology utilised in feature selection and extraction. Chapters 4 and 5 present the bodies of research dealing with ex-ante and real-time predicion respectively, with each chapter covering the following:

- Mathematical formulation of the forecasting model.
- Detailed mathematical formulation of each of the statistical / machine learning techniques.
- A breakdown of the results obtained from each of the sub-models including an in-depth discussion of the findings.
- Applications and ancillary use cases.

Finally, chapter 6 provides a summary of research, major contributions, conclusions, and recommendations for future works.

CHAPTER 2

Literature Review

2.1 Overview

In the ever evolving field of sports analytics, real time analysis has become a key area of interest. However, due to the proprietary nature of real-time data most public research is confined to ex-ante result prediction and optimal betting strategies with the goal of beating bookmakers odds. Due to this distinction current research can be classified into two categories: ex-ante prediction using static features, and real-time prediction using dynamic features. Static features consist of match information prior to the start of a match, while dynamic features are based on in-match information.

Features which are used for both approaches do not differ significantly across classification and regression methods but tend to follow a logical grouping depending on which sport is being observed. From features which are commonly utilised, such as teams or players involved to sport specific features such as rebounds or turnovers, it is clear that feature selection dictates the success of these models (Lopez and Matthews 2014). Features should be selected carefully paying attention to not only the statistical merit of each feature but also to their relevance to the sport as a whole.

Ex-ante prediction (Constantinou, Fenton, and Neil 2012; Delen, Cogdell, and Kasap 2012; Lopez and Matthews 2014; Maszczyk et al. 2014) is implemented in a variety of sports regardless of tempo (the speed at which the sport is played) and is a large part of the currently available literature. Machine learning techniques such as Artificial Neural Networks (ANN) and Support Vector Machines (SVM) were used to great success for result prediction in both American Football and Athletics (Delen, Cogdell, and Kasap 2012; Maszczyk et al. 2014).

On the other end of the spectrum different methods of regression and generalised linear models were used to accurately predict match outcome, points scored, and margin of victory (MOV) (Stefani and Clarke 1992; Goddard 2005; Rue and Salvesen 2000; Crowder et al. 2002; Lopez and Matthews 2014). Goddard (2005) was able to predict the goals scored and conceded by the home team. The results were comparable to predicting match outcomes win, loss, draw by using generalised linear models Stefani and Clarke (1992) using linear regression were able to quantify the home advantage for each team and predict the MOV for a given pairing of teams within the AFL.

Due to the cost and difficulty of simultaneous data collection real-time prediction (Min et al. 2008; Akhtar and Scarf 2012), is carried out on slower moving sports (when compared to Australian Rules Football) and those where up to date data is easily available, such as cricket (Bailey and Clarke 2006; Akhtar and Scarf 2012) and soccer (Min et al. 2008). These applications tend to use less computationally taxing methods such as multinomial linear and logistic regression, and rely heavily on pre-established methodologies such as the Duckworth-Lewis resource matrix (Duckworth and Lewis 2004; Stern 2016) and existing match strategies.

This chapter is divided into four main sections, each of which elaborate upon the key ideas and theoretical underpinnings on which this research is based. Firstly, section 2.2 investigates Australian Rules Football as a sport and explores the statistical and analytic methods employed in AFL match outcome prediction. Secondly, section 2.3 gives insight into current real-time prediction research both within and without the sporting realm. Thirdly, section 2.4 provides an overview of the methods employed in this study; including advantages, disadvantages, and current applications found in contemporary literature. Finally, section 2.5 gives a brief overview of future avenues of data acquisition.

The objectives of this chapter are:

(i) Explore Australian Rules Football as a sport.

(ii) Present the evolution of AFL match prediction methods.

(iii) Explore the current state of both ex-ante and real-time prediction.

(iv) Introduce the statistical models used in this study.

(v) Explore future sources of player and match data.

2.2 AFL

The AFL began its life in 1896 when the six strongest clubs in Victoria broke away from the then-current Victorian Football Association over administrative differences. These clubs would then go on to establish the Victorian Football League (VFL) which over the years would expand to include interstate teams and form the AFL as known today.

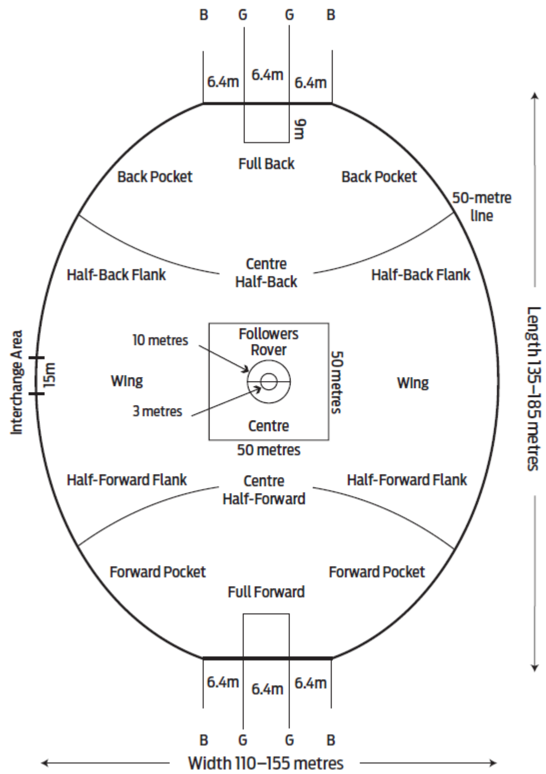
Australian Rules Football is an invasion style ball game similar to both rugby and American Football in which two teams vie for leadership by scoring points either by kicking the ball through the centre posts (scoring a goal worth 6 points) or through the outer posts (scoring a behind worth 1 point). A typical AFL (men's) season consists of two phases; a 23 round premiership season wherein each team plays 22 matches over the course of 23 weeks, and a 4 week finals series wherein the top 8 teams of the premiership season play for a place in the grand final.

During each premiership season teams are ranked based on the number of premiership points won, with a win yielding 4 points, a draw 2 points, and a loss 0 points; however, in the case of a tie teams are then ranked as a percentage of total match points scored to total match points conceded.

The finals series is played according to the AFL final eight system which is a modified version of the McIntyre final eight system. This set-up requires that the top 4 teams need only win 2 games while the bottom 4 teams need to win a total of 3 games thus ensuring an easier path to the grand final for the higher ranked teams.

Australian Rules Football is played on an oval field or varying size (Figure 2.1) by two teams of 22 players (18 on-field and 4 reserves) over the course of 4 quarters. Each quarter theoretically runs for 20 minutes, however, due to the addition of stoppage time and allowing for on-field interruptions a quarter could run for as long as 37 minutes. During play the ball is moved down the field by either kicking, passing, or handballing the ball to another player which results in a fast pace game where strategy and possession are of utmost importance.

Figure 2.1: Field Dimensions and Positions (Australian Football League 2015).



2.2.1 Prediction of AFL Match Outcomes

Research into the AFL has existed almost as long as the game has, however, as the game has matured so has the depth and breadth of its field of research. Initial studies concerned themselves with the concept of home advantage and its effect on match outcome given the intrinsic advantage awarded to the home team (Ryall 2011; Taylor and Demick 1994; Clarke 2005). In the earliest days of the AFL, when interstate travel was at a minimum, the main cause of home advantage was thought to be just as the name implied; whether a team is playing at their home stadium, or in front of a majority crowd. However, as the league expanded so did the contributing factors (Johnston et al. 2018). From amongst this myriad of factors it is possible to identify the following three criteria: psychological, tactical, and physiological.

Psychological factors are the 'typical' influences that the average fan would identify and are those which directly affect players on a psychosomatic level and can range anywhere from cognitive ability to a bad performance in a previous match (Woods et al. 2016). This is additionally confounded by the journalistic practices of the media (Pedersen 2014; Sheffer and Schultz 2013; Weedon et al. 2018) where views are attracted through sensationalist headlines and appeals to emotion. Common amongst these are; Is a team playing at their home stadium (Courneya and Carron 1992)? Do they have a majority or hostile crowd (Russell 1983)? These all play a key role in a player's on-field performance and awareness.

Tactical factors primarily deal with a team's/player's ability to convert difficult positions through tactics and superior on-field 'skill' (Rennie et al. 2020) but can also refer to a team's/player's familiarity with a stadium and their ability to alter plays and strategies to accommodate changes in field size and condition. Tactics refer to on-field feats performed at either team or player level and are engrained through specialised training drills and skirmishes; these are generally under the purview of a team's coaching staff (Johnston et al. 2018) and are most times customised on a per match basis. Skill however pertains to a player's ability to handle both themselves and the ball during the ebb and flow of a match and is generally measured in terms of the various on-field transactions (handball, intercept, mark, etc.) performed by said player (Rennie et al. 2020).

Physiological factors deal with a player's inherent physical ability as well as the strain placed upon them during the course of a season. Physiological indicators of a player's success are generally accepted as a player's body composition (height, weight, musculature) but tend to be more complex interaction between those and their fitness. A player's height and wingspan enable easier access to the ball and makes them a difficult target to capture the ball from, whereas fitness (aerobic and anaerobic) may allow players of lesser stature compensate by jumping higher or running faster than their competition (Woods et al. 2016). Adding more complexity to this is a player's propensity for injury and how quickly they recover as well as additional strain introduced through travel and training regimen (Johnston et al. 2018).

When playing at home or any familiar stadium, a team's ground familiarity is an enormous advantage as the players are assured of no disruption to their regular training routine which would otherwise result in fatigue and inability to readily rely on familiar drills and strategies (Woods and Robertson 2021). When playing interstate, travel becomes increasingly more of a concern as the season progresses and can be detrimental to both the mental and physical state of players therefore resulting in poor on-field performance, this is due to the suboptimal recovery and training times afforded to the travelling team (Robertson and Joyce 2018). The home crowd factor is of extreme importance as it is always better and easier for the players to perform at their optimal whilst the crowd is clearly on their side, there is also less chance of the umpire making unfavourable calls due to a hostile crowd (Taylor and Demick 1994).

It must be noted that there is a discrepancy in the amount of home games played which results in some teams playing in front of large home crowds, while others play to smaller crowds. It is therefore obvious that results can be related to both travel and crowd size to the amount of home team victories.

Research into home advantage is plentiful and can be seen as the cornerstone of AFL research. Stefani and Clarke (1992) wrote their first paper seeking to validate the concept of home advantage in the AFL and found significant results; not only indicating home advantage, but also that non-Victorian teams are subject to a greater advantage on average than their Victorian counterparts. It would reason that the larger proportion of Victorian teams, their disproportionate travel requirements, and shared home stadia are to account for this (Stefani and Clarke 1992). Like many of their contemporaries (Stefani 1980, 1987; Pace and Carron 1992; Harville 1980) Stefani and Clarke's findings rely on linear regression analysis to assess the home advantage for each team in the AFL and produce similar overall findings to the home advantage experienced in other sports. Further research by Clarke (2005) culminating in his highly regarded publication 'Home advantage in the Australian football league' concur with the earlier findings of Stefani and Clarke (1992) and further elaborate on the following ideas: Australian Rules Football like many other similar sports is subject to unbalanced fixtures, this leads to stronger teams having relatively easier seasons, and in combination with the aforementioned home advantage phenomena results in an approximate home team win rate of 60%, with the home team scoring approximately 10.4 points more per game than their opponent. However, there is a caveat; if teams are of vastly different ability then home advantage will not necessarily play a major role in the outcome of the match.

Following on from the study of home advantage, the next and most prevalent field of study is that of match outcome prediction. The vast majority of research into AFL match prediction is confined to ex-ante prediction and the optimisation of betting strategies with the goal of beating bookmakers odds. Due to this distinction, current research can be classified into two categories: ex-ante prediction using static features (gathered before a match), and real-time prediction using dynamic features (gathered during a match). However, due to the proprietary nature of real-time data, most publicly available research is dedicated to the former.

Theoretically, features which are used for both approaches do not differ significantly, but tend to follow a logical progression depending on the complexity of the model used and output desired. From feature which are commonly used; such as team rankings and venue location, to more specific features such as field position and angles of attack, it is clear that feature selection significantly dictates a model's success (Lopez and Matthews 2014). Features should therefore be selected carefully, not only paying attention to the statistical merit of each feature but also to their relevance to the sport as a whole.

Linear models, such as those used by Stefani and Clarke (1992), Bailey (2005), Ryall (2011), and Robertson, Back, and Bartlett (2015) make use of far more rudimentary features; such as ranking, match outcome (MOV and outcome), and home advantage. Stefani and Clarke (1992) who can be though of as the progenitors of modern AFL research make use of their previous research into home advantage as well as a novel system of AFL team rankings and a least-squares approach to facilitate their predictions. These predictions which spanned the entirety of the 1980–1989 AFL premiership seasons achieved an average accuracy of 68.1% which are comparable to those made in similar sports of the time.

Bailey (2005) and Ryall (2011) whose research share the common goal of 'financial success' each take significantly different approaches. Bailey (2005) opts for a multiple linear regression whilst Ryall (2011) utilises an adjusted ELO style system (Elo and Sloan 2008) similar to that used in the ranking of chess players. Another significant difference between the two are the features used for each model; Ryall (2011) makes use of simplistic features such as: home advantage, travel fatigue, initial rankings, and match results over the 2002–2009 AFL premiership seasons. Whilst Bailey (2005) makes use of MOV, and differences in both turn overs and inside 50s at a team level over the 1987–1999 AFL premiership seasons, hence, they can be seen as one of the first AFL researchers to experiment with measures of team momentum. Regardless of this divergence in methodology Bailey (2005) and Ryall (2011) attained comparable results with 64% and 62.1% prediction accuracies and 10.1% and 10.4% average return on wager respectively.

Finally, Robertson, Back, and Bartlett (2015) utilised a gamut of match performance indicators recorded over the course of the 2013–2014 AFL premiership seasons in combination with binomial logistic regression. Ultimately settling on a set of statistically significant 'key' performance indicators which included but were not limited to: kicks, marks, and inside 50s (see Appendix B.2 for a full list of transactions and definitions). The outcomes of this research were two-fold; firstly, identifying significant performance metrics in the form of 'key' performance indicators, and secondly, by proving that the aforementioned features are able to be used to great effect in the prediction of match outcomes by achieving a prediction accuracy of 87.1% for the 2014 AFL premiership season.

Hence, the results of Robertson, Back, and Bartlett (2015) give further credence to a comment by Stefani and Clarke (1992) "It appears that the accuracy of a prediction depends primarily upon the information content of the data used to construct the [model] and much less on the algorithm used ..."

Following on from the linear methods above Leushuis (2018) observes that Australian Rules Football is a far more complicated than traditional modelling methods can compensate for and as such suggests a hybrid model composed of two random processes. The first, which models team performance is a Gaussian autoregressive process of order one, while the second process which models team ranking is a Markov chain model. Both processes are then combined using a Kalman filter with further smoothing being done by a Kalman smoother. Regardless of this leap in model complexity, the data used is comparable to the most basic models discussed above; MOV (modelled as a function of team strengths), and home advantage. It should be noted that unlike Stefani and Clarke (1992) who suggest individual home advantages per team, Leushuis (2018) uses a common value for all teams. This study spanned the 2012–2016 AFL premiership seasons and attained a prediction accuracy of 73.62%, and whilst the data used may be seen as rudimentary —the results are promising as it validates the use state space models in AFL match outcome prediction.

The final and most promising body of prediction research is that of artificial intelligence and machine learning, and with the ever-increasing potential of modern computer processors there is no telling where the proverbial ceiling lies. McCabe and Trevathan (2008) created a generalised model for outcome predictions in four sports (Australian Rules Football, Rugby League, Super Rugby, and English Premier League) and in doing so make use of an artificial neural network —more specifically a multilayer perceptron with three layers. While artificial neural networks are highly adaptable to almost any problem, they are unfortunately referred to as black box. This is due to the fact that their underlying structure yields no insight into how the model actually works. For this model features were extracted similarly to those studies previously discussed (MOV, team ranking, venue) as well as novel metrics of team form and performance. These metrics included but were not limited to: average score over the last n matches, win percentage over the past n matches, and win percentage for both home and away matches over the last n matches. The total number of features totalled 19 and resulted in an average prediction accuracy of 65.1% for their AFL predictions. A significant result of this research is that similar accuracies were obtained across all four of the sports studied, indicating that an interoperability exists amongst the features explored.

Furthermore, Young et al. (2019) making full use of the extended capabilities of machine learning algorithms investigated 103 performance indicators as potential features from data spanning the 2001–2016 AFL premiership seasons. In addition to this they investigated the concept of seasonal clusters within the AFL such that the features exhibited similar characteristics. From this they identified 2009 as a significant boundary point which just so happens to coincide with the addition of two new teams into the league. The feature selection process utilised an amalgamation of four metrics: information gain, information gain ratio, Gini index, and correlation. From these metrics 91 significant features were identified, these features included all those identified by Robertson, Back, and Bartlett (2015) in addition to a variety of features not previously found in the literature. Two separate random forest models were constructed with MOV and match outcome respectively and achieved a prediction accuracy of 88.5% for match outcome and a root mean squared error of 21.4 ± 0.2 for MOV.

Throughout the literature explored above it is clear that while accuracies have improved over time there is a point of diminishing returns where ex-ante prediction is concerned. Standard statistical models produce accuracies of up to 70% (Bailey and Clarke 2006; Ryall 2011) while more complex machine learning methods yield accuracies into the upper 80% range (Leushuis 2018; Young et al. 2019). Whilst each sport will have sport specific features such as the number of kicks or intercepts, features which are similar across various sports are those such as team, ranking, home/away assignment, and win/loss averages over a period. These features are utilised throughout the literature and provide a thorough springboard to expand on current knowledge with the goal of a near real-time predictive model.

More recently researchers have been concerned with team performance and fixture analysis. As stated previously the AFL is inherently subject to bias in its scheduling, with the ramifications of this often offloaded onto coaching staff (Guerrero-Calderon et al. 2021; Lin, Pecotich, and Yap 2011; Rocaboy and Pavlik 2020; Ter Weel 2011; Lenten 2011). In light of this researchers have begun looking into how a team's performance varies relative to their given fixture. Robertson, Back, and Bartlett (2015) identifies various key performance indicators and explores their relative effect on match outcome.

In simplistic terms a key performance indicator is a metric which is correlated directly with a team's success, and whilst most studies have been done outside the scope of Australian Rules Football, specific studies are few and far between. Traditionally performance analysis has been conducted using modelling approaches such as multinomial logistic regression (Stewart, Mitchell, and Stavros 2007; Robertson, Back, and Bartlett 2015), unsupervised machine learning (Robertson, Back, and Bartlett 2015), and principle component analysis (Castellano, Casamichana, and Lago 2012). Results from the above studies are generally concordant and identify metrics that have become synonymous with on-field ‘momentum’ (Taylor and Demick 1994), that is to say on-field actions which allow a team to move the ball further into their opponents territory whilst maintaining ball possession or minimising their opponent’s reacquisition of ground (Hughes and Bartlett 2002). Appendix B.2 lists a complete summary of the performance indicators tracked within Australian Rules Football, with kicks, marks, and handballs being some of the most significant (Robertson, Back, and Bartlett 2015).

Traditionally a team is said to have performed well if they win a match, with varying degrees of success attributed to their MOV or an individuals exemplary on-field performance; but what of factors that are outside a team’s direct purview —a key player injured, a particularly difficult match such that one team outclasses the other. These are some of the many factors which often go unchecked by fans, punters, and investors; and as such has driven researchers to investigate what underpins a fair season or fixture (Lenten 2011). Sportspeople are typically subject to long seasons and are under constant pressure to maintain peak performance, this coupled with frequent travel, injury, and interruptions to training puts a great deal of mental and physical stress on both players and coaching staff. Therefore, coaches and training staff are under constant pressure to micromanage training and rehabilitation plans as to minimise on-field performance loss. A key stratagem often employed is that of tactical periodisation where a team’s on-field composition and training regiments are dynamically varied in preparation for, or in response to matches or events that are considered high priority. A common application of this is colloquially known as ‘tanking’ and is the act of intentionally under-performing in order to prepare for a later more significant event. This is often done by either fielding a weaker team and resting key players or by intentionally losing a match, with the latter typically being met with crowd disappointment (Tuck and Whitten 2013).

Regardless of these tactics it is far more important to ensure that a balanced fixture is enjoyed by all, however, this is easier said than done. Various financial and intra-club factors make it infeasible to achieve. Ideally, a conference structure similar to that of the NFL could be adopted to minimise travel and balance out fixtures but that would place significant strain on clubs and players as it would require a longer season and significant financial investment (Josman, Gupta, and Robertson 2016a). In the AFL’s current incarnation business is conducted in cartel-like fashion with the AFL and its members exercising absolute control over the administration and distribution of the game and its talents (Stewart, Nicholson, and Dickson 2005). As per the current broadcast contract, rights were sold after an offer in excess of \$500 million Australian dollars was made; this includes all pre and in season games as well as the grand final. If schedules were altered it is fair to say that the costs would significantly increase.

2.3 Real-Time Prediction

Real-time prediction is an ever-increasing realm of research within the sporting world —whether seeking to beat the betting market (Bailey and Clarke 2006), or to gain the upper hand on an opponent through a rapid yet efficient system of strategic changes (Gréhaigne and Godbout 2014), there is always impetus to improve be it from coaches, investors, or fans. The approaches and techniques however, vary quite significantly depending on the sport, number of input variables, and frequency at which predictions or outputs are required. Traditionally, researchers have circumnavigated this requirement by segmenting events so that predictions may be made at predetermined intervals during a match, therefore allowing discrete prediction methods to be used at the cost of the granularity afforded by real-time methodologies. For example, Akhtar and Scarf (2012) implement an evolving multinomial logistic regression model to predict the outcome probabilities during a five-day cricket test match. However, due to the limitations of regression models, outputs are required to be generated on a pseudo-real basis and as such are produced at the end of each innings over the course of play. Whilst this approach does allow for the analysis of batting and bowling trends as the match progresses it takes until the end of play on day two to reach comparable results to studies of a similar nature (McHale and Scarf 2011; Stefani and Clarke 1992; Bailey and Clarke 2006). Regardless of this time-lag, models of this nature can enable coaches and captains to tailor their team's batting and bowling strategies with respect to current match prospects. In a similar vein Bailey and Clarke (2006) produce updated MOV targets at the end of each over. This is achieved through the use of standard linear regression in conjunction with the Duckworth-Lewis method (Duckworth and Lewis 2004) and requires very little in terms of input data as the Duckworth-Lewis resource conversion scheme has remained virtually unchanged since its inception in 1999 and subsequent refinement in 2014 as the Duckworth-Lewis-Stern method (Stern 2016).

Moreover, Clarke (1988) further increases the rate at which predictions are generated, thereby producing a prediction at the end of each ball bowled during a one-day cricket match. To implement this methodology a dynamic programming approach was adapted with the objective being to calculate an optimal run rate by which a maximum final run count may be achieved. Applications of this approach are numerous; from allowing captains to dynamically structure batting orders, to performance tracking and measurement of individual players and teams, and even outcome prediction on a ball by ball basis.

Moving further towards a true real-time prediction model Oh, Keshri, and Iyengar

(2015) developed a graph based simulation model for the National Basketball Association. This was achieved through the use of both play-by-play and player location data, and whilst not an outcome prediction approach in the truest sense it allows for the simulation and 'prediction' of a match when provided with a given starting squad for each team. Similarly, both Štrumbelj and Vračar (2012) and Manner (2016) make use of play-by-play data, homogeneous Markov models, and Monte Carlo simulation to project the most probable path that the ball takes over an average number of possessions and henceforth declare a victor over a number of simulations. An advantage of this approach is that the Monte Carlo simulation provides unbiased estimates of the points scored by each team, however, it was also found to overestimate the performance of weaker teams.

2.4 Prediction Methods

Whilst the statistical theory underpinning the following models is discussed in Chapters 4 and 5, the rationale behind each model is to follow.

2.4.1 Multinomial Logistic Regression

Multinomial Logistic Regression (MLogR) is a frequently used classification algorithm (see Section 4.1.1) that adapts logistic regression to multi-class problems. It is both easy to implement and interpret, and allows for identification of feature importance whilst also allowing for easy derivation of bimodal event probabilities. Requirements of the MLogR model are such that the dependent variable is discrete and that there is independence and no multicollinearity amongst the independent variables, with a major drawback being that the data needs to be linearly separable which is rarely found in real-world scenarios.

2.4.2 Logistic Model Tree

Logistic Model Tree (LMT) is a commonly used classification algorithm (see Section 4.1.2), which performs comparatively to other classifiers whilst remaining easy to interpret (Landwehr, Hall, and Frank 2005). LMT combines two popular classification techniques: tree induction, and logistic linear regression, which when used in combination synergise to counteract the other classifiers shortcomings (Hornik, Buchta, and Zeileis 2009). To elaborate linear regression is inherently subject to low variance and high bias while tree induction is subject to high variance and low bias, with the LMT yielding both low variance and bias. At each iteration a decision tree is grown after which linear regression is performed resulting in piecewise logistic linear regression model from which the next iteration is started.

Similar to the MLogR model the LMT requires linearity as well as independence due to its logistic component, whilst not requiring any additional constraints due to the non-parametric nature of the tree induction. This implementation returns a white box model that allows for easy interpretation and still performs relatively well even if the aforementioned assumptions are invalidated. Conversely, the LMT can produce overly complicated trees that do not accurately capture the splits in the data leading to instability. Another consideration is that the LMT will generally become biased if there are structurally dominant features and classes that significantly outweigh others (Landwehr, Hall, and Frank 2005).

2.4.3 Random Forest

Random Forest (RF) is an ensemble method used in both classification and prediction and as such makes use of multiple classification and regression trees that are further integrated using some form of voting or weighting in order to provide a more accurate prediction (see Section 4.1.3). It is widely used due to its inability to overfit, low prediction misclassification rates, and efficiency with large datasets (Breiman 2001; Biau 2012; Zhou, Fenton, and Neil 2014). Algorithmically it can be seen as an extension of the basic bagging methodology which incorporates random feature sampling, with a single iteration of the RF procedure generating a single tree $r(\mathbf{X}, \Theta, \mathcal{F})$.

As a purely non-parametric method there are no underlying distribution assumptions with the RF being able to handle both discrete and continuous data as both dependent and independent variables, it is also able to map complex non-linear relationships. The final forest is an aggregation of trees built throughout the training process and as such there is little to no instability unlike the LMT above. Unfortunately, a major drawback of the RF is a black box model and does not allow one to investigate the inner workings of the model which disallows the interpretation of all but the output.

2.4.4 Support Vector Machine

Support Vector Machine (SVM) is a classification algorithm (see Section 4.1.4) which is often used due to its high accuracy with both large and small datasets, the algorithm attempts to find the best separating hyperplane between two groups within a set of descriptors (Bennett and Bredensteiner 2000). For classification of data with more than two groups the original problem is split into multiple binary problems which are then classified and compared, with the problem having the most votes per instance being assigned as the classifier (Meyer and Wien 2014).

In application the SVM is a highly tuneable algorithm with multiple parameters and the ability to switch between both parametric and non-parametric implementations. Of the various parameters, those of most importance are regularisation, gamma, and kernel; regularisation affects how sensitive the classification is to incorrect classifications, gamma affects the distance at which vectors are considered as members to the hyperplane, and the kernel is a set of functions by which calculations are performed. SVM is very effective when dealing with high dimensional data that has clearly defined classes, however, as it does not directly calculate probabilities it requires additional computational overhead and may not be as efficient as other classifiers (Pisner and Schnyer 2020).

2.4.5 Continuous Time Inhomogeneous Markov Models

A Markov model is a probabilistic graphical model used to represent the changes in a system consisting of random processes (see Section 5.1.1). Probabilistic such that it models the changes in a system consisting of random processes, and graphical in such a way that it is possible to represent the observable set of outcomes on a digraph with nodes made up of a countable set of states belonging to an overarching state space (Howard 2012). The Markov model by nature is able to model a wide variety of discrete and continuous systems including those that are infeasible to classical models and is used in many fields of research —from financial to survival analysis with a major boon being that one is able to graphically display the progress or path taken by the process being studied (Boyd and Lau 1998). As per its name, the model assumes the Markov property, that is to say that future states $\{X_{t+1}\}$ depend only on the current state of the model $\{X_t\}$. However, it should be noted that this adaptability comes at the cost of computational efficiency and the Markov assumption may not be compatible with certain systems.

2.5 Data Sources

This study relies on traditional methods of data collection and as such makes use of final match data and live match transcriptions. Final match data generally consists of final tallies for each of the statistics of interest and can be gathered either team-wise or player-wise, with these statistics being made available shortly after the conclusion of a match and are published in various forms and on multiple platforms. On the other hand live match transcriptions are not easy to come by, this is primarily due to the cost prohibitive nature of obtaining said data, to that effect Champion Data (2017) as the official statistics provider of the AFL provides all live statistics to the AFL and each club within it. These statistics consist of all facets of play on a play-by-play basis, including but not limited to players involved, type of transaction, location on field, and time of transaction.

It is, however, important to understand the overall landscape and current innovations in sports data acquisition and how it pertains to the future of sports analytics. Both final match data and live match data are purely observational and as such lack many contextual identifiers such as location and locomotive metrics. Champion Data (2017) in an effort to remedy this records the absolute quadrant in which a transaction takes place, however, in doing so ignores important spatio-temporal data with regard to the remaining players on the field.

A remedy to this is found in the deployment of Global Positioning System (GPS) devices to athletes. These GPS devices have been shown to be effective in the monitoring and classification of human locomotion in both sporting and casual settings and could open up novel avenues of player tracking and performance metric extraction (Aughey 2011).

2.6 Summary

Following on from the above literature review (summarised in Table 2.1) it is clear that whilst there is an abundance of worked focused on ex-ante outcome prediction, there is still yet work to be done in the realm of real-time outcome prediction.

Ex-ante prediction is implemented in a variety of sports regardless of tempo (the speed at which the sport is played) and is a large part of the currently available literature. Machine learning techniques such as RF and SVM were used to great success for result prediction in both American Football and Athletics.

On the other end of the spectrum different methods of regression and generalised linear models were used to accurately predict match outcome, points scored, MOV, and quantify the effect of home advantage, with results being comparable across both sports and methods.

Due to the cost and difficulty of simultaneous data collection real-time prediction is carried out on slower moving sports (when compared to Australian Rules Football) and those where up to date data is easily available such as cricket and soccer. These applications tend to use less computationally taxing methods such as multinomial linear and logistic regression and rely heavily on pre-established methodologies such as the Duckworth-Lewis resource matrix and existing match strategies.

As comprehensive as the current literature may seem there are some issues which need to be addressed. Firstly, and most importantly, in a statistical and practical sense none of the literature reviewed (bar Clarke (2005)) explicitly defines what the home team is. This is further confounded by the fact that due to factors such as home advantage a match between teams $\mathcal{H}$ and $\mathcal{A}$ is fundamentally different to a match between teams $\mathcal{A}$ and $\mathcal{H}$. Secondly, methods which use objective data are often biased and whilst sometimes more accurate, would require significant extra resources to reliably implement in a real-time scenario.

Table 2.1: Literature Review Summary.

Study	Sport / Activity	Features		Prediction Frequency	Method
		Static	Dynamic		
Akhtar and Scarf (2012)	Cricket	Home team, away team	Lead of reference (home) team, rating difference, home factor, ground effect, home team wicket resources, away team wicket resources	Once at the start of each state of play (start of day, at lunch, at tea)	Multinomial logistic regression
Bailey (2005)	Cricket	Team, score average, class, experience, score average last 10 games, neutral venue, average MOV	Runs scored, wickets taken, remaining overs	At the end of each over	Multiple linear regression, and modified Duckworth-Lewis method
Castellano, Casamichana, and Lago (2012)	Soccer	Goals scored, total shots, shots on target, shots off target, ball possession, number of off-sides committed, fouls received, corners, total shots received, shots on target received, shots off target received, off-sides received, fouls committed, corners against, yellow cards and red cards		At the end of each season	Discriminant analysis
Clarke (1988)	Cricket		Overs remaining, wickets remaining, runs scored	At the end of each ball	Dynamic programming
Clarke (2005)	Australian Rules Football	Team ratings, margin of victory, year, round, home team, away team, ground		At the end of each season	Linear regression
Constantinou (2012)	Soccer	Past performance, current points, subjective points, form, motivation, spirit, fatigue, bookkeeper's odds		Once prior to the beginning of each match	Bayesian network
Crowder et al. (2002)	Soccer		Attack and defence ratings for each team	At the start of each match	AR(1) process
Delen, Cogdell, and Kasap (2012)	American Football (college)	34 features categorised as offence/defence, outcome, team configuration, against the odds, ID features		Once after model creation	Artificial Neural Networks (ANN), Support Vector Machine (SVM), Classification and Regression Tree (CART)
Goddard (2005)	Soccer	25 features categorised as home team attack and away team defence goals covariates, home team defence away team attack goals covariates, home and away team results covariates, other covariates		Once prior to the beginning of each match	Bivariate Poisson regression, ordered logistic regression
Harville (1980)	American Football	Home advantage, team performance level, margin of victory		Prior to each match	Mixed linear models, AR(1) process
Leushuis (2018)	Australian Rules Football	Margin of victory, home team strength, away team strength, home score, away score		At the end of each season	Gaussian state space model, Kalman filter
Lopez and Matthews (2014)	Basketball	Team rating, offence, defence, adjusted offence, adjusted defence, tempo, adjusted tempo, neutral venue, point spread		Once prior to the beginning of the season	Logistic and linear regression
Manner (2016)	Basketball	Match outcome, betting odds	Team strengths and rankings	Before the start of each game	GAR(1) with Kalman filter, state space model

Study	Sport / Activity	Features		Prediction Frequency	Method
		Static	Dynamic		
Maszczyk et al. (2014)	Athletics (javelin)	Cross step with assuming the throwing stance, specific power of the arms and trunk, specific power of the abdominal muscles, grip power		Once after model creation	Artificial Neural Network (ANN), linear regression
McCabe and Trevathan (2008)	Australian Rules Football, Rugby, Soccer	Points for, points against, overall performance, home team performance, away team performance, previous game performance, performance in past n games, team ranking, points for in previous n games, points against in previous n games, location, player availability		Prior to each match	Artificial Neural Network (ANN), Multi-layer perceptron (MLP)
McHale and Morton (2011)	Tennis	Date, player names, rankings, match results, tournament, location, playing surface, tournament importance		Once at the beginning of each tournament week (with data being updated using the prior week's results)	Bradley-Terry type model
Min et al. (2008)	Soccer	Team, location, reputation, skills, teamwork, squad depth, stamina, main formation, sub formation, hard working, aggression, pass length	Formation, overlapping, fatigue, position, pressing, morale, offenders, finishing, defenders, activity level, endurance, offensive grade, defensive grade, possessive grade, fatigue modifier	10 times per game (at intervals of 9 minutes)	Bayesian network and rule-based reasoner
Oh, Keshri, and Iyengar (2015)	Basketball	Offensive team lineup, defensive team lineup, historic player tracking and play-by-play data, average possession time	Propensity to take a shot, ability to deter shot attempt, tendency to pass, shooting ability, defensive ability, ability to draw a shooting foul, foul proneness, defensive rebound ability, offensive rebound ability	After every ball touch	Graphical state model
Pace and Carron (1992)	Hockey	Number of time zones crossed, direction of travel, distance traveled, preparation/adjustment time, time of season, game number on the road trip, home stand		At the end of each season	Multiple linear regression
Robertson, Back, and Bartlett (2015)	Australian Rules Football	Match result, performance indicators (number of kicks, marks, handballs, etc.)		Once after model creation	Logistic regression, decision tree
Rue and Salvesen (2000)	Soccer	Match result, attacking skill, defending skill, goals scored, psychological team effect			Markov chain Monte Carlo, Bayesian dynamic generalised linear model, Brownian motion
Stefani and Clarke (1992)	Australian Rules Football	Rank, team, result, score, home advantage		Once prior to the beginning of each season of play	Least squares and 0.75 power method

Study	Sport / Activity	Features		Prediction Frequency	Method
		Static	Dynamic		
Štrumbelj and Vračar (2012)	Basketball	Home team, away team, Markov transition matrix for five states, effective field goal percentage, free throw factor, turnover ratio, opponent's effective field goal percentage, offensive rebound ratio, opponent's turnover ratio, defensive rebound ratio, opponent's free throw factor		Once prior to the beginning of each game	Homogenous Markov model and multinomial logistic row models
Young et al. (2019)	Australian Rules Football	Match aggregate performance indicators (kicks, handballs, possession ratio, etc.), match outcome, margin of victory			Random forest, segmented regression

CHAPTER 3

Data Acquisition and Processing

The singular most important component to a statistical model, apart from the underlying statistical framework is the data used therein. With the growing interest of both sporting fans and weekend tipsters, online repositories began gaining traction as a primary source of information as early as 1995; however, these repositories are often run and moderated by sporting fans and community members with adjacent interests and may contain anomalous or erroneous data (as often noted by disclaimers to that effect). In contrast to the aforementioned, real-time match and player data may be gathered, though inherent human and capital costs make this impractical for all but corporate entities and the special interest groups that they serve. In short, access to real-time data is primarily limited to AFL clubs through their provider Champion Data (Champion Data 2019), with costs being mostly offset through improvements gained in terms of scenario specific training regiments, and empirically optimised pre and post-match strategic planing.

3.1 Data Sources

As stated above, a major factor in any mathematical model is the quality of data used for both model creation and testing. With the issue of big data and its widespread adoption within the sporting world, it is important that heavy scrutiny be placed upon establishing the quality of data prior to its use. The two types of data utilised for this research can be summarised as follows; static data (known prior to the match) which is widely accessible and can be found on a myriad of online repositories, and dynamic data (gathered during the match) which is restricted to AFL teams and the companies that gather said data. Due to the proprietary nature of the aforementioned dynamic data, the final model has been restricted to matches played by the Western Bulldogs during the 2015 and 2017 AFL seasons.

At its core, the data collected is structured as follows; for a given match m between home team i and away team j a set of feature data $F_t = \{S, D_t\}$ is computed where $S = \{S_i, S_j\}$ and $D_t = \left\{D_{(i,t)}, D_{(j,t)}\right\}$ are the sets of static and dynamic data at time t for teams i and j respectively.

3.1.1 Static Data

Static data refers to all data which for a given match is able to be collected prior to the commencement of said match. Data of this type may include but are not limited to: match location, stadium, official team membership, team rosters, previous match results, and player performance records. From this data it is possible to calculate relevant team-based performance statistics and produce ex-ante outcome predictions using various statistical methods.

The static data utilised in this study were acquired from various sources and fall into one of the following four categories: match data, team rankings, membership numbers, and home grounds. Match data were obtained from AFL Tables (AFL Tables 2017) and contained all information pertinent to each match of the 2001–2017 AFL premiership seasons (see Appendix A.1). This dataset contains but is not limited to home team, away team, venue, season, round, and in cases where the match had already concluded, end of match statistics and result (Tables 3.1 and 3.2).

Table 3.1: Summary of relevant raw categorical AFLTables data.

Statisitc	Description
Season	The season in which a match is played.
Round	The round in which a match is played.
Date	The date on which a match is played.
Local.Start.Time	The time at which a match begins.
Venue	The venue where a match is played.
Home.team	The home team.
Away.team	The away team.

Team rankings were calculated from the aforementioned match data for each season and round similarly to the official AFL league tables, whereby a team is awarded 4 points for a win, 2 points for a draw, and 0 points for a loss, with ties being determined by a team's goal ratio (the ratio of goals for to goals against) (Australian Football League 2015).

Membership numbers and home grounds were obtained from the team summaries published in the 2001 to 2017 AFL annual reports (Australian Football League 2019) and represent the total number of people who were club members during each of the 2001–2017 seasons (Table 3.3). The rationale behind the inclusion of membership numbers is that they act as a reasonable approximation for crowd composition and as such play into two major ideas considered by this study; firstly, it is posited that crowd atmosphere directly affects a team's morale and as such impacts on field performance (Jones and Harwood

Table 3.2: Summary of relevant raw numeric AFLTables data.

Statisitc	Description	Mean	Standard Deviation
Attendance	Total stadium attendance.	23427	17740
X1Q1G	Total number of goals scored by the home team at the end of the first quarter.	3.105	1.882
X1Q1B	Total number of behinds scored by the home team at the end of the first quarter.	3.224	1.98
X1Q2G	Total number of goals scored by the home team at the end of the second quarter.	6.295	2.87
X1Q2B	Total number of behinds scored by the home team at the end of the second quarter.	6.464	2.78
X1Q3G	Total number of goals scored by the home team at the end of the third quarter.	9.517	3.931
X1Q3B	Total number of behinds scored by the home team at the end of the third quarter.	9.708	3.627
X1Q4G	Total number of goals scored by the home team at the end of the fourth quarter.	12.805	4.998
X1Q4B	Total number of behinds scored by the home team at the end of the fourth quarter.	12.937	4.38
Home.Score	Total Total number of points scored by the home team.	89.767	31.561
X2Q1G	Total number of goals scored by the away team at the end of the first quarter.	2.791	1.813
X2Q1B	Total number of behinds scored by the away team at the end of the first quarter.	2.966	1.885
X2Q2G	Total number of goals scored by the away team at the end of the second quarter.	5.669	2.751
X2Q2B	Total number of behinds scored by the away team at the end of the second quarter.	5.957	2.688
X2Q3G	Total number of goals scored by the away team at the end of the third quarter.	8.592	3.787
X2Q3B	Total number of behinds scored by the away team at the end of the third quarter.	8.926	3.524
X2Q4G	Total number of goals scored by the away team at the end of the fourth quarter.	11.519	4.747
X2Q4B	Total number of behinds scored by the away team at the end of the fourth quarter.	11.879	4.223
Away.Score	Total Total number of points scored by the away team.	80.990	29.967

2008; Roane et al. 2004); and secondly, it allows one to decide home and away allocations in the event that neither team is playing at home or both teams are playing at a shared home ground.

Table 3.3: AFL club membership numbers for the years 2001 - 2017.

Team	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Adelaide	42014	46620	47097	45642	43256	50138	50976	48720	46472	45545	44719	45105	46405	53026	52920	54307	56865
Brisbane Lions	18330	22288	24365	30221	28913	26459	21976	22737	24873	26779	20792	20762	24130	23247	25408	23286	21362
Carlton	27725	26385	33525	32095	33534	28756	35431	39360	42408	40480	43791	45800	50564	45911	47305	50130	50326
Collingwood	31455	32549	40455	41128	38612	38038	38587	42498	45972	57408	71271	72688	78427	72170	75037	74643	75879
Essendon	36227	35219	31970	33469	32734	32511	32759	41947	40412	40589	42559	47780	56173	55700	60818	57494	67768
Fremantle	23898	23775	25368	32780	34178	35666	43343	43366	39206	39854	42762	42918	43880	48000	51433	51889	51254
Geelong	25420	23756	24017	25021	30821	32290	30169	36850	37160	40326	39343	40000	42884	40666	44312	50571	54854
Gold Coast	0	0	0	0	0	0	0	0	0	0	11141	11204	12502	12806	13643	12854	11665
Greater Western Sydney	0	0	0	0	0	0	0	0	0	0	0	10241	12681	11696	13480	15312	20944
Hawthorn	30140	33319	31500	31255	29261	28003	31064	41436	52496	53978	56224	60841	63353	65494	72924	75351	75663
Melbourne	22940	20152	20555	20647	24805	24698	28077	32600	31506	33358	36937	35459	33177	33419	35953	39146	42233
North Melbourne	21409	20831	21403	23420	24154	24624	22366	29516	28340	26953	28761	33423	34607	34716	41012	45014	40343
Port Adelaide	33296	36229	35425	36340	36834	35648	34073	34185	30605	29092	32581	35543	39838	46549	54057	53743	52129
Richmond	26501	27251	25101	27133	28029	29406	30044	30820	36981	35960	40184	53027	60321	63486	70809	72278	72669
St Kilda	22248	17696	23626	30534	32043	32327	30394	30063	31906	39021	39276	35440	32707	29332	32746	38009	42052
Sydney	28022	27755	21270	25010	24955	30382	28764	26721	26269	28671	27106	29873	36358	38000	48836	56523	58838
West Coast	38649	34880	36234	40792	42406	44138	45949	44863	43927	44160	43216	57377	58501	51547	60221	65188	65064
Western Bulldogs	19085	20838	21260	19295	21974	26042	28725	28306	28215	32077	29710	30007	30209	26622	35222	39459	47653

Additionally, a comprehensive table of home grounds were collated. This table contains both current and past; major and minor AFL stadia. The need for this is thrice-fold, firstly, to monitor and account for stadium name changes, secondly, to ensure home and away status is correctly assigned for each match, and thirdly, to allow for cross-referencing and data merging between the raw data sets. The AFL since its inception has always been Victoria-centric (Blainey 2010; Pennings 2012) and as such a majority of both teams and stadia are either located in or based out of Victoria, with 51% of stadia and 56% of teams calling Victoria home (Figure 3.1).

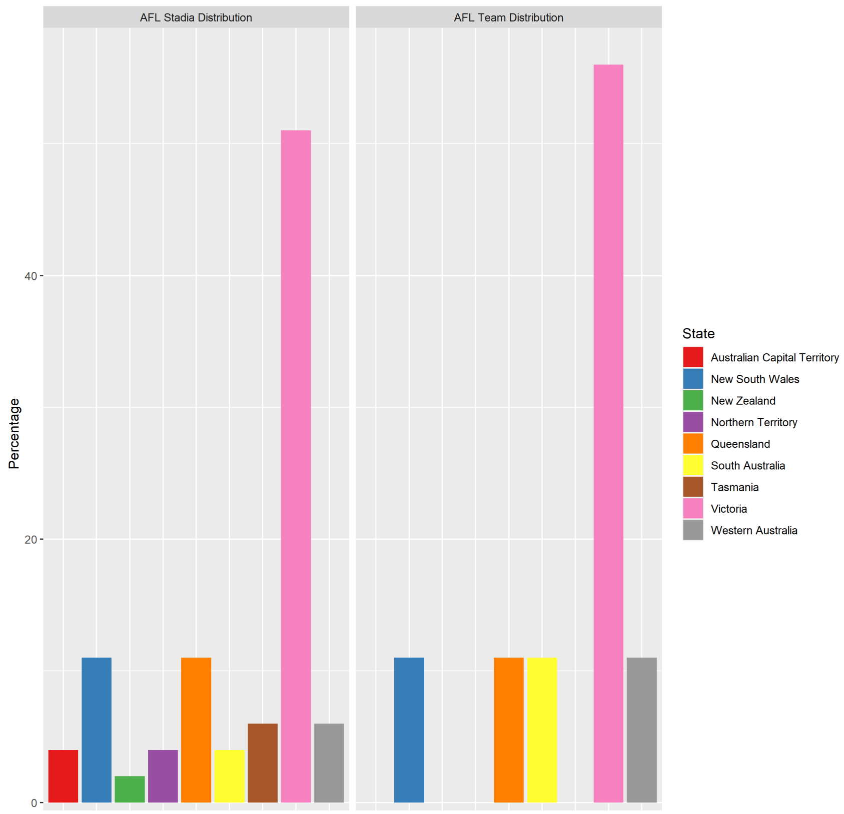


Figure 3.1: AFL Stadia and Team Distributions.

This bias is often the subject of debate (Clarke 2005; Ryall and Bedford 2011; Watson 2013) as it affects many facets of the game, both on and off the field. The most common argument (Duffield and Fowler 2017; Fowler, Duffield, and Vaile 2014; Pace and Carron 1992; Ryall 2011) is that Victorian teams are subject to far more travel, placing greater physical strain on the players who in turn have increased fatigue and less pre-match training time when compared to their non-Victorian counterparts (Stefani and Clarke 1992), potentially having a negative impact on their match performance. However, these assumptions are spurious in nature and continuously fuelled by fan and media speculation. These claims have been refuted in previous years with researchers finding that interstate travel has minimal effect on both sleep quality and performance in Australian Football at the elite level (Richmond et al. 2007).

3.1.2 Dynamic Data

Dynamic data refers to all data which for a given match is collected whilst the match is in progress. Data of this type, within the scope of the Australian Rules Football may include but are not limited to: number of kicks, number of tackles, number of fouls, and number of goals. From this data it is possible to quantify overall team performance and momentum, and as per this research develop near real-time predictive models for match outcome.

The use of dynamic data, no matter how advantageous poses many practical challenges that need to be addressed. Prime amongst these are data acquisition; as a vast majority of data is manually captured an efficient and well-trained staff are required, thus making the capture and processing of data costly. Secondly, as data is manually captured and validated there is the question of data of data accuracy and validity. Not many studies have broached this topic by quantifying the accuracy of data captures, however, Robertson, Gupta, and McIntosh (2016) performed a reliability assessment of Champion Data's data accuracy during a round of the 2014 AFL premiership season finding high agreement (ICC $\in [0.947, 1]$) between the data gathered by Champion Data and that which was manually gathered through video footage by the author.

The dynamic data utilised in this study were acquired from Champion Data (Champion Data 2017) and contained a comprehensive account of all on-field events such as kicks or goals (henceforth known as transactions) complete with timestamps, player, and team data. Each transaction constitutes an epoch for data collection purposes and additionally conditions the model as to when new forecasts may be produced. For example, if only transactions of a single type are observed throughout a match, and said transactions occur at two distinct points in time, all else being equal, it is not possible to produce an intermittent forecast as the interstitial time frames are for all intents and purposes unobserved and therefore 'unknown'. The data contained an exhaustive list of transactions as outlined in Appendix B.2 and as such certain considerations were levied with regard to the subset of data used for this study.

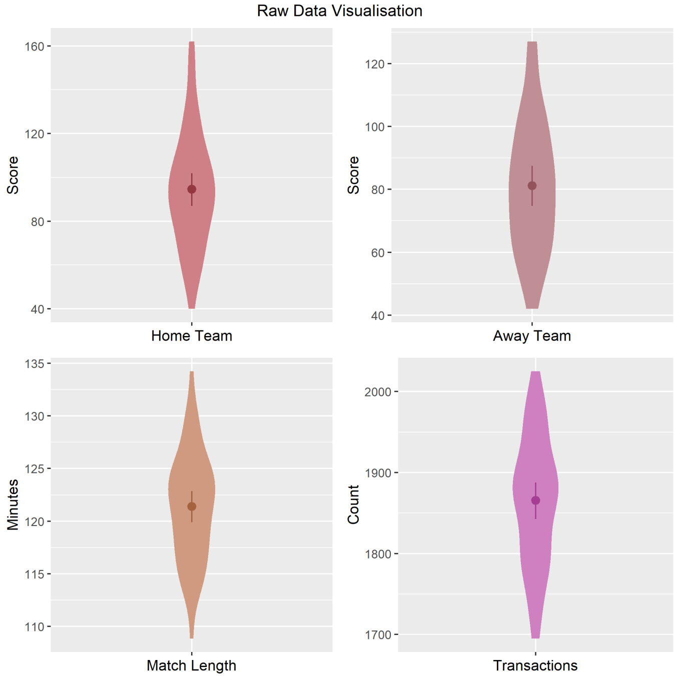
The data contained 172 unique transactions (a total of 339 when coded in reference to the team responsible for said transaction) over a period of 45 matches. On average a single match lasted $\mu = 121.406$ ($\sigma = 5.222$) minutes and consisted of $\mu = 1865.4$ ($\sigma = 80.192$) transactions. Furthermore, the score profiles for both home and away teams are as follows; for an average match the home team scores $\mu = 94.578$ ($\sigma = 26.571$) points; whereas the away team scores $\mu = 81.133$ ($\sigma = 21.130$) points (Figure 3.2).

3.2 Data Processing

3.2.1 Static Data

Static data collected from AFL Tables (2017) were truncated to include information pertaining to all 3289 matches played during the 2001 — 2017 seasons. This data was pre-processed such that the home and away team allocations provide an unbiased derivation of the home and away team assignments for each match (Stefani and Clarke 1992).

Figure 3.2: Raw Data Visualisation.



Guided by the research of Jones and Harwood (2008) and Ryall (2011) this methodology removes both inter and intra-club biases introduced through economic and political agendas, and instead relies on stadium conditions (location and crowd composition) which have been shown to directly affect player performance.

As such, for any pairing of teams $(i, j) \in (\mathcal{H}, \mathcal{A})$ in a given match m, the home and away teams are defined as follows; if either team i or j are playing at their home ground then assign the home team accordingly, however, if both teams i and j share the same home ground or are both playing an away game then assign the home team to that team which has the highest official membership number. The rationale behind this is that whilst crowd attendance numbers are available there is no real way to determine crowd composition, to that effect membership numbers are used as a proxy for crowd proportions and as a metric to decide the home team when a match is played at a neutral venue. This revised team assignment yields a home team win probability of 0.607 which whilst slightly higher than average holds with the paradigm of home advantage outlined by Stefani and Clarke (1992) and Clarke (2005). Additional statistics relating to team form and performance (Margin, Head2Head, PastHome, and PastAway) were then calculated (Equations 3.1—3.4) such that for a given match m between home and away teams i and j, the result $R_{(i,j,m)} = 1$ if team $\mathcal{H}$ wins and $R_{(i,j,m)} = 0$ if team $\mathcal{H}$ loses.

- **Margin:** The score margin by which the home team either won or lost the match.

$$Home.score - Away.score \tag{3.1}$$

- **Head2Head:** The percentage of games for which the home team has won against the away team (over the past k games), prior to match m.

$$\frac{\sum_{g=m-k}^{m-1}(R_{i,j,g})}{k} \tag{3.2}$$

- **PastHome:** The percentage of games for which the home team has won against any opponent (over the past l games), prior to match m, where $R_{i,g} = 1$ if team i won its last g^{th} match and $R_{i,g} = 0$ if team i lost its last g^{th} match.

$$\frac{\sum_{g=m-l}^{m-1}(R_{i,g})}{l} \tag{3.3}$$

- **PastAway:** The percentage of games for which the away team has won against any opponent (over the past l games), prior to match m, where $R_{j,g} = 1$ if team j won its last g^{th} match and $R_{j,g} = 0$ if team j lost its last g^{th} match.

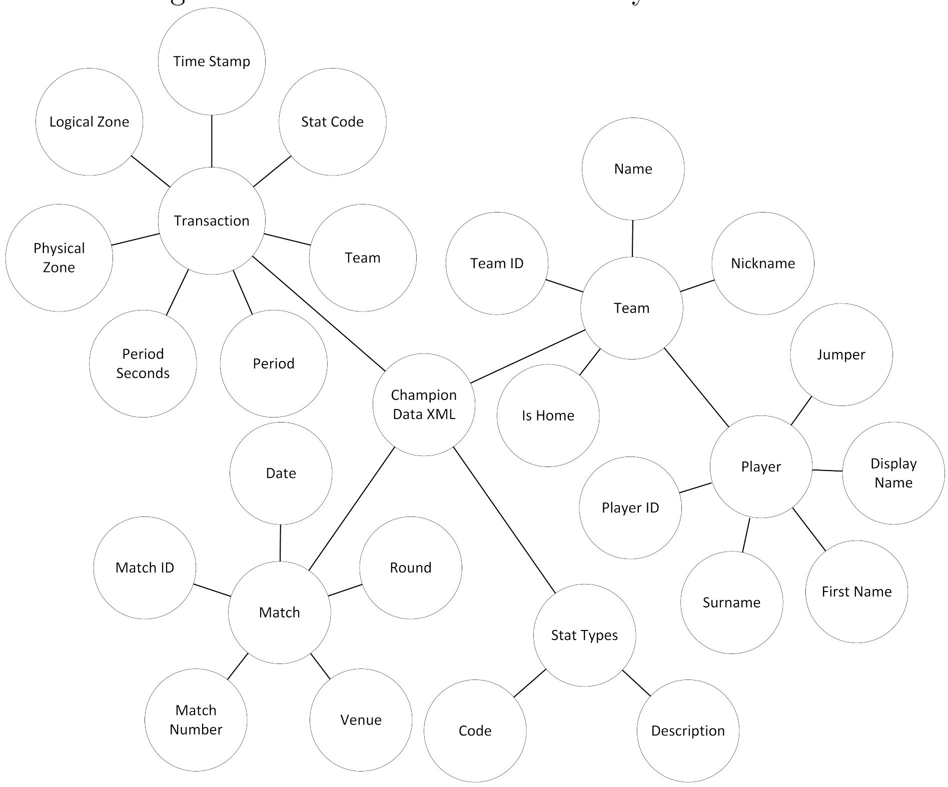
$$\frac{\sum_{g=m-l}^{m-1}(R_{j,g})}{l} \tag{3.4}$$

3.2.2 Dynamic Data

Dynamic data (see Appendix B.1 for a complete summary) collected from Champion Data (2017) were supplied in two formats: the 2015 data contained within multiple Extensible Markup Language (XML) files, and the 2017 data contained within a single comma-separated value (CSV) file. This data was extracted from both XML and CSV formats using processing routines (see Appendices C.1 and C.2) written in R (R Core Team 2018). Due to the difference in format across the two file types additional steps were needed to standardise the data and correct for any parsing inconsistencies, most notably the 2017 data included a far more granular set of transactions which had to be recoded in order to conform with the broader transaction definitions contained within the 2015 data. In addition, all duplicate epochs (an epoch such that the possessive team, player, transaction, and time are the same) were removed, with the final set of extracted data having a layout as per figure 3.3 an explanation of which can be found in Appendix B.3.

Due to the unpredictable structure of AFL match durations, the time code data (quarter and quarter time in seconds) were restructured as a single vector representing the overall time in minutes for each match. Additionally, in instances such that multiple yet different transactions occur in the same epoch (for example, the start of a match and a centre bounce), an offset was added to each relevant time code by $\Delta = 0.0001\delta$, where δ is incremented by 1 for each offending epoch. This alteration to the data is necessary

Figure 3.3: Structure of extracted dynamic data.



as the utilised Markov model (Chapter 5) is both state based and temporally dependent, that is to say, for a transaction to be observed at time t the preceding transaction at time $t-1$ must be known, with a single epoch unable to contain more than one transaction.

3.3 Feature Selection

As an aside and to further facilitate the discussion on the various features extracted from the data described in sections 3.3.1 and 3.3.2 a brief overview of the expected output of each class of model is warranted. The aim of each static model is the prediction of match outcomes probabilities with respect to the home team, whilst the aim of the dynamic model is to forecast match outcomes with respect to the home team after observing a portion of a match and allowing for various metrics of team performance and momentum in addition to prior team and venue knowledge.

3.3.1 Static Features

In order to model and predict match outcome probabilities across a variety of teams, stadia, and match conditions, a robust set of static features need to be selected. These features therefore need to both span and accurately capture key pre-match criteria. The features described below therefore show a subset of those available which, after a thorough examination of the literature and talks with industry personnel were deemed to most significantly influence match outcome (Robertson 2018, Interview with Western Bulldogs staff. April 24; Wilson 2020, Interview with Champion Data staff. February 14).

The season (represented as a year) and round in which a match is played (represented as an integer value associated with the relative week in which a match is played) are both integral in understanding a given matches place in time. Primarily, season and round, when used in conjunction replace the need for date coded data and as such facilitate the grouping of matches with respect to their relative seasonal stage as opposed to the date on which a match is played. The reason for this is twofold: firstly, due to logistical and financial reasons it is highly infeasible for all teams to play their respective matches on the same day, and secondly, there is an increasing correlation between a team's rank at the end of a given round and that of said team's rank at the end of that rounds corresponding season (Robertson and Joyce 2015).

The finals indicator (represented as a binary indicator with 0 indicating the home and away season and 1 indicating the finals series) when used in conjunction with season and round provides insight into potential strategies that may be employed by either team for a given match during a particular round. This becomes more apparent when utilised in tandem with the custom measures of team performance outlined in equations 3.2—3.4 as well as team rankings, and can potentially identify episodes of 'tanking' or intentional poor performance of a team when they are guaranteed a place in the finals series (Tuck and Whitten 2013).

The venue at which a match is played is primarily used to determine home and away team assignments (with home grounds and membership numbers as published by the AFL (Australian Football League 2019)) and augments the idea of home advantage by identifying venue bias with relation to crowd composition, travel distance, and team preference (Ryall 2011; Clarke 2005; Carbone, Corke, and Moisiadis 2016; Lenten 2011). Additionally, the ranks of both home and away teams (stylised as HomeRank and AwayRank respectively) are used in lieu of team names and provide an adequate facsimile of both a team's end of round ranking and relative strength (provided no significant alteration to a team's composition due to injury or strategy).

Finally, the home team's win percentage over the past k matches (stylised as Head2Head), the home team's win percentage over the past l matches (stylised as PastHome), and the away team's win percentage over the past l matches (stylised as PastAway) were inspired by those used in NFL and baseball, and are used as significant indicators of team form and performance (Delen, Cogdell, and Kasap 2012; Leung and Joseph 2014). These may also be used in evaluating a team's psychology and potential strategies prior to a match (Ryall 2011; Jones, Mellalieu, and James 2004; Jones and Harwood 2008; Taylor and Demick 1994).

3.3.2 Dynamic Features

As with the static features discussed previously, it is of even greater importance to capture only dynamic features which provide the most detailed overview of a team's performance and momentum during a match. The raw data provided by Champion Data (2017) contains well over 150 transactions per team, with each transaction falling into one of the following categories; possession, offence, defence, accuracy, scoring, or play reset.

To gather so much data so quickly Champion Data have developed a support system which operates simultaneously at both the stadium and Champion Data's own 'The Bunker'. From within the stadium there are four main roles; match caller, matchups caller, support/IT, and interchange operator, and within 'The Bunker' there are five main roles; back caller, graphical operator, keyboarder, pressure capturer, and pressure caller. The match caller observes the match via binoculars and reports every transaction as it occurs to the keyboarder back at 'The Bunker', all while being assisted by their own support/IT person who listens to the umpire's call and assists the media. The matchups caller observes and records the position of persons relevant to each transaction, and the interchange operator watches for and records player interchanges.

The keyboarder inputs all the basic transaction statistics whilst the back caller double checks the calls made against the ground caller to identify any possible miscalls. The graphical operator records and maps the exact on-field position of players and transactions. Finally, the pressure capturer and pressure caller provide annotated insights into the other facets of each transaction, for example, pressure on disposal, what foot was used to kick a ball, etc. (Champion Data 2017).

Due to the large number of available features and in order to not oversaturate the model, various individual transactions have been combined to form descriptive transaction groupings (Table 3.4). These groups will often combine transactions from multiple categories provided that a synergy exists between them. For example, the time (in seconds) at which a transaction occurs combines all play reset information (period start, period end, and centre bounce) whilst also integrating the quarter in which a transaction takes place.

The most thorough method for feature selection would be to train and evaluate models for every possible feature combination and compare various metrics of model fit and performance. Due to the exhaustive set of potential dynamic features (see Appendix B.1), a stepwise additive approach was adopted for model building whereby features were added in line with their prevalence in the literature as well as after consultation with industry representatives.

The preselected list of transactions described in table 3.4 were selected in line with current studies and constitute a practically acceptable subset of the most significant trans-

Table 3.4: List of Dynamic Transactions

CODE	DESCRIPTION	Notes	Classification
BEHI & RUSHN & RUSHO & RUSHP	Behind and Rushed Behind	1 Point - Merge to Behind	Scoring
BUCL & TICL & CBCL	Ball Up, Throw In, and Centre Bounce Clearance	Merge to Clearance	Possession
BUHO & BUHSK & BUHSD & BUSM & BUHAD & TMBUH & TMBSD & TMBUS & TMBUA & CBHO & CBHSK & CBHSD & CBSM & CBHAD & TIHO & TIHSK & TIHSD & TISM & TIHAD & TMTIH & TMTSD & TMTIS & TMTIA	Ball Up, Centre Bounce, and Throw In Hitout	Merge to Hitout	Possession
CEBO	Centre Bounce		Play Reset
FRAGN & FRAGO & FRAGP & FRABB	Free Against	Merge to Frag	Defensive
FRFO & FRFBB & FRFNO & FRFOB	Free For		Offensive
GOAL	Goal	6 Points	Scoring
HBEF	Handball Effective		Offensive
HBIN	Handball Ineffective		Offensive
HBRE	Handball Received		Offensive
IN50	Inside 50		Possession
KILO & KILA & KISH & KISE & KBLO & KBSH & KKBW & KKGKE & KKLO & KKLA & KKSH	Kick In and Kick Effective	Merge to Kicks	Offensive
KKIN	Kick Ineffective		
MACOO & MACOP & MAUNO & MAUNP	Mark Contested and Uncontested	Merge to Marks	Offensive
PEREN	End Of Period	End of Quarter	Play Reset
PERST	Start Of Period	Start of Quarter	Play Reset
RE50	Rebound 50m		Offensive
SPOI & SPOIO & SPOIP & SPOIG	Spoil		Defensive
TACKN & TACKO & TACKP	Tackle		Defensive

Each variable described in the table above (excluding CEBO, PEREN, and PERST) are recorded individually for each team equating to $16 \times 2 + 3 = 35$ transaction events which have been preselected for their significance as descriptors with relevance to a winning team.

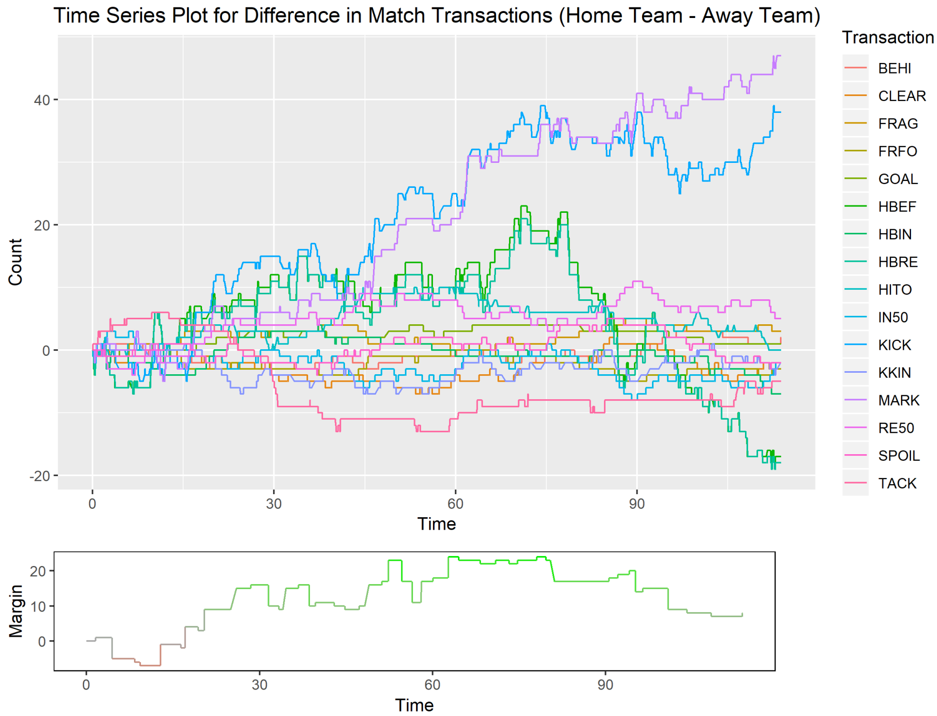
actions with relation to win probability (Robertson, Back, and Bartlett 2015; Hughes and Bartlett 2002; Young et al. 2019). This relationship can be clearly seen in figure 3.4 which is taken from match 8 of the 2017 AFL premiership season in which the West Coast Eagles played the Western Bulldogs at home. The most rudimentary performance metric, the score margin (Equation 3.1) acts as a generalised facsimile for all facets of play, incorporating both team's offensive and defensive performance.

A team's ability to score a goal or behind provide a far deeper understanding of an attacking team's momentum in relation to the defending team's defence. To score a behind, a team needs to display greater possession of the ball (particularly within their opponent's inside 50) as well as superior offence and efficient use of the ball (O'Shaughnessy 2006).

Possession is the term used to describe seizure and control of the ball by a team and is generally held to be a good identifier of strategic or strength imbalances between the teams. Hence, the team who possesses the ball longer will have a dominating field presence and be far more likely to be in a winning position (Casal et al. 2019).

Of the numerous possession metrics available, this study utilises 3 clearance, 23 hitout, and inside 50 metrics which are further consolidated and recoded as their parent metric type. A clearance is the clearing of the ball out of a stoppage situation such that a particular team retains possession at the continuance of play; a hitout is the act of knocking the ball out of the ruck contest following a stoppage with clear control, regardless of which side wins the following contest at ground level; and an inside 50 is the act of running or passing the ball inside the 50m arc.

Figure 3.4: Snapshot of Iterative Transaction Differences.



The next set of metrics explored are those which may be classified as offensive, the model makes use of 4 free for, handball effective, handball ineffective, handball received, 12 kick, 4 mark, and rebound 50 metrics. Transactions classified as free for represent instances where a possession of the ball given to a player as a result of an infringement by an opposition player. Handball effective is when a handball to a teammate that hits the intended target, whereas handball ineffective is a handball which is not advantageous to the team, but does not directly turn the ball over to the opposition, and a handball received is as uncontested possession that is the result of a teammate's handball.

Similar to the handball metrics the 12 kick metrics used are variations along the lines of effective, ineffective, and received. A mark is a clean catch of the ball after it has been kicked by another player (either by a teammate or by the opposition), before it has touched the ground, or been touched by any other player, and after it has travelled a minimum of 15 metres. A rebound 50 is the act of moving the ball from a team's own defensive zone into the midfield.

Finally, the defensive metrics used by the model are 4 free against, 4 spoil, and 3 tackle metrics. A free against is when an infringement occurs resulting in the opposition receiving a free kick from the umpires. A spoil is a punch or slap of the ball which hinders an opposition player from taking a mark. A tackle is the grabbing of an opposition player in possession of the ball, in order to impede their progress or to force them to dispose of the ball quickly.

Finally, the transactions grouped together as 'kick' are a significant component of momentum and constitute an array of possession, offence, and accuracy metrics. For example, a kick long advantage (stylised as KILA but grouped as KICK) is a kick of more than 40 meters which ends in possession by a team-mate and contributes to a team's possession and accuracy.

3.4 Summary

This chapter introduced the data used in this research. Data was gathered primarily from AFL Tables (2017) and Champion Data (2017) and contained well over 100 variables and 339 match time transactions. Each match time transaction or on-field event occurs within an epoch constituting a period of observed on-field play. These data were initially reduced to 59 variables including 35 match time transactions, after which through the elimination of confounding variables and clustering for similar transaction types resulted in 14 variables including 4 match type transactions (A.BEHI, H.BEHI, A.KICK, and B.KICK).

The data from AFL Tables (2017) were collected in CSV format whilst the data from Champion Data (2017) were collected in both XML (2015 season) and CSV (2017 season) formats. Significant processing and cleaning were required in order to collate the data into a singular dataset from which all analysis was conducted.

It is also of great importance to note that the data obtained from Champion Data (2017) were captured during actual league matches and as such any results can be seen as practically viable as opposed to simulatory. However, there is a drawback as unlike simulated data which can be homogeneously created and replicated, the observed data is inhomogeneous and as such whatever models are employed must be able to handle data of irregular time series.

Tables 3.5 and 3.6 contain summaries of all static and dynamic variables contained within the final dataset.

Table 3.5: List of Static Variables

Variable	Type	Description
Result	Discrete	Result indicator
Season	Discrete	Season in which match is played
Round	Discrete	Round in which match is played
Finals	Discrete	Indicator as to whether the match is part of the home and away or finals series team
Venue	Discrete	Match venue
HomeRank	Discrete	Current ladder rank for the home team
AwayRank	Discrete	Current ladder rank for the away team
Head2Head	Continuous	Home team's win percentage over past m games against away team
PastHome	Continuous	Win percentage over past n games
PastAway	Continuous	Win percentage over past n games

Table 3.6: List of Dynamic Variables at Each Epoch

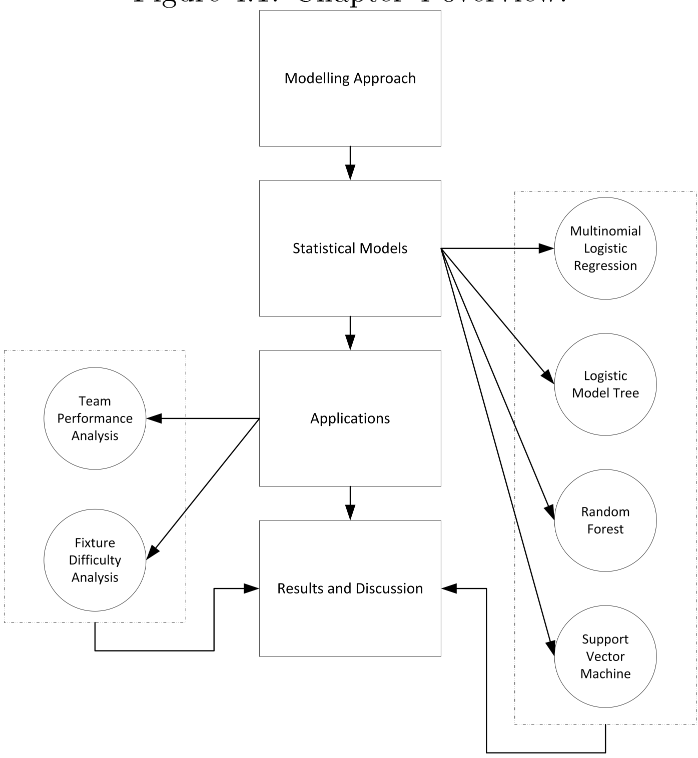
Variable	Merged Transactions	Type	Description
A.BEHI H.BEHI	BEHI, RUSHN, RUSHO, & RUSHP	Continuous Continuous	Number of behinds scored by the away team Number of behinds scored by the home team
A.KICK H.KICK	KIKIN, KKEF, KILO, KILA, KISH, KISE, KBLO, KBSH, KKBW, KKGKE, KKLO, KKLA, & KKSH	Continuous Continuous	Number of kicks executed by the away team Number of kicks executed by the home team

CHAPTER 4

Static Prediction Models

The main objective of this thesis is to devise a framework for near real-time AFL match outcome prediction which is predicated on the use of both static and dynamic match data. However, as current literature on the subject is rather lacking, it is advantageous to develop a deep understanding of ex-ante methodologies and in turn contribute novel applications which shall form a cornerstone of this thesis' true objective. As such, this chapter presents the following (Figure 4.1); a modelling approach for the prediction of ex-ante match outcomes; a brief summary and formulation of the models used within; the results of training and optimising said models, and their applications.

Figure 4.1: Chapter 4 overview.



4.1 Static Models

Mathematically a predictive model for the outcome of a match between team $\mathcal{H}$ (home) and team $\mathcal{A}$ (away) can be defined as $C(F) = f\left(S, S^{\mathcal{H}}, S^{\mathcal{A}}\right)$ where $\left\{S, S^{\mathcal{H}}, S^{\mathcal{A}}\right\} = (S_1, S_2, \ldots, S_a, S_1^{\mathcal{H}}, S_2^{\mathcal{H}}, \ldots, S_b^{\mathcal{H}}, S_1^{\mathcal{A}}, S_2^{\mathcal{A}}, \ldots, S_b^{\mathcal{A}})$ are the values of the a match specific static features and $2b$ team specific static features, and C is a representation of the predicted outcome probability for a match with respect to the home team prior to the game's start.

$$C(F) = \Pr(\mathrm{Draw}, \mathrm{Loss}, \mathrm{Win}) \tag{4.1}$$

with $f(\cdot)$ being an unknown function to be estimated using the statistical methods outlined in subsections 4.1.1–4.1.4, and the static components $\left\{S, S^{\mathcal{H}}, S^{\mathcal{A}}\right\}$ of feature set F as described in Chapter 3 subsection 3.3.1

4.1.1 Multinomial Logistic Regression

Multinomial Logistic Regression (MLogR) is a generalised linear model commonly used in both multi-class classification and probability prediction problems and has many benefits such as robustness when dealing with large feature sets (be they categorical, ordinal, or numerical), and the ability to incorporate dynamic (non-stationary) features (Penny and Roberts 1999). It is an extension of the logistic regression model which provides classification and probability prediction results for bimodal data (Hosmer Jr, Lemeshow, and Sturdivant 2013), and takes the general form for each level j of Y

$$C_j(F) = \Pr(Y = j \mid F) = \frac{e^{\sum_{i=0}^{n} \beta_{i,j} F_i}}{1 + e^{\sum_{\ell=1}^{J-1} \sum_{i=0}^{n} \beta_{i,\ell} F_i}} \tag{4.2}$$

for $j = 1, 2, \ldots, J-1$ and

$$C_J(F) = \Pr(Y = J \mid F) = \frac{1}{1 + e^{\sum_{\ell=1}^{J-1} \sum_{i=0}^{n} \beta_{i,\ell} F_i}} \tag{4.3}$$

for the last level J, which under a logit transformation becomes

$$\ln\left[\frac{C_j(F)}{C_J(F)}\right] = \beta_{0,j} + \beta_{1,j} F_1 + \ldots + \beta_{n,j} F_n \tag{4.4}$$

whereafter predictions are derived through the setting of a threshold value which aims to maximise the classification rate (Equation 4.5). This value serves as a cut-off point for assigning classifications to the probabilistic output of the model and is calculated as the point of intersection between the model's sensitivity ($\Pr(\hat{y}_i = 1 \mid y_i = 1)$) and specificity ($\Pr(\hat{y}_i = 0 \mid y_i = 0)$).

$$cv = \max\left(\Pr\left(\hat{y}_i = 1 \mid y_i = 1\right) \bigcap \Pr\left(\hat{y}_i = 0 \mid y_i = 0\right)\right) \tag{4.5}$$

Assuming the above holds for the MLogR, the process of fitting the model is as follows (Hosmer Jr, Lemeshow, and Sturdivant 2013; Neath and Johnson 2010); with observations assumed to be independent, the likelihood function is defined as

$$l\left(\beta\right) = \prod_{i=1}^{n} \prod_{j=1}^{J-1} \left(\frac{C_j\left(F\right)}{C_J\left(F\right)}\right)^{Y_{i,j}} C_J\left(F\right)^{n_i} \tag{4.6}$$

with its logit transform becoming

$$\ln\left(l\left(\beta\right)\right) = \sum_{i=1}^{n} \sum_{j=1}^{J-1} \left(Y_{i,j} \sum_{k=0}^{K} \beta_{k,j} F_{i,k}\right) - n_i \ln\left(1 + \sum_{j=1}^{J-1} e^{\sum_{k=0}^{K} \beta_{k,j} F_{i,k}}\right) \tag{4.7}$$

from this equation it is possible to derive $(J-1)(K+1)$ individual likelihood equations, one for each parameter $\beta_{k,j}$. These are then solved by taking the second order partial derivatives of the log-likelihood function

$$\frac{\partial^2 l\left(\beta\right)}{\partial \beta_{k,j} \partial \beta_{k',j'}} = -\sum_{i=1}^{n} n_i F_{i,k} C_j\left(F\right)\left(1 - C_j\left(F\right)\right) F_{i,k'}, \quad j' = j \tag{4.8}$$

and

$$\frac{\partial^2 l\left(\beta\right)}{\partial \beta_{k,j} \partial \beta_{k',j'}} = \sum_{i=1}^{n} n_i F_{i,k} C_j\left(F\right) C_{j'}\left(F\right) F_{i,k'}, \qquad j' \neq j \tag{4.9}$$

4.1.2 Logistic Model Tree

An LMT is simply a decision tree formed using the LogitBoost algorithm (Friedman, Hastie, Tibshirani, et al. 2000) with logistic regression (Hosmer Jr, Lemeshow, and Sturdivant 2013) at each node. The C4.5 splitting criterion is used to improve the purity of each node, with nodes containing fewer than 15 cases becoming terminal nodes. Algorithmically the LogitBoost performs a forward stage-wise fitting; such that during each iteration, a variable z_{ij} is computed as to capture the error of the model for its respective training data (Algorithm 1).

Mathematically a LMT is a tree containing a set of non-terminal nodes $\mathfrak{N}$ and a set of terminal nodes $\mathfrak{T}$ such that $\mathfrak{S} \in \{\mathfrak{N}, \mathfrak{T}\}$ and spanned by all data features. The tree is therefore split such that

$$\mathfrak{S} = \bigcup_{\mathfrak{t} \in \mathfrak{T}} \mathfrak{S}_{\mathfrak{t}}, \; \mathfrak{S}_{\mathfrak{t}} \bigcap \mathfrak{S}_{t'} = \varnothing \text{ for } \mathfrak{t} \neq \mathfrak{t}'$$

Algorithm 1 LogitBoost algorithm (J classes) (Friedman, Hastie, Tibshirani, et al. 2000).

Input: Weights $w_{ij} = \frac{1}{n}$, $i = \{1, \ldots, n\}$, $j = \{1, \ldots, J\}$, $H_j(x) = 0$ and $p_j(x) = \frac{1}{J}\ \forall j$,

1: **for** $m = \{1, \ldots, M\}$;

a: **for** $j = \{1, \ldots, J\}$;

b: Compute working responses and weights for the j$^{\text{th}}$ class

$$z_{ij} = \frac{y^*_{ij} - p_j(x_i)}{p_j(x_i)(1 - p_j(x_i))}$$

$$w_{ij} = p_j(x_i)(1 - p_j(x_i))$$

c: Fit $h_{mj}(x)$ by a weighted least-squares regression of z_{ij} to x_i with weights w_{ij}

2: Set $h_{mj}(x) \leftarrow \frac{J-1}{J}\left(h_{mj}(x) - \frac{1}{J}\sum_{k=1}^{J} h_{mk}(x)\right), H_j(x) \leftarrow H_j(x) + h_{mj}(x)$;

3: Update $p_j(x) = \frac{e^{H_j(x)}}{\sum_{k=1}^{J} e^{H_k(x)}}$;

4: **return** the classifier $\operatorname{argmax}_j H_j(x)$

Output: A LogitBoost decision tree

As per algorithm 1, each leaf $\mathfrak{t} \in \mathfrak{T}$ has a logistic regression function h_j rather than a class label. the logistic regression function h_j incorporates a subset $F' \in F$ of all features in the data, and each class probability calculated as

$$\Pr(G = j \mid X = x) = \frac{e^{H_j(x)}}{\sum_{k=1}^{J} e^{H_k(x)}} \tag{4.10}$$

where

$$H_j(x) = \alpha_0^j + \sum_{k=1}^{m} \alpha^j_{F'_k} F'_k \tag{4.11}$$

however, if $\alpha^j_{F'_k} = 0$ for $F'_k \in F$

$$h_j(x) = \sum_{\mathfrak{t} \in \mathfrak{T}} h_{\mathfrak{t}}(x)\, I(x \in \mathfrak{S}_{\mathfrak{t}}) \tag{4.12}$$

where

$$I(x \in \mathfrak{S}_{\mathfrak{t}}) = \begin{cases} 1 \text{ if } (x \in \mathfrak{S}_{\mathfrak{t}}) \\ 0 \text{ else} \end{cases} \tag{4.13}$$

4.1.3 Random Forest

Mathematically the process of building a RF involves the construction of a predictor $C(F) = \{r_n(\mathbf{X}, \Theta, \mathcal{F}_n), m > 1\}$ containing a set of randomised classification and regression trees such that $\mathbf{X}$ is the feature set, $\Theta = \{\Theta_1, \Theta_2, \ldots, \Theta_m\}$ is a randomised response vector consisting of i.i.d. outputs of a response variable Θ, and $\mathcal{F}_n$ is the training set (Biau

2012). This ensemble hence yields an expectation of

$$\overline{r_n}(\mathbf{X}, \mathcal{F}_n) = \mathbb{E}_\Theta \left[r_n (\mathbf{X}, \Theta, \mathcal{F}_n) \right] \tag{4.14}$$

Algorithm 2 Random tree algorithm (Zhou 2012).

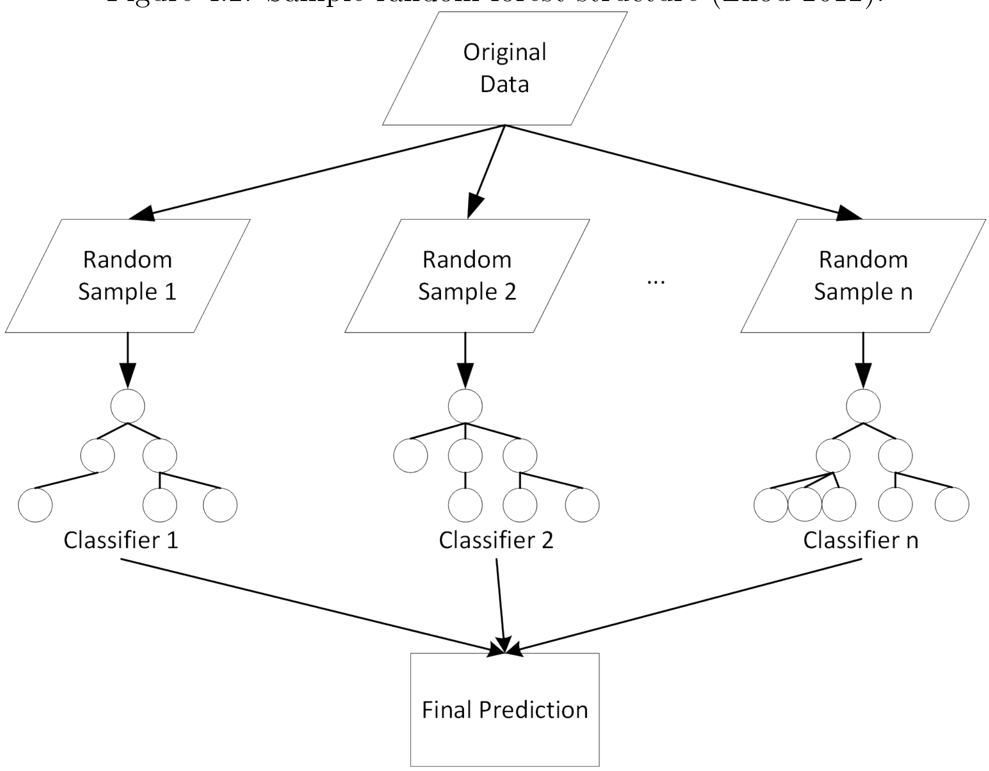
Input: Feature set $\mathbb{F} = \{F, y\} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, Feature subset size K.

```
N ← create a tree node based on 𝔽;
if all instances in the same class then return N
ℱ ← the set of features that can be split further;
if ℱ is empty then return N
ℱ̂    select K features from ℱ randomly;
N.f    the feature which has the best split point in ℱ̂;
N.p    the best split point on N.f;
𝔽_t    subset of 𝔽 with values on N.f smaller than N.p;
𝔽_u    subset of 𝔽 with values on N.f no smaller than N.p;
N_t ← call the process with parameters (𝔽_t, K);
N_u    call the process with parameters (𝔽_u, K);
return N
```

Output: A random decision tree

where $\mathbb{E}_\Theta$ is the expectation generated through majority voting, conditionally on $\mathbf{X}$ and the training set $\mathcal{F}_n$.

Figure 4.2: Sample random forest structure (Zhou 2012).



Each randomised tree (Figure 4.2) yields the average over all sampled response vectors Y_i for which the corresponding feature set $\mathbf{X}_i$ fall within the same random partition $A_n(\mathbf{X}, \Theta)$ containing $\mathbf{X}$,

$$r_n(\mathbf{X}, \Theta) = \frac{\sum_{i=1}^{n} Y_i \mathbf{1}_{[\mathbf{X}_i \in A_n(\mathbf{X}, \Theta)]}}{\sum_{i=1}^{n} \mathbf{1}_{[\mathbf{X}_i \in A_n(\mathbf{X}, \Theta)]}} \mathbf{1}_{\mathcal{E}_n(\mathbf{X}, \Theta)}, \tag{4.15}$$

where $\mathcal{E}(\mathbf{X}, \Theta)$ is defined as

$$\mathcal{E}_n(\mathbf{X}, \Theta) = \left[\sum_{i=1}^{n} \mathbf{1}_{[\mathbf{X}_i \in A_n(\mathbf{X}, \Theta)]} \neq 0 \right]. \tag{4.16}$$

with the final expectation taking the form of

$$\overline{r_n}(\mathbf{X}) = \mathbb{E}_\Theta \left[r_n(\mathbf{X}, \Theta) \right] = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{\sum_{i=1}^{n} Y_i \mathbf{1}_{[\mathbf{X}_i \in A_n(\mathbf{X}, \Theta)]}}{\sum_{i=1}^{n} \mathbf{1}_{[\mathbf{X}_i \in A_n(\mathbf{X}, \Theta)]}} \mathbf{1}_{\mathcal{E}_n(\mathbf{X}, \Theta)} \right]. \tag{4.17}$$

4.1.4 Support Vector Machine

Support Vector Machine (SVM) is a non-probabilistic classification model which is often used due to its high accuracy with both large and small datasets. This method attempts to find the best separating vector (or hyperplane) between two groups (or classes) within a set of descriptors (Bennett and Bredensteiner 2000) However, for problems with more than two classes, such as the one presented in this research, the original problem is split into multiple pairwise binary problems (Prakash et al. 2012) which are then classified and compared, with the problem having the most votes per instance being assigned as the predicted classifier (Meyer and Wien 2014).

For a given set of M training points (x_i, y_i), $i = \{1, 2, \ldots, M\}$, with x_i and y_i being the input vector and class of interest respectively. Where y_i takes the value of 1 for a positive case and -1 for a negative case. In order to calculate the desired hyperplane u is required such that

$$u = \vec{w}.x - b \tag{4.18}$$

where $\vec{w}$ is a normal vector to the separating hyperplane, x is the input vector, and the separating hyperplane is where $u = 0$. From equation 4.18 the parallel hyperplanes can be derived when $u = \pm 1$, with the margin m defined as

$$m = \frac{1}{\|w\|^2} \tag{4.19}$$

In order to maximise the margin in equation 4.19 various optimisation techniques may be used to derive the support vectors, thereafter the normal vector $\vec{w}$ and threshold b can be calculated as

$$\vec{w} = \sum_{i=1}^{N} y_i \alpha_i \vec{x_i},\ b = \vec{w}.\vec{x_i} - y_k\ \forall \alpha_k > 0 \tag{4.20}$$

where α is known as the Lagrange Multiplier, and the output of the SVM is computed as the sum of Lagrangian Multipliers

$$u = \sum_{j=1}^{N} y_j \alpha_j K\left(\vec{x_j}, \vec{x}\right) - b \tag{4.21}$$

Within this research, the datasets which are utilised are non-linear, with a "kernel trick" being applied in order to map the input space into the feature space, thus creating a hyperplane in the feature space. The kernel is therefore a function which allows such a mapping, due to the structure of the feature data implemented in this model a Radial Basis Function (RBF) is used as the kernel and is given by equation 4.22.

$$K\left(x, x_i\right) = e^{-\left(\frac{1}{\sigma^2}|x-x_i|^2\right)} \tag{4.22}$$

4.1.5 Model Settings in R

All analyses were conducted on a computer with a 64-bit Windows operating system, Intel® Core™ i7-7700K processor, and 32GB RAM. Results were obtained using routines and algorithms (see Appendix D) written in the statistical computing package R (R Core Team 2018) which makes use of the packages listed in table 4.1. All models were

Table 4.1: Static model packages.

Method	Package	Author
MLogR	nnet	Venables and Ripley (2002)
LMT	RWeka	Hornik, Buchta, and Zeileis (2009)
RF	randomForest	Liaw and Wiener (2002)
SVM	e1071	Meyer and Wien (2014)

formulated with the response variable set to match outcome (Draw, Loss, Win) and explanatory variables set to those outlined in chapter 3 section 3.3.1, a verbose output of said model function can be seen in equation 4.23.

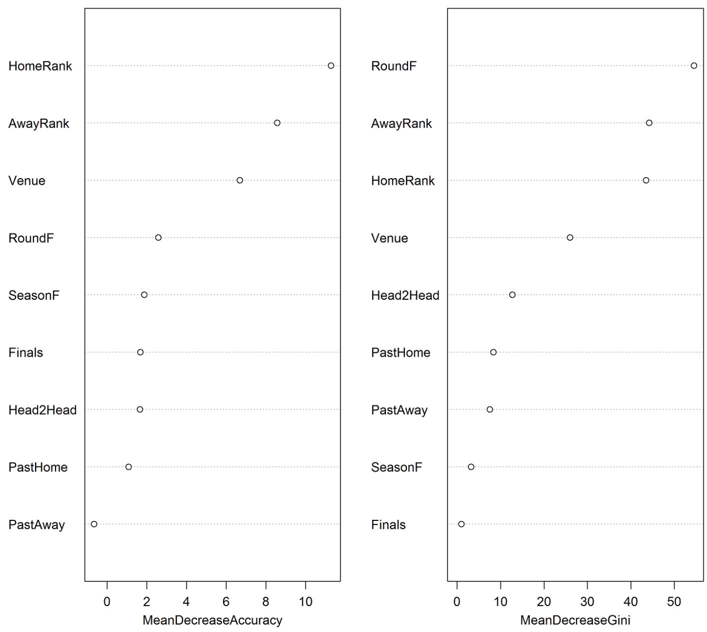
$$\begin{aligned} Result \sim Venue + Finals + Head2Head + PastHome + PastAway \\ +HomeRank + AwayRank + SeasonF + RoundF \end{aligned} \tag{4.23}$$

The MLogR model; being the simplest computationally; was run in a standard configuration. However, for each model iteration a customised threshold value (known here as a cut-off value or *cv*) was calculated in order to most efficiently parse the probabilistic output of the model to an outcome result.

The LMT model; grows trees according to the LogitBoost algorithm (Algorithm 1), with the optimal number of logistic regression iterations at each node determined using 5-fold cross validation. With splits containing additional nodes added if and only if the new node contains; greater than 15 cases, at least two subsets of two cases each, and attains an information gain score above a certain threshold.

The RF model; is a classification and regression tree method without pruning, and as such was configured to grow a forest of 500 trees. With feature sampling at each node set to 3 random features with replacement. Additionally Variable importance was calculated in terms of both Gini index and node purity (Figure 4.3). From this it is possible to see that home rank, away rank, venue, and round are most important in terms of both Gini index and node purity with only slight variations in their relative position.

Figure 4.3: Random forest variable importance.



The SVM model; having the most configurable set-up was initialised with a radial kernel (Equation 4.22), and iteratively tuned for cost and gamma hyperparameters in the ranges of $[1, 10]$ and $[10^{-6}, 1]$ respectively. In terms of the practical implications tuning the values of cost and gamma; cost dictates how much the model is penalised for similar data points within groups, and gamma parametrises the radial kernel's Gaussian distribution in terms of standard deviation.

4.1.5.1 Model Tuning

In order to get optimal results from the substantial static dataset, each model was run 234 times; 13 times for each data span combination, 6 times for each potential value of match span $\{k \mid k \in \mathbb{Z}, 5 \leq k \leq 10\}$, and 3 times for each possible value of match span $\{l \mid l \in \mathbb{Z}, 3 \leq k \leq 5\}$; where k represents the number of past games considered when looking at a pair of team's head-to-head match history and l represents the number of past games when looking at a team's overall match history. The data span was of particular interest in this research as with the ever growing supply of static data, it is important to know at what point each model reaches diminishing returns in terms of accuracy as a result of too much or too few training data. As such starting at the 2001 and ending at the 2014 AFL premiership seasons, the included training data started at a full thirteen-year span (2001¯2014) and was pruned by a year at each iteration down to a two-year span $(2013 - 2014)$, and then tested on the 2015 AFL premiership season. A graphical depiction of the accuracies obtained for each of these data spans and values of k and l, as well as a summary of the maximum accuracies attained under each data span can be seen in figure 4.4 and table 4.2 respectively.

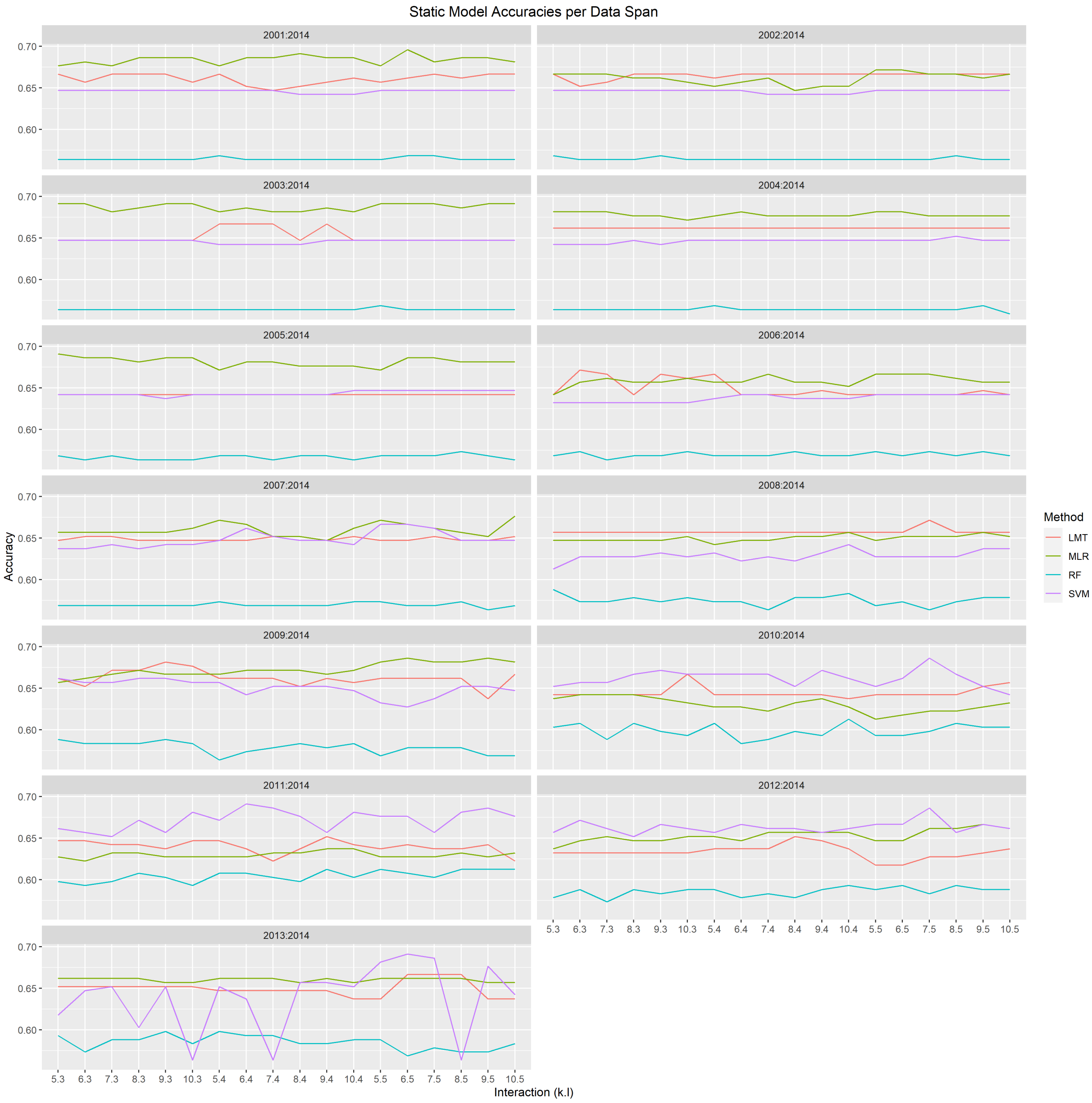
From this data we can glean the following; as the scope of data used to train a model decreases, so decreases said model's consistency. That is to say, a decrease in data span yields lower average accuracies and an increase in overall variability across each value of k, l, and data span.

Table 4.2: Optimal Results per Data Span.

Accuracy	Method	KValue	LValue	Data Span
0.696	MLogR	6	5	2001:2014
0.672	MLogR	5	5	2002:2014
0.691	MLogR	10	3	2003:2014
0.681	MLogR	5	3	2004:2014
0.691	MLogR	5	3	2005:2014
0.672	LMT	6	3	2006:2014
0.676	MLogR	10	5	2007:2014
0.672	LMT	7	5	2008:2014
0.686	MLogR	6	5	2009:2014
0.686	SVM	7	5	2010:2014
0.691	SVM	6	4	2011:2014
0.686	SVM	7	5	2012:2014
0.691	SVM	6	5	2013:2014

In order to determine optimal values for data span, k, and l, a sensitivity analysis was performed. Using these parameters, results were then iteratively calculated with models trained and tested as previously discussed. Results were then assessed using

Figure 4.4: Model Accuracies per Values of k and l for Each Data Span.



analysis of variance (Table 4.3) which indicated the following; there is no significant interaction between parameters k and l ($F\,(10, 867) = 0.771, p = 0.658$) and as such main effects for each parameter can be discussed, conversely there is a significant interaction between the method used for modelling and the amount of data supplied to the model ($F\,(36, 867) = 59.512, p < 0.000$) preventing the analysis of main effects for method and data span at this stage.

The main effects of k and l are not significant with ($F\,(5, 867) = 0.495, p = 0.780$) and ($F\,(2, 867) = 2.588, p = 0.076$) respectively, and as such hold little sway over the predictive power of each model.

Table 4.3: ANOVA for model input variations.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Method	3	1.083	0.361	5053.509	0.000
k	5	0.000	0	0.495	0.780
l	2	0.000	0.000	2.588	0.076
Data	12	0.012	0.001	14.022	0.000
k:l	10	0.001	0.000	0.771	0.658
Method:Data	36	0.153	0.004	59.512	0.000
Residuals	867	0.062	0.000		

4.1.5.2 Model Evaluations

The aforementioned models have been evaluated using RMSE (root-mean-square error) and computation time in seconds. The RMSE is defined by equation 4.24 while computation time is simply the time taken in seconds to train and obtain a prediction using each method, and is additionally used as a measure of model practicality and as a potential tiebreaker in case of similar accuracies between methods.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{n}} \tag{4.24}$$

In terms of model performance, model accuracy is inversely proportional to RMSE. And as such a model with a lower RMSE is preferable to one with a higher RMSE. Hence, when choosing a model and its corresponding parameters, one that minimises RMSE is ideal. The minimum RMSE across all models is 0.5513 which corresponds to the parameters listed in table 4.4. Isolating each model iteration for the parameters listed in table 4.4 gives us a set of four optimal models, one for each of the tested methods with differing accuracies and computation times (Table 4.5).

From the optimal parameter configuration of $k = 6$, $l = 5$, and data spanning 14 years from 2001 to 2014, the method which yields optimal results is the MLogR with

Table 4.4: Optimal model parameters based on minimum RMSE.

k	l	Data Span
6	5	2001:2014

Table 4.5: Optimal model parameters, results, and evaluation statistics.

Method	k	l	Data Span	Accuracy	RMSE	Computation Time
MLogR	6	5	2001:2014	0.696	0.551	0.425
RF	6	5	2001:2014	0.569	0.657	2.793
LMT	6	5	2001:2014	0.662	0.582	1.986
SVM	6	5	2001:2014	0.647	0.594	252.406

an accuracy of 0.696, a computation time of 0.425 seconds, and a significantly good fit at a 5% level of significance as determined by the Hosmer-Lemshow test (Hosmer Jr, Lemeshow, and Sturdivant 2013) ($\chi^2_{12} = 4.705, p = 0.967$). This configuration is 6.571, 4.672, and 593.896 times faster than the RF, LMT, and SVM respectively. The significant discrepancy between the computation times of MLogR, RF, LMT, and those of SVM can be attributed to the need to tune the SVM for the cost and gamma hyperparameters prior to the fitting of the final model.

4.2 Applications of Static Models

As with all sports there is a certain level of unhappiness when it comes to the way in which a league is administered, this mainly stems from how a season is scheduled; with factors such as travel, venue, and relative opposition strength being among the most common concerns. The current AFL ladder scoring system (which can also serve as a proxy for a team's overall performance) has remained unchanged since the inception of the Victorian Football League (VFL) in 1897 and produces point totals which are heavily favoured towards teams which win more matches regardless of match difficulty (Figure 4.5). In other words, currently, a team will be awarded four points for a win, two points for a draw, and deducted zero points for a loss regardless of opposition - where surely it makes more sense for a team to have points awarded and deducted proportionally to the difficulty of the match being played (Aldous 2017; Csató 2020).

Making use of the optimal model configuration in subsection 4.1.5.2, this section of the research sought to develop a methodology to objectively quantify both team performance and fixture difficulty. There are currently no formally documented methods within the AFL to quantify such things, and as such, firstly, inspired by elements of the ELO rating system (Hvattum and Arntzen 2010) and probabilistic Bradley-Terry type models (McHale and Morton 2011) a method was developed to quantify team performance, and

Figure 4.5: Density Plots for Current and Proposed Point Models.

secondly, inspired by rank differentials and Bernoulli simulation (Law, Kelton, and Kelton 1991) a method was developed to not only simulate the outcome of a given season, but to also quantify the mathematically perceived difficulty of said season.

4.2.1 Team Performance Analysis

A team's ladder score and by proxy their performance (and anecdotally that of their coach) $\mathbb{P}_{\mathbb{T},y}$ is currently defined as

$$\mathbb{P}_{\mathbb{T},y} = \sum_{m=1}^{22} \mathcal{P}_{\mathbb{T},m,\mathcal{Y}} \tag{4.25}$$

where $\mathcal{P}_{\mathbb{T},m,\mathcal{Y}}$ is the point value awarded to team $\mathbb{T}$ after match m during season $\mathcal{Y}$. As previously discussed, the canonical value of $\mathcal{P}_{\mathbb{T},m,\mathcal{Y}}$ is 4 for a win, 2 for a draw, and 0 for a loss. Presented here are two models which rely on outcome probabilities obtained by first building an MLogR prediction model $C(F)$ as outlined in section 4.1 and using optimal method and parameter configurations discussed in subsection 4.1.5.2.

The first model, named the Static Performance Model (SPM) assigns points according to the difficulty (assessed as probability of winning) of a given match. If a team wins they are awarded $\min\left(25, \frac{1}{C(F)}\right)$ points, while if they lose they are awarded $\max\left(-25, -\frac{1}{1-C(F)}\right)$ points.

The second model, named the Variable Performance Model (VPM) makes use of the same probabilistic model $C(F)$ as well as parameters specifying both point and probability thresholds. In addition, it is also possible to modify the above SPM to allow for more granulated control of the weighting between wins and losses. Where $\mathfrak{p}$ and $\mathfrak{q}$ are the upper and lower probability thresholds and $\mathfrak{p}_1$ and $\mathfrak{p}_2$ are the upper and lower point thresholds. After which the predicted outcome probabilities $C(F)$ generated by the optimal model specified in section 4.1.5.2 and combined with varying parameters $\mathfrak{p}, \mathfrak{q}, \mathfrak{p}_1$, and, $\mathfrak{p}_2$ provide a methodology where a team is assigned points according to the match difficulty (Table 4.6).

Table 4.6: Match difficulty template.

Match Difficulty	Probability Ranges
Easy	$C(F) < \mathfrak{q}$
Average	$\mathfrak{q} \leq C(F) \leq \mathfrak{p}$
Difficult	$C(F) > \mathfrak{p}$

The aforementioned VPM model is now formulated as follows; if a team wins, the points they are awarded are defined as

$$\mathcal{P}_{\mathbb{T},m,y} = \begin{cases} \mathfrak{p}_1 & \text{if } C(F) < \mathfrak{q} \\ \mathfrak{p}_2 & \text{if } C(F) > \mathfrak{p} \\ \frac{1}{C(F)} + \mathfrak{p}_1 & \text{otherwise} \end{cases} \tag{4.26}$$

likewise, if a team loses, the points they are awarded are defined as

$$\mathcal{P}_{\mathbb{T},m,y} = \begin{cases} -\mathfrak{p}_1 & \text{if } C(F) > \mathfrak{p} \\ -\mathfrak{p}_2 & \text{if } C(F) < \mathfrak{q} \\ -\frac{1}{1-C(F)} - \mathfrak{p}_1 & \text{otherwise} \end{cases} \tag{4.27}$$

A sensitivity analysis was conducted on the VPM for parameters $\mathfrak{p} \in \{0.9, 0.8, 0.7, 0.6, 0.5\}$, $\mathfrak{q} \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, $\mathfrak{p}_1 \in \{5, 6, 7, 8, 9, 10, 11, 12\}$, and $\mathfrak{p}_2 \in \{0, 1, 2, 3, 4, 5\}$; after which the results were then analysed using ANOVA. The results of the ANOVA (Table 4.7) indicate that, unsurprisingly a change in team significantly changes the team performance rating $(F(2080.86, 17), p < 2e - 16)$, however the maximum and minimum point parameters $\mathfrak{p}_1$ and $\mathfrak{p}_2$ are not significant and there are no three and four-way interactions between the parameters with each combination yielding a p-value of 1. The parameters of importance are therefore the maximum and minimum probability thresholds $\mathfrak{p}$ and $\mathfrak{q}$, and any two-way interaction that contains one or the other. From this, optimal parameters of $\mathfrak{p} = 0.7$, $\mathfrak{q} = 0.3$, $\mathfrak{p}_1 = 12$, and $\mathfrak{p}_2 = 5$ were chosen for use in the VPM model.

Table 4.7: VPM ANOVA.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Team	17	11950251	702956	2080.86	<0.00
$\mathfrak{q}$	4	1085164	271291	803.06	<2e-16
$\mathfrak{p}_1$	7	0	0	0	1
$\mathfrak{p}$	4	1085164	271291	803.06	<2e-16
$\mathfrak{p}_2$	5	0	0	0	1
$\mathfrak{q} : \mathfrak{p}_1$	28	167384	5978	17.7	<2e-16
$\mathfrak{q} : \mathfrak{p}$	16	0	0	0	1
$\mathfrak{p}_1 : \mathfrak{p}$	28	167384	5978	17.7	<2e-16
$\mathfrak{q} : \mathfrak{p}_2$	20	92991	4650	13.76	<2e-16
$\mathfrak{p}_1 : \mathfrak{p}_2$	35	0	0	0	1
$\mathfrak{p} : \mathfrak{p}_2$	20	92991	4650	13.76	<2e-16
$\mathfrak{q} : \mathfrak{p}_1 : \mathfrak{p}$	112	0	0	0	1
$\mathfrak{q} : \mathfrak{p}_1 : \mathfrak{p}_2$	140	0	0	0	1
$\mathfrak{q} : \mathfrak{p} : \mathfrak{p}_2$	80	0	0	0	1
$\mathfrak{p}_1 : \mathfrak{p} : \mathfrak{p}_2$	140	0	0	0	1
$\mathfrak{q} : \mathfrak{p}_1 : \mathfrak{p} : \mathfrak{p}_2$	560	0	0	0	1
Residuals	20383	6885785	338		

4.2.2 Fixture Difficulty Analysis

The difficulty $\mathcal{D}_{\mathbb{T},\mathcal{R}}$ of a season for a given team $\mathbb{T}$, starting the season at rank $\mathcal{R}$ can be defined using one of two models formulated within this research. The previous season ranking model (PSR) which is a simple linear style model, and the season ranking simulation model (SRS) which is predicated on the principles of the MLogR model described in subsection 4.1.1.

The difficulty $\mathcal{D}_{\mathbb{T},\mathcal{R}}$ derived from the PSR model is defined as the sum of the differences in the ranking in ranking between the reference team (the team whose difficulty is being calculated) and their opponents during their 11 home and 11 away games (hg and ag respectively) during a given season.

$$\mathcal{D}_{\mathbb{T},\mathcal{R}} = \sum_{hg=1}^{11} (\mathcal{R}_{\mathbb{T},hg} - \mathcal{R}_{\mathcal{A},ag}) + \sum_{ag=1}^{11} (\mathcal{R}_{\mathbb{T},ag} - \mathcal{R}_{\mathcal{H},ag}) \tag{4.28}$$

Scores are then approximated as standard random variables as per equation 4.29 by setting both mean and standard deviation as the arithmetic mean and range of $\mathbb{A}_{\mathbb{T},\mathcal{R}}$ and $\mathbb{B}_{\mathbb{T},\mathcal{R}}$ respectively, where $\mathbb{A}_{\mathbb{T},\mathcal{R}}$ and $\mathbb{B}_{\mathbb{T},\mathcal{R}}$ are the minimum and maximum possible difficulty ratings for a given team and starting rank (with fixtures as outlined by the AFL Commission) respectively, with values less than 0 indicating an easier than average season and vice versa.

$$\mathcal{D}^*_{\mathbb{T},\mathcal{R}} = \frac{\mathcal{D}_{\mathbb{T},\mathcal{R}} - \mu_{\mathbb{T},\mathcal{R}}}{\sigma_{\mathbb{T},\mathcal{R}}}, \text{ where } \mu_{\mathbb{T},\mathcal{R}} = \frac{\mathbb{A}_{\mathbb{T},\mathcal{R}} - \mathbb{B}_{\mathbb{T},\mathcal{R}}}{2}, \text{ and } \sigma_{\mathbb{T},\mathcal{R}} = \mathbb{B}_{\mathbb{T},\mathcal{R}} - \mathbb{A}_{\mathbb{T},\mathcal{R}} \tag{4.29}$$

The AFL Commission (Australian Football League 2015) have outlined the following guidelines for the setting of fixtures (accurate as of the 2015 AFL season); each team is to play 22 games over a period of 25 weeks with each team playing each other team at least once. Teams ranked 1 to 6 at the beginning of the season will then play either 2 or 3 additional games against other teams ranked 1 to 6, either 1 or 2 additional games against teams ranked 7 to 12, or either 0 or 1 additional games against teams ranked 13 to 18. Teams ranked 7 to 12 at the beginning of the season will then play either 1 or 2 additional games against teams ranked 1 to 6, either 2 or 3 additional games against other teams ranked 7 to 12, or either 1 or 2 additional games against teams ranked 13 to 18. Teams ranked 13 to 18 at the beginning of the season will then play either 0 or 1 additional games against teams ranked 1 to 6, either 1 or 2 additional games against teams ranked 7 to 12, or either 2 or 3 additional games against other teams ranked 13 to 18.

From the above guidelines it is possible to generate a list (Table 4.8) of maximum ($\mathbb{B}_{\mathbb{T},\mathcal{R}}$) and minimum ($\mathbb{A}_{\mathbb{T},\mathcal{R}}$) difficulty rating values for each team given their starting rank and number of scheduled games $G^{\min}_{\mathbb{T},j}$ and $G^{\max}_{\mathbb{T},j}$ against team j, where $G^{\min}_{\mathbb{T},j}$ and $G^{\max}_{\mathbb{T},j}$ are the easiest and hardest sets of scheduled games respectively.

$$\mathbb{A}_{\mathbb{T},\mathcal{R}} = 22\mathcal{R}_{\mathbb{T}} - \sum_{\substack{j=1 \\ j\neq\mathbb{T}}}^{18} \mathcal{R}_j G^{\min}_{\mathbb{T},j} \tag{4.30}$$

$$\mathbb{B}_{\mathbb{T},\mathcal{R}} = 22\mathcal{R}_{\mathbb{T}} - \sum_{\substack{j=1 \\ j\neq\mathbb{T}}}^{18} \mathcal{R}_j G^{\max}_{\mathbb{T},j} \tag{4.31}$$

The SRS model is a hybrid simulation model combining aspects of result prediction, Bernoulli simulation, linear regression, and heuristic clustering. Using this model the difficulty $\mathcal{D}_{\mathbb{T},\mathcal{R}}$ is derived as the difference between a team's rankings $\mathcal{R}_{\mathbb{T},\mathcal{Y}}$ at the end of the current and previous seasons.

$$\mathcal{D}_{\mathbb{T},\mathcal{R}} = \mathcal{R}_{\mathbb{T},\mathcal{Y}} - \mathcal{R}_{\mathbb{T},\mathcal{Y}-1} \tag{4.32}$$

A team's current ranking $\mathcal{R}_{\mathbb{T},\mathcal{Y}}$ is obtained by first building a classification model as per subsection 4.1.1, and from the obtained win probabilities the season's results are obtained through a Bernoulli simulation conducted 10000 times. $X \sim \text{Bern}\left(1, C\left(F\right)\right)$ such that a

Table 4.8: Fixture difficulty distribution values per starting rank.

Start of Season Rank	Easiest Rating $\mathbb{A}_{\mathbb{T},\mathcal{R}}$	Hardest Rating $\mathbb{B}_{\mathbb{T},\mathcal{R}}$	Mean Rating $\mu_{\mathbb{T},\mathcal{R}}$	SD (Range) $\sigma_{\mathbb{T},\mathcal{R}}$
1	-200	-172	-186	28
2	-177	-148	-162.5	29
3	-154	-124	-139	30
4	-131	-100	-115.5	31
5	-107	-77	-92	30
6	-83	-54	-68.5	29
7	-74	-43	-58.5	31
8	-51	-19	-35	32
9	-28	5	-11.5	33
10	-5	28	11.5	33
11	19	51	35	32
12	43	74	58.5	31
13	54	83	68.5	29
14	77	107	92	30
15	100	131	115.5	31
16	124	154	139	30
17	148	177	162.5	29
18	172	200	186	28

team is awarded 4 points for a win, with the total number of points being averaged over all trials within the simulation. Differences are then calculated as outlined above with negative differences indicating an easier season and vice versa. Teams are then clustered using heuristic clustering in order to group teams with similar season difficulties.

4.2.3 Results and Discussion

A cursory look at the win probabilities generated by the MLogR model (Figure 4.6) would indicate that teams such as Carlton and Melbourne have the hardest season and teams such as Adelaide and Fremantle have the easiest season. However, as is the nature of a competitive game such as the Australian Rules Football, the team with the easiest season does not necessarily perform the best.

Tables 4.9 and 4.10 present the results from the SPM and the VPM respectively, whilst graphical representations may be found in figure 4.7. Via the SPM; Richmond and Western Bulldogs occupy the top two positions while Brisbane Lions and St Kilda the bottom two, with the VPM yielding similar results with Richmond and Sydney occupying the top two positions. The Spearman's rank correlation test revealed no significant difference between the rankings generated with either the SPM or VPM and a moderately positive correlation ($\rho = 0.395, n = 18, p = 0.105$).

Figure 4.6: Per match win probabilities for the 2015 AFL season.

As there is no significant difference in the rankings predicted by the SPM and VPM, further analysis was conducted using the VPM. Hence, figure 4.8(a) depicts a team's actual rank at the end of the 2015 season as opposed to the rank predicted through the VPM, with 7, 6, and 5 teams performing better, worse, and at parity respectively.

Whilst this would indicate that only 5 out of 18 teams were accurately predicted, it is important to be note that any errors herein have been compounded over 10000 simulations and yet the maximum observed error bound is a relatively low 4 ranks with and 11 out of 18 results occurring within an error bound of $[-1, 1]$. In addition figure 4.8(b) makes use of expectation theory to calculate the expected number of points awarded to each team at the end of the season in addition to the points awarded by the simulation. From this it can be seen that teams above the line $y = x$ are predicted to preform better than expected as per the initial MLogR model and VPM simulation, with 10 and 8 teams predicted to perform better and worse than expected respectively.

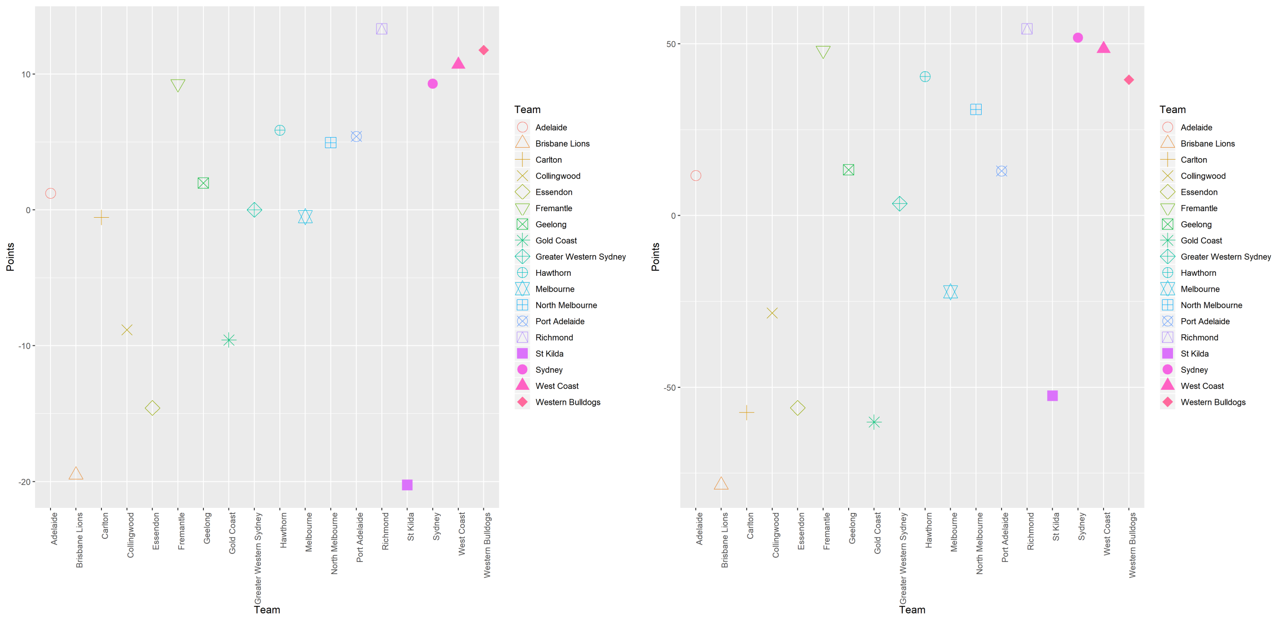
Tables 4.11 and 4.12 present the fixture difficulty results (PSR and SRS respectively) for each team during the 2015 AFL premiership season, contrary to out cursory analysis the PSR model predicts St Kilda and Geelong to have the easiest and hardest seasons respectively. However, it can be seen that these difficulty ratings are outliers and can be attributed to the simplistic nature of the model. Another observation that can be made

Table 4.9: SPM results for the 2015 AFL season.

Team	Points
Richmond	13.324
Western Bulldogs	11.772
West Coast	10.699
Sydney	9.296
Fremantle	9.281
Hawthorn	5.864
Port Adelaide	5.412
North Melbourne	4.949
Geelong	1.981
Adelaide	1.219
Greater Western Sydney	0.009
Melbourne	-0.495
Carlton	-0.557
Collingwood	-8.830
Gold Coast	-9.578
Essendon	-14.583
Brisbane Lions	-19.511
St Kilda	-20.254

Table 4.10: VPM results for the 2015 AFL season.

Team	Points
Richmond	54.321
Sydney	51.759
West Coast	48.493
Fremantle	48.099
Hawthorn	40.450
Western Bulldogs	39.539
North Melbourne	30.884
Geelong	13.304
Port Adelaide	12.981
Adelaide	11.652
Greater Western Sydney	3.459
Melbourne	-22.203
Collingwood	-28.355
St Kilda	-52.477
Essendon	-55.995
Carlton	-57.351
Gold Coast	-60.119
Brisbane Lions	-78.443



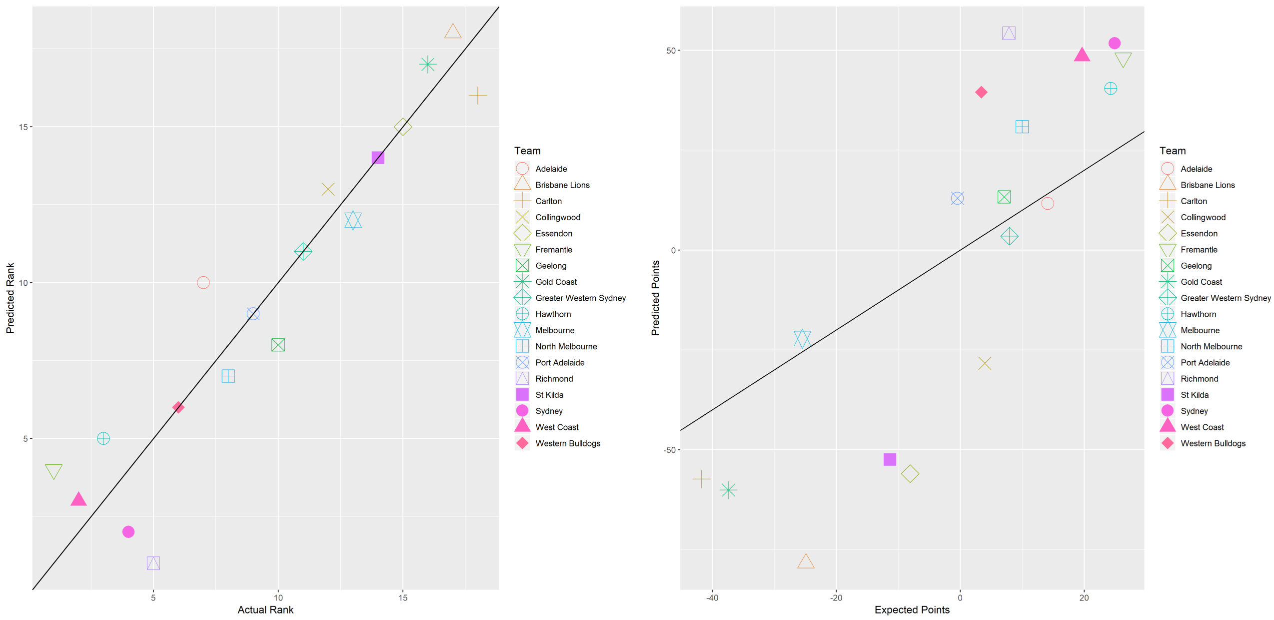
(a) SPM team performance.

(b) VPM team performance.

Figure 4.7: Team performance results for the 2015 AFL season.

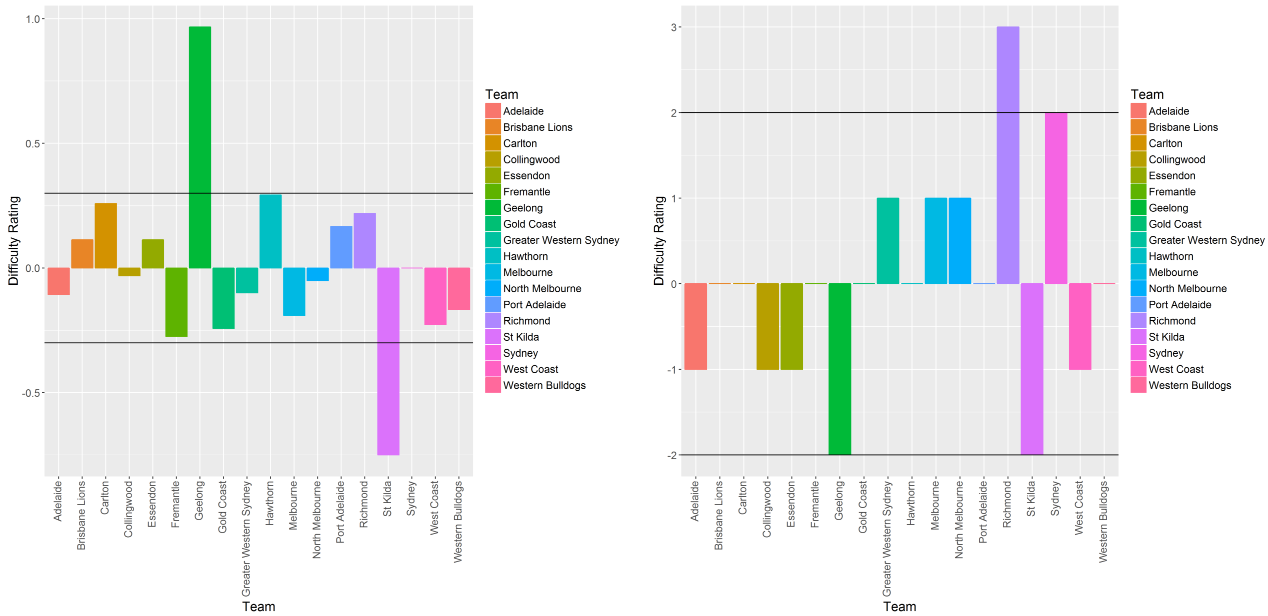
is that the remaining 16 teams have a difficulty rating between -0.3 and 0.3 and as such can be said to have a relatively fair season fixture.

Using the results generated by the SRS model it can be seen that most of the results lie within the range of -2 to 2 and can therefore it can once again be concluded that the season is of average difficulty for all teams other than Richmond - who in this case are only subjected to a fixture difficulty marginally higher than the other teams.



(a) VPM team rank (predicted vs. actual). (b) VPM team points (predicted vs. expected).

Figure 4.8: Team performance analysis for the 2015 AFL season.



(a) PSR fixture difficulty. (b) SRS fixture difficulty.

Figure 4.9: Fixture difficulty results for the 2015 AFL season (difficulties within horizontal boundaries represent fixtures of average difficulty).

The aim of this section of research was to determine whether it was possible to mathematically quantify both team performance and fixture difficulty. With respect to the MLogR model's accuracy, it achieves similar results to those in the literature. Baker and McHale (2013) achieved accuracies of 63.6 and 66.9% respectively using a contin-

Table 4.11: PSR results for the 2015 AFL season.

Team	Difficulty Rating
St Kilda	-0.75
Fremantle	-0.274
Gold Coast	-0.242
West Coast	-0.227
Melbourne	-0.189
Western Bulldogs	-0.167
Adelaide	-0.106
Greater Western Sydney	-0.1
North Melbourne	-0.052
Collingwood	-0.031
Sydney	0
Brisbane Lions	0.113
Essendon	0.113
Port Adelaide	0.167
Richmond	0.219
Carlton	0.259
Hawthorn	0.293
Geelong	0.967

Table 4.12: SRS results for the 2015 AFL season.

Team	Points	Simulated Rank	Previous Rank	Difficulty
Geelong	58.888	1	3	-2
Hawthorn	57.116	2	2	0
Sydney	56.636	3	1	2
Fremantle	54.96	4	4	0
Port Adelaide	52.748	5	5	0
Essendon	51.68	6	7	-1
North Melbourne	49.336	7	6	1
West Coast	47.868	8	9	-1
Adelaide	47.416	9	10	-1
Collingwood	44.996	10	11	-1
Richmond	41.56	11	8	3
Gold Coast	39.38	12	12	0
Carlton	36.704	13	13	0
Western Bulldogs	35.16	14	14	0
Brisbane Lions	32.932	15	15	0
St Kilda	27.78	16	18	-2
Greater Western Sydney	23.1	17	16	1
Melbourne	21.74	18	17	1

uous time Markov process to predict the outcomes of National Football League (NFL) games, Akhtar and Scarf (2012) achieved a 59.6% accuracy for predicting ex-ante outcomes of cricket matches when using a MLogR model, and Carbone, Corke, and Moisiadis (2016) achieved accuracies of 63 and 55.7% respectively using an ELO based method for predicting National Rugby League (NRL) match outcomes.

Whilst the predictive accuracy of the aforementioned models compare similarly to those in the literature – all of these models assume independence between matches. However, it can be safe to say that match results are subject to some form of dependence. Nevertheless, violation of the independence assumption does not significantly impact the final results due to the scale of our data (Heo and Leon 2005).

The SPM was designed using a truncated risk matrix such that the points assigned to a team who wins a very easy match ($\Pr(Win) > 0.7$) are significantly smaller in magnitude than points assigned to a team who wins a very hard match ($\Pr(Win) < 0.3$) with the inverse true for a team who loses a match. The rationale behind this design is that it is believed to be able to more accurately capture the real world implications of winning and losing matches of varying difficulty. The significantly larger negative results from the VPM are due to the heavier weightings assigned for winning and losing hard and easy games respectively. The coefficients and parameters of the risk matrix can also be altered in accordance with the MLogR model and coaching decisions. Hence, this methodology can be utilised for other competitive team sports.

The season difficulty models were initially designed with model simplicity in mind (PSR) and then graduated to a more complex simulation model (SRS), the rationale behind the PSR model is that it provides a model based on the most simplistic (and in this case most telling) metric of opponent difficulty (previous season ranking), while the SRS model attempts to simulate the outcome of a given season and then draw inferences with respect to relative fixture difficulty.

4.3 Summary

This chapter introduced the static data prediction models used to create ex-ante outcome forecasts for the 2015 AFL premiership season. Four candidate models were considered and underwent a significant degree of validation and significance testing. A total of 936 model variations were generated via the variation of method, data span, and match span parameters. The results obtained throughout the candidate models were similar to those found in the literature and demonstrates that the previously held paradigm of ex-ante prediction is relevantly stable regardless of the introduction of novel performance indicators. Following this two applications were investigated, namely team performance and fixture difficulty analysis. Two sub-models were considered for each of the applications with their findings being statistically similar within their respective groupings. Most notably the 2015 fixture was found to have 162 'fair' matches such that neither team was significantly favoured with the remaining 234 matches having some bias either way. In addition to this there was also a clear delineation within team performance consistent with current league standings.

CHAPTER 5

Dynamic Prediction Model

This chapter presents a near real-time model for the prediction of match outcome probabilities whilst a match is in progress. The aforementioned model is said to be both dynamic and near real-time as it allows for model parameters to evolve in time with events that transpire within a given match. The model makes use of both static and dynamic features as defined in Chapter 3 and relies on various computational optimisations afforded by the model's Markovian nature. The results obtained were computed over an entire AFL season with the model displaying robustness in regard to both initial probabilities and responsiveness to on-field events as a match is in progress. In addition, the overall accuracy of the model far surpasses that of those methods currently used in the literature.

5.1 Real-Time Prediction Models

As per Chapter 4 we can similarly define a predictive model for the outcome of a match between team $\mathcal{H}$ (home) and team $\mathcal{A}$ (away) at time $t \in [0, T]$ as $C_t(F_t) = f\left(S, S^{\mathcal{H}}, S^{\mathcal{A}}, D^{\mathcal{H}}(t), D^{\mathcal{A}}(t)\right)$ where $\left\{D^{\mathcal{H}}(t), D^{\mathcal{A}}(t)\right\} = \left(D_1^{\mathcal{H}}(t), D_2^{\mathcal{H}}(t), \ldots, D_c^{\mathcal{H}}(t), D_1^{\mathcal{A}}(t), D_2^{\mathcal{A}}(t), \ldots, D_c^{\mathcal{A}}(t)\right)$ are values of the the $2c$ team specific dynamic match features, and C_t is a representation of the predicted outcome probability for a match given that dynamic feature data is observed up to and including time t.

$$C_t(F_t) = \Pr\left(\{\text{Draw}, \text{Loss}, \text{Win}\} \,|t = T\right) \tag{5.1}$$

with $f(\cdot)$ being an unknown function to be estimated using the statistical methods outlined in subsection 5.1.1, and the dynamic components $\left\{D^{\mathcal{H}}(t), D^{\mathcal{A}}(t)\right\}$ of feature set F_t as described in Chapter 3 subsection 3.3.2

5.1.1 Continuous Time Inhomogeneous Markov Models

Although not prevalent within current sporting literature, Markov models are able to capture complex time and covariate interactions in both homogeneous and inhomogeneous observation cases and in absorbing or non-absorbing state space structures. A Continuous Time Inhomogeneous Markov Model is a stochastic model which describes the changes in a system consisting of random processes. In this application the model forecasts over a discrete state space as a sequence of Markov chains where the interval between successive state transitions is irregular (Ibe 2013). A Markov chain is a sequence of discrete observations satisfying the Markov property

$$\Pr\left(X_{t_{j+1}}|F_1,\ldots,F_{t_j}\right) = \Pr\left(X_{t_{j+1}}|F_{t_j}\right) \tag{5.2}$$

such that $X_t = \{D, L, W\}$ is the state space, $F_{t_j} = \left\{S, D_{t_j}\right\}$ is the set of static and dynamic features observed during an AFL match and

$$\Pr\left(X_{t_{j+1}}|F_{t_j}\right) = \begin{bmatrix} p_{DD} & p_{DL} & p_{DW} \\ p_{LD} & p_{LL} & p_{LW} \\ p_{WD} & p_{WL} & p_{WW} \end{bmatrix} = P\left(t_j, t_{j+1}\right) = \exp\left(Q\left(F_{t_j}\right) t_j\right) \tag{5.3}$$

is the probability of observing an outcome X at time t_{j+1} given observed feature data F up to and including time t_j where

$$Q\left(F_{t_j}\right) = \begin{bmatrix} q_{DD} & q_{DL} & q_{DW} \\ q_{LD} & q_{LL} & q_{LW} \\ q_{WD} & q_{WL} & q_{WW} \end{bmatrix} \tag{5.4}$$

is the transition intensity matrix after observing feature data F up to and including time t_j (Logofet and Lesnaya 2000), which is solved using the Kolmogorov forward equation making use of partial differential equations and eigenvalue decomposition to solve for each

$q_{\mathbb{S}(t_j)\mathbb{S}(t_{j+1})}$ (Marshall and Jones 1995).

$$\begin{aligned}
\frac{\partial P(t)}{\partial t} &= P(t) \\
\Rightarrow \frac{1}{P(t)}\frac{\partial P(t)}{\partial t} &= Q \\
\Rightarrow \frac{\partial \ln(P(t))}{\partial t} &= Q \\
\Rightarrow \int \ln(P(t)) &= \int Q \partial t \\
\Rightarrow P(t) &= \exp(Qt) \\
\Rightarrow P(t) &= ADA^{-1}
\end{aligned} \tag{5.5}$$

The decomposition in Equation 5.5 is such that D is a diagonal matrix with element $\{(i,j) : i = j\}$ corresponding to the exponential of the i^{th} distinct eigenvalue of Q and A is a matrix with eigenvectors corresponding to the aforementioned eigenvalues. Hence, to forecast $\Pr(t_1, t_2)$; that is the probability transition matrix from t_1 to t_2, the decomposition $AD^{t_2-t_1}A^{-1}$ is derived and solved. In addition to this the values for the transition matrix Q are found by maximising the likelihood function $L(Q|\theta)$ for each of the unknown parameter values $\theta = \left\{q_{(\mathbb{S}_{t_j})(\mathbb{S}_{t_{j+1}})}, \beta_{(\mathbb{S}_{t_j})(\mathbb{S}_{t_{j+1}})} | \left\{\mathbb{S}_{t_j}, \mathbb{S}_{t_{j+1}}\right\} \in X_t\right\}$, where $\beta_{(\mathbb{S}_{t_j})(\mathbb{S}_{t_{j+1}})}$ are the coefficients for the transition between states from time t_j to t_{j+1}. The likelihood described above can be expressed by Equations 5.6–5.7 where successive states $\mathbb{S}(t_j)$ and $\mathbb{S}(t_{j+1})$ occur at times t_j and t_{j+1} for an index i which contains the set of all observable matches $M = \{1, 2, \ldots, m\}$

$$L(Q|\theta) = \prod_{i=1}^{M} L_{i,j} \tag{5.6}$$

$$L_{i,j} = p_{\mathbb{S}(t_j)\mathbb{S}(t_{j+1})}(t_{j+1} - t_j|\theta) \tag{5.7}$$

Equivalently, once the values for Q have been found it is possible to calculate $P(t)$ by using the matrix exponential $\exp(Qt)$ such that

$$\exp(Qt) = \sum_{k=0}^{\infty} \frac{1}{k!}(Qt)^k \tag{5.8}$$

with $(Qt)^k = Qt \times Qt \times \cdots \times Qt$. However, while this formulation is simple enough it does not allow for the inclusion of time varying covariates, this is achieved by altering the above formulation as follows;

$$q_{(\mathbb{S}_{t_j})(\mathbb{S}_{t_{j+1}})}(t_j) = q^0 e^{\beta^T_{(\mathbb{S}_{t_j})(\mathbb{S}_{t_{j+1}})} z(t_j)} \tag{5.9}$$

$$L_{i,j} = e^{q_{\mathbb{S}(t_j)\mathbb{S}(t_j)}(t_{j+1}-t_j|\theta)} q_{\mathbb{S}(t_j)\mathbb{S}(t_{j+1})} \tag{5.10}$$

where $z(t_j) = Cov_{t_j} - Cov_{Mean}$ is the difference between the observed covariate values at time t_j and the mean model values, and q^0 is the log-baseline estimate of $Q(t_j)$. This enables the approximation of

$$Q(t_j) = e^{\sum q^0 + q^0_{\mathbb{S}(t_j)} z(t_j)} \tag{5.11}$$

from which it is possible to calculate $P(t_1, t_2)$ as

$$P(t_1, t_2) = P(t_1, t_2) \times P(t_2, t_{j-1}) \times P(t_{j-1}, t_j) \tag{5.12}$$

where the epochs $\{t_1, \ldots, t_j\}$ are the inhomogeneous time points at which on-field transactions are recorded, and

$$\begin{aligned} P(t_1, t_2) &= \exp(t_2 - t_1) Q(t_1) \\ P(t_2, t_{j-1}) &= \prod_{t_2}^{t_{j-1}} \exp(t_i - t_{i-1}) Q(t_i) \\ P(t_{j-1}, t_j) &= \exp(t_j - t_{j-1}) Q(t_{j-1}) \end{aligned} \tag{5.13}$$

5.1.2 Results and Discussion

As per the static models in section 4.1 all analyses were once again conducted on a computer with a 64-bit Windows operating system, Intel® Core™ i7-7700K processor, and 32GB RAM. Results were obtained using an algorithm and routine (see Appendix E.1) written in the statistical computing package R (R Core Team 2018) which makes use of the packages listed in table 5.1. Each model took approximately 3 hours and 11 minutes to build which is practically acceptable for a study with such complex data and models, however, evaluation of a new case only takes approximately 11 seconds for a full match (approximately 120 minutes and 1900 epochs) which is practically acceptable as in practice only a single epoch will be evaluated at a time.

Table 5.1: Dynamic model packages.

Package	Author
msm	Jackson (2011)
doParallel	Microsoft Corporation and Weston (2018)
ggplot2	Wickham (2009)
zoo	Zeileis and Grothendieck (2005)
expm	Goulet et al. (2017)

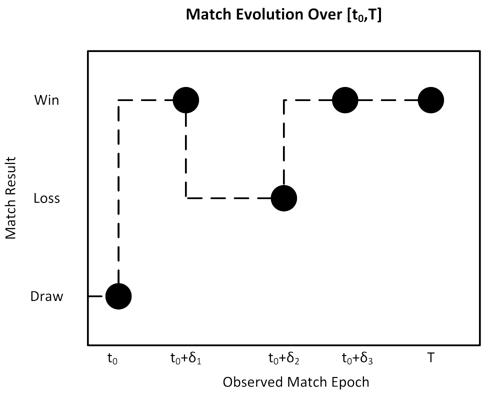


Figure 5.1: Evolution of the Markov model.

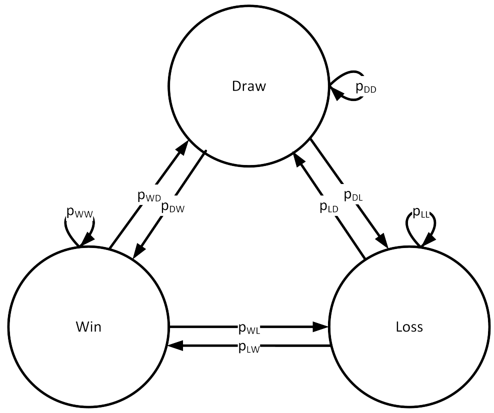


Figure 5.2: State space model.

The Markov model described in section 5.1.1 was implemented as a three state non-absorbing system, such that it is possible to model the transitions to and from each state at a given epoch, where an epoch is an observable instance during a match such that a transition between states occurs. An example of this can be seen in figures 5.1 and 5.2 such that when a match is observed at an epoch the model transitions between states, following the general process outlined in figure 5.3. It is important to note that as the model is both time inhomogeneous and continuous, the time between each observed epoch is uncertain and non integer resulting in an evolution approximated by equations 5.12 and 5.13. The initial transition matrix is an important parameter for any stochastic state model, therefore to generate an practically acceptable generalisation, the transition matrix is determined through the use of averaged match evolutions over the 2015 premiership season, and formulated as follows. Given a specific state transition from state $S(t_j)$ to state $S(t_{j+1})$ the transition probability is calculated as the proportion of the number of transitions from $S(t_j)$ to $S(t_{j+1})$ with respect to the total number of transitions starting at state $S(t_j)$.

$$\pi_0 = \text{from } S(t_j) \begin{array}{c} \\ \\ \text{Draw} \\ \text{Loss} \\ \text{Win} \end{array} \begin{array}{ccc} \multicolumn{3}{l}{\text{to } S(t_{j+1})} \\ \text{Draw} & \text{Loss} & \text{Win} \\ \frac{\sum(\text{Draw,Draw})}{\sum \text{Draw}} & \frac{\sum(\text{Draw,Loss})}{\sum \text{Draw}} & \frac{\sum(\text{Draw,Win})}{\sum \text{Draw}} \\ \frac{\sum(\text{Loss,Draw})}{\sum \text{Loss}} & \frac{\sum(\text{Loss,Loss})}{\sum \text{Loss}} & \frac{\sum(\text{Loss,Win})}{\sum \text{Loss}} \\ \frac{\sum(\text{Win,Draw})}{\sum \text{Win}} & \frac{\sum(\text{Win,Loss})}{\sum \text{Win}} & \frac{\sum(\text{Win,Win})}{\sum \text{Win}} \end{array} \tag{5.14}$$

The next initialisation parameter of importance is the vector of initial probabilities for each state. In order to assess both the stability and convergence of the Markov model, two types of initial probabilities were considered. Deterministic initial probabilities such that each match used the same probability vector

$$\mathbf{u_d}(1) = \Pr(\text{Draw}, \text{Loss}, \text{Win})_{\mathbf{d}} = \{0.1, 0.3, 0.6\} \tag{5.15}$$

and static initial probabilities as determined by the optimal MLogR outlined in subsection 4.1.5.2

$$C(F) = \mathbf{u_s}(1) = \left\{ \frac{e^{\beta_1 x_k}}{\sum_{c=1}^{3} e^{\beta_c x_k}}, \frac{e^{\beta_2 x_k}}{\sum_{c=1}^{3} e^{\beta_c x_k}}, \frac{e^{\beta_3 x_k}}{\sum_{c=1}^{3} e^{\beta_c x_k}} \right\} = \Pr(\text{Draw}, \text{Loss}, \text{Win})_{\mathbf{s}} \tag{5.16}$$

Utilising equations 5.3, 5.6, 5.9, and 5.10 a Markov chain model is constructed with match outcome probabilities as a function of both static and dynamic features described in chapter 3. This model yields an average Q matrix for the system built using games played by the Western Bulldogs during the 2015 AFL premiership season

$$Q\left(F_{t_j}\right) = \begin{pmatrix} -0.273 & 0.091 & 0.182 \\ 0.018 & -0.054 & 0.036 \\ 0.000 & 0.006 & -0.006 \end{pmatrix} \tag{5.17}$$

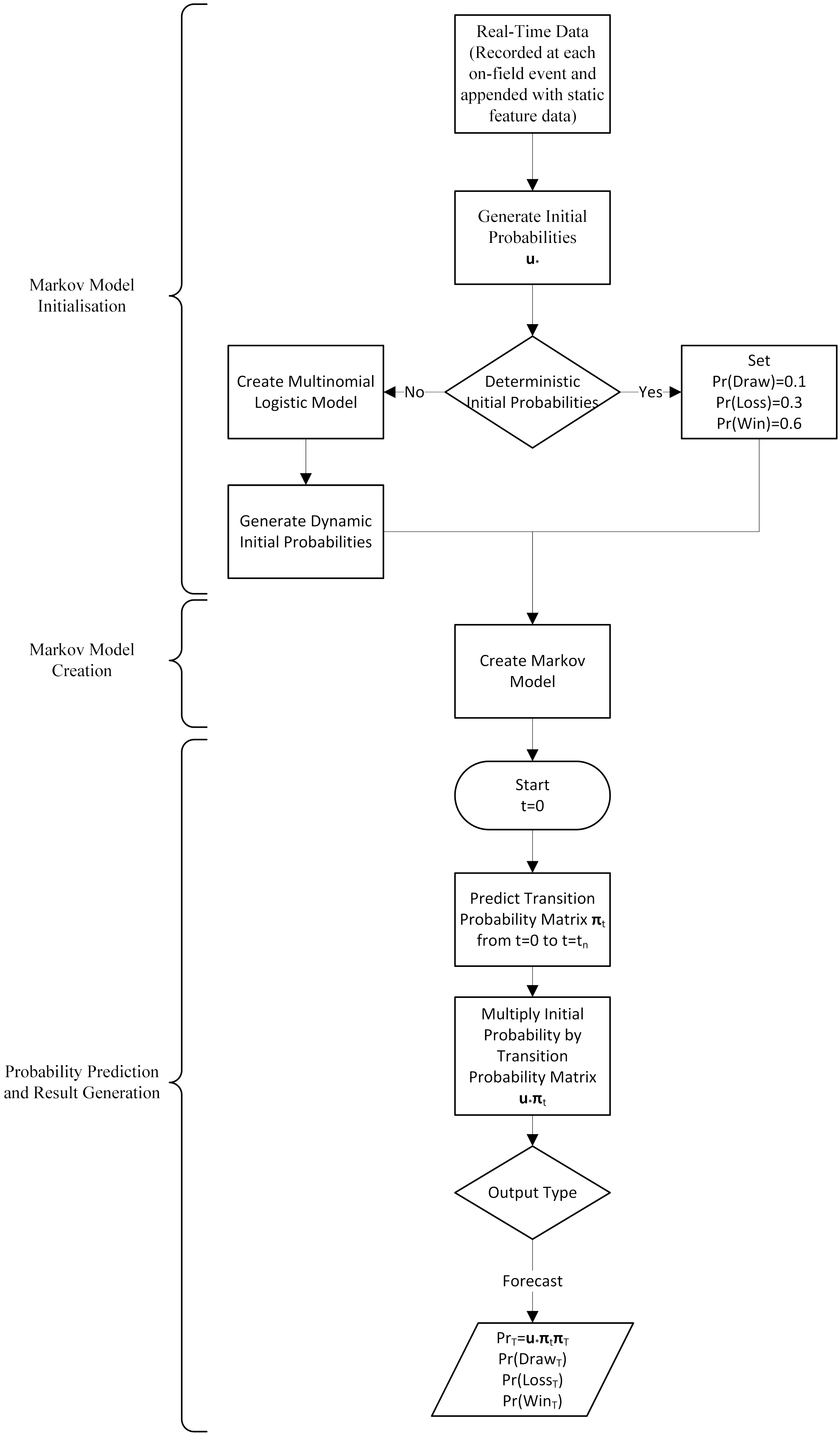
and hazard ratios $e^{\beta_{S(t_j)S(t_{j+1})}}$ for the model covariates, where the hazard ratios are computed by exponentiating the estimated covariate effects on the log-transition intensities.

$$e^{\beta_{DL}} = \{0.979, 1.038, 0.559, 0.440, 0.082, 1.046, 0.986, 0.812, 1.255, 1.171, 0.850\} \tag{5.18}$$
$$e^{\beta_{DW}} = \{1.008, 0.985, 2.351, 0.431, 0.463, 1.032, 1.027, 2.084, 0.507, 0.642, 1.530\} \tag{5.19}$$
$$e^{\beta_{LD}} = \{1.054, 1.028, 0.997, 0.389, 0.020, 1.019, 1.029, 0.871, 0.932, 0.948, 1.067\} \tag{5.20}$$
$$e^{\beta_{LW}} = \{1.155, 0.978, 1.422, 0.201, 2.225, 0.973, 1.043, 0.840, 1.033, 0.988, 1.028\} \tag{5.21}$$
$$e^{\beta_{WD}} = \{1.020, 0.951, 2.032, 4.048, 1.926, 1.021, 1.002, 0.628, 1.361, 1.067, 0.870\} \tag{5.22}$$
$$e^{\beta_{WL}} = \{1.408, 1.086, 0.879, 5.802, 1.047, 1.041, 0.981, 1.008, 0.754, 1.064, 0.960\} \tag{5.23}$$

At each step of the forecasting algorithm, in such a way that each observation takes place at an epoch $t \in [0, T]$ where t is the currently observed epoch in the match and T is the final observed epoch in the match. The algorithm then produces a forecast of the match outcome probabilities at time T. With these results being generated using equation 5.24

$$g\left(F_{t_j}\right) = \mathbf{u}_*(1)\, \pi_{0,t_j} \pi^*_{t_j,T} = \Pr(\text{Draw}, \text{Loss}, \text{Win})_T \tag{5.24}$$

Figure 5.3: Markov model overview.



5.1.2.1 Model Evaluation

As the Markov model described above is non-absorbing, time inhomogeneous, and continuous; evaluation of the model's goodness of fit becomes significantly more difficult. Hence, each model was evaluated for both epoch prediction accuracy and final result outcome. Epoch prediction accuracy is measured as the percentage of epochs correctly forecast relative to the final outcome at time T, while final result outcome is measured as the percentage of forecasts at time T which match the final result outcome. For example, in the case of a match which ends in a win for the home team; the epoch prediction accuracy is measured as the percentage of epochs forecast as a win for the home team, while the final result outcome is a correct classification if the forecast outcome at time T matches that of the actual final outcome (in this case a win for the home team).

The Markov model was trained on match data for games played by the Western Bulldogs during the 2015 AFL premiership season and subsequently tested using match data for games played by the Western Bulldogs from the 2017 AFL premiership season using both deterministic (Equation 5.15) and static initial probabilities (Table 5.2). These results are listed in tables 5.3 and 5.4.

Table 5.2: Per match static initial probabilities.

Match	Draw	Loss	Win
1	0.016	0.503	0.481
2	0.004	0.304	0.692
3	0.000	0.676	0.324
4	0.001	0.072	0.926
5	0.002	0.200	0.798
6	0.000	0.572	0.428
7	0.000	0.585	0.415
8	0.001	0.304	0.695
9	0.000	0.177	0.823
10	0.006	0.452	0.541
11	0.100	0.711	0.189
12	0.000	0.437	0.563
13	0.000	0.162	0.838
14	0.000	0.491	0.509
15	0.003	0.221	0.776
16	0.000	0.838	0.162
17	0.000	0.222	0.778
18	0.006	0.418	0.577
19	0.000	0.765	0.235
20	0.005	0.603	0.392
21	0.000	0.278	0.722
22	0.000	0.351	0.649

Table 5.3: Deterministic initial probability Markov model results.

Initial Draw Probability	Initial Loss Probability	Initial Win Probability	Final Draw Probability	Final Loss Probability	Final Win Probability	Epoch Accuracy	Actual Result	Forecast Result	Home Rank	Away Rank
0.100	0.300	0.600	0.003	0.147	0.850	0.482	Loss	Win	12	7
0.100	0.300	0.600	0.000	0.000	1.000	0.911	Win	Win	8	14
0.100	0.300	0.600	0.000	0.129	0.871	0.480	Win	Win	18	8
0.100	0.300	0.600	0.000	0.018	0.982	0.973	Win	Win	9	17
0.100	0.300	0.600	0.000	0.001	0.999	0.936	Win	Win	6	15
0.100	0.300	0.600	0.000	0.173	0.827	0.983	Win	Win	4	5
0.100	0.300	0.600	0.001	0.554	0.445	0.009	Win	Loss	7	4
0.100	0.300	0.600	0.000	0.039	0.961	0.826	Win	Win	4	5
0.100	0.300	0.600	0.002	0.191	0.807	0.763	Win	Win	6	9
0.100	0.300	0.600	0.000	0.227	0.773	0.520	Win	Win	6	7
0.100	0.300	0.600	0.000	0.001	0.999	0.991	Win	Win	16	6
0.100	0.300	0.600	0.000	1.000	0.000	1.000	Loss	Loss	8	6
0.100	0.300	0.600	0.000	0.009	0.991	0.000	Win	Win	9	16
0.100	0.300	0.600	0.000	0.733	0.267	1.000	Loss	Loss	9	7
0.100	0.300	0.600	0.000	0.000	1.000	0.982	Win	Win	2	10
0.100	0.300	0.600	0.036	0.849	0.114	1.000	Loss	Loss	16	11
0.100	0.300	0.600	0.000	0.001	0.999	0.997	Win	Win	11	15
0.100	0.300	0.600	0.000	0.926	0.074	1.000	Loss	Loss	8	10
0.100	0.300	0.600	0.002	0.985	0.013	1.000	Loss	Loss	18	9
0.100	0.300	0.600	0.000	0.998	0.002	1.000	Loss	Loss	7	2
0.100	0.300	0.600	0.000	0.017	0.983	0.927	Win	Win	6	9
0.100	0.300	0.600	0.000	0.996	0.004	1.000	Loss	Loss	11	12
Accuracy						0.808		0.909		
(SD)						(0.312)		(0.294)		

Table 5.4: Static initial probability Markov model results.

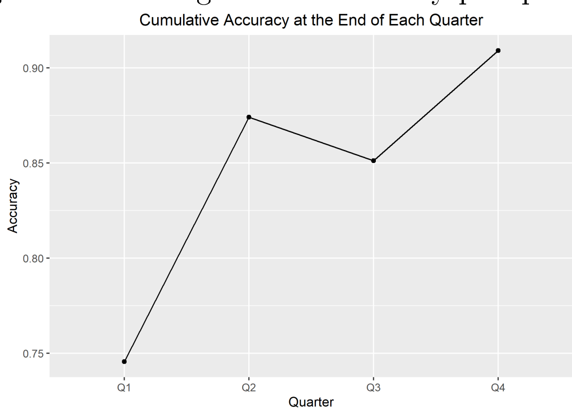
Initial Draw Probability	Initial Loss Probability	Initial Win Probability	Final Draw Probability	Final Loss Probability	Final Win Probability	Epoch Accuracy	Actual Result	Forecast Result	Home Rank	Away Rank
0.016	0.503	0.481	0.003	0.147	0.850	0.482	Loss	Win	12	7
0.004	0.304	0.692	0.000	0.000	1.000	0.911	Win	Win	8	14
0.000	0.676	0.324	0.000	0.129	0.871	0.480	Win	Win	18	8
0.001	0.072	0.926	0.000	0.018	0.982	0.973	Win	Win	9	17
0.002	0.200	0.798	0.000	0.001	0.999	0.936	Win	Win	6	15
0.000	0.572	0.428	0.000	0.221	0.779	0.983	Win	Win	4	5
0.000	0.585	0.415	0.001	0.558	0.440	0.009	Win	Loss	7	4
0.001	0.304	0.695	0.000	0.038	0.962	0.826	Win	Win	4	5
0.000	0.177	0.823	0.001	0.181	0.818	0.763	Win	Win	6	9
0.006	0.452	0.541	0.000	0.227	0.773	0.520	Win	Win	6	7
0.100	0.711	0.189	0.000	0.001	0.999	0.991	Win	Win	16	6
0.000	0.437	0.563	0.000	1.000	0.000	1.000	Loss	Loss	8	6
0.000	0.162	0.838	0.000	0.009	0.991	0.000	Win	Win	9	16
0.000	0.491	0.509	0.000	0.733	0.267	1.000	Loss	Loss	9	7
0.003	0.221	0.776	0.000	0.000	1.000	0.982	Win	Win	2	10
0.000	0.838	0.162	0.036	0.849	0.114	1.000	Loss	Loss	16	11
0.000	0.222	0.778	0.000	0.001	0.999	0.997	Win	Win	11	15
0.006	0.418	0.577	0.000	0.926	0.074	1.000	Loss	Loss	8	10
0.000	0.765	0.235	0.002	0.985	0.013	1.000	Loss	Loss	18	9
0.005	0.603	0.392	0.000	0.998	0.002	1.000	Loss	Loss	7	2
0.000	0.278	0.722	0.000	0.017	0.983	0.927	Win	Win	6	9
0.000	0.351	0.649	0.000	0.996	0.004	1.000	Loss	Loss	11	12
Accuracy						0.808		0.909		
(SD)						(0.312)		(0.294)		

In spite of the reduced breadth of available data (restricted to Western Bulldogs matches for the 2015 and 2017 seasons) both deterministic and static initial probability models performed exceedingly well and each attained an average epoch accuracy of 80.81% and a final result accuracy of 90.90%. This duplication of results is an important observation as it confirms that the model is both stable and convergent for differing initial probabilities. From this it is also possible to conclude that the overall model procedure can be streamlined by removing the generation of static initial probabilities as they have no perceivable impact on the model.

5.2 Application of Real-Time Models

Further to the above results it is possible to see the model's increasing accuracy as a match progresses, with the model attaining an average accuracy of 74.57% at the end of the first quarter which then increases to an average of 90.09% at the end of the fourth quarter (Figure 5.4), these results are both far greater than that of those produced by other studies in the literature and even those produced in chapter 4 of this study whose accuracy peaked at 69.6%. It should be noted that the model experiences a drop in average accuracy from quarters 2 to 3. A possible cause being that the model yields higher accuracies when exposed to data exhibiting greater variability with respect to the response variable. That is to say, when observing a quarter that is more 'active' (producing more inter-state transitions than intra-state transitions) the model performs better. In this application and throughout the data available to this study the variance of the match result at time t fluctuates in accordance with the changes in the cumulative accuracy at the end of each quarter (Table 5.5).

Figure 5.4: Average model accuracy per quarter.



Whilst these results are significant, a key importance of the model is that it is responsive to on-field transactions so that it may be used responsively as a training, coaching, and tactical toolbox. Currently, match data is fed directly to the coaching team as a match progresses with various key parameters monitored and codified according to pre-

Table 5.5: Per quarter variance with respect to match outcome.

Quarter	Variance
1	3.552
2	3.732
3	1.647
4	2.522

determined thresholds (Figure 5.5). With this information coaches can then tailor their team's strategy; bolstering defence or exploiting newly discovered weaknesses in the opposing side. After the fact analysis is also possible whereby new drills and tactics can be developed to optimally prepare for a given opponent or to train players in the handling of certain scenarios.

Figure 5.5: Rocket Dashboard.

With the application of the methodology and framework identified herein, the above can be further extended by enabling coaches to see how any single action or sequence of actions will affect their team's outcome probabilities. This allows for the quantification of coaching decisions whereby the quality of each decision can be measured in terms of the change in the observed outcome probabilities. Further applications could be seen in a training context where multiple scenarios or sequences of transactions could be permuted; each with a set difficulty or victory odds and then replicated in a controlled training environment enabling the players to learn how to respond to or limit the influence of superior opposition play.

In figure 3.4 it has already been shown that the margin is directly correlated with the on-field transactions of interest and as such shall be used their stead in the discussions to follow. An example of the output generated by the model can be seen in figures 5.6 and 5.7 below and are a representation of a match between Fremantle and Western Bulldogs which took place in round 3 of the 2017 AFL premiership season. In the upper section of Figure 5.6 the outcome probabilities generated by the Markov model are plotted against their respective epoch times (with breaks in the plot representing unobserved epochs), below that is a visual depiction of the predicted outcome (once again generated by the Markov model) against the actual match outcome, whereas the upper section of figure 5.7 plots the outcome probabilities generated by the Markov model as match time progresses with a running margin below. The margin is plotted with respect to the home team as the match progresses and acts as a facsimile to in-match events and also serves as an approximation of team form and possession.

Figure 5.6: Outcome prediction over time.

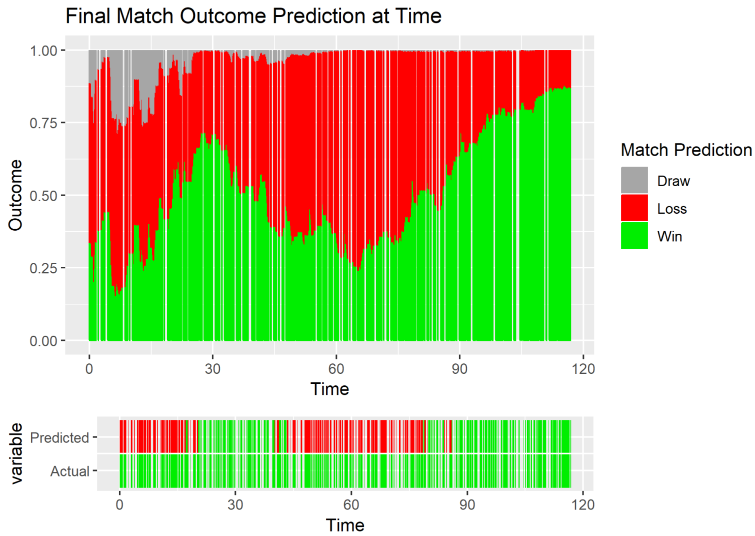


Figure 5.7: Prediction probabilities and margin over time.

Figure 5.6 presents an attractive visual representation of a match in play and would most likely increase fan engagement by providing fans with a graphical representation

of their team's current form/performance relative to their opponent, whereas figure 5.7 presents a more streamlines probabilistic output with respect to the home team's margin of victory and could be used to compare bookmakers odds across a wide variety of sources or even allow bettors to make better informed decisions when placing bets.

This could become a powerful tool to any sporting team as it enables one to alter match strategies on the fly and even possibly play the meta-game in such a way that could potentially increase both the psychological pressure on an opposing team and the excitement for fans as new and innovative strategies and training practices are formulated and become available.

5.3 Summary

This chapter presented the three-state continuous time inhomogeneous Markov model used to create near real-time outcome predictions for the 2017 AFL premiership season. This model made use of both dynamic and static data and was conditioned on both static and deterministic initial probabilities for which the models produced convergent results. Model forecasts responded significantly to the transactional inputs of the model and the predictive accuracy of the model far surpassed that of current ex-ante methods found in the literature and even of those produced in chapter 4. This is promising as it would indicate that coaches and training staff would be able to use this to dynamically alter their strategies and make far more informed decisions in response to real-time match conditions.

CHAPTER 6

Conclusions, Contributions, and Future Works

6.1 Summary of the Work

Sports analysis has always been a real talking point amongst both statisticians and sports personnel. However the complexity of creating an efficient and accurate model coupled with the difficulties in acquiring in-game statistics has resulted in most research being focused on before the fact result prediction. This research presents a framework for the near real-time prediction of match outcomes at various strategic points within an AFL match. This was achieved through the acquisition of in-game statistics, data on past performances, and using statistical modelling methods with the final goal being the development of a robust and efficient prediction of match results. The outcome of this research will aid coaches and training staff by allowing them to quantify how specific sequences of on-field transactions, player actions, and coaching calls directly affect the outcome probabilities for a given match. Additionally, coaches may decide to rest key players if their win probability is high or try risky strategies when faced with a low win probability. This will further accentuate retrospective analysis by enabling the development of strategies and drills that accentuate features that are most influential in the model.

The methodological frameworks developed herein can be easily transferred across other sports. The static methods (Chapter 4) can be applied in a variety of sporting scenarios, both fast moving and slow paced; with similar work being undertaken in soccer, rugby, baseball, and tennis to name a few (Castellano, Casamichana, and Lago 2012; Carbone, Corke, and Moisiadis 2016; Horvat and Job 2020; Clarke and Dyte 2000). The dynamic methods (Chapter 5) whilst applicable primarily to Australian Football due to the specific nature of the statistics utilised, could be generalised to similar fast moving invasion style games such as rugby and soccer as similarities may be drawn between offensive and defensive metrics with only minor restructuring needed to adapt game specific metrics.

A survey of the literature revealed that whilst real-time analysis is a key area of interest in fields such as medicine and finance, the proprietary nature of real-time sporting data restricts most public research to ex-ante result prediction and optimal betting strategies with the goal of beating bookmakers odds. Meanwhile, the features used for predictions across various sports do not differ significantly across methods but tend to follow a logical grouping depending on which sport is being observed. From this it is clear that feature selection dictates the success of these models. Ex-ante prediction is implemented in a variety of sports regardless of the speed at which the sport is played and is a large part of the currently available literature. Both machine learning and generalised linear techniques have been used to great success for result prediction in a variety of sporting applications.

Due to the cost and difficulty of simultaneous data collection, real-time prediction is carried out on slower moving sports and those where up to date data is easily available. These applications tend to use less computationally taxing methods such as multinomial linear and logistic regression and rely heavily on pre-established methodologies such as the Duckworth-Lewis resource matrix and existing match strategies.

A major factor in any mathematical model is the quality of data used for both model creation and testing. With the issue of big data and its widespread adoption within the sporting world, it is important that heavy scrutiny be placed upon data prior to its use. The two types of data utilised for this research can be summarised as follows; static data (known prior to the match) which is widely accessible and can be found on a myriad of online repositories, and dynamic data (gathered during the match) which is restricted to AFL teams and the companies that gather said data.

Data was gathered from various online repositories (static data) as well as Champion Data (dynamic data) after which the data was cleaned, processed, and relevant features extracted. In terms of data accuracy, both static and dynamic data originate from Champion Data either directly or indirectly where historically Champion Data have boasted a 99% accuracy through the use of their multi-phase data entry strategy (Champion Data 2017). Whilst there is no publicly available audit to attest to the accuracy of this claim, the fact still remains that Champion Data has been and still remains retained by both the AFL and their participant clubs for considerable financial compensation. The data were then subjected to further quality control measures during processing to ensure that no erroneous or duplicate data existed within the final dataset. Following this various static feature models were explored with the goal of feature selection and comparative ex-ante prediction. The results obtained were in line with the literature in terms of both features used and model accuracies, with the most accurate model achieving an overall accuracy of 69.60%. Applications of the static model were then explored, with the goal being the development of new methods to quantify team performance and fixture difficulty.

The next phase of analysis was concerned with the dynamic prediction model in which a continuous time inhomogeneous Markov model was selected and as such allows for the irregular frequency at which on-field transactions are observed. The model performed notably well with an average epoch accuracies in excess of 80% and match outcome results in excess of 90%. The results of this study demonstrate that accurate near real-time prediction is achievable under real world conditions using non-simulated on-field transactional data.

In conclusion the Markov model implemented within this study has shown to be practically acceptable, obtaining far greater accuracies than that of static only ex-ante models. Further research and exploration is however still needed, and as more data is made available it is theorised that far more robust and accurate models may be created. In addition to this more automation in terms of variable selection would also be preferable.

6.2 Contributions

The main focus of this study was to provide a robust and efficient framework for the prediction of near real-time AFL match outcome probabilities. Through this research, a number of contributions specific to Australian Rules Football analytics were made. These contributions are as follows:

- To the best of our knowledge, the framework and methodologies presented within this thesis are the first publicly available of their kind within the realm of Australian Rules Football prediction.
- The research herein addressed the need for real-time analysis within the AFL. More specifically this research focused on outcome prediction based on data extracted as a match progresses and is additionally supplemented by data available prior to the start of a match.
- A variation to the structure of AFL rankings was proposed such that each team is no longer awarded a fixed number of points after a match but instead awarded points according to the relative difficulty of the fixture. This could be further augmented to provide an alternative measure of team ‘form’ and may even lead to an increase in fan engagement.
- It was demonstrated that a novel yet computationally complex methodology was able to accurately model match outcomes as a function of in-match transactions and as such confirms that currently available technologies can significantly augment the decision-making process of coaches and team staff.
- The frameworks used in this study have the potential to be applied in a wide variety of sports.

6.3 Future Work

Whilst this research provides a novel framework for accurately forecasting the outcome of an AFL match as it is in progress, future extensions to the current work could include the following:

- Update the current database of static and dynamic data so that further and more in-depth studies can be conducted.
- In addition to the above update it would be worthwhile to run comparative studies on data pertaining to pre, during, and post the COVID-19 pandemic to see if changes to scheduling, crowd capacity, and on-field rules had any significant effect on the sport.
- With the widespread adoption of new monitoring technologies, the scope of available data is ever-increasing. As such it would be advantageous to incorporate as many new sources of data as possible; most notably amongst these are GPS and LPS receivers which can relay locomotive and positional data.
- Further development of the framework to bundle data importation, feature extraction, forecasting, and analysis as a standalone application therefore simplifying the process and making it suitable for the end-user.

Bibliography

AFL Tables. 2017. *AFL-VFL match, player and coaching stats, records and lists.* Dataset. http://afltables.com/afl/afl_index.html.

Stats glossary: Every stat explained. 2017, December. https://www.afl.com.au/news/2017-12-28/stats-glossary-every-stat-explained.

Akhtar, S., and P. Scarf. 2012. "Forecasting test cricket match outcomes in play." *International Journal of Forecasting* 28 (3): 632–43. ISSN: 0169-2070. https://doi.org/http://dx.doi.org/10.1016/j.ijforecast.2011.08.005. http://www.sciencedirect.com/science/article/pii/S0169207011001622.

Aldous, D. 2017. "Elo ratings and the sports model: A neglected topic in applied probability?" *Statistical science* 32 (4): 616–29.

Aughey, R. J. 2011. "Applications of GPS technologies to field sports." *International journal of sports physiology and performance* 6 (3): 295–310.

Australian Football League. 2015. *Laws of Australian football 2015.*

———. 2019. *AFL Annual Reports 1993-2017.* Annual Report. https://www.clearinghouseforsport.gov.au/Library/archive/digital_archive/australian_football.

Bailey, M. 2005. "Predicting sporting outcomes: A statistical approach." Thesis.

Bailey, M., and S. R. Clarke. 2006. "Predicting the Match Outcome in One Day International Cricket Matches, while the Game is in Progress." *Journal of Sports Science and Medicine* 5:480–87.

Baker, R. D., and I. G. McHale. 2013. "Forecasting exact scores in National Football League games." *International Journal of Forecasting* 29 (1): 122–30. ISSN: 0169-2070. https://doi.org/http://dx.doi.org/10.1016/j.ijforecast.2012.07.002. http://www.sciencedirect.com/science/article/pii/S0169207012001070.

Baumer, B., and A. Zimbalist. 2014. *The sabermetric revolution.* University of Pennsylvania Press.

Bennett, K. P., and E. J. Bredensteiner. 2000. "Duality and geometry in SVM classifiers." In *ICML,* 57–64.

Biau, G. 2012. "Analysis of a random forests model." *Journal of Machine Learning Research* 13 (Apr): 1063–95.

Blainey, G. 2010. *A game of our own: The origins of Australian football.* Black Inc.

Boyd, M. A., and S. Lau. 1998. "An introduction to Markov modeling: Concepts and uses." In *Reliability and Maintainability Symposium.*

Breiman, L. 2001. "Random forests." *Machine learning* 45 (1): 5–32. ISSN: 0885-6125.

Carbone, J., T. Corke, and F. Moisiadis. 2016. "The Rugby League Prediction Model: Using an ELO-Based Approach to Predict the Outcome of National Rugby League (NRL) Matches." *International Educational Scientific Research Journal* 2 (5).

Casal, C. A., M. T. Anguera, R. Maneiro, and J. L. Losada. 2019. "Possession in Football: More Than a Quantitative Aspect - A Mixed Method Study." *Frontiers in Psychology* 10:501. ISSN: 1664-1078. https://doi.org/10.3389/fpsyg.2019.00501. https://www.frontiersin.org/article/10.3389/fpsyg.2019.00501.

Castellano, J., D. Casamichana, and C. Lago. 2012. "The use of match statistics that discriminate between successful and unsuccessful soccer teams." *Journal of human kinetics* 31:137–47.

Champion Data. 2017. *Real-time AFL match data.* Dataset. https://www.championdata.com/.

———. 2019. *About Us - Champion Data.* https://www.championdata.com/about-us/.

Clarke, S. R. 1988. "Dynamic programming in one-day cricket-optimal scoring rates." *Journal of the Operational Research Society* 39 (4): 331–37.

———. 2005. "Home advantage in the Australian football league." *Journal of Sports Sciences* 23 (4): 375–85.

Clarke, S. R., and D. Dyte. 2000. "Using official ratings to simulate major tennis tournaments." *International transactions in operational research* 7 (6): 585–94. ISSN: 1475-3995.

Constantinou, A. C., N. E. Fenton, and M. Neil. 2012. "Pi-football: A Bayesian network model for forecasting Association Football match outcomes." *Knowledge-Based Systems* 36:322–39. https://doi.org/10.1016/j.knosys.2012.07.008. http://www.scopus.com/inward/record.url?eid=2-s2.0-84867880743&partnerID=40&md5=266efd1dbc55a553ac66beec03a20fb6.

Constantinou, A. C. 2012. "Bayesian networks for prediction, risk assessment and decision making in an inefficient Association Football gambling market." Thesis.

Courneya, K. S., and A. V. Carron. 1992. "The home advantage in sport competitions: a literature review." *Journal of Sport & Exercise Psychology* 14 (1).

Crowder, M., M. Dixon, A. Ledford, and M. Robinson. 2002. "Dynamic modelling and prediction of English Football League matches for betting." *Journal of the Royal Statistical Society: Series D (The Statistician)* 51 (2): 157–68.

Csató, L. 2020. "The UEFA Champions League seeding is not strategy-proof since the 2015/16 season." *Annals of Operations Research* 292 (1): 161–69.

Delen, D., D. Cogdell, and N. Kasap. 2012. "A comparative analysis of data mining methods in predicting NCAA bowl outcomes." *International Journal of Forecasting* 28 (2): 543–52. ISSN: 0169-2070. https://doi.org/http://dx.doi.org/10.1016/j.ijforecast.2011.05.002. http://www.sciencedirect.com/science/article/pii/S0169207011000914.

Duckworth, F., and T. Lewis. 2004. *Your Comprehensive Guide to the Duckworth/Lewis Method for Resetting Targets in One-day Cricket:(standard Ed.)* Acumen Books.

Duffield, R., and P. M. Fowler. 2017. "Domestic and international travel: implications for performance and recovery in team-sport athletes." *Sport, recovery, and performance: Interdisciplinary insights.*

Elo, A., and S. Sloan. 2008. *The Rating of Chess Players, Past and Present.* Ishi Press International. ISBN: 9780923891275. https://books.google.com.au/books?id=syjcPQAACAAJ.

Fowler, P., R. Duffield, and J. Vaile. 2014. "Effects of domestic air travel on technical and tactical performance and recovery in soccer." *International Journal of Sports Physiology and Performance* 9 (3): 378–86.

Friedman, J., T. Hastie, R. Tibshirani, et al. 2000. "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)." *The annals of statistics* 28 (2): 337–407.

Goddard, J. 2005. "Regression models for forecasting goals and match results in association football." *International Journal of Forecasting* 21 (2): 331–40. ISSN: 0169-2070. https://doi.org/http://dx.doi.org/10.1016/j.ijforecast.2004.08.002. http://www.sciencedirect.com/science/article/pii/S0169207004000676.

Goulet, V., C. Dutang, M. Maechler, D. Firth, M. Shapira, and M. Stadelmann. 2017. *expm: Matrix Exponential, Log, 'etc'.* R package version 0.999-2. https://CRAN.R-project.org/package=expm.

Gréhaigne, J.-F., and P. Godbout. 2014. "Dynamic systems theory and team sport coaching." *Quest* 66 (1): 96–116.

Guerrero-Calderon, B., A. Owen, J. A. Morcillo, and A. Castillo-Rodriguez. 2021. "How does the mid-season coach change affect physical performance on top soccer players?" *Physiology & Behavior* 232:113328.

Harville, D. 1980. "Predictions for National Football League Games via Linear-Model Methodology." *Journal of the American Statistical Association* 75 (371): 516–24. ISSN: 0162-1459. https://doi.org/10.1080/01621459.1980.10477504.

Heo, M., and A. C. Leon. 2005. "Comparison of statistical methods for analysis of clustered binary observations." *Statistics in medicine* 24 (6): 911–23.

Hornik, K., C. Buchta, and A. Zeileis. 2009. "Open-source machine learning: R meets Weka." *Computational Statistics* 24 (2): 225–32. https://doi.org/10.1007/s00180-008-0119-7.

Horvat, T., and J. Job. 2020. "The use of machine learning in sport outcome prediction: A review." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 10 (5): e1380.

Hosmer Jr, D. W., S. Lemeshow, and R. X. Sturdivant. 2013. *Applied logistic regression.* Vol. 398. John Wiley & Sons.

Howard, R. A. 2012. *Dynamic probabilistic systems: Markov models.* Vol. 1. Courier Corporation.

Hughes, M. D., and R. M. Bartlett. 2002. "The use of performance indicators in performance analysis." *Journal of sports sciences* 20 (10): 739–54.

Hvattum, L. M., and H. Arntzen. 2010. "Using ELO ratings for match result prediction in association football." *International Journal of forecasting* 26 (3): 460–70.

Ibe, O. C. 2013. "5 - Continuous-Time Markov Chains." In *Markov Processes for Stochastic Modeling (Second Edition),* Second Edition, edited by O. C. Ibe, 85–102. Oxford: Elsevier. ISBN: 978-0-12-407795-9. https://doi.org/https://doi.org/10.1016/B978-0-12-407795-9.00005-0. https://www.sciencedirect.com/science/article/pii/B9780124077959000050.

Jackson, C. H. 2011. "Multi-State Models for Panel Data: The msm Package for R." *Journal of Statistical Software* 38 (8): 1–29. http://www.jstatsoft.org/v38/i08/.

Johnston, R. D., G. M. Black, P. W. Harrison, N. B. Murray, and D. J. Austin. 2018. "Applied sport science of Australian football: a systematic review." *Sports Medicine* 48 (7): 1673–94.

Jones, M. I., and C. Harwood. 2008. "Psychological momentum within competitive soccer: Players' perspectives." *Journal of Applied Sport Psychology* 20 (1): 57–72.

Jones, N. M., S. D. Mellalieu, and N. James. 2004. "Team performance indicators as a function of winning and losing in rugby union." *International Journal of Performance Analysis in Sport* 4 (1): 61–71.

Josman, C., R. Gupta, and S. Robertson. 2016a. "Fixture Difficulty and Team Performance Models for use in the Australian Football League." In *Proceedings of the 13th Australasian Conference on Mathematics and Computers in Sport,* 15–20. ANZIAM MathSport. ISBN: 978-0-646-95741-8.

———. 2020a. "Markov Chain Models for the Near Real-Time Forecasting of Australian Football League Match Outcomes." In *Soft Computing for Problem Solving 2019,* 111–25. Springer. https://doi.org/10.1007/978-981-15-3287-0_9.

Landwehr, N., M. Hall, and E. Frank. 2005. "Logistic model trees." *Machine Learning* 59 (1-2): 161–205. ISSN: 0885-6125.

Law, A. M., W. D. Kelton, and W. D. Kelton. 1991. *Simulation modeling and analysis.* Vol. 2. McGraw-Hill New York.

Lenten, L. J. 2011. "The extent to which unbalanced schedules cause distortions in sports league tables." *Economic Modelling* 28 (1): 451–58.

Leung, C. K., and K. W. Joseph. 2014. “Sports data mining: predicting results for the college football games.” Leung, Carson K. Joseph, Kyle W. Jedrzejowicz, P Czarnowski, I Howlett, RJ Jain, LC 18th Annual International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES) SEP 15-17, 2014 Gdynia, POLAND Gdynia Maritime Univ, KES Int, *Knowledge-Based and Intelligent Information & Engineering Systems 18th Annual Conference, Kes-2014* 35:710–19. ISSN: 1877-0509. https://doi.org/10.1016/j.procs.2014.08.153. %3CGo%20to%20ISI%3E://WOS:000345394100073.

Leushuis, C. 2018. “Beating the Odds-A State Space Model for predicting match results in the Australian Football League.” *MaRBLe* 2.

Liaw, A., and M. Wiener. 2002. “Classification and Regression by randomForest.” *R News* 2 (3): 18–22. http://CRAN.R-project.org/doc/Rnews/.

Lin, L. Y., A. Pecotich, and K. B. Yap. 2011. “The effect of coaching succession in Australian Rules Football.”

Logofet, D. O., and E. V. Lesnaya. 2000. “The mathematics of Markov models: what Markov chains can really predict in forest successions.” *Ecological Modelling* 126 (2): 285–98. ISSN: 0304-3800. https://doi.org/https://doi.org/10.1016/S0304-3800(00)00269-6. http://www.sciencedirect.com/science/article/pii/S0304380000002696.

Lopez, M. J., and G. Matthews. 2014. “Building an NCAA mens basketball predictive model and quantifying its success.” *arXiv preprint arXiv:1412.0248.*

Manner, H. 2016. “Modeling and forecasting the outcomes of NBA basketball games.” *Journal of Quantitative Analysis in Sports* 12 (1): 31–41.

Marshall, G., and R. H. Jones. 1995. “Multi-state models and diabetic retinopathy.” *Statistics in medicine* 14 (18): 1975–83.

Maszczyk, A., A. Gołas, P. Pietraszewski, R. Roczniok, A. Zając, and A. Stanula. 2014. “Application of Neural and Regression Models in Sports Results Prediction.” *Procedia - Social and Behavioral Sciences* 117 (0): 482–87. ISSN: 1877-0428. https://doi.org/http://dx.doi.org/10.1016/j.sbspro.2014.02.249. http://www.sciencedirect.com/science/article/pii/S1877042814017790.

McCabe, A., and J. Trevathan. 2008. “Artificial Intelligence in Sports Prediction.” In *Fifth International Conference on Information Technology: New Generations, 2008,* 1194–97. IEEE. ISBN: 0769530990.

McHale, I., and A. Morton. 2011. “A Bradley-Terry type model for forecasting tennis match results.” *International Journal of Forecasting* 27 (2): 619–30. ISSN: 0169-2070. https://doi.org/http://dx.doi.org/10.1016/j.ijforecast.2010.04.004. http://www.sciencedirect.com/science/article/pii/S0169207010001019.

McHale, I., and P. Scarf. 2011. “Modelling the dependence of goals scored by opposing teams in international soccer matches.” *Statistical Modelling* 11 (3): 219–36.

Meyer, D., and F. T. Wien. 2014. “Support vector machines.” *The Interface to libsvm in package e1071.*

Microsoft Corporation and S. Weston. 2018. *doParallel: Foreach Parallel Adaptor for the 'parallel' Package.* R package version 1.0.14. https://CRAN.R-project.org/package=doParallel.

Min, B., J. Kim, C. Choe, H. Eom, and R. B. McKay. 2008. “A compound framework for sports results prediction: A football case study.” *Knowledge-Based Systems* 21 (7): 551–62. ISSN: 0950-7051.

Neath, R., and M. Johnson. 2010. “Discrimination and Classification.” In *International Encyclopedia of Education (Third Edition),* Third Edition, edited by P. Peterson, E. Baker, and B. McGaw, 135–41. Oxford: Elsevier. ISBN: 978-0-08-044894-7. https://doi.org/https://doi.org/10.1016/B978-0-08-044894-7.01312-9. https://www.sciencedirect.com/science/article/pii/B9780080448947013129.

O'Shaughnessy, D. M. 2006. “Possession versus position: strategic evaluation in AFL.” *Journal of sports science & medicine* 5 (4): 533.

Oh, M.-h., S. Keshri, and G. Iyengar. 2015. “Graphical model for baskeball match simulation.” In *2015 MIT Sloan Sports Analytics Conference.*

Pace, A., and A. V. Carron. 1992. “Travel and the home advantage.” *Canadian Journal of Sport Sciences.*

Pedersen, P. M. 2014. “The changing role of sports media producers.” In *Routledge handbook of sport and new media,* 119–27. Routledge.

Pennings, M. 2012. *Origins of Australian football: Victoria's early history, Volume 1: Amateur heroes and the rise of clubs, 1858 to 1876.* Connor Court Publishing.

Penny, W. D., and S. J. Roberts. 1999. “Dynamic Logistic Regression.”

Pisner, D. A., and D. M. Schnyer. 2020. “Support vector machine.” In *Machine learning,* 101–21. Elsevier.

Prakash, J. S., K. A. Vignesh, C. Ashok, and R. Adithyan. 2012. "Multi class Support Vector Machines classifier for machine vision application." In *Machine Vision and Image Processing (MVIP), 2012 International Conference on,* 197–99. IEEE.

R Core Team. 2018. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Rennie, M. J., S. J. Kelly, S. Bush, R. W. Spurrs, D. J. Austin, and M. L. Watsford. 2020. "Phases of match-play in professional Australian Football: Distribution of physical and technical performance." *Journal of sports sciences* 38 (14): 1682–89.

Richmond, L. K., B. Dawson, G. Stewart, S. Cormack, D. R. Hillman, and P. R. Eastwood. 2007. "The effect of interstate travel on the sleep patterns and performance of elite Australian Rules footballers." *Journal of Science and Medicine in Sport* 10 (4): 252–58.

Roane, H. S., M. E. Kelley, N. M. Trosclair, and L. S. Hauer. 2004. "Behavioral Momentum in Sports: A Partial Replication with Women's Basketball." *Journal of Applied Behavior Analysis* 37 (3): 385–90. ISSN: 1938-3703. https://doi.org/10.1901/jaba.2004.37-385. http://dx.doi.org/10.1901/jaba.2004.37-385.

Robertson, S., N. Back, and J. D. Bartlett. 2015. "Explaining match outcome in elite Australian Rules football using team performance indicators." *Journal of sports sciences,* no. ahead-of-print, 1–8.

Robertson, S., R. Gupta, and S. McIntosh. 2016. "A method to assess the influence of individual player performance distribution on match outcome in team sports." *Journal of sports sciences* 34 (19): 1893–900.

Robertson, S., and D. Joyce. 2018. "Evaluating strategic periodisation in team sport." *Journal of sports sciences* 36 (3): 279–85.

Robertson, S., and D. Joyce. 2015. "Informing in-season tactical periodisation in team sport: development of a match difficulty index for Super Rugby." *Journal of sports sciences* 33 (1): 99–107.

Rocaboy, Y., and M. Pavlik. 2020. "Performance expectations of professional sport teams and in-season head coach dismissals-Evidence from the English and French Men's Football first divisions." *Economies* 8 (4): 82.

Rue, H., and O. Salvesen. 2000. "Prediction and retrospective analysis of soccer matches in a league." *Journal of the Royal Statistical Society: Series D (The Statistician)* 49 (3): 399–418.

Russell, G. W. 1983. "Crowd size and density in relation to athletic aggression and performance." *Social Behavior and Personality: an international journal* 11 (1): 9–15.

Ryall, R. 2011. "Predicting Outcomes in Australian Rules Football." PhD diss., Citeseer.

Ryall, R., and A. Bedford. 2011. "The Intra-Match Home Advantage in Australian Rules Football." *Journal of Quantitative Analysis in Sports* 7 (2). https://doi.org/doi:10.2202/1559-0410.1314. https://doi.org/10.2202/1559-0410.1314.

Sheffer, M. L., and B. Schultz. 2013. "The new world of social media and broadcast sports reporting." In *Routledge handbook of sport communication,* 224–32. Routledge.

Stefani, R. 1980. "Improved least squares football, basketball, and soccer predictions." *IEEE transactions on systems, man, and cybernetics* 10 (2): 116–23.

———. 1987. "Applications of statistical methods to American football." *Journal of Applied Statistics* 14 (1): 61–73. ISSN: 0266-4763.

Stefani, R., and S. Clarke. 1992. "Predictions and home advantage for Australian rules football." *Journal of Applied Statistics* 19 (2): 251–61. ISSN: 0266-4763.

Stern, S. E. 2016. "The Duckworth-Lewis-Stern method: extending the Duckworth-Lewis methodology to deal with modern scoring rates." *Journal of the Operational Research Society* 67 (12): 1469–80.

Stewart, B., M. Nicholson, and G. Dickson. 2005. "The Australian Football League's recent progress: A study in cartel conduct and monopoly power." *Sport Management Review* 8 (2): 95–117.

Stewart, M., H. Mitchell, and C. Stavros. 2007. "Moneyball applied: Econometrics and the identification and recruitment of elite Australian footballers." *International Journal of Sport Finance* 2 (4): 231–48.

Štrumbelj, E., and P. Vračar. 2012. "Simulating a basketball match with a homogeneous Markov model and forecasting the outcome." *International Journal of Forecasting* 28 (2): 532–42. ISSN: 0169-2070. https://doi.org/http://dx.doi.org/10.1016/j.ijforecast.2011.01.004. http://www.sciencedirect.com/science/article/pii/S0169207011000458.

Taylor, J., and A. Demick. 1994. "A multidimensional model of momentum in sports." *Journal of Applied Sport Psychology* 6 (1): 51–70.

Ter Weel, B. 2011. "Does manager turnover improve firm performance? Evidence from Dutch soccer, 1986–2004." *De Economist* 159 (3): 279–303.

Tuck, G. N., and A. R. Whitten. 2013. "Lead us not into tanktation: A simulation modelling approach to gain insights into incentives for sporting teams to tank." *PloS one* 8 (11): e80798.

Venables, W. N., and B. D. Ripley. 2002. *Modern Applied Statistics with S.* ISBN 0-387-95457-0. New York: Springer. Book.

Watson, J. 2013. "Australian Football League: "home advantage", "umpire bias", or both?" *Sport, Business, and Management: An International Journal.*

Weedon, G., B. Wilson, L. Yoon, and S. Lawson. 2018. "Where's all the 'good' sports journalism? Sports media research, the sociology of sport, and the question of quality sports reporting." *International Review for the Sociology of Sport* 53 (6): 639–67.

Wickham, H. 2009. *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. ISBN: 978-0-387-98140-6. http://ggplot2.org.

Woods, C. T., A. J. Raynor, L. Bruce, Z. McDonald, and S. Robertson. 2016. "The application of a multi-dimensional assessment approach to talent identification in Australian football." *Journal of sports sciences* 34 (14): 1340–45.

Woods, C. T., and S. Robertson. 2021. "Is Playing at Home Really an Advantage? An Australian Football, Rugby League, and Rugby Union Perspective." In *Home Advantage in Sport,* 194–203. Routledge.

Young, C., W. Luo, P. Gastin, J. Tran, and D. Dwyer. 2019. "Modelling Match Outcome in Australian Football: Improved accuracy with large databases." *International Journal of Computer Science in Sport* 18 (1): 80–92.

Zeileis, A., and G. Grothendieck. 2005. "zoo: S3 Infrastructure for Regular and Irregular Time Series." *Journal of Statistical Software* 14 (6): 1–27. https://doi.org/10.18637/jss.v014.i06.

Zhou, Y., N. Fenton, and M. Neil. 2014. "Bayesian network approach to multinomial parameter learning using data and expert judgments." *International Journal of Approximate Reasoning* 55 (5): 1252–68. ISSN: 0888-613X. https://doi.org/http://dx.doi.org/10.1016/j.ijar.2014.02.008. http://www.sciencedirect.com/science/article/pii/S0888613X14000371.

Zhou, Z.-H. 2012. *Ensemble methods: foundations and algorithms.* Chapman / Hall/CRC.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear from any copyright owner who has been omitted or incorrectly acknowledged.

Static Data

A.1 Match Data

MatchData
26 Variables 593473 Observations

Season

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	121	1	1968	38	1908	1918	1942	1974	1998	2009	2012

`lowest : 1897 1898 1899 1900 1901, highest: 2013 2014 2015 2016 2017`

Round

n	missing	distinct
593473	0	29

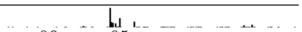
`lowest : 1  10 11 12 13, highest: EF GF PF QF SF`

Date

n	missing	distinct
593473	0	4649

```
lowest : 1897-05-08 1897-05-15 1897-05-22 1897-05-24 1897-05-29
highest: 2017-09-15 2017-09-16 2017-09-22 2017-09-23 2017-09-30
```

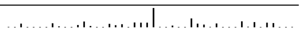
Local.start.time



n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	80	0.963	1489	138.8	1408	1410	1410	1420	1445	1910	1940

`lowest : 1030 1045 1100 1140 1210, highest: 2010 2015 2038 2040 2100`

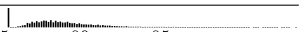
Venue



n	missing	distinct
593473	0	46

```
lowest : Adelaide Oval   Albury          Arden St        Bellerive Oval  Blacktown
highest: Western Oval    Windy Hill      Yallourn        Yarraville Oval York Park
```

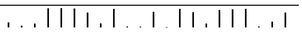
Attendance



n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	9051	0.999	23716	18562	0	3800	12250	20475	31517	44627	57849

`lowest :      0   1071   1327   2000   2127, highest: 116828 116956 118192 119165 121696`

Home.team



n	missing	distinct
593473	0	22

```
lowest : Adelaide        Brisbane Bears  Brisbane Lions  Carlton         Collingwood
highest: St Kilda        Sydney          University      West Coast      Western Bulldogs
```

X1Q1G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	15	0.974	3.121	2.091	0	1	2	3	4	6	7

lowest : 0 1 2 3 4, highest: 10 11 12 13 15

Value	0	1	2	3	4	5	6	7	8	9	10	11
Frequency	36477	85730	119437	123013	97417	66275	34933	18878	7914	2184	767	324
Proportion	0.061	0.144	0.201	0.207	0.164	0.112	0.059	0.032	0.013	0.004	0.001	0.001

Value	12	13	15
Frequency	40	40	44
Proportion	0.000	0.000	0.000

X1Q1B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	16	0.975	3.232	2.178	0	1	2	3	4	6	7

lowest : 0 1 2 3 4, highest: 11 12 13 14 15

Value	0	1	2	3	4	5	6	7	8	9	10	11
Frequency	34088	82048	117069	122606	95675	66675	38978	18921	10292	4351	1442	858
Proportion	0.057	0.138	0.197	0.207	0.161	0.112	0.066	0.032	0.017	0.007	0.002	0.001

Value	12	13	14	15
Frequency	196	116	118	40
Proportion	0.000	0.000	0.000	0.000

X1Q2G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	22	0.989	6.334	3.216	2	3	4	6	8	10	11

lowest : 0 1 2 3 4, highest: 17 18 19 20 21

X1Q2B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	23	0.988	6.476	3.093	2	3	5	6	8	10	11

lowest : 0 1 2 3 4, highest: 18 19 20 21 24

X1Q3G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	29	0.994	9.58	4.417	4	5	7	9	12	15	16

lowest : 0 1 2 3 4, highest: 24 25 26 27 28

X1Q3B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	28	0.993	9.731	4.049	4	5	7	10	12	15	16

lowest : 0 1 2 3 4, highest: 23 24 25 26 32

X1Q4G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	38	0.996	12.89	5.614	5	7	9	13	16	19	22

lowest : 0 1 2 3 4, highest: 33 34 35 36 37

X1Q4B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	34	0.995	12.97	4.894	6	8	10	13	16	19	21

lowest : 0 1 2 3 4, highest: 29 30 31 32 41

Home.score

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	208	1	90.3	35.51	41	51	68	88	110	132	145

lowest : 3 7 8 9 10, highest: 228 229 233 236 238

Away.team

n	missing	distinct
593473	0	22

lowest :	Adelaide	Brisbane Bears	Brisbane Lions	Carlton	Collingwood
highest:	St Kilda	Sydney	University	West Coast	Western Bulldogs

---

X2Q1G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	14	0.971	2.813	2.01	0	1	1	3	4	5	6

lowest : 0 1 2 3 4, highest: 9 10 11 12 13

Value	0	1	2	3	4	5	6	7	8	9	10	11
Frequency	48102	104530	131722	119437	86173	54628	28383	12580	5374	1728	648	84
Proportion	0.081	0.176	0.222	0.201	0.145	0.092	0.048	0.021	0.009	0.003	0.001	0.000

Value	12	13
Frequency	44	40
Proportion	0.000	0.000

X2Q1B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	14	0.972	2.971	2.068	0	1	2	3	4	5	6

lowest : 0 1 2 3 4, highest: 9 10 11 12 13

Value	0	1	2	3	4	5	6	7	8	9	10	11
Frequency	40259	97362	127664	119494	94168	56885	30921	15902	5965	2867	1366	312
Proportion	0.068	0.164	0.215	0.201	0.159	0.096	0.052	0.027	0.010	0.005	0.002	0.001

Value	12	13
Frequency	228	80
Proportion	0.000	0.000

X2Q2G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	21	0.987	5.71	3.064	2	2	4	5	7	9	10

lowest : 0 1 2 3 4, highest: 16 17 18 19 20

X2Q2B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	19	0.987	5.972	2.996	2	3	4	6	8	9	11

lowest : 0 1 2 3 4, highest: 14 15 16 17 18

Value	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Frequency	3045	13777	34221	56717	76286	90735	84832	74160	58735	41836	25147	16684	8168	4988
Proportion	0.005	0.023	0.058	0.096	0.129	0.153	0.143	0.125	0.099	0.070	0.042	0.028	0.014	0.008

Value	14	15	16	17	18
Frequency	2300	1167	400	119	156
Proportion	0.004	0.002	0.001	0.000	0.000

X2Q3G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	28	0.993	8.651	4.233	3	4	6	8	11	14	15

lowest : 0 1 2 3 4, highest: 23 24 25 26 29

X2Q3B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	28	0.992	8.948	3.929	4	5	6	9	11	14	15

lowest : 0 1 2 3 4, highest: 23 24 25 26 27

X2Q4G

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	37	0.996	11.6	5.297	4	6	8	11	15	18	20

lowest : 0 1 2 3 4, highest: 32 34 35 36 37

X2Q4B

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	35	0.995	11.91	4.718	5	7	9	12	15	17	19

lowest : 0 1 2 3 4, highest: 30 31 32 34 35

Away.score

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
593473	0	202	1	81.52	33.51	36	45	61	79	100	120	134

lowest : 1 2 3 5 6, highest: 211 216 222 231 239

A.2 Team Rankings

RankData
4 Variables 6729 Observations

Season

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
6729	0	18	0.997	2009	6.032	2000	2001	2004	2009	2013	2016	2017

```
lowest : 2000 2001 2002 2003 2004, highest: 2013 2014 2015 2016 2017
```

Value	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Frequency	352	352	352	352	352	352	352	352	352	352	352	373	414	414
Proportion	0.052	0.052	0.052	0.052	0.052	0.052	0.052	0.052	0.052	0.052	0.052	0.055	0.062	0.062

Value	2014	2015	2016	2017
Frequency	414	414	414	414
Proportion	0.062	0.062	0.062	0.062

Round

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
6729	0	23	0.998	11.69	7.449	2	3	6	12	17	21	22

```
lowest :  1  2  3  4  5, highest: 19 20 21 22 23
```

Team

n	missing	distinct
6729	0	34

```
lowest : AD         Adelaide   BL         Brisbane Lions   CA
highest: Sydney     WB         WC         West Coast       Western Bulldogs
```

Rank

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
6729	0	20	0.997	8.894	5.613	1	2	5	9	13	16	16

```
lowest :  1  2  3  4  5, highest: 16 17 18 19 20
```

Value	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Frequency	402	402	402	403	401	403	402	403	401	403	401	402	402	402
Proportion	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060	0.060

Value	15	16	17	18	19	20
Frequency	401	402	158	137	1	1
Proportion	0.060	0.060	0.023	0.020	0.000	0.000

A.3 Membership Numbers

Membership
24 Variables 20 Observations

Team

n	missing	distinct
20	0	20

```
lowest : Adelaide         Brisbane Bears   Brisbane Lions   Carlton          Collingwood
highest: Richmond         St Kilda         Sydney           West Coast       Western Bulldogs
```

X1995

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.996	13398	11257	0	0	6893	12470	18138	24132	27563

lowest : 0 6088 6893 8806 8870, highest: 18456 22543 23833 26821 41654

Value	0	6088	6893	8806	8870	9544	12212	12728	14027	14647	15922	18032	18456	22543
Frequency	3	1	2	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.15	0.05	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	23833	26821	41654
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X1996

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.996	14910	11381	0	0	10082	13670	20419	24660	28411

lowest : 0 7628 9525 10267 10650, highest: 20752 23278 24324 27681 42283

Value	0	7628	9525	10267	10650	12484	12964	14375	14438	17346	19622	20308	20752	23278
Frequency	3	1	1	2	1	1	1	1	1	1	1	1	1	1
Proportion	0.15	0.05	0.05	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	24324	27681	42283
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X1997

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	19117	13560	0	0	15276	19659	25489	33538	36088

lowest : 0 15054 15350 16610 16769, highest: 27005 28063 33286 35809 41395

Value	0	15054	15350	16610	16769	18858	19368	19949	22109	22761	24975	24984	27005	28063
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	33286	35809	41395
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X1998

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	16	0.992	21141	14354	0	0	17430	22695	27237	37577	38489

lowest : 0 16108 17870 19971 20064, highest: 27649 31089 37496 38305 41985

Value	0	16108	17870	19971	20064	20196	22186	23204	25402	27092	27099	27649	31089	37496
Frequency	4	1	1	1	1	1	1	1	1	1	2	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.10	0.05	0.05	0.05

Value	38305	41985
Frequency	1	1
Proportion	0.05	0.05

X1999

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	22086	14684	0	0	19018	23488	31411	36307	37414

lowest : 0 16931 19713 20491 20793, highest: 32120 32358 36212 37166 42120

Value	0	16931	19713	20491	20793	21032	22080	24896	25719	29047	29858	31175	32120	32358
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	36212	37166	42120
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2000

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	21925	14694	0	0	18006	25260	29243	35319	39069

lowest : 0 17855 18056 18227 20295, highest: 30177 34278 34925 38868 42896

Value	0	17855	18056	18227	20295	22156	24925	25595	26869	26879	27571	28932	30177	34278
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	34925	38868	42896
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2001

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	22368	14533	0	0	18896	24659	30469	36469	38817

lowest : 0 18330 19085 21409 22248, highest: 31455 33296 36227 38649 42014

Value	0	18330	19085	21409	22248	22940	23898	25420	26501	27725	28022	30140	31455	33296
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	36227	38649	42014
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2002

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	22477	15051	0	0	19538	23766	32742	35320	36749

lowest : 0 17696 20152 20831 20838, highest: 33319 34880 35219 36229 46620

Value	0	17696	20152	20831	20838	22288	23756	23775	26385	27251	27755	32549	33319	34880
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	35219	36229	46620
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2003

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	23159	15378	0	0	21084	24191	32359	36656	40787

lowest : 0 20555 21260 21270 21403, highest: 33525 35425 36234 40455 47097

Value	0	20555	21260	21270	21403	23626	24017	24365	25101	25368	31500	31970	33525	35425
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	36234	40455	47097
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2004

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	24739	15933	0	0	20309	28677	32952	40826	41354

lowest : 0 19295 20647 23420 25010, highest: 33469 36340 40792 41128 45642

Value	0	19295	20647	23420	25010	25021	27133	30221	30534	31255	32095	32780	33469	36340
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	40792	41128	45642
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2005

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	25325	15364	0	0	23609	29087	33695	38991	42449

lowest : 0 21974 24154 24805 24955, highest: 34178 36834 38612 42406 43256

Value	0	21974	24154	24805	24955	28029	28913	29261	30821	32043	32734	33534	34178	36834
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	38612	42406	43256
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2006

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	25956	15841	0	0	24680	29081	33295	38648	44438

lowest : 0 24624 24698 26042 26459, highest: 35648 35666 38038 44138 50138

Value	0	24624	24698	26042	26459	28003	28756	29406	30382	32290	32327	32511	35648	35666
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	38038	44138	50138
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2007

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	26635	16865	0	0	22269	30107	34413	43604	46200

lowest : 0 21976 22366 28077 28725, highest: 35431 38587 43343 45949 50976

Value	0	21976	22366	28077	28725	28764	30044	30169	30394	31064	32759	34073	35431	38587
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	43343	45949	50976
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2008

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	28699	17701	0	0	25725	31710	41564	43516	45056

lowest : 0 22737 26721 28306 29516, highest: 41947 42498 43366 44863 48720

Value	0	22737	26721	28306	29516	30063	30820	32600	34185	36850	39360	41436	41947	42498
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	43366	44863	48720
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2009

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	29337	18467	0	0	25920	31706	40911	46022	46773

lowest : 0 24873 26269 28215 28340, highest: 42408 43927 45972 46472 52496

Value	0	24873	26269	28215	28340	30605	31506	31906	36981	37160	39206	40412	42408	43927
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	45972	46472	52496
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2010

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	17	0.992	30713	19464	0	0	26910	34659	40507	46388	54150

lowest : 0 26779 26953 28671 29092, highest: 40589 44160 45545 53978 57408

Value	0	26779	26953	28671	29092	32077	33358	35960	39021	39854	40326	40480	40589	44160
Frequency	4	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.20	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	45545	53978	57408
Frequency	1	1	1
Proportion	0.05	0.05	0.05

X2011

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	18	0.997	32519	20865	0	0	25528	38107	42876	45870	56976

lowest : 0 11141 20792 27106 28761, highest: 43216 43791 44719 56224 71271

Value	0	11141	20792	27106	28761	29710	32581	36937	39276	39343	40184	42559	42762	43216
Frequency	3	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.15	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	43791	44719	56224	71271
Frequency	1	1	1	1
Proportion	0.05	0.05	0.05	0.05

X2012

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	19	0.999	35374	22492	0	9217	27595	35501	46295	57723	61433

lowest : 0 10241 11204 20762 29873, highest: 47780 53027 57377 60841 72688

Value	0	10241	11204	20762	29873	30007	33423	35440	35459	35543	40000	42918	45105	45800
Frequency	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	47780	53027	57377	60841	72688
Frequency	1	1	1	1	1
Proportion	0.05	0.05	0.05	0.05	0.05

X2013

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	19	0.999	37836	24176	0	11252	28689	38098	51966	60624	64107

lowest : 0 12502 12681 24130 30209, highest: 56173 58501 60321 63353 78427

Value	0	12502	12681	24130	30209	32707	33177	34607	36358	39838	42884	43880	46405	50564
Frequency	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	56173	58501	60321	63353	78427
Frequency	1	1	1	1	1
Proportion	0.05	0.05	0.05	0.05	0.05

X2014

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	19	0.999	37619	24229	0	10526	25778	39333	51917	63687	65828

lowest : 0 11696 12806 23247 26622, highest: 53026 55700 63486 65494 72170

Value	0	11696	12806	23247	26622	29332	33419	34716	38000	40666	45911	46549	48000	51547
Frequency	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	53026	55700	63486	65494	72170
Frequency	1	1	1	1	1
Proportion	0.05	0.05	0.05	0.05	0.05

X2015

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	19	0.999	41807	26006	0	12132	30912	45809	55598	71021	73030

lowest : 0 13480 13643 25408 32746, highest: 60221 60818 70809 72924 75037

Value	0	13480	13643	25408	32746	35222	35953	41012	44312	47305	48836	51433	52920	54057
Frequency	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	60221	60818	70809	72924	75037
Frequency	1	1	1	1	1
Proportion	0.05	0.05	0.05	0.05	0.05

X2016

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	19	0.999	43760	26220	0	11569	34328	50351	56766	72515	74678

lowest : 0 12854 15312 23286 38009, highest: 57494 65188 72278 74643 75351

Value	0	12854	15312	23286	38009	39146	39459	45014	50130	50571	51889	53743	54307	56523
Frequency	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	57494	65188	72278	74643	75351
Frequency	1	1	1	1	1
Proportion	0.05	0.05	0.05	0.05	0.05

X2017

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
20	0	19	0.999	45378	26737	0	10499	35598	50790	60395	72968	75674

lowest : 0 11665 20944 21362 40343, highest: 65064 67768 72669 75663 75879

Value	0	11665	20944	21362	40343	42052	42233	47653	50326	51254	52129	54854	56865	58838
Frequency	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Proportion	0.10	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Value	65064	67768	72669	75663	75879
Frequency	1	1	1	1	1
Proportion	0.05	0.05	0.05	0.05	0.05

A.4 Home Grounds

HomeGrounds
9 Variables 21 Observations

Team

n	missing	distinct
21	0	21

```
lowest : Adelaide        Brisbane Bears   Brisbane Lions   Carlton          Collingwood
highest: St Kilda        Sydney           University       West Coast       Western Bulldogs
```

Home1

n	missing	distinct
21	0	10

```
lowest : Adelaide Oval     Brunswick St      Carrara           Docklands         East Melbourne
highest: Gabba             Kardinia Park     M.C.G.            Stadium Australia Subiaco
```

```
Adelaide Oval (2, 0.095), Brunswick St (1, 0.048), Carrara (2, 0.095), Docklands (5, 0.238),
East Melbourne (1, 0.048), Gabba (1, 0.048), Kardinia Park (1, 0.048), M.C.G. (4, 0.190),
Stadium Australia (2, 0.095), Subiaco (2, 0.095)
```

Home2

n	missing	distinct
18	3	16

```
lowest : Bellerive Oval    Cazaly's Stadium  Corio Oval        East Melbourne    Football Park
highest: Sydney Showground Victoria Park     W.A.C.A.          Westpac Stadium   York Park
```

```
Bellerive Oval (1, 0.056), Cazaly's Stadium (1, 0.056), Corio Oval (1, 0.056), East Melbourne
(1, 0.056), Football Park (2, 0.111), Gabba (1, 0.056), Junction Oval (1, 0.056), Marrara Oval
(1, 0.056), Princes Park (1, 0.056), Punt Rd (1, 0.056), S.C.G. (1, 0.056), Sydney Showground
(1, 0.056), Victoria Park (1, 0.056), W.A.C.A. (2, 0.111), Westpac Stadium (1, 0.056), York
Park (1, 0.056)
```

Home3

n	missing	distinct
12	9	10

```
lowest : Arden St            Brisbane Exhibition Domain Stadium      Euroa               Glenferrie Oval
highest: Junction Oval       Manuka Oval         Princes Park        Traeger Park        Windy Hill
```

```
Arden St (1, 0.083), Brisbane Exhibition (1, 0.083), Domain Stadium (2, 0.167), Euroa (1,
0.083), Glenferrie Oval (1, 0.083), Junction Oval (1, 0.083), Manuka Oval (1, 0.083), Princes
Park (2, 0.167), Traeger Park (1, 0.083), Windy Hill (1, 0.083)
```

Home4

n	missing	distinct
10	11	9

```
lowest : Blacktown           Brisbane Exhibition Coburg Oval         Etihad Stadium      Moorabbin Oval
highest: Moorabbin Oval      Princes Park        Punt Rd             UNSW Canberra Oval  Victoria Park
```

```
Blacktown (1, 0.1), Brisbane Exhibition (1, 0.1), Coburg Oval (1, 0.1), Etihad Stadium (2,
0.2), Moorabbin Oval (1, 0.1), Princes Park (1, 0.1), Punt Rd (1, 0.1), UNSW Canberra Oval (1,
0.1), Victoria Park (1, 0.1)
```

Home5

n	missing	distinct
7	14	6

```
lowest : Albury          Etihad Stadium  Mars Stadium    Simonds Stadium Waverley Park
highest: Etihad Stadium  Mars Stadium    Simonds Stadium Waverley Park   Western Oval
```

Value	Albury	Etihad Stadium	Mars Stadium	Simonds Stadium	Waverley Park
Frequency	1	1	1	1	2
Proportion	0.143	0.143	0.143	0.143	0.286

Value	Western Oval
Frequency	1
Proportion	0.143

Home6

n	missing	distinct
4	17	4

Value	Bruce Stadium	Etihad Stadium	Euroa	Toorak Park
Frequency	1	1	1	1
Proportion	0.25	0.25	0.25	0.25

Home7

n	missing	distinct	value
1	20	1	Yallourn

Value	Yallourn
Frequency	1
Proportion	1

Home8

n	missing	distinct	value
1	20	1	Etihad Stadium

Value	Etihad Stadium
Frequency	1
Proportion	1

Champion Data Statistics

B.1 Summary of Raw Champion Data

afl.club.trx
67 Variables 739224 Observations

FIXED_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25
739224	0	225	1	100741592	1463	100740202	100740303	100740607

.50	.75	.90	.95
100741301	100741902	100742207	100742308

lowest : 100740101 100740102 100740103 100740104 100740105
highest: 100750201 100750202 100750301 100750302 100750401

MATCH_DATE

n	missing	distinct
739224	0	89

lowest : 2017-03-23 2017-03-24 2017-03-25 2017-03-26 2017-03-30
highest: 2017-09-15 2017-09-16 2017-09-22 2017-09-23 2017-09-30

MATCH_TIME

n	missing	distinct
739224	0	24

lowest : 13:10 13:15 13:40 13:45 14:10, highest: 19:10 19:20 19:25 19:50 20:00

SEASON_ID

n	missing	distinct	Info	Mean	Gmd
739224	0	1	0	2017	0

Value	2017
Frequency	739224
Proportion	1

GROUP_ROUND_NO

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739224	0	27	0.998	12.52	8.218	2	3	6	13	19	22	23

lowest : 1 2 3 4 5, highest: 23 24 25 26 27

VENUE_NAME

n	missing	distinct
739224	0	18

lowest : Adelaide Oval Blundstone Arena Cazaly's Stadium Domain Stadium
highest: Spotless Stadium TIO Stadium TIO Traeger Park University of Tasmania Stadium UNSW Canberra Oval

---

HOME_SQUAD

n	missing	distinct
739224	0	18

lowest : Adelaide Crows Brisbane Lions Carlton Collingwood Essendon
highest: Richmond St Kilda Sydney Swans West Coast Eagles Western Bulldogs

---

HOME_SCORE

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739206	18	84	1	92.69	28.09	56	61	77	89	110	127	138

```
lowest :  38  40  44  48  50, highest: 143 145 147 153 160
```

AWAY_SQUAD

n	missing	distinct
739224	0	19

```
lowest : Adelaide Crows    Brisbane Lions    BYE               Carlton           Collingwood
highest: Richmond          St Kilda          Sydney Swans      West Coast Eagles Western Bulldogs
```

AWAY_SCORE

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739206	18	86	1	85.69	27.12	47	56	70	84	100	117	130

```
lowest :  20  39  42  43  45, highest: 146 150 153 155 163
```

MATCH_TRX_ID

```
n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50
739206      18    11795        1  1525364  1258789    11800    24410    65310  1067405
.75      .90      .95
2066400  3036400  3048800

Value            0   50000 1000000 1050000 1100000 2000000 2050000 3000000 3050000 4000000
Frequency    75637  111844   75903  107016      28   75853  109309   76040  107100     255
Proportion   0.102   0.151   0.103   0.145   0.000   0.103   0.148   0.103   0.145   0.000

Value      5000000
Frequency      221
Proportion   0.000
```

SEQUENCE

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739206	18	10	0.351	1.338	0.6245	1	1	1	1	1	2	3

```
Value           1      2      3      4      5      6      7      8     10     11
Frequency  639795  60020   6446   6051   6050   6049   6044   6041   1421   1289
Proportion  0.866  0.081  0.009  0.008  0.008  0.008  0.008  0.008  0.002  0.002
```

PERIOD

n	missing	distinct	Info	Mean	Gmd
739206	18	6	0.938	2.495	1.254

```
Value           1      2      3      4      5      6
Frequency  187481 182947 185162 183140    255    221
Proportion  0.254  0.247  0.250  0.248  0.000  0.000
```

PERIOD_SECS

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739206	18	2232	1	901.4	629.1	58	151	430	896	1366	1649	1755

```
lowest :    0    1    2    3    4, highest: 2324 2328 2332 2337 2339
```

STATISTIC_CODE

n	missing	distinct
739224	0	173

```
lowest :       BALKD BAULK BEHI  BHAS , highest: TIHO  TIHSD TIHSK TISM  TIVS
```

PERSON_ID

```
n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50
677629   61595      661        1   374038   182112   240072   250222   270917   291533
.75      .90      .95
296078   992462   996483

Value       200000  210000  220000  230000  240000  250000  260000  270000  280000  290000
Frequency     1153    4813   12579   11491   28757   36215   48292   49383   77748  207815
Proportion   0.002   0.007   0.019   0.017   0.042   0.053   0.071   0.073   0.115   0.307

Value       300000  990000 1000000 1010000
Frequency   109066   50704   39253     360
Proportion   0.161   0.075   0.058   0.001
```

FULLNAME

n	missing	distinct
739224	0	662

```
lowest :                Aaron Black   Aaron Francis Aaron Hall    Aaron Mullett
highest: Zach Guthrie   Zach Merrett  Zach Tuohy    Zaine Cordy   Zak Jones
```

SQUAD_NAME

n	missing	distinct
739224	0	19

```
lowest :                  Adelaide Crows    Brisbane Lions    Carlton           Collingwood
highest: Richmond         St Kilda          Sydney Swans      West Coast Eagles Western Bulldogs
```

OPP_SQUAD

n	missing	distinct
739224	0	19

```
lowest :                  Adelaide Crows    Brisbane Lions    Carlton           Collingwood
highest: Richmond         St Kilda          Sydney Swans      West Coast Eagles Western Bulldogs
```

AR_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
18759	720465	164	0.999	328503	112852	250088	250298	270811	280763

.75	.90	.95
293957	298174	992752

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	365	276	48	143	1367	2169	1868	4392	4755	2035
Proportion	0.019	0.015	0.003	0.008	0.073	0.116	0.100	0.234	0.253	0.108

Value	990000	1000000
Frequency	550	791
Proportion	0.029	0.042

AR

n	missing	distinct
739224	0	165

```
lowest :                 Aaron Francis     Aaron Sandilands Adam Treloar     Alex Rance
highest: Will Langford    Wylie Buzza       Zac Smith        Zac Williams     Zaine Cordy
```

H1_ID 

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6052	733172	225	1	354534	157530	230231	240712	270896	290683

.75	.90	.95
294429	990704	993979

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	37	183	112	280	380	434	679	598	1902	789
Proportion	0.006	0.030	0.019	0.046	0.063	0.072	0.112	0.099	0.314	0.130

Value	990000	1000000
Frequency	389	269
Proportion	0.064	0.044

H1

n	missing	distinct
739224	0	226

```
lowest :                 Aaron Hall        Adam Treloar     Alex Neal-Bullen Andrew Gaff
highest: Will Langford    Will Setterfield Zac Williams     Zach Merrett     Zak Jones
```

H2_ID 

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6050	733174	229	1	354835	156793	240052	250105	270908	290671

.75	.90	.95
294472	990704	993979

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	40	122	131	290	358	452	644	699	1877	780
Proportion	0.007	0.020	0.022	0.048	0.059	0.075	0.106	0.116	0.310	0.129

Value	990000	1000000
Frequency	413	244
Proportion	0.068	0.040

H2

n	missing	distinct
739224	0	230

```
lowest :                 Aaron Hall        Aaron Young      Adam Treloar     Alex Neal-Bullen
highest: Will Langford    Will Setterfield Zac Williams     Zach Merrett     Zak Jones
```

H3_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6044	733180	237	1	367292	176055	230231	240417	270896	290832

.75	.90	.95
295445	992462	996701

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	57	155	100	320	308	478	577	590	1888	809
Proportion	0.009	0.026	0.017	0.053	0.051	0.079	0.095	0.098	0.312	0.134

Value	990000	1000000
Frequency	389	373
Proportion	0.064	0.062

H3

n	missing	distinct
739224	0	238

lowest :		Aaron Hall	Aaron Young	Adam Treloar	Alex Neal-Bullen
highest:	Will Langford	Will Setterfield	Zac Williams	Zach Merrett	Zak Jones

---

A1_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6051	733173	240	1	365569	171655	240027	240712	270912	290847

.75	.90	.95
295467	992016	994539

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	43	146	113	306	247	443	639	587	1935	850
Proportion	0.007	0.024	0.019	0.051	0.041	0.073	0.106	0.097	0.320	0.140

Value	990000	1000000	1010000
Frequency	445	292	5
Proportion	0.074	0.048	0.001

A1

n	missing	distinct
739224	0	241

lowest :		Aaron Hall	Aaron Young	Adam Treloar	Alex Neal-Bullen
highest:	Will Brodie	Will Langford	Zac Williams	Zach Merrett	Zak Jones

---

A2_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6049	733175	236	1	354162	152570	240124	250298	270912	290778

.75	.90	.95
295136	990704	993979

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	31	90	84	281	339	473	616	708	1897	890
Proportion	0.005	0.015	0.014	0.046	0.056	0.078	0.102	0.117	0.314	0.147

Value	990000	1000000	1010000
Frequency	376	259	5
Proportion	0.062	0.043	0.001

A2

n	missing	distinct
739224	0	237

lowest :		Aaron Hall	Adam Treloar	Alex Neal-Bullen	Alex Sexton
highest:	Will Langford	Will Setterfield	Zac Williams	Zach Merrett	Zak Jones

---

A3_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6041	733183	243	1	365062	172414	240124	250105	261911	290778

.75	.90	.95
295313	992016	996483

Value	210000	220000	230000	240000	250000	260000	270000	280000	290000	300000
Frequency	40	88	134	323	372	565	475	682	1800	819
Proportion	0.007	0.015	0.022	0.053	0.062	0.094	0.079	0.113	0.298	0.136

Value	990000	1000000	1010000
Frequency	424	314	5
Proportion	0.070	0.052	0.001

A3

n	missing	distinct
739224	0	244

lowest :		Aaron Hall	Aaron Young	Adam Treloar	Alex Neal-Bullen
highest:	Will Langford	Will Setterfield	Zac Williams	Zach Merrett	Zak Jones

---

ZONE_LOGICAL_AFL

n	missing	distinct
739224	0	7

Value		AM	CB	D50	DM	F50	U
Frequency	26	194032	31745	131139	182237	109303	90742
Proportion	0.000	0.262	0.043	0.177	0.247	0.148	0.123

ZONE_PHYSICAL_AFL

n	missing	distinct
739224	0	6

Value		L50	LM	M	R50	RM
Frequency	18	125665	192727	99253	129832	191729
Proportion	0.000	0.170	0.261	0.134	0.176	0.259

TRUEX [2]

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
348617	390607	1700	1	0.6205	48.32	-65.1	-56.3	-35.3	0.2	36.6	58.0	66.5

lowest : -86.6 -86.4 -86.0 -85.6 -85.4, highest: 85.8 85.9 86.1 86.3 86.6

TRUEY [2]

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
348617	390607	1413	1	-0.5899	37.99	-51.7	-44.8	-28.2	-1.0	26.2	45.4	53.0

lowest : -70.6 -70.5 -70.4 -70.3 -70.2, highest: 70.2 70.3 70.4 70.5 70.6

VENUE_LENGTH

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739206	18	11	0.88	162.9	5.777	155	156	160	160	167	175	175

Value	155	156	160	161	162	164	165	167	168	170	175
Frequency	42126	38089	360141	35906	11355	32070	3355	91248	3714	39477	81725
Proportion	0.057	0.052	0.487	0.049	0.015	0.043	0.005	0.123	0.005	0.053	0.111

VENUE_WIDTH

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
739206	18	13	0.97	131.3	8.563	122	122	123	129	140	141	141

Value	115	122	123	124	128	129	132	134	135	136	138	140
Frequency	25032	78033	91248	11248	32070	171136	3714	35906	7047	42126	49444	14445
Proportion	0.034	0.106	0.123	0.015	0.043	0.232	0.005	0.049	0.010	0.057	0.067	0.020

Value	141
Frequency	177757
Proportion	0.240

STDX [2]

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
348617	390607	1601	1	0.6105	47.6	-64.1	-55.3	-35.0	0.2	36.2	57.1	65.5

lowest : -79.9 -79.8 -79.7 -79.6 -79.5, highest: 79.7 79.8 79.9 80.0 80.1

STDY [2]

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
348617	390607	1383	1	-0.5945	39.12	-53.8	-46.3	-28.5	-1.0	26.5	47.0	55.1

lowest : -69.1 -69.0 -68.9 -68.8 -68.7, highest: 68.7 68.8 68.9 69.0 69.1

XY_FLIP

n	missing	distinct	Info	Mean	Gmd
739224	0	2	0.75	0.002746	1

Value	-1	1
Frequency	368597	370627
Proportion	0.499	0.501

INITIAL_TRX_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
737472	1752	3478	1	1524905	1258524	11400	23900	64900	1067100

.75	.90	.95
2065700	3035900	3048200

Value	0	50000	1000000	1050000	1100000	2000000	2050000	3000000	3050000	4000000
Frequency	76732	110234	77399	105144	11	77264	107585	77459	105172	254
Proportion	0.104	0.149	0.105	0.143	0.000	0.105	0.146	0.105	0.143	0.000

Value	5000000
Frequency	218
Proportion	0.000

FINAL_TRX_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
737472	1752	5254	1	1526061	1258525	12600	25300	65900	1068000

.75	.90	.95
2066900	3036910	3049500

Value	0	50000	1000000	1050000	1100000	2000000	2050000	3000000	3050000	4000000
Frequency	73179	113787	73960	108558	36	73603	111246	74050	108581	254
Proportion	0.099	0.154	0.100	0.147	0.000	0.100	0.151	0.100	0.147	0.000

Value	5000000
Frequency	218
Proportion	0.000

CHAIN_SQUAD

n	missing	distinct
739224	0	19

```
lowest :                 Adelaide Crows   Brisbane Lions    Carlton           Collingwood
highest: Richmond        St Kilda         Sydney Swans      West Coast Eagles Western Bulldogs
```

INITIAL_STATE

n	missing	distinct
739224	0	6

Value		BU	CB	KI	PG	TI
Frequency	1752	88394	150738	50371	318061	129908
Proportion	0.002	0.120	0.204	0.068	0.430	0.176

FINAL_STATE

n	missing	distinct
739224	0	9

Value		BEHI	BU	GOAL	OOB	ORUSH	PC	RUSH	TO
Frequency	8063	68387	68211	96063	91829	501	47315	13560	345295
Proportion	0.011	0.093	0.092	0.130	0.124	0.001	0.064	0.018	0.467

ZONE_LOGICAL_INITIAL

n	missing	distinct
739224	0	6

Value		AM	CB	D50	DM	F50
Frequency	3430	154400	148744	217265	172440	42945
Proportion	0.005	0.209	0.201	0.294	0.233	0.058

FINAL_ZONE_LOGICAL

n	missing	distinct
739224	0	6

Value		AM	CB	D50	DM	F50
Frequency	3430	168549	6265	26058	129315	405607
Proportion	0.005	0.228	0.008	0.035	0.175	0.549

LAUNCH_PERSON_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
717863	21361	648	1	367294	173436	240052	250267	270896	290832

.75	.90	.95
295461	992016	996442

Value	200000	210000	220000	230000	240000	250000	260000	270000	280000	290000
Frequency	981	4894	13986	14986	26974	40148	55902	57896	89646	215735
Proportion	0.001	0.007	0.019	0.021	0.038	0.056	0.078	0.081	0.125	0.301

Value	300000	990000	1000000	1010000
Frequency	107253	50215	39035	212
Proportion	0.149	0.070	0.054	0.000

LAUNCH_PLAYER

n	missing	distinct
739224	0	649

```
lowest :              Aaron Black   Aaron Francis Aaron Hall    Aaron Mullett
highest: Zach Guthrie Zach Merrett  Zach Tuohy    Zaine Cordy   Zak Jones
```

GUILTY_PERSON_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
342629	396595	650	1	387873	200309	240072	250134	270951	291784

.75	.90	.95
296322	993798	996580

Value	200000	210000	220000	230000	240000	250000	260000	270000	280000	290000
Frequency	894	2089	6211	5444	15646	17233	21061	22814	33178	107511
Proportion	0.003	0.006	0.018	0.016	0.046	0.050	0.061	0.067	0.097	0.314

Value	300000	990000	1000000	1010000
Frequency	58501	30176	21596	275
Proportion	0.171	0.088	0.063	0.001

GUILTY_PLAYER

n	missing	distinct
739224	0	651

```
lowest :              Aaron Black   Aaron Francis Aaron Hall    Aaron Mullett
highest: Zach Guthrie  Zach Merrett  Zach Tuohy    Zaine Cordy   Zak Jones
```

PARAM1

n	missing	distinct
739224	0	26

```
lowest :                          CENTRE                      CENTRE_BOUNCE_INFRINGEMENT CHOPPING_THE_ARMS          CORRIDOR
highest: PUSH_IN_BACK             RIGHT                       RUN_TOO_FAR                THROWING_THE_BALL          TRIP_SLIDE
```

PARAM2

n	missing	distinct
739224	0	16

```
(691267, 0.935), BOMB (3722, 0.005), DELIBERATE_SNAP (403, 0.001), GENERAL (7286, 0.010),
GO_TO (12593, 0.017), GO_TO_NO_CHANCE (1363, 0.002), MARK_PLAY_ON (417, 0.001), MARKING (3901,
0.005), OFF_GROUND (224, 0.000), ON_RUN_IN_GENERAL_PLAY (2381, 0.003), OTHER (3786, 0.005),
RUCK (825, 0.001), SCORE (94, 0.000), SET_SHOT (5036, 0.007), SNAP (2591, 0.004), TACKLING
(3335, 0.005)
```

PARAM3

n	missing	distinct
739224	0	8

Value		BOUNDARY_LEFT	BOUNDARY_RIGHT	DIRECTLY_IN_FRONT
Frequency	728172	321	322	1683
Proportion	0.985	0.000	0.000	0.002

Value	ZONE_1	ZONE_2	ZONE_3	ZONE_4
Frequency	1625	2968	2822	1311
Proportion	0.002	0.004	0.004	0.002

PARAM4

n	missing	distinct
739224	0	6

Value		M0_15	M15_30	M30_40	M40_50	M50_PLUS
Frequency	728172	1189	2506	2677	3094	1586
Proportion	0.985	0.002	0.003	0.004	0.004	0.002

KICK_FOOT

n	missing	distinct
739224	0	3

Value		Left	Right
Frequency	648558	25051	65615
Proportion	0.877	0.034	0.089

KICK_INTENT

n	missing	distinct
739224	0	9

Value		Backwards Lead	Covered	Distance	Goal
Frequency	648558	401	62418	3024	10919
Proportion	0.877	0.001	0.084	0.004	0.015

Value	Goal Smothered	Lead	Open	Pack
Frequency	86	5523	2412	5883
Proportion	0.000	0.007	0.003	0.008

KICK_DISTANCE

n	missing	distinct
739224	0	4

Value		Chip	Long	Short
Frequency	648555	11538	36670	42461
Proportion	0.877	0.016	0.050	0.057

KICK_DIRECTION

n	missing	distinct
739224	0	4

Value		Backward	Forward	Lateral
Frequency	648556	5528	78219	6921
Proportion	0.877	0.007	0.106	0.009

PRESSURE_LEVEL

n	missing	distinct
739224	0	7

Value		Chasing	Closing	Corralling	None	Physical	Set
Frequency	555504	4236	29326	46502	18241	40793	44622
Proportion	0.751	0.006	0.040	0.063	0.025	0.055	0.060

PRESSURE_PLAYER_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
120734	618490	656	1	402769	218947	240124	250298	271078	291806

.75	.90	.95
296420	993905	997033

Value	200000	210000	220000	230000	240000	250000	260000	270000	280000	290000
Frequency	144	785	1835	1694	5180	5777	8055	7429	12521	35732
Proportion	0.001	0.007	0.015	0.014	0.043	0.048	0.067	0.062	0.104	0.296

Value	300000	990000	1000000	1010000
Frequency	20782	11489	9184	127
Proportion	0.172	0.095	0.076	0.001

PRESSURE_PLAYER

n	missing	distinct
739224	0	657

lowest :		Aaron Black	Aaron Francis	Aaron Hall	Aaron Mullett
highest:	Zach Guthrie	Zach Merrett	Zach Tuohy	Zaine Cordy	Zak Jones

---

PRESSURE_PLAYER2_ID

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50
6508	732716	628	1	400481	214480	240226	250298	280109	291806

.75	.90	.95
296420	993903	997100

Value	200000	210000	220000	230000	240000	250000	260000	270000	280000	290000
Frequency	6	29	94	86	284	305	388	370	670	2057
Proportion	0.001	0.004	0.014	0.013	0.044	0.047	0.060	0.057	0.103	0.316

Value	300000	990000	1000000	1010000
Frequency	1125	573	508	13
Proportion	0.173	0.088	0.078	0.002

PRESSURE_PLAYER2

n	missing	distinct
739224	0	629

lowest :		Aaron Black	Aaron Francis	Aaron Hall	Aaron Mullett
highest:	Zach Guthrie	Zach Merrett	Zach Tuohy	Zaine Cordy	Zak Jones

---

PRESSURE_POINTS

n	missing	distinct	Info	Mean	Gmd
183720	555504	6	0.953	1.812	1.199

Value	0.75	1.00	1.20	1.50	2.25	3.75
Frequency	44622	18241	46502	4236	29326	40793
Proportion	0.243	0.099	0.253	0.023	0.160	0.222

B.2 Description of Supplied Transaction Data

Table B.1: Descriptions of Champion Data transactional data. (*Stats glossary: Every stat explained* 2017)

Statistic	Description
BAULK	Using deception as the ball carrier to beat an opponent, by sidestepping or feigning disposal.
BEHIND	A minor score, as judged by the goal umpire. Behinds are worth one point to a team's total score.
BEHIND ASSIST	Creating a behind by getting the ball to a teammate either via a disposal, knock-on, ground kick or hit-out, or by winning a free kick before the advantage is paid to the goal scorer.
BLOCK	Effectively shepherding an opponent out of a contest to the benefit of a teammate.
BROKEN TACKLE	Evading a tackle attempt by an opponent and legally disposing of the ball in space.
CLANGER HANDBALL	Handballs that give possession directly to the opposition.
CLANGER KICK	Kicks that give possession directly to the opposition.
CLEARANCE	Credited to the player who has the first effective disposal in a chain that clears the stoppage area, or an ineffective kick or clanger kick that clears the stoppage area.
CONTESTED KNOCK ON	Using the hand to knock the ball to a teammate's advantage rather than attempting to take possession from a contested situation.
CONTESTED MARK	When a player takes a mark under physical pressure of an opponent or in a pack.
CONTESTED MARK FROM OPP	
CONTESTED MARK FROM TEAM	
CONTESTED POSSESSION	A possession which has been won when the ball is in dispute. Includes looseball-gets, hardball-gets, contested marks, gathers from a hit-out and frees for.
CRUMB	A type of groundball-get that is won by a player at ground level after a marking contest. The player must not be involved in the original contest. Crumbing Possessions can be either hardball or looseball-gets.
DISPOSAL	Legally getting rid of the ball, via a handball or kick.
EFFECTIVE DISPOSAL	
EFFECTIVE HANDBALL	A handball to a teammate that hits the intended target.
EFFECTIVE KICK	A kick of more than 40 metres to a 50/50 contest or better for the team or a kick of less than 40 metres that results in the intended target retaining possession.
FIRST POSSESSION	The initial possession that follows a stoppage, including a looseball-get, hardball-get, intended ball-get (gather), free kick or ground kick.
FREE AGAINST	When an infringement occurs resulting in the opposition receiving a free kick from the umpires.
FREE FOR	When a player is interfered with and is awarded a free kick by the umpires.
GATHER	Possessions that were a result of a teammate deliberately directing the ball in the player's direction, via a hit-out, disposal or knock-on, excluding marks and handball receives. Gathers from a hit-out are contested possessions the rest are uncontested.
GATHER FROM HIT-OUT	A possession gained from a teammate's hit-out to advantage. Counted as a contested possession.
GOAL	A major score, as judged by the goal umpire. Worth six points to a team's total score.

GOAL ASSIST	Creating a goal by getting the ball to a teammate either via a disposal, knock-on, ground kick or hitout, or by winning a free kick before the advantage is paid to the goal scorer.
GROUND BALL GET	Contested possessions won at ground level, excluding free kicks. Groundball gets can either be hardball gets or looseball gets.
GROUND KICK	A deliberate kick without taking possession that gains either significant distance from the point of contact or an uncontested possession for a teammate.
HANDBALL	Disposing of the ball by hand.
HARDBALL GET	A disputed ball at ground level under direct physical pressure or out of a ruck contest, resulting in an opportunity to effect a legal disposal.
HIT-OUT	Knocking the ball out of a ruck contest following a stoppage with clear control, regardless of which side wins the following contest at ground level.
HIT-OUT SHARK	Winning clear possession of the ball from the opposition ruck's hit-out.
HIT-OUT SHARKED	A hit-out that directly results in an opponent's possession.
HIT-OUT TO ADVANTAGE	A hit-out that reaches an intended teammate.
HOLD	Holding the ball in when the umpire calls for a ball up.
INEFFECTIVE GROUND KICK	Ground kicks that are not advantageous to the team, but do not directly turn the ball over to the opposition.
INEFFECTIVE HANDBALL	Handballs that are not advantageous to the team, but do not directly turn the ball over to the opposition.
INEFFECTIVE KICK	Kicks that are not advantageous to the team, but do not directly turn the ball over to the opposition.
INSIDE 50	Moving the ball from the midfield into the forward zone. Excludes multiple entries within the same chain of possession.
INSIDE 50 TARGET	Recorded when a player inside the forward 50 is clearly the sole target of a teammate's kick into the forward 50. The inside 50 target player will be recorded regardless of the outcome of the kick.
KICK	
KICK BACKWARDS	
KICK-IN	When a player kicks the ball back into play after an opposition behind. Kick-ins are regarded as a function of the team and do not count as kicks, although they are similarly graded for quality.
KICK INSIDE 50	When a player records an inside 50 for his team by kicking the ball from the midfield zone into the forward line.
KICK LONG ADVANTAGE	A long kick that results in an uncontested possession by a teammate. If an error is made by the player 'receiving' the kick, a 'kick long to advantage' is still recorded for the player kicking the ball.
KNOCK ON	When a player uses his hand to knock the ball to a teammate's advantage rather than attempting to take possession within his team's chain of play.
LONG KICK	A kick of more than 40 metres to a 50/50 contest or better for the team.
LOOSEBALL GET	A disputed ball at ground level not under direct physical pressure that results in an opportunity to record a legal disposal.
MARK	When a player cleanly catches (is deemed to have controlled the ball for sufficient time) a kicked ball that has travelled more than 15 metres without anyone else touching it or the ball hitting the ground.
MARK FROM OPP KICK	
MARK FUMBLED	Mark Fumbled
MARK ON LEAD	An uncontested mark taken after outsprinting an opponent.
MARK PLAY ON	Playing on immediately without retreating behind the mark.

MISSED TACKLES	Attempted tackles that are missed, allowing the ball carrier to break into space.
ONE ON ONE CONTEST DEFENDER	Being isolated in a one-on-one contest as the defender.
ONE ON ONE CONTEST TARGET	Being isolated in a one-on-one contest as the target of the kick.
OUT ON THE FULL	
REBOUND 50	Moving the ball from the defensive zone into the midfield.
RECEIVE HANDBALL	An uncontested possession that is the result of a teammate's handball.
RUCK HARDBALL GET	Taking possession of the ball directly out of the ruck.
RUNNING BOUNCE	Touching the ball to the ground, either directly or via a bounce, to allow a player to avoid being penalised for running too far.
SCORE ASSIST	Creating a score by getting the ball to a teammate either via a disposal, knock-on, ground kick or hitout, or by winning a free kick before the advantage is paid to the goal scorer.
SHORT KICK	A kick of less than 40 metres that results in the intended target retaining possession. Does not include kicks that are spoiled by the opposition.
SHOT AT GOAL	
SMOTHER	Suppressing an opposition disposal by either changing the trajectory of the ball immediately after the disposal or by blocking the disposal altogether.
SPOIL	Knocking the ball away from a marking contest preventing an opponent from taking a mark.
SPOIL GAINING POSSESSION	Spoils directed straight to a teammate.
SPOIL INEFFECTIVE	Spoils directed straight to an opposition player.
TACKLE	Using physical contact to prevent an opponent in possession of the ball from getting an effective disposal.
UNCONTESTED GATHER	Winning possession of the ball uncontested at ground level.
UNCONTESTED MARK	Marks taken under no physical pressure from an opponent. Includes marks taken on a lead and from opposition kicks.
UNCONTESTED MARK FROM OPP	
UNCONTESTED MARK FROM TEAM	
UNCONTESTED POSSESSION	Possessions gained whilst under no physical pressure, either from a teammate's disposal or an opposition's clanger kick. Includes handball receives, uncontested marks (including lead marks) and intended ball gets from a disposal.

B.3 Champion Data XML Dictionary

Table B.2: Descriptions of Champion Data raw XML data.

Term	Definition
Transaction	
Period	The quarter in which the transaction takes place.
Period Seconds	The number of seconds elapsed since the beginning of the quarter
Physical Zone	The zone on the field in which the transaction took place relative to stadium position (as if watching the match on television).
Logical Zone	The zone on the field in which the transaction took place relative to the team in possession of the ball.
Time Stamp	The time at which the transaction occurred.
Stat Code	The abbreviated transaction code.
Team	The team for which the transaction occurred.
Match	
Match ID	A unique match identifier.
Date	The data on which the match took place.
Round	The round in which the match took place.
Venue	The venue at which the match took place.
Match Number	A sequenced integer representation of the match.
Team	
Team ID	A unique team identifier.
Name	Team name
Nickname	Team nickname
Is Home	Is the team considered the home team for this match.
Player	
Player ID	A unique player identifier.
Jumper	The number printed on the players guernsey.
Display Name	The player's name as it is to be displayed.
First Name	The player's first name.
Surname	The player's surname.
Stat Types	
Code	The abbreviated transaction code.
Description	A description of the abbreviated code.

R Code for Champion Data Extraction

C.1 XML Data

```r
##CHAMPIONDATA PREPARATION
##CREATED BY: CASEY JOSMAN
##LAST EDITED: 19/09/2016

##LIBRARIES
library(XML)
library(car)
library(lme4)

##PREAMBLE
StaticData<- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Historic Sensitivity/Draws/6-5.csv",header=TRUE) #read in static data
StaticData$Home.team<-recode(StaticData$Home.team, '"Adelaide" = "Adelaide Crows";"Brisbane Lions" = "Brisbane Lions";"Carlton" = "Carlton";"Collingwood" = "Collingwood";"Essendon" = "Essendon";"Fremantle" = "Fremantle";"Geelong" = "Geelong Cats";"Gold Coast" = "Gold Coast Suns";"Greater Western Sydney" = "GWS Giants";"Hawthorn" = "Hawthorn";"Melbourne" = "Melbourne";"North Melbourne" = "North Melbourne";"Port Adelaide" = "Port Adelaide";"Richmond" = "Richmond";"St Kilda" = "St Kilda";"Sydney" = "Sydney Swans";"West Coast" = "West Coast Eagles";"Western Bulldogs" = "Western Bulldogs"') #make sure team names are consistent
StaticData$Away.team<-recode(StaticData$Away.team, '"Adelaide" = "Adelaide Crows";"Brisbane Lions" = "Brisbane Lions";"Carlton" = "Carlton";"Collingwood" = "Collingwood";"Essendon" = "Essendon";"Fremantle" = "Fremantle";"Geelong" = "Geelong Cats";"Gold Coast" = "Gold Coast Suns";"Greater Western Sydney" = "GWS Giants";"Hawthorn" = "Hawthorn";"Melbourne" = "Melbourne";"North Melbourne" = "North Melbourne";"Port Adelaide" = "Port Adelaide";"Richmond" = "Richmond";"St Kilda" = "St Kilda";"Sydney" = "Sydney Swans";"West Coast" = "West Coast Eagles";"Western Bulldogs" = "Western Bulldogs"') #make sure team names are consistent
setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Data\\ChampionData\\Bulldogs Data Feed 2015")
filenames <- list.files(pattern=".xml") #fetches file list from above directory
VenueData<-NULL
HomeNames<-NULL
AwayNames<-NULL
FullData<-NULL

##DATA MINING AND PREPARATION
for (f in filenames){ # for each file extracts data and transforms into workable dataframe
```

```
game <- xmlRoot(xmlTreeParse(f,getDTD=F,addAttributeNamespaces=T))
Match <- data.frame(t(unlist(xmlApply(game,xmlValue)[3:7])))

HomeData <- xmlApply(game[[8]],xmlChildren)
Home <- data.frame(t(unlist(lapply(lapply(HomeData[names(HomeData)!='PLAYER'],unlist), function(x) as.character(x[names(x)=='text.value'])))))
HPlayers <- data.frame(do.call(rbind,lapply(HomeData[names(HomeData)=='PLAYER'],unlist))[,c(3,6,9,12,15)],stringsAsFactors=FALSE)
names(HPlayers) <- sapply(names(HPlayers),function(x) sub('.children.text.value','',x))


AwayData <- xmlApply(game[[9]],xmlChildren)
Away <- data.frame(t(unlist(lapply(lapply(AwayData[names(AwayData)!='PLAYER'],unlist), function(x) as.character(x[names(x)=='text.value'])))))
APlayers <- data.frame(do.call(rbind,lapply(AwayData[names(AwayData)=='PLAYER'],unlist))[,c(3,6,9,12,15)],stringsAsFactors=FALSE)
names(APlayers) <- sapply(names(APlayers),function(x) sub('.children.text.value','',x))

StatData <- xmlApply(game[[10]],xmlChildren)
Stats <- data.frame(do.call(rbind,lapply(StatData,unlist))[,c(3,6)])
names(Stats) <- sapply(names(Stats),function(x) sub('.children.text.value','',x))

TransData <- xmlApply(game[[11]],xmlChildren)
TRX <- data.frame(do.call(rbind,lapply(TransData,unlist))[,c(3,6,9,12,15,18,21,24)])
names(TRX) <- sapply(names(TRX),function(x) sub('.children.text.value','',x))

TRX$FULLNAME <- TRX$TRX_PLAYER
levels(TRX$FULLNAME) <- sapply(levels(TRX$FULLNAME),function(x) tail(c(x,HPlayers$DISPLAYNAME[HPlayers$PLAYER_ID==as.character(x)]),1))
levels(TRX$FULLNAME) <- sapply(levels(TRX$FULLNAME),function(x) tail(c('',APlayers$DISPLAYNAME[APlayers$PLAYER_ID==as.character(x)]),1))

#Retrieve venue list, home, and away teams
VenueData<-c(VenueData,as.character(Match$VENUE))
HomeNames<-c(HomeNames,as.character(Home$NAME))
AwayNames<-c(AwayNames,as.character(Away$NAME))

#Select appropriate TRX
Ind<-which(TRX$STAT_CODE=="BEHI" | TRX$STAT_CODE=="RUSH" | TRX$STAT_CODE=="BUCL" | TRX$STAT_CODE=="TICL" | TRX$STAT_CODE=="CBCL" | TRX$STAT_CODE=="BUHO" | TRX$STAT_CODE=="CBHO" | TRX$STAT_CODE=="TIHO" | TRX$STAT_CODE=="CEBO" | TRX$STAT_CODE=="FRAG" | TRX$STAT_CODE=="FRFO" | TRX$STAT_CODE=="GOAL" | TRX$STAT_CODE=="HBEF" | TRX$STAT_CODE=="HBIN" | TRX$STAT_CODE=="HBRE" | TRX$STAT_CODE=="IN50" | TRX$STAT_CODE=="KIKIN" | TRX$STAT_CODE=="KKEF" | TRX$STAT_CODE=="KKIN" | TRX$STAT_CODE=="MACO" | TRX$STAT_CODE=="MAUN" | TRX$STAT_CODE=="PEREN" | TRX$STAT_CODE=="PERST" | TRX$STAT_CODE=="RE50" | TRX$STAT_CODE=="SPOIL" | TRX$STAT_CODE=="TACK")

TRX<-TRX[Ind,]
#Merge ramaining transactions (RUSH -> BEHI, BUCL;TICL;CBCL -> CLEAR, BUHO;CBHO;TIHO -> HITO, KIKIN;KKEF -> KICK, MACO;MAUN -> MARK)
TRX$STAT_CODE<-recode(TRX$STAT_CODE, '"RUSH"="BEHI";"BUCL"="CLEAR";"TICL"="CLEAR";"CBCL"="CLEAR";"BUHO"="HITO";"CBHO"="HITO";"TIHO"="HITO";"KIKIN"="KICK";"KKEF"="KICK";"MACO"="MARK";"MAUN"="MARK"')
#Append missing player information to TRX (transaction data)
TRX$TRX_PLAYER<-as.character(TRX$TRX_PLAYER) #set TRX_PLAYER to character
TRX$FULLNAME<-as.character(TRX$FULLNAME) #set FULLNAME to character

PlayState<-which(TRX$TRX_TEAM==0) #generate list of play reset states
```

```r
TRX$TRX_PLAYER[PlayState]<-"0"
TRX$FULLNAME[PlayState]<-"Reset"

HomePlays<-which(TRX$TRX_TEAM==as.character(Home$TEAM_ID)) #pick up plays according to home team code
for (i in 1:length(TRX$FULLNAME[HomePlays])){
TRX$FULLNAME[HomePlays][i]<-HPlayers$DISPLAYNAME[match(TRX$TRX_PLAYER[HomePlays][i],HPlayers$PLAYER_ID)]
}

AwayPlays<-which(TRX$TRX_TEAM==as.character(Away$TEAM_ID)) #pick up plays according to away team code
for (i in 1:length(TRX$FULLNAME[AwayPlays])){
TRX$FULLNAME[AwayPlays][i]<-APlayers$DISPLAYNAME[match(TRX$TRX_PLAYER[AwayPlays][i],APlayers$PLAYER_ID)]
}

TRX$TRX_PLAYER<-as.factor(TRX$TRX_PLAYER) #revert to factor
TRX$FULLNAME<-as.factor(TRX$FULLNAME) #revert to factor

##Create Team Specific (H/A) STAT_CODE
TRX$STAT_HA<-as.character(TRX$STAT_CODE) #transform TRX$STAT_HA to character list
TRX$STAT_HA[which(TRX$TRX_TEAM==as.character(Home$TEAM_ID))]<-paste("H.",as.character(TRX$STAT_HA[which(TRX$TRX_TEAM==as.character(Home$TEAM_ID))]),sep="") #add H. for all home TRX
TRX$STAT_HA[which(TRX$TRX_TEAM==as.character(Away$TEAM_ID))]<-paste("A.",as.character(TRX$STAT_HA[which(TRX$TRX_TEAM==as.character(Away$TEAM_ID))]),sep="") #add A. for all away TRX
TRX$STAT_HA[grep("CEBO",TRX$STAT_HA)]<-"CEBO" #remove team assignment from center bounce
TRX$STAT_HA<-as.factor(TRX$STAT_HA) #revert back to factor
##Create Dummy Variables for Summation
DummyTemp<-dummy(rbind("A",as.character(TRX$STAT_HA)))
DummyTemp<-DummyTemp[-seq(from=1,to=nrow(DummyTemp),by=2),] #remove extra rows added in by dummy function (need a more elegant way to create dummy variables)
##Create Play by Play Data (FULL DATA)
TempData<-NULL
SumData<-NULL
SumData<-as.data.frame(t(DummyTemp[1,]))
DateTemp<-as.Date(paste(strsplit(strsplit(as.character(Match$DATE),",")[[1]][2]," ")[[1]][3],strsplit(strsplit(as.character(Match$DATE),",")[[1]][2]," ")[[1]][2],substring(strsplit(as.character(Match$DATE),",")[[1]][3],first=2),collapse="",sep=""),"%d%B%Y")
HomeTemp<-as.character(Home$NAME)
AwayTemp<-as.character(Away$NAME)
if (nrow(StaticData[which(as.character(StaticData$Date)==DateTemp & StaticData$Home.team==HomeTemp & StaticData$Away.team==AwayTemp),])==0){
StaticTemp<-StaticData[which(as.character(StaticData$Date)==DateTemp & StaticData$Home.team==AwayTemp & StaticData$Away.team==HomeTemp),]
Swap<-1
} else {StaticTemp<-StaticData[which(as.character(StaticData$Date)==DateTemp & StaticData$Home.team==HomeTemp & StaticData$Away.team==AwayTemp),]
Swap<-0
}

TempData<-cbind.data.frame(StaticTemp,SumData,TRX$PERIOD[1],TRX$PERIODSECONDS[1],TRX$STAT_HA[1],row.names=NULL) #initialization of TempData
colnames(TempData)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")
```

```
for (i in 2:nrow(TRX)){

SumData<-as.data.frame(t(colSums(DummyTemp[1:i,])))
Temp<-cbind.data.frame(StaticTemp[1,],SumData,TRX$PERIOD[i],TRX$PERIODSECONDS[i],TRX$STAT_HA[i],row.names=NULL)
colnames(Temp)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")
TempData<-rbind.data.frame(TempData,Temp)

}

##Reorder Home/Away Teams as per Static Definition
if (Swap==1){
TempData<-TempData[,c(1:16,34:49,33,17:32,50:54)]
colnames(TempData)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")
} else {colnames(TempData)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")}

FullData<-rbind.data.frame(FullData,TempData)

}
```

C.2 CSV Data

```r
##CHAMPIONDATA PREPARATION
##CREATED BY: CASEY JOSMAN
##LAST EDITED: 08/12/2018

##LIBRARIES
library(car)
library(lme4)

##PREAMBLE
StaticData<- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Historic Sensitivity/Draws/6-5.csv",header=TRUE)
StaticData$Home.team<-recode(StaticData$Home.team, '"Adelaide" = "Adelaide Crows";"Brisbane Lions" = "Brisbane Lions";"Carlton" = "Carlton";"Collingwood" = "Collingwood";"Essendon" = "Essendon";"Fremantle" = "Fremantle";"Geelong" = "Geelong Cats";"Gold Coast" = "Gold Coast Suns";"Greater Western Sydney" = "GWS Giants";"Hawthorn" = "Hawthorn";"Melbourne" = "Melbourne";"North Melbourne" = "North Melbourne";"Port Adelaide" = "Port Adelaide";"Richmond" = "Richmond";"St Kilda" = "St Kilda";"Sydney" = "Sydney Swans";"West Coast" = "West Coast Eagles";"Western Bulldogs" = "Western Bulldogs"')
StaticData$Away.team<-recode(StaticData$Away.team, '"Adelaide" = "Adelaide Crows";"Brisbane Lions" = "Brisbane Lions";"Carlton" = "Carlton";"Collingwood" = "Collingwood";"Essendon" = "Essendon";"Fremantle" = "Fremantle";"Geelong" = "Geelong Cats";"Gold Coast" = "Gold Coast Suns";"Greater Western Sydney" = "GWS Giants";"Hawthorn" = "Hawthorn";"Melbourne" = "Melbourne";"North Melbourne" = "North Melbourne";"Port Adelaide" = "Port Adelaide";"Richmond" = "Richmond";"St Kilda" = "St Kilda";"Sydney" = "Sydney Swans";"West Coast" = "West Coast Eagles";"Western Bulldogs" = "Western Bulldogs"')

TRX <- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/To Be Processed/AFL Club TRX 2017.csv",header=TRUE)
TRX$VENUE_NAME<-recode(TRX$VENUE_NAME, '"MCG"="M.C.G.";"SCG"="S.C.G."')

VenueData<-NULL
HomeNames<-NULL
AwayNames<-NULL
FullData<-NULL

#get only Western Bulldogs data
games<-subset(TRX[!duplicated(TRX[,c(1:10)]),1:10],HOME_SQUAD=="Western Bulldogs" & AWAY_SQUAD!="BYE" | AWAY_SQUAD=="Western Bulldogs")

##DATA MINING AND PREPARATION
for (j in 1:22){ # for each match extracts data and transforms into workable dataframe

Match<-games$GROUP_ROUND_NO[j]
DateChar<-as.character(games$MATCH_DATE[j])
DateChar<-paste(strsplit(DateChar,"-")[[1]][1],strsplit(DateChar,"-")[[1]][2],as.numeric(strsplit(DateChar,"-")[[1]][3])+2000,sep="")
Venue<-as.character(games$VENUE[j])
Home<-as.character(games$HOME_SQUAD[j])
Away<-as.character(games$AWAY_SQUAD[j])
TRXtmp<-subset(TRX,GROUP_ROUND_NO==Match & HOME_SQUAD==Home & AWAY_SQUAD==Away,select=c(1:19))

```

```
#Retrieve venue list, home, and away teams
VenueData<-c(VenueData,Venue)
HomeNames<-c(HomeNames,Home)
AwayNames<-c(AwayNames,Away)

#Remove unnecessary transactions (BLOC, BOUN, BUBO, BUCA, FR50, FRF5, OOBO, OOFU, THIN) ! CEBO KEPT FOR NOW! TRX$STAT_CODE=="CEBO"
#RemInd<-which(TRX$STAT_CODE=="BLOC" | TRX$STAT_CODE=="BOUN" | TRX$STAT_CODE=="BUBO" | TRX$STAT_CODE=="BUCA" | TRX$STAT_CODE=="FR50" | TRX$STAT_CODE=="FRF5" | TRX$STAT_CODE=="OOBO" | TRX$STAT_CODE=="OOFU" | TRX$STAT_CODE=="THIN")

#Select appropriate TRX
Ind<-which(TRXtmp$STATISTIC_CODE=="BEHI" | TRXtmp$STATISTIC_CODE=="RUSHN" | TRXtmp$STATISTIC_CODE=="RUSHO" | TRXtmp$STATISTIC_CODE=="RUSHP" | TRXtmp$STATISTIC_CODE=="BUCL" | TRXtmp$STATISTIC_CODE=="TICL" | TRXtmp$STATISTIC_CODE=="CBCL" | TRXtmp$STATISTIC_CODE=="BUHO" | TRXtmp$STATISTIC_CODE=="BUHSK" | TRXtmp$STATISTIC_CODE=="BUHSD" |
TRXtmp$STATISTIC_CODE=="BUSM" | TRXtmp$STATISTIC_CODE=="BUHAD" | TRXtmp$STATISTIC_CODE=="TMBUH" | TRXtmp$STATISTIC_CODE=="TMBSD" | TRXtmp$STATISTIC_CODE=="TMBUS" | TRXtmp$STATISTIC_CODE=="TMBUA" | TRXtmp$STATISTIC_CODE=="CBHO" | TRXtmp$STATISTIC_CODE=="CBHSK" | TRXtmp$STATISTIC_CODE=="CBHSD" | TRXtmp$STATISTIC_CODE=="CBSM" |
TRXtmp$STATISTIC_CODE=="CBHAD" | TRXtmp$STATISTIC_CODE=="TIHO" | TRXtmp$STATISTIC_CODE=="TIHSK" | TRXtmp$STATISTIC_CODE=="TIHSD" | TRXtmp$STATISTIC_CODE=="TISM" | TRXtmp$STATISTIC_CODE=="TIHAD" | TRXtmp$STATISTIC_CODE=="TMTIH" | TRXtmp$STATISTIC_CODE=="TMTSD" | TRXtmp$STATISTIC_CODE=="TMTIS"  | TRXtmp$STATISTIC_CODE=="TMTIA" |
TRXtmp$STATISTIC_CODE=="CEBO" | TRXtmp$STATISTIC_CODE=="FRAGN" | TRXtmp$STATISTIC_CODE=="FRAGO" | TRXtmp$STATISTIC_CODE=="FRAGP" | TRXtmp$STATISTIC_CODE=="FRABB" | TRXtmp$STATISTIC_CODE=="FRFO" | TRXtmp$STATISTIC_CODE=="FRFBB" | TRXtmp$STATISTIC_CODE=="FRFNO" | TRXtmp$STATISTIC_CODE=="FRFOB" | TRXtmp$STATISTIC_CODE=="GOAL" |
TRXtmp$STATISTIC_CODE=="HBEF" | TRXtmp$STATISTIC_CODE=="HBIN" | TRXtmp$STATISTIC_CODE=="HBRE" | TRXtmp$STATISTIC_CODE=="IN50" | TRXtmp$STATISTIC_CODE=="KILO" | TRXtmp$STATISTIC_CODE=="KILA" | TRXtmp$STATISTIC_CODE=="KISH" | TRXtmp$STATISTIC_CODE=="KISE" | TRXtmp$STATISTIC_CODE=="KBLO" | TRXtmp$STATISTIC_CODE=="KBSH" |
TRXtmp$STATISTIC_CODE=="KKBW" | TRXtmp$STATISTIC_CODE=="KKGKE" | TRXtmp$STATISTIC_CODE=="KKLO" | TRXtmp$STATISTIC_CODE=="KKLA" | TRXtmp$STATISTIC_CODE=="KKSH" | TRXtmp$STATISTIC_CODE=="KKIN" | TRXtmp$STATISTIC_CODE=="MACOO" | TRXtmp$STATISTIC_CODE=="MACOP" | TRXtmp$STATISTIC_CODE=="MAUNO" | TRXtmp$STATISTIC_CODE=="MAUNP" |
TRXtmp$STATISTIC_CODE=="PEREN" | TRXtmp$STATISTIC_CODE=="PERST" | TRXtmp$STATISTIC_CODE=="RE50" | TRXtmp$STATISTIC_CODE=="SPOI" | TRXtmp$STATISTIC_CODE=="SPOIO" | TRXtmp$STATISTIC_CODE=="SPOIP" | TRXtmp$STATISTIC_CODE=="SPOIG" | TRXtmp$STATISTIC_CODE=="TACKN" | TRXtmp$STATISTIC_CODE=="TACKO" | TRXtmp$STATISTIC_CODE=="TACKP" |
TRXtmp$STATISTIC_CODE=="CETU"
)

TRXtemp<-TRXtmp[Ind,]
#Merge appropriate transactions
TRXtemp$STATISTIC_CODE<-recode(TRXtemp$STATISTIC_CODE, '"BEHI"="BEHI";"RUSHN"="BEHI";"RUSHO"="BEHI";"RUSHP"="BEHI";"BUCL"="CLEAR";"TICL"="CLEAR";"CBCL"="CLEAR";
"BUHO"="HITO";"BUHSK"="HITO";"BUHSD"="HITO";"BUSM"="HITO";"BUHAD"="HITO";"TMBUH"="HITO";"TMBSD"="HITO";
"TMBUS"="HITO";"TMBUA"="HITO";"CBHO"="HITO";"CBHSK"="HITO";"CBHSD"="HITO";"CBSM"="HITO";"CBHAD"="HITO";
"TIHO"="HITO";"TIHSK"="HITO";"TIHSD"="HITO";"TISM"="HITO";"TIHAD"="HITO";"TMTIH"="HITO";"TMTSD"="HITO";
"TMTIS"="HITO";"TMTIA"="HITO";"FRAGN"="FRAG";"FRAGO"="FRAG";"FRAGP"="FRAG";"FRABB"="FRAG";"FRFO"="FRFO";
"FRFBB"="FRFO";"FRFNO"="FRFO";"FRFOB"="FRFO";"KILO"="KICK";"KILA"="KICK";"KISH"="KICK";"KISE"="KICK";
```

```
"KBLO"="KICK";"KBSH"="KICK";"KKBW"="KICK";"KKGKE"="KICK";"KKLO"="KICK";"KKLA"="KICK";"KKSH"="KICK";
"MACOO"="MARK";"MACOP"="MARK";"MAUNO"="MARK";"MAUNP"="MARK";
"SPOI"="SPOIL";"SPOIO"="SPOIL";"SPOIP"="SPOIL";"SPOIG"="SPOIL";
"TACKN"="TACK";"TACKO"="TACK";"TACKP"="TACK";"CETU"="CEBO"'
)

##Create Team Specific (H/A) STAT_CODE
TRXtemp$STAT_HA<-as.character(TRXtemp$STATISTIC_CODE) #transform TRXtemp$STAT_HA to character list
TRXtemp$STAT_HA[which(TRXtemp$SQUAD_NAME==Home)]<-paste("H.",as.character(TRXtemp$STAT_HA[which(TRXtemp$SQUAD_NAME==Home)]),sep="") #add H. for all home TRXtemp
TRXtemp$STAT_HA[which(TRXtemp$SQUAD_NAME==Away)]<-paste("A.",as.character(TRXtemp$STAT_HA[which(TRXtemp$SQUAD_NAME==Away)]),sep="") #add A. for all away TRXtemp
TRXtemp$STAT_HA[grep("CEBO",TRXtemp$STAT_HA)]<-"CEBO" #remove team assignment from center bounce
TRXtemp$STAT_HA<-as.factor(TRXtemp$STAT_HA) #revert back to factor

dup<-which(duplicated(TRXtemp[,c("PERIOD_SECS","STAT_HA")]))
TRXtemp<-TRXtemp[-dup,]

##Create Dummy Variables for Summation
DummyTemp<-dummy(rbind("A",as.character(TRXtemp$STAT_HA)))
DummyTemp<-DummyTemp[-seq(from=1,to=nrow(DummyTemp),by=2),] #remove extra rows added in by dummy function (need a more elegant way to create dummy variables)
##Create Play by Play Data (FULL DATA)
TempData<-NULL
SumData<-NULL
SumData<-as.data.frame(t(DummyTemp[1,]))
DateTemp<-as.Date(DateChar,"%d%B%Y")
HomeTemp<-Home
AwayTemp<-Away
if (nrow(StaticData[which(as.character(StaticData$Date)==DateTemp & StaticData$Home.team==HomeTemp & StaticData$Away.team==AwayTemp),])==0){
StaticTemp<-StaticData[which(as.character(StaticData$Date)==DateTemp & StaticData$Home.team==AwayTemp & StaticData$Away.team==HomeTemp),]
Swap<-1
} else {StaticTemp<-StaticData[which(as.character(StaticData$Date)==DateTemp & StaticData$Home.team==HomeTemp & StaticData$Away.team==AwayTemp),]
Swap<-0
}

TempData<-cbind.data.frame(StaticTemp,SumData,TRXtemp$PERIOD[1],TRXtemp$PERIOD_SECS[1],TRXtemp$STAT_HA[1],row.names=NULL) #initialization of TempData
colnames(TempData)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")

for (i in 2:nrow(TRXtemp)){

SumData<-as.data.frame(t(colSums(DummyTemp[1:i,])))
Temp<-cbind.data.frame(StaticTemp[1,],SumData,TRXtemp$PERIOD[i],TRXtemp$PERIOD_SECS[i],TRXtemp$STAT_HA[i],row.names=NULL)
colnames(Temp)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")
TempData<-rbind.data.frame(TempData,Temp)

}

```

```
##Reorder Home/Away Teams as per Static Definition
if (Swap==1){
TempData<-TempData[,c(1:16,34:49,33,17:32,50:54)]
colnames(TempData)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")
} else {colnames(TempData)<-c(colnames(StaticData),colnames(DummyTemp),"QUARTER","TIME_SEC","STAT_HA")}

FullData<-rbind.data.frame(FullData,TempData)

}
```

C.3 Time Code Preprocessing

```
CumulTime<-function(data){ #Calculates full game time (adds previous quarter end time)
tempTime<-NULL
StartIndex<-as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))
EndIndex<-c(as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))[-1]-1,nrow(data))
for (i in 1:length(StartIndex)){
tempind<-StartIndex[i]:EndIndex[i]
tempData<-data[tempind,c("TIME_SEC","QUARTER")]
t1<-as.numeric(subset(tempData,QUARTER==1)$TIME_SEC)
t2<-as.numeric(subset(tempData,QUARTER==2)$TIME_SEC)+max(t1)
t3<-as.numeric(subset(tempData,QUARTER==3)$TIME_SEC)+max(t2)
t4<-as.numeric(subset(tempData,QUARTER==4)$TIME_SEC)+max(t3)
tempCalc<-c(t1,t2,t3,t4)
tempTime<-c(tempTime,tempCalc)
}
return(tempTime)
}

OffsetTime<-function(data,delta=0.0001){ #adds a multiple of delta to differing transactions occuring on the same epoch
TimeOff<-NULL
sig<-nchar(gsub("(.*)(\\.)|([0]*$)","",format(delta,scientific=FALSE)))
StartIndex<-as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))
EndIndex<-c(as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))[-1]-1,nrow(data))
for (i in 1:length(StartIndex)){
tempTime<-round(data$CumulT[StartIndex[i]:EndIndex[i]],digits=sig)
IndE<-which(duplicated(tempTime)) #gives location of second value in duplicate (need to get value before)
for (j in IndE){
IndS<-which(tempTime==tempTime[j]) #gives location of all matching duplicates
if(length(IndS)==0){
} else{
tempTime[IndS]<-tempTime[which(tempTime==tempTime[j])]+seq(0,(length(which(tempTime==tempTime[j]))-1)*delta,delta)
}
}
TimeOff<-c(TimeOff,tempTime)
}
return(TimeOff)
}
```

R Code for Static Models

D.1 Static Model R Code

```r
##Static Feature Models
##Created By: Casey Josman
##Last Edited: 17/03/2016

##LIBRARIES
library(bnlearn)
library(deal)
library(Rgraphviz)
library(gee)
library(MuMIn)
library(binomTools)
library(randomForest)
library(RWeka)
library(e1071)
library(fmsb)
library(caret)

##FUNCTIONS
logistic.regression.or.ci <- function(regress.out, level=0.95) #FUNCTION FROM http://www.medicine.mcgill.ca/epidemiology/joseph/courses/EPIB-621/logistic.regression.or.ci.txt
{
  usual.output <- summary(regress.out)
  z.quantile <- qnorm(1-(1-level)/2)
  number.vars <- length(regress.out$coefficients)
  OR <- exp(regress.out$coefficients[-1])
  temp.store.result <- matrix(rep(NA, number.vars*2), nrow=number.vars)
  for(i in 1:number.vars)
  {
    temp.store.result[i,] <- summary(regress.out)$coefficients[i] +
    c(-1, 1) * z.quantile * summary(regress.out)$coefficients[i+number.vars]
  }
  intercept.ci <- temp.store.result[1,]
  slopes.ci <- temp.store.result[-1,]
  OR.ci <- exp(slopes.ci)
  output <- list(regression.table = usual.output, intercept.ci = intercept.ci,
  slopes.ci = slopes.ci, OR=OR, OR.ci = OR.ci)
  return(output)
}

```

```r
predtab<-function(pred,actual){ #pred=PREDICTION OF GEE MODEL, actual=RESULT COLUMN FROM DATASET, MUST SET SCALE PARAMETER INTERNALLY
  count<-0
  newtab<-data.frame()
  len<-length(pred)
  for(i in 1:len){
    if(pred[i]>=0.7){
      newtab[i,1]=1}
    else{
      newtab[i,1]=0}
  }
  for (j in 1:len){
    if (actual[j]==1){
      count<-count+1
    }
    else{
    }
  }
  combtab<-cbind(newtab,actual)
  acctab<-table(combtab[,2],combtab[,1]) #TABULATES ACTUAL VS PREDICTED
  acc=sum(diag(acctab))/len
  list(ConfusionMatrix=acctab,Accuracy=acc,Pred=newtab)
}

chsq<-function(model,data,metric){ #MODEL=GLM OUTPUT, DATA=DATA, METRIC="RESULT BEING MODELLED"
  mlrfit<-model$fitted
  r<-(data[,metric] - mlrfit)/(sqrt(mlrfit*(1-mlrfit)))
  r2<-sum(r^2)
  df1<-nrow(data)-length(model$coeff)
  pval1<-1-pchisq(r2,df1)
  return(print(paste("Chi-square goodness of fit test with df=",df1,":"," p-value = ",pval1,sep="")))
}

splitdf <- function(dataframe, seed=NULL,sizeprop) { #DATAFRAME=DATA TO BE SPLIT, SEED=SEED, SIZEPROP=SIZE OF TRAINING SET
  if (!is.null(seed)) set.seed(seed)
  index <- 1:nrow(dataframe)
  trainindex <- sample(index, trunc(length(index)*(sizeprop)))
  trainset <- dataframe[trainindex, ]
  testset <- dataframe[-trainindex, ]
  list(trainset=trainset,testset=testset)
}

matchsplit<-function(dataframe,season,rnd){
  if(rnd==1){
    trainset<-subset(dataframe,dataframe$Season<season)
    testset<-subset(dataframe,dataframe$Season==season & dataframe$Round==rnd)
  }else{
    trainset<-subset(dataframe,dataframe$Season<season | dataframe$Season==season & dataframe$Round<rnd)
    testset<-subset(dataframe,dataframe$Season==season & dataframe$Round==rnd)
  }
  list(trainset=trainset,testset=testset)
}

MultiLogLoss <- function(act, pred){ #FUNCTION FROM https://www.kaggle.com/wiki/
```

```r
    LogarithmicLoss
  eps = 1e-15;
  nr <- nrow(pred)
  pred = matrix(sapply( pred, function(x) max(eps,x)), nrow = nr)
  pred = matrix(sapply( pred, function(x) min(1-eps,x)), nrow = nr)
  ll = sum(act*log(pred) + (1-act)*log(1-pred))
  ll = ll * -1/(nrow(act))
  return(ll);
}


rf.eval<-function(dataset, season=NULL, rnd=NULL, rf.fn=NULL, metric){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  rf.temp<-randomForest(formula=rf.fn, data = training.temp, ntree=nrow(training.temp)*
      10, importance=TRUE)
  ind<-match(metric,colnames(dataset))
  predict.temp<-predict(rf.temp,testing.temp[,-ind],type="response")
  confusion.temp<-table(testing.temp[,ind],predict.temp)
  accuracy.temp<-sum(diag(confusion.temp))/sum(confusion.temp)
  if(metric=="Result"){
    return(predict.temp)
  }else {
    return(predict.temp)
    #abserror.temp<-mean(abs(testing.temp[,ind]-predict.temp))
    #return(abserror.temp)
  }
}

lmt.eval<-function(dataset, season=NULL, rnd=NULL, lmt.fn=NULL, metric){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  ind<-match(metric,colnames(dataset))
  lmt.temp<-LMT(formula=lmt.fn, data = training.temp)
  predict.temp<-predict(lmt.temp, newdata=testing.temp)
  confusion.temp<-table(testing.temp[,ind],predict.temp)
  accuracy.temp<-sum(diag(confusion.temp))/sum(confusion.temp)
  if(metric=="Result"){
    return(predict.temp)
  }else {
    return(predict.temp)
    #abserror.temp<-mean(abs(testing.temp[,ind]-predict.temp))
    #return(abserror.temp)
  }
}

svm.eval<-function(dataset, season=NULL, rnd=NULL, svm.fn=NULL, metric){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  ind<-match(metric,colnames(dataset))
  tuned.temp <- tune.svm(svm.fn, data = training.temp, gamma = 10^(-6:-1), cost =
      10^(-1:1))
  G<-tuned.temp$best.parameters$gamma #best performing gamma
  C<-tuned.temp$best.parameters$cost  #best performing cost
  svm.temp<-svm(svm.fn,data = training.temp, kernel = "radial", gamma = G, cost = C)
```

```r
  predict.temp<-predict(svm.temp, newdata=testing.temp[,-match(metric,colnames(testing.temp))])
  confusion.temp<-table(testing.temp[,ind],predict.temp)
  accuracy.temp<-sum(diag(confusion.temp))/sum(confusion.temp)
  if(metric=="Result"){
    return(predict.temp)
  }else {
    return(predict.temp)
    #abserror.temp<-mean(abs(testing.temp[,ind]-predict.temp))
    #return(abserror.temp)
  }
}

result.venue.independence<-function(dataset, gee.fn=NULL, season=NULL, rnd=NULL,metric){ #gee.fn=gee model, gee.id=set internally, gee.cor=set internally
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  training.temp<-training.temp[order(training.temp$Venue),]
  testing.temp<-split.temp$testset
  fit.temp<-gee(data=dataset, gee.fn, maxiter=100, family=binomial(logit), id=Venue, corstr="independence")
  if (metric=="Result"){
    predict.temp<-predict(fit.temp, testing.temp, type="response")
    predict.tab<-predtab(pred=predict.temp, actual=testing.temp[,match(metric,colnames(testing.temp))])
    accuracy.temp<-predict.tab$Accuracy
    return(predict.tab$Pred)
  } else{
    predict.temp<-predict(fit.temp, testing.temp, type="scale")
    abserror.temp<-mean(abs(testing.temp$Margin-predict.temp))
    list(abserror.temp,predict.temp)
  }
}

margin.venue.independence<-function(dataset, gee.fn=NULL, season=NULL, rnd=NULL,metric){ #gee.fn=gee model, gee.id=set internally, gee.cor=set internally
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  training.temp<-training.temp[order(training.temp$Venue),]
  testing.temp<-split.temp$testset
  fit.temp<-gee(data=dataset, gee.fn, maxiter=100, family=binomial(logit), id=Venue, corstr="independence")
  if (metric=="Result"){
    predict.temp<-predict(fit.temp, testing.temp, type="response")
    predict.tab<-predtab(pred=predict.temp, actual=testing.temp[,match(metric,colnames(testing.temp))])
    accuracy.temp<-predict.tab$Accuracy
    return(predict.tab$Pred)
  } else{
    predict.temp<-predict(fit.temp, testing.temp, type="scale")
    abserror.temp<-mean(abs(testing.temp$Margin-predict.temp))
    list(abserror.temp,predict.temp)
  }
}

result.mlr.eval<-function(dataset,mlr.fn=NULL, season=NULL, rnd=NULL){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
```

```r
  testing.temp<-split.temp$testset
  fit.temp<-glm(data=training.temp, mlr.fn, family=binomial(logit))
  predict.temp<-predict(fit.temp, testing.temp, type="response")
  predict.tab<-predtab(pred=predict.temp, actual=testing.temp[,match("Result",colnames(testing.temp))])
  accuracy.temp<-predict.tab$Accuracy
  return(predict.tab$Pred)
}

##READ DATA FILE
matchRes<-NULL
teamRes<-NULL
#setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Data")
#StaticData<-read.csv("Static SeasonData.csv",header=TRUE)
StaticData <- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Sensitivity Analysis/5-5.csv",header=TRUE)
#StaticData <- subset(StaticData,Season==2014 | Season==2015)
#StaticData$Season<-as.factor(StaticData$Season)
#StaticData$Round<-as.factor(StaticData$Round)
StaticData$Finals<-as.factor(StaticData$Finals)
ResN<-StaticData$Result
StaticData$Result<-as.factor(StaticData$Result)
StaticData$HomeRank<-as.factor(StaticData$HomeRank)
StaticData$AwayRank<-as.factor(StaticData$AwayRank)
StaticData<-cbind(StaticData,ResN)
##SET GLOBAL VARIABLES

set.seed(314)
cvseed<-c(866,933,828,955,978,805,959,878,831,910)
nrep<-10


##MODELS (ALL FEATURES) - TEAM SPECIFIC
nonfeat<-match(c("Date","Result","ResN","Margin","Home.score","Away.score"),colnames(StaticData))
Resultfn=as.formula(paste("Result~",paste(colnames(StaticData[,-nonfeat]),collapse="+")))
Marginfn=as.formula(paste("Margin~",paste(colnames(StaticData[,-nonfeat]),collapse="+")))

##MODELS (ALL FEATURES) - MATCH SPECIFIC
matchnonfeat<-match(c("Date","Result","ResN","Margin","Home.score","Away.score","Home.team","Away.team"),colnames(StaticData))
dummyhome<-predict(dummyVars(~Home.team,data=StaticData),StaticData)
colnames(dummyhome)<-make.names(colnames(dummyhome), unique=TRUE)
dummyaway<-predict(dummyVars(~Away.team,data=StaticData),StaticData)
colnames(dummyaway)<-make.names(colnames(dummyaway), unique=TRUE)
matchResultfn<-as.formula(paste("Result~",paste(colnames(StaticData[,-matchnonfeat]),collapse="+"),paste("+"),paste(colnames(dummyhome),collapse="+"),paste("+"),paste(colnames(dummyaway),collapse="+")))
matchMarginfn<-as.formula(paste("Margin~",paste(colnames(StaticData[,-matchnonfeat]),collapse="+"),paste("+"),paste(colnames(dummyhome),collapse="+"),paste("+"),paste(colnames(dummyaway),collapse="+")))

##MODELS - GEE (TEAM SPECIFIC)
geenonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Venue"),colnames(StaticData))
geeResultfn=as.formula(paste("Result~",paste(colnames(StaticData[,-geenonfeat]),collapse="+")))
geeMarginfn=as.formula(paste("Margin~",paste(colnames(StaticData[,-geenonfeat]),collapse=
```

```
"+")))

##MODELS - GEE (MATCH SPECIFIC)
geematnonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Venue","Home.team","Away.team"),colnames(StaticData))
geematResultfn<-as.formula(paste("Result~",paste(colnames(StaticData[,-geematnonfeat]),collapse="+"),paste("+"),paste(colnames(dummyhome),collapse="+"),paste("+"),paste(colnames(dummyaway),collapse="+")))
geematMarginfn<-as.formula(paste("Margin~",paste(colnames(StaticData[,-geematnonfeat]),collapse="+"),paste("+"),paste(colnames(dummyhome),collapse="+"),paste("+"),paste(colnames(dummyaway),collapse="+")))

#TEAM SPECIFIC

##MLR
mlrmod1<-glm(Resultfn,data=subset(StaticData,Season<=2014),family=binomial(logit))
mlr1Res<-predtab(actual=subset(StaticData,Season==2015)$Result,pred=predict(mlrmod1,subset(StaticData,Season==2015)[,-match("Result",colnames(StaticData))],type="response"))
mlracc1<-mlr1Res$Accuracy

#GEE
#geemod3<-gee(data=subset(StaticData,Season<=2014), formula=geeResultfn, maxiter=100, family=binomial(logit), id=Venue, corstr="unstructured")
#gee3Res<-predtab(actual=subset(StaticData,Season==2015)$Result,pred=predict(geemod3,subset(StaticData,Season==2015),type="response"))
#gee3acc<-gee3Res$Accuracy

#geemod4<-gee(data=subset(StaticData,Season<=2014), formula=geeMarginfn, maxiter=100, family=binomial(logit), id=Venue, corstr="unstructured")
#gee4Res<-predict(geemod4,subset(StaticData,Season==2015),type="scale")
#geeacc4<-mean(abs(subset(StaticData,Season==2015)$Margin-gee4Res))

#RF
rfmod1<-randomForest(formula=Resultfn, data = subset(StaticData,Season<=2014), importance=TRUE)
rf1Res<-predict(rfmod1,subset(StaticData,Season==2015)[,-match("Result",colnames(StaticData))],type="response")
rf1temp<-cbind(rf1Res,subset(StaticData,Season==2015)$Result)
rfacc1<-length(which(rf1temp[,1]==rf1temp[,2]))/length(rf1temp[,1])

rfmod2<-randomForest(formula=Marginfn, data = subset(StaticData,Season<=2014), importance=TRUE)
rf2Res<-predict(rfmod2,subset(StaticData,Season==2015)[,-match("Margin",colnames(StaticData))],type="response")
rfacc2<-mean(abs(as.numeric(subset(StaticData,Season==2015)$Margin)-as.numeric(unlist(rf2Res))))

#LMT
lmtmod1<-LMT(formula=Resultfn, data = subset(StaticData,Season<=2014))
lmt1Res<-predict(lmtmod1, newdata=subset(StaticData,Season==2015))
lmt1temp<-cbind(lmt1Res,subset(StaticData,Season==2015)$Result)
lmtacc1<-length(which(lmt1temp[,1]==lmt1temp[,2]))/length(lmt1temp[,1])

#SVM
tune1<-tune.svm(Resultfn, data = subset(StaticData,Season<=2014), gamma = 10^(-6:-1), cost = 10^(-1:1))
G<-tune1$best.parameters$gamma
```

```
C<-tune1$best.parameters$cost
svmmod1<-svm(Resultfn, data = subset(StaticData,Season<=2014), kernel="radial", gamma=G, cost=C)
svm1Res<-predict(svmmod1,newdata=subset(StaticData,Season==2015)[,-match("Result", colnames(StaticData))])
svm1temp<-cbind(svm1Res,subset(StaticData,Season==2015)$Result)
svmacc1<-length(which(svm1temp[,1]==svm1temp[,2]))/length(svm1temp[,1])

tune2<-tune.svm(Marginfn, data = subset(StaticData,Season<=2014), gamma = 10^(-6:-1), cost = 10^(-1:1))
G<-tune2$best.parameters$gamma
C<-tune2$best.parameters$cost
svmmod2<-svm(Marginfn, data = subset(StaticData,Season<=2014), kernel="radial", gamma=G, cost=C)
svm2Res<-predict(svmmod2,newdata=subset(StaticData,Season==2015)[,-match("Margin", colnames(StaticData))])
svmacc2<-mean(abs(as.numeric(subset(StaticData,Season==2015)$Margin)-as.numeric(unlist(svm2Res))))

teamRes<-cbind(mlracc1,rfacc1,rfacc2,lmtacc1,svmacc1,svmacc2)
#MATCH SPECIFIC
MatchData<-cbind(StaticData,dummyhome,dummyaway)


##MLR
mlrmod1<-glm(matchResultfn,data=subset(MatchData,Season<=2014),family=binomial(logit))
mlr1Res<-predtab(actual=subset(MatchData,Season==2015)$Result,pred=predict(mlrmod1,subset(MatchData,Season==2015)[,-match("Result",colnames(MatchData))],type="response"))
mlracc1<-mlr1Res$Accuracy

#GEE
#geemod3<-gee(data=subset(MatchData,Season<=2014), function=geematchResultfn, maxiter=100, family=binomial(logit), id=Venue, corstr="unstructured")
#gee3Res<-predtab(actual=subset(MatchData,Season==2015)$Result,pred=predict(geemod3, subset(MatchData,Season==2015)),type="response")
#gee3acc<-gee3Res$Accuracy

#geemod4<-gee(data=subset(MatchData,Season<=2014), function=geematchMarginfn, maxiter=100, family=binomial(logit), id=Venue, corstr="unstructured")
#gee4Res<-predict(geemod4,subset(MatchData,Season==2015),type="scale")
#geeacc4<-mean(abs(subset(MatchData,Season==2015)$Margin-gee4Res))

#RF
rfmod1<-randomForest(formula=matchResultfn, data = subset(MatchData,Season<=2014), importance=TRUE)
rf1Res<-predict(rfmod1,subset(MatchData,Season==2015)[,-match("Result",colnames(MatchData))],type="response")
rf1temp<-cbind(rf1Res,subset(MatchData,Season==2015)$Result)
rfacc1<-length(which(rf1temp[,1]==rf1temp[,2]))/length(rf1temp[,1])

rfmod2<-randomForest(formula=matchMarginfn, data = subset(MatchData,Season<=2014), importance=TRUE)
rf2Res<-predict(rfmod2,subset(MatchData,Season==2015)[,-match("Margin",colnames(MatchData))],type="response")
rfacc2<-mean(abs(as.numeric(subset(MatchData,Season==2015)$Margin)-as.numeric(unlist(rf2Res))))

#LMT
```

```
lmtmod1<-LMT(formula=matchResultfn, data = subset(MatchData,Season<=2014))
lmt1Res<-predict(lmtmod1, newdata=subset(MatchData,Season==2015))
lmt1temp<-cbind(lmt1Res,subset(MatchData,Season==2015)$Result)
lmtacc1<-length(which(lmt1temp[,1]==lmt1temp[,2]))/length(lmt1temp[,1])

#SVM
tune1<-tune.svm(matchResultfn, data = subset(MatchData,Season<=2014), gamma = 10^(-6:-1), cost = 10^(-1:1))
G<-tune1$best.parameters$gamma
C<-tune1$best.parameters$cost
svmmod1<-svm(matchResultfn, data = subset(MatchData,Season<=2014), kernel="radial", gamma=G, cost=C)
svm1Res<-predict(svmmod1,newdata=subset(MatchData,Season==2015)[,-match("Result",colnames(MatchData))])
svm1temp<-cbind(svm1Res,subset(MatchData,Season==2015)$Result)
svmacc1<-length(which(svm1temp[,1]==svm1temp[,2]))/length(svm1temp[,1])

tune2<-tune.svm(matchMarginfn, data = subset(MatchData,Season<=2014), gamma = 10^(-6:-1), cost = 10^(-1:1))
G<-tune2$best.parameters$gamma
C<-tune2$best.parameters$cost
svmmod2<-svm(matchMarginfn, data = subset(MatchData,Season<=2014), kernel="radial", gamma=G, cost=C)
svm2Res<-predict(svmmod2,newdata=subset(MatchData,Season==2015)[,-match("Margin",colnames(MatchData))])
svmacc2<-mean(abs(as.numeric(subset(MatchData,Season==2015)$Margin)-as.numeric(unlist(svm2Res))))


matchRes<-cbind(mlracc1,rfacc1,rfacc2,lmtacc1,svmacc1,svmacc2)
```

D.2 Sensitivity Analysis

```r
##Static Sensitivity Analysis (Final)
##Created By: Casey Josman
##Last Edited: 30/01/2017

library(bnlearn)
library(deal)
library(Rgraphviz)
library(gee)
library(MuMIn)
library(binomTools)
library(randomForest)
library(RWeka)
library(e1071)
library(fmsb)
library(caret)
library(stringr)
library(psych)
library(agricolae)
library(xtable)

##FUNCTIONS
logistic.regression.or.ci <- function(regress.out, level=0.95) #FUNCTION FROM http://www.medicine.mcgill.ca/epidemiology/joseph/courses/EPIB-621/logistic.regression.or.ci.txt
{
  usual.output <- summary(regress.out)
  z.quantile <- qnorm(1-(1-level)/2)
  number.vars <- length(regress.out$coefficients)
  OR <- exp(regress.out$coefficients[-1])
  temp.store.result <- matrix(rep(NA, number.vars*2), nrow=number.vars)
  for(i in 1:number.vars)
  {
    temp.store.result[i,] <- summary(regress.out)$coefficients[i] +
    c(-1, 1) * z.quantile * summary(regress.out)$coefficients[i+number.vars]
  }
  intercept.ci <- temp.store.result[1,]
  slopes.ci <- temp.store.result[-1,]
  OR.ci <- exp(slopes.ci)
  output <- list(regression.table = usual.output, intercept.ci = intercept.ci,
  slopes.ci = slopes.ci, OR=OR, OR.ci = OR.ci)
  return(output)
}

perf = function(cut, pred, y)
{
  if (is.factor(y)){
    y<-as.numeric(as.character(y))
  } else {y<-y}
  yhat = (pred>cut)
  w = which(y==1)
  sensitivity = mean( yhat[w] == 1 )
  specificity = mean( yhat[-w] == 0 )
  c.rate = mean( y==yhat )
  d = cbind(sensitivity,specificity)-c(1,1)
  d = sqrt( d[1]^2 + d[2]^2 )
```

```
  out = t(as.matrix(c(sensitivity, specificity, c.rate,d)))
  colnames(out) = c("sensitivity", "specificity", "c.rate", "distance")
  return(out)
}

predtab<-function(pred,actual){ #pred=PREDICTION OF GEE MODEL, actual=RESULT COLUMN FROM DATASET, MUST SET SCALE PARAMETER INTERNALLY
  count<-0
  newtab<-data.frame()
  len<-length(pred)
  s = seq(.01,.99,length=1000)
  OUT = matrix(0,1000,4)
  for(o in 1:1000){
    OUT[o,]=perf(s[o],pred=pred,y=actual)
  }
  cp<-mean(s[which(OUT[,4]==min(OUT[,4]))])
  for(i in 1:len){
    if(pred[i]>=cp){
      newtab[i,1]=1}
    else{
      newtab[i,1]=0}
  }
  for (j in 1:len){
    if (actual[j]==1){
      count<-count+1
    }
    else{
    }
  }
  combtab<-cbind(newtab,actual)
  acctab<-table(combtab[,2],combtab[,1]) #TABULATES ACTUAL VS PREDICTED
  acc=sum(diag(acctab))/len
  list(ConfusionMatrix=acctab,Accuracy=acc,Pred=newtab,out=OUT)
}

chsq<-function(model,data,metric){ #MODEL=GLM OUTPUT, DATA=DATA, METRIC="RESULT BEING MODELLED"
  mlrfit<-model$fitted
  r<-(data[,metric] - mlrfit)/(sqrt(mlrfit*(1-mlrfit)))
  r2<-sum(r^2)
  df1<-nrow(data)-length(model$coeff)
  pval1<-1-pchisq(r2,df1)
  return(print(paste("Chi-square goodness of fit test with df=",df1,":"," p-value = ", pval1,sep="")))
}

splitdf <- function(dataframe, seed=NULL,sizeprop) { #DATAFRAME=DATA TO BE SPLIT, SEED=SEED, SIZEPROP=SIZE OF TRAINING SET
  if (!is.null(seed)) set.seed(seed)
  index <- 1:nrow(dataframe)
  trainindex <- sample(index, trunc(length(index)*(sizeprop)))
  trainset <- dataframe[trainindex, ]
  testset <- dataframe[-trainindex, ]
  list(trainset=trainset,testset=testset)
}

matchsplit<-function(dataframe,season,rnd){
  if(rnd==1){
```

```
    trainset<-subset(dataframe,dataframe$Season<season)
    testset<-subset(dataframe,dataframe$Season==season & dataframe$Round==rnd)
  }else{
    trainset<-subset(dataframe,dataframe$Season<season | dataframe$Season==season & dataframe$Round<rnd)
    testset<-subset(dataframe,dataframe$Season==season & dataframe$Round==rnd)
  }
  list(trainset=trainset,testset=testset)
}

MultiLogLoss <- function(act, pred){ #FUNCTION FROM https://www.kaggle.com/wiki/LogarithmicLoss
  eps = 1e-15;
  nr <- nrow(pred)
  pred = matrix(sapply( pred, function(x) max(eps,x)), nrow = nr)
  pred = matrix(sapply( pred, function(x) min(1-eps,x)), nrow = nr)
  ll = sum(act*log(pred) + (1-act)*log(1-pred))
  ll = ll * -1/(nrow(act))
  return(ll);
}


rf.eval<-function(dataset, season=NULL, rnd=NULL, rf.fn=NULL, metric){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  rf.temp<-randomForest(formula=rf.fn, data = training.temp, ntree=nrow(training.temp)*10, importance=TRUE)
  ind<-match(metric,colnames(dataset))
  predict.temp<-predict(rf.temp,testing.temp[,-ind],type="response")
  confusion.temp<-table(testing.temp[,ind],predict.temp)
  accuracy.temp<-sum(diag(confusion.temp))/sum(confusion.temp)
  if(metric=="Result"){
    return(predict.temp)
  }else {
    return(predict.temp)
    #abserror.temp<-mean(abs(testing.temp[,ind]-predict.temp))
    #return(abserror.temp)
  }
}

lmt.eval<-function(dataset, season=NULL, rnd=NULL, lmt.fn=NULL, metric){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  ind<-match(metric,colnames(dataset))
  lmt.temp<-LMT(formula=lmt.fn, data = training.temp)
  predict.temp<-predict(lmt.temp, newdata=testing.temp)
  confusion.temp<-table(testing.temp[,ind],predict.temp)
  accuracy.temp<-sum(diag(confusion.temp))/sum(confusion.temp)
  if(metric=="Result"){
    return(predict.temp)
  }else {
    return(predict.temp)
    #abserror.temp<-mean(abs(testing.temp[,ind]-predict.temp))
    #return(abserror.temp)
  }
}
```

```r

svm.eval<-function(dataset, season=NULL, rnd=NULL, svm.fn=NULL, metric){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  ind<-match(metric,colnames(dataset))
  tuned.temp <- tune.svm(svm.fn, data = training.temp, gamma = 10^(-6:-1), cost = 10^(-1:1))
  G<-tuned.temp$best.parameters$gamma #best performing gamma
  C<-tuned.temp$best.parameters$cost  #best performing cost
  svm.temp<-svm(svm.fn,data = training.temp, kernel = "radial", gamma = G, cost = C)
  predict.temp<-predict(svm.temp, newdata=testing.temp[,-match(metric,colnames(testing.temp))])
  confusion.temp<-table(testing.temp[,ind],predict.temp)
  accuracy.temp<-sum(diag(confusion.temp))/sum(confusion.temp)
  if(metric=="Result"){
    return(predict.temp)
  }else {
    return(predict.temp)
    #abserror.temp<-mean(abs(testing.temp[,ind]-predict.temp))
    #return(abserror.temp)
  }
}

result.venue.independence<-function(dataset, gee.fn=NULL, season=NULL, rnd=NULL,metric){ #gee.fn=gee model, gee.id=set internally, gee.cor=set internally
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  training.temp<-training.temp[order(training.temp$Venue),]
  testing.temp<-split.temp$testset
  fit.temp<-gee(data=dataset, gee.fn, maxiter=100, family=binomial(logit), id=Venue, corstr="independence")
  if (metric=="Result"){
    predict.temp<-predict(fit.temp, testing.temp, type="response")
    predict.tab<-predtab(pred=predict.temp, actual=testing.temp[,match(metric,colnames(testing.temp))])
    accuracy.temp<-predict.tab$Accuracy
    return(predict.tab$Pred)
  } else{
    predict.temp<-predict(fit.temp, testing.temp, type="scale")
    abserror.temp<-mean(abs(testing.temp$Margin-predict.temp))
    list(abserror.temp,predict.temp)
  }
}

margin.venue.independence<-function(dataset, gee.fn=NULL, season=NULL, rnd=NULL,metric){ #gee.fn=gee model, gee.id=set internally, gee.cor=set internally
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  training.temp<-training.temp[order(training.temp$Venue),]
  testing.temp<-split.temp$testset
  fit.temp<-gee(data=dataset, gee.fn, maxiter=100, family=binomial(logit), id=Venue, corstr="independence")
  if (metric=="Result"){
    predict.temp<-predict(fit.temp, testing.temp, type="response")
    predict.tab<-predtab(pred=predict.temp, actual=testing.temp[,match(metric,colnames(testing.temp))])
    accuracy.temp<-predict.tab$Accuracy
```

```
    return(predict.tab$Pred)
  } else{
    predict.temp<-predict(fit.temp, testing.temp, type="scale")
    abserror.temp<-mean(abs(testing.temp$Margin-predict.temp))
    list(abserror.temp,predict.temp)
  }
}

result.mlr.eval<-function(dataset,mlr.fn=NULL, season=NULL, rnd=NULL){
  split.temp<-matchsplit(dataframe=dataset, season=season, rnd=rnd)
  training.temp<-split.temp$trainset
  testing.temp<-split.temp$testset
  fit.temp<-glm(data=training.temp, mlr.fn, family=binomial(logit))
  predict.temp<-predict(fit.temp, testing.temp, type="response")
  predict.tab<-predtab(pred=predict.temp, actual=testing.temp[,match("Result",colnames(
      testing.temp))])
  accuracy.temp<-predict.tab$Accuracy
  return(predict.tab$Pred)
}

##READ DATA FILE
#setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Data\\Sensitivity Analysis") #
    years 2010-2015
setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Data\\Historic Sensitivity") #
    years 2001-2015
files<-list.files(pattern=".csv")
files<-files[-match("HistData.csv",files)]
Res<-NULL
MOV<-NULL
f<-files[9]
setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Data\\Historic Sensitivity")

StaticData<-read.csv(f,header=TRUE)
StaticData<-StaticData[-which(StaticData$Result=="Draw"),]
StaticData$SeasonF<-as.factor(StaticData$Season)
StaticData$RoundF<-as.factor(StaticData$Round)
StaticData$Finals<-as.factor(StaticData$Finals)
StaticData$ResN<-StaticData$Result
StaticData$Result<-as.factor(StaticData$Result)
StaticData$HomeRank<-as.factor(StaticData$HomeRank)
StaticData$AwayRank<-as.factor(StaticData$AwayRank)


g<-2001
TrainData<-subset(StaticData,Season>=g & Season<=2014)#& Round<=24)
TestData<-subset(StaticData,Season==2015)#& Round<=24)
Kval<-str_extract(f,"(^[0-9]+)")
Lval<-substring(str_extract(f,"(-+[0-9]+)"),2)
Dataval<-paste(g,":","2014",sep="")

##SET GLOBAL VARIABLES
set.seed(314)
cvseed<-c(866,933,828,955,978,805,959,878,831,910)
nrep<-10

##MODELS (ALL FEATURES) - MATCH SPECIFIC
#Finals indicator was removed due to negligible importance
nonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Home.team","Away.
```

```
team","Season","Round","ResN"),colnames(StaticData))
Resultfn=as.formula(paste("Result~",paste(colnames(StaticData[,-nonfeat]),collapse="+")))
Marginfn=as.formula(paste("Margin~",paste(colnames(StaticData[,-nonfeat]),collapse="+")))

##MODELS - GEE (MATCH SPECIFIC)
geenonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Home.team","Away.
    team","Season","Round","ResN"),colnames(StaticData))
geeResultfn=as.formula(paste("ResN~",paste(colnames(StaticData[,-geenonfeat]),collapse="+
    ")))
geeMarginfn=as.formula(paste("Margin~",paste(colnames(StaticData[,-geenonfeat]),collapse=
    "+")))

##MLR
mlrstart<-Sys.time()
mlrmod1<-glm(Resultfn,data=TrainData,family=binomial(logit))
mlrmod1$xlevels[["SeasonF"]]<-union(mlrmod1$xlevels[["SeasonF"]],levels(TestData$SeasonF)
    )
#mlrmod1$xlevels[["Venue"]]<-union(mlrmod1$xlevels[["Venue"]],levels(TestData$Venue))
#mlrmod1$xlevels[["HomeRank"]]<-union(mlrmod1$xlevels[["HomeRank"]],levels(TestData$
    HomeRank))
#mlrmod1$xlevels[["AwayRank"]]<-union(mlrmod1$xlevels[["AwayRank"]],levels(TestData$
    AwayRank))
mlr1Res<-predtab(actual=TestData$Result,pred=predict(mlrmod1,TestData[,-match("Result",
    colnames(StaticData))],type="response"))
mlracc1<-mlr1Res$Accuracy
RMSQmlr<-sqrt(mean((as.numeric(as.character(TestData$Result))-mlr1Res$Pred)^2))
mlrend<-Sys.time()
CTmlr<-difftime(mlrend,mlrstart,units="secs")
#R2mlr<-

#RF
rfstart<-Sys.time()
rfmod1<-randomForest(formula=Resultfn, data = TrainData, importance=TRUE)
rf1Res<-predict(rfmod1,TestData[,-match("Result",colnames(StaticData))],type="response")
rf1temp<-cbind(rf1Res,TestData$Result)
rfacc1<-length(which(rf1temp[,1]==rf1temp[,2]))/length(rf1temp[,1])
RMSQrf<-sqrt(mean((as.numeric(as.character(TestData$Result))-as.numeric(as.character(
    rf1Res)))^2))
rfend<-Sys.time()
CTrf<-difftime(rfend,rfstart,units="secs")
#R2rf<-

rfmod2<-randomForest(formula=Marginfn, data = TrainData, importance=TRUE)
rf2Res<-predict(rfmod2,TestData[,-match("Margin",colnames(StaticData))],type="response")
rfacc2<-mean(abs(as.numeric(TestData$Margin)-as.numeric(unlist(rf2Res))))

#LMT
lmtstart<-Sys.time()
lmtmod1<-LMT(formula=Resultfn, data = TrainData)
lmt1Res<-predict(lmtmod1, newdata=TestData)
lmt1temp<-cbind(lmt1Res,TestData$Result)
lmtacc1<-length(which(lmt1temp[,1]==lmt1temp[,2]))/length(lmt1temp[,1])
RMSQlmt<-sqrt(mean((as.numeric(as.character(TestData$Result))-as.numeric(as.character(
    lmt1Res)))^2))
lmtend<-Sys.time()
CTlmt<-difftime(lmtend,lmtstart,units="secs")
#R2lmt<-
```

```
#SVM
svmstart<-Sys.time()
tune1<-tune.svm(Resultfn, data = TrainData, gamma = 10^(-6:-1), cost = 10^(-1:1))
G<-tune1$best.parameters$gamma #0.01 from 10^(-6:-1)
C<-tune1$best.parameters$cost #10 from 10^(-1:1)
svmmod1<-svm(Resultfn, data = TrainData, kernel="radial", gamma=G, cost=C)
svm1Res<-predict(svmmod1,newdata=TestData[,-match("Result",colnames(StaticData))])
svm1temp<-cbind(svm1Res,TestData$Result)
svmacc1<-length(which(svm1temp[,1]==svm1temp[,2]))/length(svm1temp[,1])
RMSQsvm<-sqrt(mean((as.numeric(as.character(TestData$Result))-as.numeric(as.character(
    svm1Res)))^2))
svmend<-Sys.time()
CTsvm<-difftime(svmend,svmstart,units="secs")
#R2svm<-

tune2<-tune.svm(Marginfn, data = TestData, gamma = 10^(-6:-1), cost = 10^(-1:1))
G<-tune2$best.parameters$gamma
C<-tune2$best.parameters$cost
svmmod2<-svm(Marginfn, data = TrainData, kernel="radial", gamma=G, cost=C)
svm2Res<-predict(svmmod2,newdata=TestData[,-match("Margin",colnames(StaticData))])
svmacc2<-mean(abs(as.numeric(TestData$Margin)-as.numeric(unlist(svm2Res))))


ResTemp<-data.frame(matrix(ncol=7,nrow=4))
ResTemp[,1]<-c(mlracc1,rfacc1,lmtacc1,svmacc1)
ResTemp[,2]<-c("MLR","RF","LMT","SVM")
ResTemp[,3]<-rep(Kval,4)
ResTemp[,4]<-rep(Lval,4)
ResTemp[,5]<-rep(Dataval,4)
ResTemp[,6]<-c(RMSQmlr,RMSQrf,RMSQlmt,RMSQsvm)
ResTemp[,7]<-c(CTmlr,CTrf,CTlmt,CTsvm)
#ResTemp[,7]<-c(R2mlr,R2rf,R2lmt,R2svm)
MOVTemp<-data.frame(matrix(ncol=5,nrow=2))
MOVTemp[,1]<-c(rfacc2,svmacc2)
MOVTemp[,2]<-c("RF","SVM")
MOVTemp[,3]<-rep(Kval,2)
MOVTemp[,4]<-rep(Lval,2)
MOVTemp[,5]<-rep(Dataval,2)

Res<-rbind(Res,ResTemp)
MOV<-rbind(MOV,MOVTemp)




colnames(Res)<-c("Result","Method","KValue","LValue","Data","RMSQ","CT")
colnames(MOV)<-c("Result","Method","KValue","LValue","Data")

setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Results\\2017\\Static Sensitivity
    (Rerun 2017 - With Finals)\\2001-2014")
write.csv(Res,row.names=FALSE,file="Static Results (Historic).csv")
write.csv(MOV,row.names=FALSE,file="Static MOV (Historic).csv")

Res$Method<-as.factor(Res$Method)
Res$KValue<-as.factor(Res$KValue)
Res$LValue<-as.factor(Res$LValue)
Res$Data<-as.factor(Res$Data)

```

```
MOV$Method<-as.factor(MOV$Method)
MOV$KValue<-as.factor(MOV$KValue)
MOV$LValue<-as.factor(MOV$LValue)
MOV$Data<-as.factor(MOV$Data)

##ANOVA ANALYSIS

ResAOV<-aov(formula = Result ~ Method + KValue * LValue + Data + RMSQ + CT, data = Res)
MOVAOV<-aov(formula = Result ~ Method + KValue * LValue + Data, data = MOV)

TukeyHSD(ResAOV)
```

D.3 Team Performance R Code

```r
##Penalty Models
##Created By: Casey Josman
##Last Edited: 08/04/2016

##LIBRARIES
library(car)
library(ggplot2)

##FUNCTIONS

GameSum<-function(prediction,team,minprob,maxprob){

  GameMat<-matrix(nrow=2,ncol=4)
  colnames(GameMat)<-c(paste("P(win)<",minprob,sep=""),paste(minprob,"<P(win)<",maxprob,sep=""),paste("P(win)>",maxprob,sep=""),"Total")
  rownames(GameMat)<-c("Win","Lose")
  wintemp<-subset(prediction,Home.team==team | Away.team==team)
  GameMat[1,]<-c(nrow(wintemp[which(wintemp$Home.team==team & wintemp$Result==1 & wintemp$WinProb<minprob),])+nrow(wintemp[which(wintemp$Away.team==team & wintemp$Result==0 & 1-wintemp$WinProb<minprob),]),nrow(wintemp[which(wintemp$Home.team==team & wintemp$Result==1 & wintemp$WinProb>minprob & wintemp$WinProb<maxprob),])+nrow(wintemp[which(wintemp$Away.team==team & wintemp$Result==0 & 1-wintemp$WinProb>minprob & 1-wintemp$WinProb<maxprob),]),nrow(wintemp[which(wintemp$Home.team==team & wintemp$Result==1 & wintemp$WinProb>maxprob),])+nrow(wintemp[which(wintemp$Away.team==team & wintemp$Result==0 & 1-wintemp$WinProb>maxprob),]),nrow(subset(wintemp,wintemp$Home.team==team & wintemp$Result==1 | wintemp$Away.team==team & wintemp$Result==0)))
  losetemp<-subset(prediction,Away.team==team)
  GameMat[2,]<-c(nrow(wintemp[which(wintemp$Home.team==team & wintemp$Result==0 & wintemp$WinProb<minprob),])+nrow(wintemp[which(wintemp$Away.team==team & wintemp$Result==1 & 1-wintemp$WinProb<minprob),]),nrow(wintemp[which(wintemp$Home.team==team & wintemp$Result==0 & wintemp$WinProb>minprob & wintemp$WinProb<maxprob),])+nrow(wintemp[which(wintemp$Away.team==team & wintemp$Result==1 & 1-wintemp$WinProb>minprob & 1-wintemp$WinProb<maxprob),]),nrow(wintemp[which(wintemp$Home.team==team & wintemp$Result==0 & wintemp$WinProb>maxprob),])+nrow(wintemp[which(wintemp$Away.team==team & wintemp$Result==1 & 1-wintemp$WinProb>maxprob),]),nrow(subset(wintemp,wintemp$Home.team==team & wintemp$Result==0 | wintemp$Away.team==team & wintemp$Result==1)))

  return(GameMat)

}

ProbPlot<-function(RawData,Fixture,MLRfn,lowbound=0.3,upbound=0.7){

  library(ggplot2)
  library(reshape2)

  RawData$SeasonF<-as.factor(RawData$Season)
  RawData$RoundF<-as.factor(RawData$Round)
  RawData$Finals<-as.factor(RawData$Finals)
  RawData$Result<-as.factor(RawData$Result)
  RawData$HomeRank<-as.factor(RawData$HomeRank)
  RawData$AwayRank<-as.factor(RawData$AwayRank)
```

```
  teams<-levels(RawData$Home.team)
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  mlrtemp<-glm(MLRfn,data=subset(RawData,Season<=seatemp & Round<=24),family=binomial(logit))
  mlrtemp$xlevels[["SeasonF"]]<-union(mlrtemp$xlevels[["SeasonF"]],levels(Fixture$SeasonF))
  if (as.character(substitute(Fixture))=="Fixture2014"){
    mlrtemp$xlevels[["Venue"]]<-union(mlrtemp$xlevels[["Venue"]],"Traeger Park")
  } else {}
  predtemp<-predict(mlrtemp,Fixture,type="response")

  preddata<-cbind(subset(RawData,Season==seatemp+1 & Round<=24),WinProb=predtemp)

  ind<-0
  plotdata<-data.frame(matrix(nrow=18,ncol=22))
  colnames(plotdata)<-paste(1:22,sep="")
  rownames(plotdata)<-teams

  for (t in teams){

    ind<-ind+1
    temp<-subset(preddata, Home.team==t | Away.team==t)
    temp[which(temp$Away.team==t),]$WinProb<-1-temp[which(temp$Away.team==t),]$WinProb
    plotdata[ind,]<-temp$WinProb

  }
  plotdata<-round(plotdata, digits=2)
  plotdata$Team<-teams
  plotmelt<-melt(plotdata,id.vars="Team")
  colnames(plotmelt)<-c("Team","Match","WinProb")
  plotmelt$Probability<-cut(plotmelt$WinProb,breaks = c(-Inf,lowbound,upbound,Inf),labels=as.character(c(paste("Pr<",lowbound,sep=""),paste(lowbound,"<Pr<",upbound,sep=""),paste("Pr>",upbound,sep=""))),right = FALSE)
  plotmelt$Team <- as.factor(plotmelt$Team)
  plotmelt$Team = with(plotmelt, factor(Team, levels = rev(levels(Team))))

  perf_cols <- c("red","white","green")
  perf_text_cols <- c("black","black","black")

  gg <- ggplot(data=plotmelt, aes(x=Match, y=Team, fill=Probability))
  gg <- gg + geom_tile()
  gg <- gg + geom_text(aes(label=WinProb, color=Probability), show.legend=FALSE)
  gg <- gg + labs(title = "Per Match Win Probabilities")
  gg <- gg + coord_equal()
  gg <- gg + scale_colour_manual(values = perf_text_cols)
  gg <- gg + scale_fill_manual(values=perf_cols)
  gg <- gg + theme_minimal(base_size = 12, base_family = "")

  return(gg)
}

StaticPen<-function(RawData,Fixture,MLRfn){

  RawData$SeasonF<-as.factor(RawData$Season)
  RawData$RoundF<-as.factor(RawData$Round)
  RawData$Finals<-as.factor(RawData$Finals)
  RawData$Result<-as.factor(RawData$Result)
  RawData$HomeRank<-as.factor(RawData$HomeRank)
```

```r
  RawData$AwayRank<-as.factor(RawData$AwayRank)
  teams<-levels(RawData$Home.team)
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  mlrtemp<-glm(MLRfn,data=subset(RawData,Season<=seatemp & Round<=24),family=binomial(logit))
  mlrtemp$xlevels[["SeasonF"]]<-union(mlrtemp$xlevels[["SeasonF"]],levels(Fixture$SeasonF))
  if (as.character(substitute(Fixture))=="Fixture2014"){
    mlrtemp$xlevels[["Venue"]]<-union(mlrtemp$xlevels[["Venue"]],"Traeger Park")
  } else {}
  predtemp<-predict(mlrtemp,Fixture,type="response")

  preddata<-cbind(subset(RawData,Season==seatemp+1 & Round<=24),WinProb=predtemp) #bound fixture and predicted probabilities

  ind<-0
  SimTemp<-matrix(ncol=2,nrow=18)
  for (t in teams){
    ind<-ind+1    #divide into four categories home win, home loss, away win, away loss
    homewin<-sum(apply(cbind(1/(subset(preddata,Home.team==t & Result==1)$WinProb),rep(25,nrow(subset(preddata,Home.team==t & Result==1)))),1,min))
    homeloss<-sum(apply(cbind(-1/(1-subset(preddata,Home.team==t & Result==0)$WinProb),rep(-25,nrow(subset(preddata,Home.team==t & Result==0)))),1,max))
    awaywin<-sum(apply(cbind(1/(1-subset(preddata,Away.team==t & Result==0)$WinProb),rep(25,nrow(subset(preddata,Away.team==t & Result==0)))),1,min))
    awayloss<-sum(apply(cbind(-1/(subset(preddata,Away.team==t & Result==1)$WinProb),rep(-25,nrow(subset(preddata,Away.team==t & Result==1)))),1,max))
    temppoints<-homewin+homeloss+awaywin+awayloss
    SimTemp[ind,]<-cbind(t,temppoints)

  }
  colnames(SimTemp)<-c("Team","Points")
  SimTemp<-as.data.frame(SimTemp)
  SimTemp$Points<-as.character(SimTemp$Points)
  SimTemp$Points<-as.numeric(SimTemp$Points)
  SimTemp<-SimTemp[order(SimTemp[,2],decreasing=TRUE),]
  return(SimTemp)
}

VariablePen<-function(RawData,Fixture,MLRfn,minpr,maxpr,minpts,maxpts){

  RawData$SeasonF<-as.factor(RawData$Season)
  RawData$RoundF<-as.factor(RawData$Round)
  RawData$Finals<-as.factor(RawData$Finals)
  RawData$Result<-as.factor(RawData$Result)
  RawData$HomeRank<-as.factor(RawData$HomeRank)
  RawData$AwayRank<-as.factor(RawData$AwayRank)
  teams<-levels(RawData$Home.team)
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  mlrtemp<-glm(MLRfn,data=subset(RawData,Season<=seatemp & Round<=24),family=binomial(logit))
  mlrtemp$xlevels[["SeasonF"]]<-union(mlrtemp$xlevels[["SeasonF"]],levels(Fixture$SeasonF))
  if (as.character(substitute(Fixture))=="Fixture2014"){
    mlrtemp$xlevels[["Venue"]]<-union(mlrtemp$xlevels[["Venue"]],"Traeger Park")
  } else {}
  predtemp<-predict(mlrtemp,Fixture,type="response")

```

```
  preddata<-cbind(subset(RawData,Season==seatemp+1 & Round<=24),WinProb=predtemp) #bound fixture and predicted probabilities

  ind<-0
  SimTemp<-matrix(ncol=2,nrow=18)
  for (t in teams){
    ind<-ind+1    #divide into four categories home win, home loss, away win, away loss
    homewin<-sum(ifelse(subset(preddata,Home.team==t & Result==1)$WinProb<minpr,maxpts,ifelse(subset(preddata,Home.team==t & Result==1)$WinProb>maxpr,minpts,(1/subset(preddata,Home.team==t & Result==1)$WinProb)+minpts)))
    homeloss<-sum(ifelse(subset(preddata,Home.team==t & Result==0)$WinProb<minpr,-minpts,ifelse(subset(preddata,Home.team==t & Result==0)$WinProb>maxpr,-maxpts,(-1/(1-subset(preddata,Home.team==t & Result==0)$WinProb))-minpts)))
    awaywin<-sum(ifelse((1-subset(preddata,Away.team==t & Result==0)$WinProb)<minpr,maxpts,ifelse((1-subset(preddata,Away.team==t & Result==0)$WinProb)>maxpr,minpts,(1/(1-subset(preddata,Away.team==t & Result==0)$WinProb))+minpts)))
    awayloss<-sum(ifelse((1-subset(preddata,Away.team==t & Result==1)$WinProb)<minpr,-minpts,ifelse((1-subset(preddata,Away.team==t & Result==1)$WinProb)>maxpr,-maxpts,(-1/(subset(preddata,Away.team==t & Result==1)$WinProb))-minpts)))
    temppoints<-homewin+homeloss+awaywin+awayloss
    SimTemp[ind,]<-cbind(t,temppoints)

  }
  colnames(SimTemp)<-c("Team","Points")
  SimTemp<-as.data.frame(SimTemp)
  SimTemp$Points<-as.character(SimTemp$Points)
  SimTemp$Points<-as.numeric(SimTemp$Points)
  SimTemp<-SimTemp[order(SimTemp[,2],decreasing=TRUE),]
  return(SimTemp)
}

ExpVariablePen<-function(RawData,Fixture,MLRfn,minpr,maxpr,minpts,maxpts){

  RawData$SeasonF<-as.factor(RawData$Season)
  RawData$RoundF<-as.factor(RawData$Round)
  RawData$Finals<-as.factor(RawData$Finals)
  RawData$Result<-as.factor(RawData$Result)
  RawData$HomeRank<-as.factor(RawData$HomeRank)
  RawData$AwayRank<-as.factor(RawData$AwayRank)
  teams<-levels(RawData$Home.team)
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  mlrtemp<-glm(MLRfn,data=subset(RawData,Season<=seatemp & Round<=24),family=binomial(logit))
  mlrtemp$xlevels[["SeasonF"]]<-union(mlrtemp$xlevels[["SeasonF"]],levels(Fixture$SeasonF))
  if (as.character(substitute(Fixture))=="Fixture2014"){
    mlrtemp$xlevels[["Venue"]]<-union(mlrtemp$xlevels[["Venue"]],"Traeger Park")
  } else {}
  predtemp<-predict(mlrtemp,Fixture,type="response")

  preddata<-cbind(subset(RawData,Season==seatemp+1 & Round<=24),WinProb=predtemp) #bound fixture and predicted probabilities

  ind<-0
  SimTemp<-matrix(ncol=2,nrow=18)
  for (t in teams){
    ind<-ind+1    #divide into four categories home win, home loss, away win, away loss
    homewin<-sum(subset(preddata,Home.team==t & Result==1)$WinProb*ifelse(subset(preddata
```

```
,Home.team==t & Result==1)$WinProb<minpr,maxpts,ifelse(subset(preddata,Home.team
==t & Result==1)$WinProb>maxpr,minpts,(1/subset(preddata,Home.team==t & Result
==1)$WinProb)+minpts)))+sum((1-subset(preddata,Home.team==t & Result==1)$WinProb)
*ifelse(subset(preddata,Home.team==t & Result==1)$WinProb<minpr,-minpts,ifelse(
subset(preddata,Home.team==t & Result==1)$WinProb>maxpr,-maxpts,(-1/(1-subset(
preddata,Home.team==t & Result==1)$WinProb))-minpts)))
  #sum(prob(win)*points if win)+sum(prob(lose)*points if lose) ------ prob(win)=p prob
  (lose)=1-p
  homeloss<-sum((1-subset(preddata,Home.team==t & Result==0)$WinProb)*ifelse(subset(
  preddata,Home.team==t & Result==0)$WinProb<minpr,-minpts,ifelse(subset(preddata,
  Home.team==t & Result==0)$WinProb>maxpr,-maxpts,(-1/(1-subset(preddata,Home.team
  ==t & Result==0)$WinProb))-minpts)))+sum(subset(preddata,Home.team==t & Result
  ==0)$WinProb*ifelse(subset(preddata,Home.team==t & Result==0)$WinProb<minpr,
  maxpts,ifelse(subset(preddata,Home.team==t & Result==0)$WinProb>maxpr,minpts,(1/
  subset(preddata,Home.team==t & Result==0)$WinProb)+minpts)))
  #sum(prob(lose)*points if lose)+sum(prob(win)*points if win) ------ prob(win)=p prob
  (lose)=1-p
  awaywin<-sum((1-subset(preddata,Away.team==t & Result==0)$WinProb)*ifelse((1-subset(
  preddata,Away.team==t & Result==0)$WinProb)<minpr,maxpts,ifelse((1-subset(
  preddata,Away.team==t & Result==0)$WinProb)>maxpr,minpts,(1/(1-subset(preddata,
  Away.team==t & Result==0)$WinProb))+minpts)))+sum((subset(preddata,Away.team==t &
   Result==0)$WinProb)*ifelse((1-subset(preddata,Away.team==t & Result==0)$WinProb)
  <minpr,-minpts,ifelse((1-subset(preddata,Away.team==t & Result==0)$WinProb)>maxpr
  ,-maxpts,(-1/(subset(preddata,Away.team==t & Result==0)$WinProb))-minpts)))
  #sum(prob(win)*points if win)+sum(prob(lose)*points if lose) ------ prob(win)=1-p
  prob(lose)=p
  awayloss<-sum((subset(preddata,Away.team==t & Result==1)$WinProb)*ifelse((1-subset(
  preddata,Away.team==t & Result==1)$WinProb)<minpr,-minpts,ifelse((1-subset(
  preddata,Away.team==t & Result==1)$WinProb)>maxpr,-maxpts,(-1/(subset(preddata,
  Away.team==t & Result==1)$WinProb))-minpts)))+sum((1-subset(preddata,Away.team==t
   & Result==1)$WinProb)*ifelse((1-subset(preddata,Away.team==t & Result==1)$
  WinProb)<minpr,maxpts,ifelse((1-subset(preddata,Away.team==t & Result==1)$WinProb
  )>maxpr,minpts,(1/(1-subset(preddata,Away.team==t & Result==1)$WinProb))+minpts))
  )
  #sum(prob(lose)*points if lose)+sum(prob(win)*points if win) ------ prob(win)=1-p
  prob(lose)=p
  temppoints<-homewin+homeloss+awaywin+awayloss
  SimTemp[ind,]<-cbind(t,temppoints)

 }
 colnames(SimTemp)<-c("Team","Points")
 SimTemp<-as.data.frame(SimTemp)
 SimTemp$Points<-as.character(SimTemp$Points)
 SimTemp$Points<-as.numeric(SimTemp$Points)
 SimTemp<-SimTemp[order(SimTemp[,2],decreasing=TRUE),]
 return(SimTemp)
}

##READ DATA
StaticData <- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Historic
  Sensitivity/6-5.csv",header=TRUE)
StaticData<-subset(StaticData,Season>=2001)
StaticData$SeasonF<-as.factor(StaticData$Season)
StaticData$RoundF<-as.factor(StaticData$Round)
StaticData$Finals<-as.factor(StaticData$Finals)
StaticData$ResN<-StaticData$Result
StaticData$Result<-as.factor(StaticData$Result)
StaticData$HomeRank<-as.factor(StaticData$HomeRank)
```

```
StaticData$AwayRank<-as.factor(StaticData$AwayRank)
Ranking <- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Ranking Table.csv",
    header=TRUE)
#Ranking$Team<-recode(Ranking$Team,'"AD"="Adelaide";"BL"="Brisbane Lions";"CA"="Carlton
    ";"CW"="Collingwood";"ES"="Essendon";"FR"="Fremantle";"GC"="Gold Coast";"GE"="Geelong
    ";"GW"="Greater Western Sydney";"HW"="Hawthorn";"ME"="Melbourne";"NM"="North
    Melbourne";"PA"="Port Adelaide";"RI"="Richmond";"SK"="St Kilda";"SY"="Sydney";"WB"="
    Western Bulldogs";"WC"="West Coast"')

Fixture2015<-subset(StaticData,Season==2015 & Round<=23,select=c
    (2,3,4,5,6,10,12,13,14,15,16,17,18))


##MODELS
nonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Home.team","Away.
    team","Season","Round","Finals","ResN"),colnames(StaticData))
Resultfn=as.formula(paste("Result~",paste(colnames(StaticData[,-nonfeat]),collapse="+")))


StaticPen2015<-StaticPen(RawData=StaticData,Fixture=Fixture2015,MLRfn=Resultfn)
VariablePen2015<-VariablePen(RawData=StaticData,Fixture=Fixture2015,MLRfn=Resultfn,minpr
    =0.3,maxpr=0.7,minpts=5,maxpts=12)
ExpVariablePen2015<-ExpVariablePen(RawData=StaticData,Fixture=Fixture2015,MLRfn=Resultfn,
    minpr=0.3,maxpr=0.7,minpts=5,maxpts=12)


##PLOTS
#predicted vs expected plot
setwd(dir = "C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Results\\2017\\Penalty
    Models")

expplotdata<-cbind(ExpVariablePen2015,ActualPoints=VariablePen2015[match(
    ExpVariablePen2015$Team,VariablePen2015$Team),]$Points)
colnames(expplotdata)<-c("Team","Expected","Predicted")


p1<-ggplot(data=expplotdata, aes(x=Expected,y=Predicted,group=Team,colour=Team,shape=Team
    )) + geom_point(size=6) + geom_abline(slope=1) + scale_shape_manual(values=1:18) +
    labs(x="Expected Points",y="Predicted Points",title="2015 Season Simulation (Variable
     Penalty) - Predicted vs Expected")

predposdata<-as.data.frame(cbind(Team=as.character(VariablePen2015$Team),PredictedRank=c
    (1:18),ActualRank=subset(Ranking,Season==2015&Round==23)[match(VariablePen2015$Team,
    subset(Ranking,Season==2015&Round==23)$Team),]$Rank))
predposdata$PredictedRank<-as.numeric(predposdata$PredictedRank)#;predposdata$
    PredictedRank<-factor(predposdata$PredictedRank)
predposdata$ActualRank<-as.numeric(predposdata$ActualRank)#;predposdata$ActualRank<-
    factor(predposdata$ActualRank)
p2<-ggplot(data=predposdata, aes(x=ActualRank,y=PredictedRank,group=Team,colour=Team,
    shape=Team)) + geom_point(size=6) + geom_abline(slope=1) + scale_shape_manual(values
    =1:18) + labs(x="Actual Rank",y="Predicted Rank",title="2015 Season Simulation (
    Variable Penalty) - Ranking Prediction")

#point overview plot
p3<-ggplot(data=VariablePen2015, aes(y=Points,x=Team,colour=Team,shape=Team)) + geom_
    point(size=5)+ scale_shape_manual(values=1:18) + theme(axis.text.x = element_text(
    angle = 90, hjust = 1)) +  labs(title="2015 Season Simulation (Variable Penalty)")
p4<-ggplot(data=StaticPen2015, aes(y=Points,x=Team,colour=Team,shape=Team)) + geom_point(
```

```
size=5)+ scale_shape_manual(values=1:18) + theme(axis.text.x = element_text(angle = 90, hjust = 1)) + labs(title="2015 Season Simulation (Static Penalty)")



##Sensitivity Analysis

library(reshape2)
library(stringr)

RankDelta<-NULL
names<-NULL
for (i in 2009:2015){
  temp<-subset(Ranking,Season==i & Round==max(subset(Ranking,Season==i)$Round))[order(subset(Ranking,Season==i & Round== max(subset(Ranking,Season==i)$Round))$Team),]$Rank
  length(temp)<-18
  names<-c(names,paste(as.character(i)))
  RankDelta<-cbind(RankDelta,temp)
}
rownames(RankDelta)<-levels(Ranking$Team)
colnames(RankDelta)<-names

for (i in 1:6){

  RankDelta<-cbind(RankDelta,RankDelta[,i+1]-RankDelta[,i])
}

colnames(RankDelta)[8:13]<-c("2009-2010","2010-2011","2011-2012","2012-2013","2013-2014","2014-2015")

OvrDelta<-rowMeans(RankDelta[,8:13],na.rm = TRUE)
SdDelta<-apply(RankDelta[,8:13],1,sd,na.rm=TRUE)
SumStats<-cbind(as.data.frame(OvrDelta),as.data.frame(SdDelta),rownames(as.data.frame(SdDelta)))
colnames(SumStats)<-c("MeanDelta","SdDelta","Team")
rownames(SumStats)<-NULL



diffplot<-melt(RankDelta[,8:13],id.vars="Team", value.name="Diff", variable.name="Season")
colnames(diffplot)<-c("Team","Season","Diff")

rankplot<-melt(RankDelta[,1:7],id.vars="Team", value.name="Rank", variable.name="Season")
colnames(rankplot)<-c("Team","Season","Rank")

ggplot(data=diffplot, aes(x=Season,y=Diff,group=Team,colour=Team,shape=Team)) + geom_line() + geom_point(size=6,alpha=1/3) + scale_shape_manual(values=1:18) + labs(y="Rank Difference",title="Change in End of Season Ranking")
ggplot(data=rankplot, aes(x=Season,y=Rank,group=Team,colour=Team,shape=Team)) + geom_line() + geom_point(size=6,alpha=1) + scale_shape_manual(values=1:18) + labs(y="Ladder Rank",title="End of Season Ranking")
ggplot(data=SumStats, aes(x=Team, y=MeanDelta,colour=Team)) + geom_point(size=4,alpha=1) + labs(y="Change in Ladder Rank",x="Team",title="Average Change in Team Ranking") + geom_errorbar(aes(ymin=MeanDelta-SdDelta, ymax=MeanDelta+SdDelta), width=.1) + scale_x_discrete(labels = function(x) str_wrap(x, width = 10))
#ggplot(data=expplotdata, aes(x=Expected,y=Predicted,group=Team,colour=Team,shape=Team))
```

```
+ geom_point(size=6) + geom_abline(slope=1) + scale_shape_manual(values=1:18) + labs(x="Expected Points",y="Predicted Points",title="Variable Penalty Simulation for the 2015 AFL Season")
#ggplot(data=StaticPen2015, aes(y=Points,x=Team,colour=Team,shape=Team)) + geom_point(size=5)+ scale_shape_manual(values=1:18) + theme(axis.text.x = element_text(angle = 90, hjust = 1)) + labs(title="Static Penalty Performance Model for the 2015 AFL Season")


VarANOVA<-NULL #original did not work due to large amount of similar data
## if change seq to
VarHSD<-NULL
VarAOV<-NULL
for (minprob in seq(0.1,0.5,0.1)){ #min prob minprob
  for (maxpt in seq(5,12,1)){ # max pts maxpt
    for (maxprob in seq(0.9,0.5,-0.1)){ # max prob maxprob
      for (minpt in seq(0,5,1)){ # min pts minpt
        temp<-VariablePen(RawData=StaticData,Fixture=Fixture2015,MLRfn=Resultfn,minpr=minprob,maxpr=maxprob,minpt=minpt,maxpt=maxpt)
        tempbind<-cbind(temp,rep(minprob,18),rep(maxpt,18),rep(maxprob,18),rep(minpt,18))
        VarANOVA<-rbind(VarANOVA,tempbind)
      }
    }
  }

}

colnames(VarANOVA)[3:6]<-c("minprob","maxpt","maxprob","minpt")
VarANOVA$Team<-as.factor(VarANOVA$Team)
VarANOVA$minprob<-as.factor(VarANOVA$minprob)
VarANOVA$maxpt<-as.factor(VarANOVA$maxpt)
VarANOVA$maxprob<-as.factor(VarANOVA$maxprob)
VarANOVA$minpt<-as.factor(VarANOVA$minpt)
VarAOV<-aov(Points~Team+minprob*maxpt*maxprob*minpt,data=VarANOVA)
summary(VarAOV)
VarHSD<-TukeyHSD(VarAOV)

##Manual Interactions
library(agricolae)

man<-VarANOVA

man$i1<-with(man,interaction(minprob,maxpt))
man$i2<-with(man,interaction(minprob,maxprob))
man$i3<-with(man,interaction(maxpt,maxprob))
man$i4<-with(man,interaction(minprob,minpt))
man$i5<-with(man,interaction(maxpt,minpt))
man$i6<-with(man,interaction(maxprob,minpt))
man$i7<-with(man,interaction(minprob,maxpt,maxprob))
man$i8<-with(man,interaction(minprob,maxpt,minpt))
man$i9<-with(man,interaction(minprob,maxprob,minpt))
man$i10<-with(man,interaction(maxpt,maxprob,minpt))
man$i11<-with(man,interaction(minprob,maxpt,maxprob,minpt))

manAOV<-aov(Points~.,data=man)
summary(manAOV)

HSD.test(manAOV,trt="i2",group=TRUE)
```

```
##Not Used But A Useful Density Plot
#i2<-HSD.test(manAOV, trt="i2")
#i2plot<-ggplot(i2$groups,aes(x=means)) + geom_density() + coord_cartesian(ylim = c(0.0141, 0.045))
#i2d<-ggplot_build(i2plot)$data[[1]]
#i2plot + geom_area(data=subset(i2d,x< -5.01),aes(x=x,y=y),fill="steelblue",alpha=0.5) + geom_area(data=subset(i2d,x> 5.01),aes(x=x,y=y),fill="steelblue",alpha=0.5) + labs(title="Variable Penalty Model")

##Similarities Removed
newman<-VarANOVA
newman$i1<-with(newman,interaction(minprob,maxpt))
newman$i2<-with(newman,interaction(minprob,maxprob))
newman$i3<-with(newman,interaction(maxpt,maxprob))
newman$i4<-with(newman,interaction(minprob,minpt))
newman$i5<-with(newman,interaction(maxpt,minpt))
newman$i6<-with(newman,interaction(maxprob,minpt))
newman$i7<-with(newman,interaction(minprob,maxpt,maxprob))
newman$i8<-with(newman,interaction(minprob,maxpt,minpt))
newman$i9<-with(newman,interaction(minprob,maxprob,minpt))
newman$i10<-with(newman,interaction(maxpt,maxprob,minpt))
newman$i11<-with(newman,interaction(minprob,maxpt,maxprob,minpt))

trtexcl<-as.character(subset(i2$groups, M=="c" | M=="cd" | M=="de" | M=="e" | M=="ef" | M=="fg" | M=="g")$trt)
newman<-newman[! newman$i2 %in% trtexcl ,]

newman$i1<-factor(newman$i1)
newman$i2<-factor(newman$i2)
newman$i3<-factor(newman$i3)
newman$i4<-factor(newman$i4)
newman$i5<-factor(newman$i5)
newman$i6<-factor(newman$i6)
newman$i7<-factor(newman$i7)
newman$i8<-factor(newman$i8)
newman$i9<-factor(newman$i9)
newman$i10<-factor(newman$i10)
newman$i11<-factor(newman$i11)

newmanAOV<-aov(Points~.,data=newman)
summary(newmanAOV)

##Distribution Analysis
oldpoints<-subset(Ranking, Season==2015 & Round==23)$Points
staticnewpoints<-StaticPen2015$Points
variablenewpoints<-VariablePen2015$Points

densitydata<-as.data.frame(cbind(oldpoints,staticnewpoints,variablenewpoints))

dp<-ggplot(data=densitydata) + geom_density(aes(x=oldpoints,colour="Current Point Model",linetype="Current Point Model",geom="line")) + geom_density(aes(x=staticnewpoints,colour="Static Penalty Model",linetype="Static Penalty Model",geom="line")) + geom_density(aes(x=variablenewpoints,colour="Variable Penalty Model",linetype="Variable Penalty Model",geom="line")) + labs(x="Points",y="Density",title="Model Density Plots") + scale_colour_manual(values=c("Current Point Model"="red","Static Penalty Model"="blue","Variable Penalty Model"="black"),name="Model") + scale_linetype_manual(values=c("Current Point Model"=1,"Static Penalty Model"=1,"Variable Penalty Model"=1),name=
```

```
    "Model")

library(moments)

skewness(oldpoints) #
skewness(staticnewpoints) #
skewness(variablenewpoints) #
```

D.4 Fixture Difficulty R Code

```r
##Fixture Difficulty
##Created By: Casey Josman
##Last Edited: 10/01/2016

##LIBRARIES
library(car)
library(ggplot2)
library(stringr)

##FUNCTIONS

read.excel <- function(header=TRUE,...) {
  read.table("clipboard",sep="\t",header=header,...)
}

write.excel <- function(x,row.names=FALSE,col.names=TRUE,...) {
  write.table(x,"clipboard",sep="\t",row.names=row.names,col.names=col.names,...)
}

#predtab<-function(pred){ #pred=PREDICTION OF GEE MODEL, actual=RESULT COLUMN FROM DATASET, MUST SET SCALE PARAMETER INTERNALLY
  #  count<-0
  #  newtab<-data.frame()
  #  len<-length(pred)
  #
  # for(i in 1:len){
    #     if(pred[i]>=0.45095){
      #        newtab[i,1]=1}
    #     else{
      #        newtab[i,1]=0}
    #  }
  #     list(Pred=newtab)
  #}

SimStat<-function(Fixture,RankData){
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  temprank<-subset(RankData,Season==seatemp & Round==23)
  teams<-levels(Fixture$Home.team)
  StatLadder<-matrix(ncol=2,nrow=18)
  ind<-0

  for (t in teams){

    ind<-ind+1
    temphome<-subset(Fixture,Home.team==t)
    homediff<-0
    for (i in 1:nrow(temphome)){

      homerank<-as.numeric(subset(temprank,Team==temphome[i,]$Home.team)$Rank)
      awayrank<-as.numeric(subset(temprank,Team==temphome[i,]$Away.team)$Rank)

      homediff<-homediff+(homerank-awayrank)

    }
```

```
    tempaway<-subset(Fixture,Away.team==t)
    awaydiff<-0
    for (j in 1:nrow(tempaway)){

      homerank<-as.numeric(subset(temprank,Team==tempaway[j,]$Home.team)$Rank)
      awayrank<-as.numeric(subset(temprank,Team==tempaway[j,]$Away.team)$Rank)

      awaydiff<-awaydiff+(awayrank-homerank)

    }
    Difficulty<-homediff+awaydiff
    StatLadder[ind,]<-cbind(t,Difficulty)

  }
  standardize<-matrix(c(-186,-162.5,-139,-115.5,-92,-68.5,-58.5,-35,-11.5,11.5,35,58.5,68.5,92,115.5,139,162.5,186,2,ncol=2,byrow=FALSE)
  colnames(standardize)<-c("Mean","SD")
  colnames(StatLadder)<-c("Team","DifficultyRating")
  StatLadder<-as.data.frame(StatLadder)
  StatLadder$DifficultyRating<-as.character(StatLadder$DifficultyRating)
  StatLadder$DifficultyRating<-as.numeric(StatLadder$DifficultyRating)
  StatLadder<-StatLadder[match(temprank$Team,StatLadder$Team),]
  StatLadder[,2]<-(StatLadder[,2]-standardize[,1])/standardize[,2]
  StatLadder<-StatLadder[order(StatLadder[,2],decreasing=FALSE),]
  #rownames(StatLadder)<-StatLadder$Team
  #StatLadder<-StatLadder[,-1]
  return(StatLadder)
} #<0 easier fixture >0 harder fixture

SimLad<-function(RawData,Fixture,RankData,MLRfn,n=20){
  RawData$SeasonF<-as.factor(RawData$Season)
  RawData$RoundF<-as.factor(RawData$Round)
  RawData$Finals<-as.factor(RawData$Finals)
  RawData$Result<-as.factor(RawData$Result)
  RawData$HomeRank<-as.factor(RawData$HomeRank)
  RawData$AwayRank<-as.factor(RawData$AwayRank)
  teams<-levels(RawData$Home.team)
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  mlrtemp<-glm(MLRfn,data=subset(RawData,Season<=seatemp & Round<=24),family=binomial(logit))
  mlrtemp$xlevels[["SeasonF"]]<-union(mlrtemp$xlevels[["SeasonF"]],levels(Fixture$SeasonF))
  if (as.character(substitute(Fixture))=="Fixture2014"){
    mlrtemp$xlevels[["Venue"]]<-union(mlrtemp$xlevels[["Venue"]],"Traeger Park")
  } else {}
  predtemp<-predict(mlrtemp,Fixture,type="response")
  SimLadder<-data.frame(matrix(ncol=n,nrow=18))    #create data.frame(matrix(ncol=n,nrow=18)) name SimLadder
  set.seed(314)
  global.seed<-runif(n,0,10000)          #add precomputed list of seeds to make data reproducible

  for (i in 1:n){                   #add in loop from 1:n where n is = 20
    ind<-0
    SimTemp<-matrix(ncol=2,nrow=18)
    set.seed(global.seed[i])               #set.seed(seed.list[n])
```

```r
    restemp<-rbinom(length(predtemp),1,predtemp)     #replace with rbinom(n,1,predtemp)
    SimRes<-restemp                            #as above -> SimRes==restemp
    FixSim<-cbind(Fixture,SimRes)
    rownames(SimLadder)<-teams                    #rownames(data.frame)<-teams

    for (t in teams){
      ind<-ind+1
      tempwinsh<-nrow(subset(FixSim,Home.team==t & SimRes==1))
      tempwinsa<-nrow(subset(FixSim,Away.team==t & SimRes==0))
      temppoints<-4*(tempwinsh+tempwinsa)
      SimTemp[ind,]<-cbind(t,temppoints)

    }
    SimLadder[,i]<-as.numeric(SimTemp[,2])          #data.frame[,n]<-SimLadder$temppoints
  }                            #end new loop from 1:n

  MeanLadder<-cbind(teams,rowMeans(SimLadder))      #data.frame.new<-cbind(teams,rowMeans(data.frame))
  #change SimLadder below to data.frame.new

  colnames(MeanLadder)<-c("Team","Points")
  MeanLadder<-as.data.frame(MeanLadder)
  MeanLadder$Points<-as.character(MeanLadder$Points)
  MeanLadder$Points<-as.numeric(MeanLadder$Points)
  MeanLadder<-MeanLadder[order(MeanLadder[,2],decreasing=TRUE),]
  MeanLadder<-cbind(MeanLadder,Rank=rank(-MeanLadder[,2],ties.method = "average"))
  MeanLadder$Rank<-as.character(MeanLadder$Rank)
  MeanLadder$Rank<-as.numeric(MeanLadder$Rank)

  difftemp<-MeanLadder[match(teams,MeanLadder$Team),]$Rank #end of season rank
  ranktemp<-subset(RankData,Season==seatemp & Round==23)
  ranktemp1<-subset(RankData,Season==seatemp & Round==23)[,match(c("Team","Rank"),colnames(ranktemp))]
  ranktemp1<-ranktemp1[match(MeanLadder$Team,ranktemp1[,1]),]
  ranktemp<-ranktemp[match(teams,ranktemp$Team),]$Rank #beginning of season rank
  rawdiff<-cbind(teams,difftemp-ranktemp) #this is where the difference is calculated, it should be changes to output a correlation (both Pearson and Spearman) between beginning and end of season
  rawdiff<-rawdiff[match(MeanLadder$Team,rawdiff[,1]),]
  MeanLadder<-cbind(MeanLadder,PrevRank=ranktemp1[,2],Difficulty=rawdiff[,2])
  rownames(MeanLadder)<-MeanLadder$Team
  MeanLadder<-MeanLadder[,-1]
  return(MeanLadder)
} #smaller result means easier season

SimProb<-function(RawData,Fixture,MLRfn){
  RawData$SeasonF<-as.factor(RawData$Season)
  RawData$RoundF<-as.factor(RawData$Round)
  RawData$Finals<-as.factor(RawData$Finals)
  RawData$Result<-as.factor(RawData$Result)
  RawData$HomeRank<-as.factor(RawData$HomeRank)
  RawData$AwayRank<-as.factor(RawData$AwayRank)
  teams<-levels(RawData$Home.team)
  seatemp<-as.numeric(substr(deparse(substitute(Fixture)),start=8,stop=11))-1
  SimLadder<-matrix(ncol=2,nrow=18)
  ind<-0
  mlrtemp<-glm(MLRfn,data=subset(RawData,Season<=seatemp & Round<=24),family=binomial(logit))
```

```
  mlrtemp$xlevels[["SeasonF"]]<-union(mlrtemp$xlevels[["SeasonF"]],levels(Fixture$SeasonF))
  if (as.character(substitute(Fixture))=="Fixture2014"){
    mlrtemp$xlevels[["Venue"]]<-union(mlrtemp$xlevels[["Venue"]],"Traeger Park")
  } else {}
  restemp<-predict(mlrtemp,Fixture,type="response")
  SimRes<-restemp
  FixSim<-cbind(Fixture,SimRes)
  ProbLadder<-matrix(ncol=2,nrow=18)
  for (t in teams){

    ind<-ind+1
    temphome<-subset(FixSim,Home.team==t)
    tempaway<-subset(FixSim,Away.team==t)

    #WinProb<-prod(temphome$SimRes)*prod(1-tempaway$SimRes) #we could also take the sum of the log(prob)
    WinProb<-(mean(temphome$SimRes)+mean(1-tempaway$SimRes))/2

    ProbLadder[ind,]<-cbind(t,WinProb)

  }
  colnames(ProbLadder)<-c("Team","WinProb")
  ProbLadder<-as.data.frame(ProbLadder)
  ProbLadder$WinProb<-as.character(ProbLadder$WinProb)
  ProbLadder$WinProb<-as.numeric(ProbLadder$WinProb)
  ProbLadder<-ProbLadder[order(ProbLadder[,2],decreasing=TRUE),]
  #rownames(ProbLadder)<-ProbLadder$Team
  #ProbLadder<-ProbLadder[,-1]
  return(ProbLadder)
}

##READ DATA
StaticData <- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Historic Sensitivity/6-5.csv",header=TRUE)
StaticData<-subset(StaticData,Season>=2001)
StaticData$SeasonF<-as.factor(StaticData$Season)
StaticData$RoundF<-as.factor(StaticData$Round)
StaticData$Finals<-as.factor(StaticData$Finals)
StaticData$Result<-as.factor(StaticData$Result)
StaticData$HomeRank<-as.factor(StaticData$HomeRank)
StaticData$AwayRank<-as.factor(StaticData$AwayRank)
Season2016<-read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/2016 Raw Data Pre-Season.csv",header=TRUE)
Season2016$SeasonF<-as.factor(Season2016$Season)
Season2016$RoundF<-as.factor(Season2016$Round)
Ranking <- read.csv("C:/Users/Casey Josman/Dropbox/PhD. Research/Data/Ranking Table.csv",header=TRUE)
#Ranking$Team<-recode(Ranking$Team,'"AD"="Adelaide";"BL"="Brisbane Lions";"CA"="Carlton";"CW"="Collingwood";"ES"="Essendon";"FR"="Fremantle";"GC"="Gold Coast";"GE"="Geelong";"GW"="Greater Western Sydney";"HW"="Hawthorn";"ME"="Melbourne";"NM"="North Melbourne";"PA"="Port Adelaide";"RI"="Richmond";"SK"="St Kilda";"SY"="Sydney";"WB"="Western Bulldogs";"WC"="West Coast"')

FixVars<-match(c("Head2Head","PastHome","PastAway","HomeRank","AwayRank"),colnames(StaticData))
FixNames<-c("Head2Head","PastHome","PastAway","HomeRank","AwayRank","Finals")

```

```r
Fixture2014<-subset(StaticData,Season==2014 & Round<=23,select=c(2,3,4,5,6,17,18)) # update fixture to iterate for (season-1) home.team away.team then take tail for statistics
Fixture2014[,FixNames]<-NA

for (i in 1:nrow(Fixture2014)){
  con<-data.frame(matrix(ncol=5,nrow=1))
  hometemp<-Fixture2014[i,]$Home.team
  awaytemp<-Fixture2014[i,]$Away.team
  tempcon<-tail(subset(StaticData,Season<2014 & Home.team==hometemp & Away.team==awaytemp),n=1)

  if(nrow(tempcon)==0){ #if no recent match home vs away is detected takes inverse of latest away vs home
    tempcon<-tail(subset(StaticData,Season<2014 & Home.team==awaytemp & Away.team==hometemp),n=1)
    con<-cbind(1-tempcon[12],tempcon[14],tempcon[13],as.character(tempcon[16]),as.character(tempcon[15]))
    colnames(con)<-FixNames[-6] #-6 removes finals label
  } else {
    con<-tempcon[FixVars]
  }

  Fixture2014[i,FixNames]<-c(con,0) #0 indicates home and away series (not finals)
}
Fixture2014$HomeRank<-as.factor(Fixture2014$HomeRank)
Fixture2014$AwayRank<-as.factor(Fixture2014$AwayRank)
Fixture2014$Finals<-as.factor(Fixture2014$Finals)

Fixture2015<-subset(StaticData,Season==2015 & Round<=23,select=c(2,3,4,5,6,17,18))
Fixture2015[,FixNames]<-NA

for (i in 1:nrow(Fixture2015)){
  con<-data.frame(matrix(ncol=5,nrow=1))
  hometemp<-Fixture2015[i,]$Home.team
  awaytemp<-Fixture2015[i,]$Away.team
  tempcon<-tail(subset(StaticData,Season<2015 & Home.team==hometemp & Away.team==awaytemp),n=1)

  if(nrow(tempcon)==0){
    tempcon<-tail(subset(StaticData,Season<2015 & Home.team==awaytemp & Away.team==hometemp),n=1)
    con<-cbind(1-tempcon[12],tempcon[14],tempcon[13],as.character(tempcon[16]),as.character(tempcon[15]))
    colnames(con)<-FixNames[-6] #-6 removes finals label
  } else {
    con<-tempcon[FixVars]
  }

  Fixture2015[i,FixNames]<-c(con,0)
}
Fixture2015$HomeRank<-as.factor(Fixture2015$HomeRank)
Fixture2015$AwayRank<-as.factor(Fixture2015$AwayRank)
Fixture2015$Finals<-as.factor(Fixture2015$Finals)

Fixture2016<-Season2016[,-3]
Fixture2016[,FixNames]<-NA

```

```
for (i in 1:nrow(Fixture2016)){
  con<-data.frame(matrix(ncol=5,nrow=1))
  hometemp<-Fixture2016[i,]$Home.team
  awaytemp<-Fixture2016[i,]$Away.team
  tempcon<-tail(subset(StaticData,Season<2016 & Home.team==hometemp & Away.team==awaytemp
      ),n=1)

  if(nrow(tempcon)==0){
    tempcon<-tail(subset(StaticData,Season<2016 & Home.team==awaytemp & Away.team==
        hometemp),n=1)
    con<-cbind(1-tempcon[12],tempcon[14],tempcon[13],as.character(tempcon[16]),as.
        character(tempcon[15]))
    colnames(con)<-FixNames[-6] #-6 removes finals label
  } else {
    con<-tempcon[FixVars]
  }

  Fixture2016[i,FixNames]<-c(con,0)
}
Fixture2016$HomeRank<-as.factor(Fixture2016$HomeRank)
Fixture2016$AwayRank<-as.factor(Fixture2016$AwayRank)
Fixture2016$Finals<-as.factor(Fixture2016$Finals)


teams<-levels(StaticData$Home.team)

##MODELS
nonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Home.team","Away.
    team","Finals","Season","Round"),colnames(StaticData))
Resultfn=as.formula(paste("Result~",paste(colnames(StaticData[,-nonfeat]),collapse="+")))

##Difficulty Using Static Rank (Final of Previous Season)

(SimStat2014<-SimStat(Fixture=Fixture2014,RankData=Ranking))
#cor(cbind(SimStat2014,Rank=subset(Ranking,Season==2013 & Round==23)[match(SimStat2014$
    Team,subset(Ranking,Season==2013 & Round==23)$Team),7])[,2:3])
(SimStat2015<-SimStat(Fixture=Fixture2015,RankData=Ranking))
#cor(cbind(SimStat2015,Rank=subset(Ranking,Season==2014 & Round==23)[match(SimStat2015$
    Team,subset(Ranking,Season==2014 & Round==23)$Team),7])[,2:3])
(SimStat2016<-SimStat(Fixture=Fixture2016,RankData=Ranking))
#cor(cbind(SimStat2016,Rank=subset(Ranking,Season==2015 & Round==23)[match(SimStat2016$
    Team,subset(Ranking,Season==2015 & Round==23)$Team),7])[,2:3])

##Difficulty Using Simulated Results and Ranks
(SimLad2014<-SimLad(Fixture=Fixture2014,RawData=StaticData,RankData=Ranking,MLRfn=
    Resultfn,n=1000))
cor2014spearman<-cor(x=SimLad2014$PrevRank,y=SimLad2014$Rank,method="spearman")
cor2014pearson<-cor(x=SimLad2014$PrevRank,y=SimLad2014$Rank,method="pearson")
(SimLad2015<-SimLad(Fixture=Fixture2015,RawData=StaticData,RankData=Ranking,MLRfn=
    Resultfn,n=1000))
cor2015spearman<-cor(x=SimLad2015$PrevRank,y=SimLad2015$Rank,method="spearman")
cor2015pearson<-cor(x=SimLad2015$PrevRank,y=SimLad2015$Rank,method="pearson")
(SimLad2016<-SimLad(Fixture=Fixture2016,RawData=StaticData,RankData=Ranking,MLRfn=
    Resultfn,n=1000))
cor2016spearman<-cor(x=SimLad2016$PrevRank,y=SimLad2016$Rank,method="spearman")
cor2016pearson<-cor(x=SimLad2016$PrevRank,y=SimLad2016$Rank,method="pearson")

##Difficulty Using Simulated Probabilities
```

```
(SimProb2014<-SimProb(Fixture=Fixture2014,RawData=StaticData,MLRfn=Resultfn))
(SimProb2015<-SimProb(Fixture=Fixture2015,RawData=StaticData,MLRfn=Resultfn))
(SimProb2016<-SimProb(Fixture=Fixture2016,RawData=StaticData,MLRfn=Resultfn))

#Plots and Cluster Analysis
setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Results\\2017\\Fixture Difficulty")

SimStat2015<-cbind(SimStat2015,Ranking=subset(Ranking,Season==2014 & Round==23)[match(SimStat2015$Team,subset(Ranking,Season==2014 & Round==23)$Team),]$Rank)
ggplot(SimStat2015,aes(x=Ranking,y=DifficultyRating,group=Team,colour=Team,shape=Team)) + geom_point(size=6) + scale_shape_manual(values=1:18) + labs(x="Starting Rank",y="Difficulty Rating",title="Previous Season Ranking Model for the 2015 AFL Season")
#Upd2014<-
Upd2015<-read.csv("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Results\\2017\\PlotData (Fixture Difficulty and Performance Models).csv")

plot(SimLad2014)
#ggplot(Upd2014, aes(y=Predicted.Rank,x=Actual.Rank,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Actual Rank",y="Predicted Rank",title="Season Ranking Simulation for the 2014 AFL Season") + geom_abline(slope = 1)
p1<-ggplot(Upd2014, aes(y=Points,x=Season.Difficulty.Stat,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Points",title="Season Ranking Simulation for the 2014 AFL Season")
p2<-ggplot(Upd2014, aes(y=Points,x=Season.Difficulty.Sim,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Points",title="Season Ranking Simulation for the 2014 AFL Season")
p3<-ggplot(Upd2014, aes(y=Static.Performance,x=Season.Difficulty.Stat,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Team Performance",title="Performance Evaluation for the 2014 AFL Season")
p4<-ggplot(Upd2014, aes(y=Variable.Performance,x=Season.Difficulty.Stat,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Team Performance",title="Performance Evaluation for the 2014 AFL Season")

plot(SimLad2015)
#ggplot(Upd2015, aes(y=Predicted.Rank,x=Actual.Rank,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Actual Rank",y="Predicted Rank",title="Season Ranking Simulation for the 2015 AFL Season") + geom_abline(slope = 1)
p5<-ggplot(Upd2015, aes(y=Points,x=SeasonDifficultyStat,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Points",title="Season Ranking Simulation for the 2015 AFL Season")
p6<-ggplot(Upd2015, aes(y=Points,x=SeasonDifficultySim,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Points",title="Season Ranking Simulation for the 2015 AFL Season")
p7<-ggplot(Upd2015, aes(y=StaticPerformance,x=SeasonDifficultyStat,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Team Performance",title="Performance Evaluation for the 2015 AFL Season")
p8<-ggplot(Upd2015, aes(y=VariablePerformance,x=SeasonDifficultyStat,group=Team,colour=Team,shape=Team)) + geom_point(size=5) + scale_shape_manual(values=1:18) + labs(x="Season Difficulty",y="Team Performance",title="Performance Evaluation for the 2015 AFL Season")
```

```r
dp1<-ggplot(SimStat2015, aes(x = Team, y = DifficultyRating, fill=Team,color=Team)) + geom_bar(stat = "identity") + theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust=0.3),text = element_text(size=14)) + labs(y="Difficulty Rating",title="Season Difficulty (Static) for the 2015 AFL Season") + geom_hline(yintercept = 0.3) + geom_hline(yintercept = -0.3)
dp2<-ggplot(SimLad2015, aes(x = Team, y = as.numeric(as.character(Difficulty)), fill=Team,color=Team)) + geom_bar(stat = "identity") + theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust=0.3),text = element_text(size=14)) + labs(y="Difficulty Rating",title="Season Difficulty (Simulation) for the 2015 AFL Season") + geom_hline(yintercept = 2) + geom_hline(yintercept = -2)
dp3<-ggplot(SimProb2015, aes(x = Team, y = WinProb, fill=Team,color=Team)) + geom_bar(stat = "identity") + theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust=0.3),text = element_text(size=14)) + labs(y="Average Win Percentage",title="Season Difficulty (Probabilistic) for the 2015 AFL Season")

tiff("2015 Season Ranking Simulation (pts-statdiff).tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
p5
dev.off()

tiff("2015 Season Ranking Simulation (pts-simdiff).tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
p6
dev.off()

tiff("2015 Performance Evaluation (staticperf-statdiff).tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
p7
dev.off()

tiff("2015 Performance Evaluation (varperf-statdiff).tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
p8
dev.off()

tiff("2015 Static Difficulty.tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
dp1
dev.off()

tiff("2014 Dendrogram.tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
dp2
dev.off()

tiff("2015 Probabilistic Difficulty.tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
dp3
dev.off()

plot(SimLad2016)

dist2014<-dist(SimLad2014[,c(3,2)]) #dist between previous rank and end of season rank
hc2014<-hclust(dist2014)
tiff("2014 Dendrogram.tiff", width = 24, height = 24, units = 'cm', res = 300, compression = 'lzw')
plot(hc2014,main="Cluster Dendrogram for the 2014 AFL Season") #grouped relative performance in 2014 season
```

```
dev.off()


dist2015<-dist(SimLad2015[,c(3,2)]) #dist between previous rank and end of season rank
hc2015<-hclust(dist2015)
tiff("2015 Dendrogram.tiff", width = 24, height = 24, units = 'cm', res = 300,
    compression = 'lzw')
plot(hc2015,main="Cluster Dendrogram for the 2015 AFL Season") #grouped relative
    performance in 2015 season
dev.off()


dist2016<-dist(SimLad2016[,c(3,2)]) #dist between previous rank and end of season rank
hc2016<-hclust(dist2016)
tiff("2016 Dendrogram.tiff", width = 24, height = 24, units = 'cm', res = 300,
    compression = 'lzw')
plot(hc2016,main="Cluster Dendrogram for the 2016 AFL Season") #grouped relative
    performance in 2016 season
dev.off()
```

R Code for Dynamic Models

E.1 Dynamic Model R Code

```r
##MARKOV MODEL WORKING VER
##CREATED BY: CASEY JOSMAN
##LAST EDITED: 08/12/2018

##LIBRARIES
library(msm)
library(doParallel)
library(ggplot2)
library(reshape2)
library(zoo)
library(car)
library(expm)

##FUNCTIONS

read.excel <- function(header=TRUE,...) {
  read.table("clipboard",sep="\t",header=header,...)
}

write.excel <- function(x,row.names=FALSE,col.names=TRUE,...) {
  write.table(x,"clipboard",sep="\t",row.names=row.names,col.names=col.names,...)
}

lay_out = function(...) { #source https://github.com/cran/wq/blob/8223
    da687d8daff2ad612f9a07926f412a08ba82/R/layOut.R
  x <- list(...)
  n <- max(sapply(x, function(x) max(x[[2]])))
  p <- max(sapply(x, function(x) max(x[[3]])))
  grid::pushViewport(grid::viewport(layout = grid::grid.layout(n, p)))

  for (i in seq_len(length(x))) {
    print(x[[i]][[1]], vp = grid::viewport(layout.pos.row = x[[i]][[2]],
    layout.pos.col = x[[i]][[3]]))
  }
}

RealTimeResult<-function(data){

  tempHome<-1*data$H.BEHI+6*data$H.GOAL
```

```
  tempAway<-1*data$A.BEHI+6*data$A.GOAL
  tempMargin<-tempHome-tempAway
  tempResult<-rep(0,length(tempMargin))
  tempResult[which(tempMargin==0)]<-1 #Draw
  tempResult[which(tempMargin<0)]<-2 #Loss
  tempResult[which(tempMargin>0)]<-3 #Win

  return(tempResult)

}

CumulTime<-function(data){ #Calculates full game time (adds previous quarter end time)

  tempTime<-NULL
  StartIndex<-as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))
  EndIndex<-c(as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))[-1]-1,nrow(data))

  for (i in 1:length(StartIndex)){
    tempind<-StartIndex[i]:EndIndex[i]

    tempData<-data[tempind,c("TIME_SEC","QUARTER")]
    t1<-as.numeric(subset(tempData,QUARTER==1)$TIME_SEC)
    t2<-as.numeric(subset(tempData,QUARTER==2)$TIME_SEC)+max(t1)
    t3<-as.numeric(subset(tempData,QUARTER==3)$TIME_SEC)+max(t2)
    t4<-as.numeric(subset(tempData,QUARTER==4)$TIME_SEC)+max(t3)

    tempCalc<-c(t1,t2,t3,t4)


    tempTime<-c(tempTime,tempCalc)
  }

  return(tempTime)

}

MatchInd<-function(data){ #Assigns unique MatchNo indicator

  tempNo<-NULL
  StartIndex<-as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))
  EndIndex<-c(as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")])))[-1]-1,nrow(data))

  for (i in 1:length(StartIndex)){
    tempind<-StartIndex[i]:EndIndex[i]
    matchtemp<-rep(i,length(tempind))

    tempNo<-c(tempNo,matchtemp)
  }

  return(tempNo)

}

OffsetTime<-function(data,delta=0.0001){
```

```
  TimeOff<-NULL
  sig<-nchar(gsub("(.*)(\\.)|([0]*$)","",format(delta,scientific=FALSE)))
  StartIndex<-as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")
      ])))
  EndIndex<-c(as.numeric(rownames(unique(data[,c("Date","Round","Home.team","Away.team")
      ])))[-1]-1,nrow(data))

  for (i in 1:length(StartIndex)){

    tempTime<-round(data$CumulT[StartIndex[i]:EndIndex[i]],digits=sig)

    IndE<-which(duplicated(tempTime)) #gives location of second value in duplicate (need
        to get value before)

    #IndT<-c(IndS,IndE)

    for (j in IndE){

      IndS<-which(tempTime==tempTime[j]) #gives location of all matching duplicates

      if(length(IndS)==0){

      } else{

        tempTime[IndS]<-tempTime[which(tempTime==tempTime[j])]+seq(0,(length(which(
            tempTime==tempTime[j]))-1)*delta,delta)

      }


    }


    TimeOff<-c(TimeOff,tempTime)
  }

  return(TimeOff)

}
##WE NEED TO EXTRACT THE CONFIDENCE INTERVALS FOR EACH ITERATION TO BE USED LATER
PredictMSM<-function(model=NULL,covariates=NULL,data=NULL,initialprobs=NULL,lengthThresh
    =50){ #must also produce plot (try ggplot)
  ProbRes<-NULL
  ForeRes<-NULL
  if (is.null(covariates)==FALSE){
    tempPredData<-data[,match(c("CumulT",covariates),colnames(data))]
    lenR<-nrow(tempPredData)

    initCov<-list()
    initCov[[1]]<-as.list(rep(0,length(covariates))) #creates an extra initial null
        covariate as we need times+1 covariates
    #covariateList<-lapply(1:3,function(n) list(treat1=FullMarkovDataT0$CumulT[n],treat2=
        FullMarkovDataT0$TIME_SEC[n])) #two case example
    covariateList<-lapply(1:lenR,function(x) as.list(tempPredData[,match(c(covariates),
        colnames(tempPredData))][x,])) #this is the generalisation that replaces the
        above
    covariateList<-c(initCov,covariateList) #joins initial covariates with full list set
```

```r
  covariateList<-lapply(covariateList,function(x) setNames(x,covariates)) #gives each element of list an appropriate name

  pb <- txtProgressBar(min = 0, max = nrow(data), style = 3)
  PMatShort<-list() #Stepwise P-Matrices
  Pcomp<-list()
  Pfore<-list()

  for (i in 1:nrow(data)){#needs to be in loop x=model, t1=0 , t2=CumulT[i], times=c(0,CumulT[i-1],CumulT[i]), covariate=covariateList[1:(i+1)]

    #PMatTemp<-pmatrix.piecewise.msm(x=model,t1=0,t2=tempPredData$CumulT[i],times=tempPredData$CumulT[1:i],covariates=covariateList[1:(i+1)],cores=4)

    ##insert optimised stages here##
    if(i==1){
      PMatShort[[1]]<-pmatrix.piecewise.msm(x=model,t1=0,t2=tempPredData$CumulT[i],times=tempPredData$CumulT[i],covariates=covariateList[i:(i+1)],cores=4)
      #PMatFore<-pmatrix.piecewise.msm(x=model,t1=i,t2=tempPredData$CumulT[nrow(tempPredData)],times=tempPredData$CumulT[1:i],covariates=covariateList[1:(i+1)],cores=4)
    }else{
      PMatShort[[i]]<-pmatrix.piecewise.msm(x=model,t1=tempPredData$CumulT[i-1],t2=tempPredData$CumulT[i],times=tempPredData$CumulT[(i-1):i],covariates=covariateList[(i-1):(i+1)],cores=4)
      #PMatFore<-pmatrix.piecewise.msm(x=model,t1=i,t2=tempPredData$CumulT[nrow(tempPredData)],times=tempPredData$CumulT[1:i],covariates=covariateList[1:(i+1)],cores=4)
    }

    PMatFore<-NULL
    if (model$qmodel$nstates==3){
      PMatFore<-matrix(diag(3),nrow=3,ncol=3)
      colnames(PMatFore)<-c("Draw","Loss","Win")
      rownames(PMatFore)<-c("Draw","Loss","Win")
    }else{
      PMatFore<-matrix(diag(2),nrow=2,ncol=2)
      colnames(PMatFore)<-c("Loss","Win")
      rownames(PMatFore)<-c("Loss","Win")
    }

    # REMOVED FOR OPTIMISED STEP BELOW
    # for (fc in i:(nrow(data)-1)){ #forward prediction of P Matrix from observed point i to end point T
      #PMatFore<-PMatFore%*%pmatrix.piecewise.msm(x=model,t1=tempPredData$CumulT[fc],t2=tempPredData$CumulT[(fc+1)],times=tempPredData$CumulT[1:i],covariates=covariateList[1:(i+1)],cores=4)
      # }

    PMatFore<-pmatrix.piecewise.msm(x=model,t1=tempPredData$CumulT[i],t2=tempPredData$CumulT[nrow(data)],times=tempPredData$CumulT[1:i],covariates=covariateList[1:(i+1)],cores=4)

    if (i==1){
      Pcomp[[1]]<-PMatShort[[1]]
      Pfore[[1]]<-PMatShort[[1]]%*%PMatFore
    }else{
      Pcomp[[i]]<-Pcomp[[(i-1)]]%*%PMatShort[[i]] #P(t)=P(0,t-1)P(t-1,t)
```

```r
          Pfore[[i]]<-Pcomp[[(i-1)]]%*%PMatShort[[i]]%*%PMatFore
        }



        if (i==1){
          ProbTemp<-initialprobs%*%Pcomp[[1]] #u(1)=u(0)P(0)
          ProbForeTemp<-initialprobs%*%Pfore[[1]]
        } else {
          ProbTemp<-initialprobs%*%Pcomp[[i]] #u(t)=u(0)P(0,t-1) => u(t+1)=u(0)P(0,t)
          ProbForeTemp<-initialprobs%*%Pfore[[i]]
        }

        ProbRes<-rbind(ProbRes,ProbTemp)
        ForeRes<-rbind(ForeRes,ProbForeTemp)

        setTxtProgressBar(pb, i)
      }



      close(pb)
      rownames(ProbRes)<-1:nrow(ProbRes)
      ProbRes<-cbind(ProbRes,Time=tempPredData$CumulT)
      ForeRes<-cbind(ForeRes,Time=tempPredData$CumulT)

    }else{
      tempPredData<-data[,match(c("CumulT"),colnames(data))]

      pb <- txtProgressBar(min = 0, max = length(data), style = 3)
      PMatShort<-list() #Stepwise P-Matrices
      Pcomp<-list()
      Pfore<-list()

      for (i in 1:length(data)){

        #PMatTemp<-pmatrix.msm(x=model,t=tempPredData[i],t1=0,covariates=0,cores=4)

        ##insert optimised stages here##
        if(i==1){
          PMatShort[[1]]<-pmatrix.msm(x=model,t=tempPredData[i],t1=0,covariates=0,cores=4)
        }else{
          PMatShort[[i]]<-pmatrix.msm(x=model,t=tempPredData[i],t1=tempPredData[i-1],
              covariates=0,cores=4)
        }

        PMatFore<-NULL
        if (model$qmodel$nstates==3){
          PMatFore<-matrix(diag(3),nrow=3,ncol=3)
          colnames(PMatFore)<-c("Draw","Loss","Win")
          rownames(PMatFore)<-c("Draw","Loss","Win")
        }else{
          PMatFore<-matrix(diag(2),nrow=2,ncol=2)
          colnames(PMatFore)<-c("Loss","Win")
          rownames(PMatFore)<-c("Loss","Win")
        }

        # REMOVED FOR OPTIMISED STEP BELOW
```

```
      # for (fc in i:(nrow(data)-1)){ #forward prediction of P Matrix from observed point
          i to end point T
        # PMatFore<-PMatFore%*%pmatrix.msm(x=model,t=tempPredData[(fc+1)],t1=tempPredData
            [fc],covariates=0,cores=4)
        # }

      PMatFore<-pmatrix.msm(x=model,t=tempPredData[nrow(data)],t1=tempPredData[i],
          covariates=0,cores=4)

      if (i==1){
        Pcomp[[1]]<-PMatShort[[1]]
        Pfore[[1]]<-PMatShort[[1]]%*%PMatFore
      }else{
        Pcomp[[i]]<-Pcomp[[(i-1)]]%*%PMatShort[[i]]
        Pfore[[i]]<-Pcomp[[(i-1)]]%*%PMatShort[[i]]%*%PMatFore
      }

      if (i==1){
        ProbTemp<-initialprobs%*%Pcomp[[1]] #u(1)=u(0)P(0)
        ProbForeTemp<-initialprobs%*%Pfore[[1]]
      } else {
        ProbTemp<-initialprobs%*%Pcomp[[i]] #u(t)=u(0)P(0,t-1) => u(t+1)=u(0)P(0,t)
        ProbForeTemp<-initialprobs%*%Pfore[[i]]
      }

      ProbRes<-rbind(ProbRes,ProbTemp)
      ForeRes<-rbind(ForeRes,ProbForeTemp)
      setTxtProgressBar(pb, i)
    }

    close(pb)
    rownames(ProbRes)<-1:nrow(ProbRes)
    ProbRes<-cbind(ProbRes,Time=tempPredData)
    ForeRes<-cbind(ForeRes,Time=tempPredData)

  }

  templong<-as.data.frame(ProbRes)
  longlong <- melt(templong, id.vars = "Time")

  foretemplong<-as.data.frame(ForeRes)
  forelonglong<-melt(foretemplong, id.vars = "Time")

  predResT<-data.frame(matrix(nrow=nrow(ProbRes),ncol=1))
  foreResT<-data.frame(matrix(nrow=nrow(ProbRes),ncol=1))

  if (length(initialprobs)==3){
    for (p in 1:nrow(ProbRes)){
      predResT[p,]<-names(which.max(ProbRes[p,1:3]))
      foreResT[p,]<-names(which.max(ForeRes[p,1:3]))
    }
  } else{
    for (p in 1:nrow(ProbRes)){
      predResT[p,]<-names(which.max(ProbRes[p,1:2]))
      foreResT[p,]<-names(which.max(ForeRes[p,1:2]))
    }
  }

```

```
names(predResT)<-"predResT"
names(foreResT)<-"foreResT"

actRes<-as.data.frame(recode(data$ResT,'"1"="Draw";"2"="Loss";"3"="Win"'), stringsAsFactors=FALSE)
names(actRes)<-"actRes"

endRes<-as.data.frame(matrix(actRes[nrow(data),],ncol=1,nrow=nrow(data)))
names(endRes)<-"endRes"
endRes$endRes<-as.character(endRes$endRes)

endlong<-as.data.frame(cbind(Actual=actRes$actRes,Predicted=predResT$predResT,Time=templong$Time))
endlong$Actual<-as.character(endlong$Actual)
endlong$Predicted<-as.character(endlong$Predicted)
endlong$Time<-as.numeric(as.character(endlong$Time))

foreendlong<-as.data.frame(cbind(Actual=endRes$endRes,Predicted=foreResT$foreResT,Time=foretemplong$Time))
foreendlong$Actual<-as.character(foreendlong$Actual)
foreendlong$Predicted<-as.character(foreendlong$Predicted)
foreendlong$Time<-as.numeric(as.character(foreendlong$Time))

endlonglong<-melt(endlong,id.vars="Time")
foreendlonglong<-melt(foreendlong,id.vars="Time")
#templong<-melt(ProbRes) #not sure why this was here (breaks plot)


##NEED TO ADD EXTRA PLOTS FOR END OF MATCH FORECAST
if (length(initialprobs)==3){ #predicted probability plot
  gg<-ggplot(data=templong,aes(Time)) + geom_line(aes(y=Draw,colour="Draw")) + geom_line(aes(y=Loss,colour="Loss")) + geom_line(aes(y=Win,colour="Win")) + labs(title="Probability of Match Outcome Over Time",y="Probability", color="Match Outcome") + scale_colour_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2"))
} else {if (length(initialprobs)==2){
    gg<-ggplot(data=templong,aes(Time)) + geom_line(aes(y=Loss,colour="Loss")) + geom_line(aes(y=Win,colour="Win")) + labs(title="Probability of Match Outcome Over Time",y="Probability", color="Match Outcome") + scale_colour_manual(values=c("Loss"="Red","Win"="Green2"))
  }

}

##actual (text) and predicted (bar)

if (length(initialprobs)==3){ #predicted probability plot
  hh<-ggplot() + geom_bar(data=longlong,aes(x=Time,y=value,colour=variable,fill=variable), position = "fill", stat = "identity") + geom_point(data = data,aes(y=scale(as.numeric(ResT),center=0.5,scale=3),x=CumulT)) + labs(title="Match Results Over Time",x="Time",y="Outcome",legend="Match Outcome") + scale_colour_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + scale_fill_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + labs(fill="Match Prediction", colour="Match Prediction") + scale_y_continuous(breaks=c(1/6,3/6,5/6),labels=c("Draw","Loss","Win"))
} else {if (length(initialprobs)==2){
    hh<-ggplot() + geom_bar(data=longlong,aes(x=Time,y=value,colour=variable,fill=variable), position = "fill", stat = "identity") + geom_point(data = data,aes(y=scale(as.numeric(ResT),center=0.5,scale=3),x=CumulT)) + labs(title="Match
```

```
Results Over Time",x="Time",y="Outcome",legend="Match Outcome") + scale_colour_manual(values=c("Loss"="Red","Win"="Green2")) + scale_fill_manual(values=c("Loss"="Red","Win"="Green2")) + labs(fill="Match Prediction",colour="Match Prediction") + scale_y_continuous(breaks=c(1/6,3/6,5/6),labels=c("Draw","Loss","Win"))
  }

}

##outcome heatmap
ii<-ggplot() + geom_tile(data=endlonglong,aes(Time,variable, fill=value,colour=value)) + labs(ylab("Outcome")) + geom_hline(yintercept=1.5,colour="white") + scale_colour_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + scale_fill_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + labs(fill="",colour="") + theme(plot.margin = unit(c(0.2,3.9,0.2,0.2), "cm"),legend.position="none")

##margin plot
jj<-ggplot(data=data,aes(x=CumulT,y=MarginT,colour=MarginT)) + geom_line() + scale_color_gradient2(midpoint=0, low="red", mid="grey65",high="green2") + theme(panel.background = element_rect(fill="white", colour="black"),panel.grid.major = element_blank(), panel.grid.minor = element_blank(),legend.position="none") + labs(x="Time",y="Margin") + theme(plot.margin = unit(c(0.2,3.9,0.2,0.2), "cm"))

u<-union(actRes$actRes,predResT$predResT)
tempTable<-table(Actual=factor(actRes$actRes,u),Predicted=factor(predResT$predResT,u))
CumulTAccuracy<-sum(diag(tempTable))/sum(tempTable) #prediction accuracy over every epoch

uFore<-union(endRes$endRes,foreResT$foreResT)
tempTableFore<-table(Actual=factor(endRes$endRes,uFore),Predicted=factor(foreResT$foreResT,uFore))
CumulTAccuracyFore<-sum(diag(tempTableFore))/sum(tempTableFore) #prediction accuracy over every epoch

if (length(initialprobs)==3){ #predicted probability plot
  kk<-ggplot() + geom_bar(data=forelonglong,aes(x=Time,y=value,colour=variable,fill=variable), position = "fill", stat = "identity") + labs(title="Final Match Outcome Prediction at Time",x="Time",y="Outcome",legend="Match Outcome") + scale_colour_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + scale_fill_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + labs(fill="Match Prediction",colour="Match Prediction")
} else {if (length(initialprobs)==2){
    kk<-ggplot() + geom_bar(data=forelonglong,aes(x=Time,y=value,colour=variable,fill=variable), position = "fill", stat = "identity") + labs(title="Final Match Outcome Prediction at Time",x="Time",y="Outcome",legend="Match Outcome") + scale_colour_manual(values=c("Loss"="Red","Win"="Green2")) + scale_fill_manual(values=c("Loss"="Red","Win"="Green2")) + labs(fill="Match Prediction",colour="Match Prediction")
  }

}

##forecast outcome heatmap
ll<-ggplot() + geom_tile(data=foreendlonglong,aes(Time,variable, fill=value,colour=value)) + labs(ylab("Outcome")) + geom_hline(yintercept=1.5,colour="white") + scale_colour_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + scale_fill_manual(values=c("Draw"="Grey65","Loss"="Red","Win"="Green2")) + labs(fill="",colour="") + theme(plot.margin = unit(c(0.2,3.9,0.2,0.2), "cm"),legend.position="none")
```

```r


  # PMatLong<-list() #Reconstructed list of P-Matrices
  # PMatLong[[1]]<-PMatShort[[1]]
  # for(i in 2:nrow(data)){
    # PMatLong[[i]]<-PMatLong[[(i-1)]]%*%PMatShort[[i]]
    # }

  # Steady<-lapply(1:length(PMatLong),function(x) (all(abs(PMatLong[[x]][1,]-colMeans(PMatLong[[x]]))<=delta)))
  # SteadyTime<-data$CumulT[min(which(Steady==TRUE))]

  if (length(initialprobs)==3){
    FinPred<-names(which.max(ProbRes[nrow(ProbRes),1:3]))
    FinFore<-names(which.max(ForeRes[nrow(ForeRes),1:3]))
  } else{
    FinPred<-names(which.max(ProbRes[nrow(ProbRes),1:2]))
    FinFore<-names(which.max(ForeRes[nrow(ForeRes),1:2]))
  }



  ##Cluster Matching
  ObsRes<-as.data.frame(recode(data$Result,'"1"="Win";"0"="Loss"'),stringsAsFactors=FALSE)
  FinRes<-unique(ObsRes) #get observed end of match result
  FinVec<-rep(FinRes,nrow(ObsRes))

  ResLogic<-FinVec==predResT$predResT #check for FinVec==ResT

  indLogic<-cbind(rle(ResLogic)$values,cumsum(rle(ResLogic)$length)-(rle(ResLogic)$lengths-1),cumsum(rle(ResLogic)$length))
  crit<-which(rle(ResLogic)$values==TRUE & rle(ResLogic)$lengths>=lengthThresh)
  # if(length(indLogic[crit,])==3){
    # indMatch<-matrix(0,nrow=1,ncol=3)
    # } else {
    # }
  indMatch<-as.data.frame(matrix(indLogic[crit,],ncol=3)) #returns a matrix of length(l) which contains number of intervals which are longer than lengthThresh as well as start and end points

  colnames(indMatch)<-c("Vec","Beg","End")


  list(Prediction=ProbRes,PredProbPlot=gg,LineBarPlot=hh,HeatPlot=ii,MarginPlot=jj,EndBarPlot=kk,ForecastHeat=ll,CumulAcc=CumulTAccuracy,CumulAccFore=CumulTAccuracyFore,PMatrices=Pcomp,ClusterMatch=indMatch,ObservedResult=FinRes,FinalPrediction=FinPred,FinalForecast=FinFore,HomeRank=unique(data$HomeRank),AwayRank=unique(data$AwayRank),ForeData=ForeRes)#,SteadyStateTime=SteadyTime)
}

##Underside Margin Plot
#ggplot(data=FullMarkovDataT0[1:2000,],aes(x=CumulT,y=MarginT,colour=MarginT)) + geom_line() + scale_color_gradient2(midpoint=0, low="red", mid="grey",high="green") + theme(panel.background = element_rect(fill="white", colour="black"),panel.grid.major = element_blank(), panel.grid.minor = element_blank(),legend.position="none") + labs(xlab("Time")) + labs(ylab("Margin"))

```

```
##REWRITTEN AND OPTIMISED INTO PredictMSM
# SteadyState<-function(Pred1=NULL,Pred2=NULL,w=100,delta=0.0001){
  # TempVar<-NULL
  # b=w/2
  # Pred1Temp<-Pred1[,-4]
  # Pred2Temp<-Pred2[,-4]
  # TimeTemp<-Pred1$Time

  # TempDiff<-Pred1Temp-Pred2Temp

  # TempVar<-rollapply(data=TempDiff,width=w,by=b,FUN=var,by.column=TRUE)
  # TempMean<-colMeans(TempVar)
  # TempSteady<-isTRUE(TempMean<delta)
  # }

##CREATE AND FORMAT DATA
FullMarkovDataT0<-read.csv("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\Data\\ChampionData\\FullMarkovDataFinal.csv",header=TRUE)

FullMarkovDataT0$Season<-as.factor(FullMarkovDataT0$Season)
FullMarkovDataT0$Season<-as.integer(as.character((FullMarkovDataT0$Season)))
FullMarkovDataT0$Round<-as.factor(FullMarkovDataT0$Round)
FullMarkovDataT0$Round<-as.integer(as.character((FullMarkovDataT0$Round)))
FullMarkovDataT0$Venue<-as.factor(FullMarkovDataT0$Venue)
FullMarkovDataT0$Finals<-as.factor(FullMarkovDataT0$Finals)
#FullMarkovDataT0$Finals<-as.integer(as.character((FullMarkovDataT0$Finals)))
FullMarkovDataT0$Result<-as.factor(FullMarkovDataT0$Result)
FullMarkovDataT0$HomeRank<-as.factor(FullMarkovDataT0$HomeRank)
FullMarkovDataT0$HomeRank<-as.integer(as.character((FullMarkovDataT0$HomeRank)))
FullMarkovDataT0$AwayRank<-as.factor(FullMarkovDataT0$AwayRank)
FullMarkovDataT0$AwayRank<-as.integer(as.character((FullMarkovDataT0$AwayRank)))
FullMarkovDataT0$QUARTER<-as.integer(as.character(FullMarkovDataT0$QUARTER))
FullMarkovDataT0$MatchNo<-MatchInd(FullMarkovDataT0) #Creates MatchNo indicator
FullMarkovDataT0$ResT<-RealTimeResult(FullMarkovDataT0)
FullMarkovDataT0$CumulT<-CumulTime(FullMarkovDataT0)
FullMarkovDataT0$Res1<-as.numeric(FullMarkovDataT0$Result)
FullMarkovDataT0$MarginT<-(FullMarkovDataT0$H.GOAL*6+FullMarkovDataT0$H.BEHI)-(FullMarkovDataT0$A.GOAL*6+FullMarkovDataT0$A.BEHI)


#Model claims " Different states observed at the same time on the same subject at observations
#Therefore we offset offending rows
##FIXED AS OF OffsetTime()

#FullMarkovDataT0<-FullMarkovDataT0[-c(1330,34346),]

#As per manual advice (considering time periods >1000) we also scale CumulT from seconds into minutes and fix using OffsetTime()

FullMarkovDataT0$CumulT<-FullMarkovDataT0$CumulT/60
FullMarkovDataT0$CumulT<-OffsetTime(FullMarkovDataT0)


##INITIALISE BASIC (WIN/LOSS/DRAW) MODEL y=time , time=quarter+seconds -> y=time+covariates , covariates=static+dynamic
#model can either take the form of ResT~QUARTER+TIME_SEC or Rest~CumulT --> we also need to check why the function returns different results at the same time point (even
```

```
though the data disagrees)
StateNamesB<-c("Draw","Loss","Win")
StateTableB<-matrix(statetable.msm(state=ResT,subject = MatchNo,data = FullMarkovDataT0), byrow = FALSE,nrow=3,ncol=3)

#StateNamesC<-c("Loss","Win") #1 == Loss 2 == Win
#StateTableC<-matrix(statetable.msm(state=Res1,subject = MatchNo,data = FullMarkovDataT0),byrow = FALSE,nrow=2,ncol=2)

#Initial transition probability matrix
Basictrans<-matrix(NA,nrow=3,ncol=3)
Basictrans<-matrix(StateTableB/rowSums(StateTableB),nrow=3,ncol=3)
colnames(Basictrans)<-StateNamesB
rownames(Basictrans)<-StateNamesB


#nonfeat<-match(c("Date","Result","Margin","Home.score","Away.score","Home.team","Away.team","MatchNo","TIME_SEC","STAT_HA","ResT","CumulT","QUARTER","Season","PEREN","PERST","Res1"),colnames(FullMarkovDataT0)) #remove season temporarily
covfn=as.formula(paste("~HomeRank + AwayRank + PastHome + PastAway + Head2Head + Round + Margin + A.BEHI + H.BEHI + A.KICK + H.KICK"))
#covfn=as.formula(paste("~HomeRank + AwayRank + PastHome + PastAway + Head2Head + Round + Margin + A.BEHI + A.FRAG + A.HBIN + A.HITO + H.BEHI + H.CLEAR + H.FRFO + H.HITO + H.TACK + A.TACK + A.CLEAR"))
#covfn=as.formula(paste("~Margin + A.BEHI + A.CLEAR + A.FRAG + A.FRFO + A.GOAL + A.HBEF + A.HBIN + A.HBRE + A.HITO + A.IN50 + A.KICK + A.KKIN + A.MARK + A.RE50 + A.SPOIL + A.TACK + H.BEHI + H.CLEAR + H.FRAG + H.FRFO + H.GOAL + H.HBEF + H.HBIN + H.HBRE + H.HITO + H.IN50 + H.KICK + H.KKIN + H.MARK + H.RE50 + H.SPOIL + H.TACK"))

#war<-warnings()
##unique(na.omit(as.numeric(unlist(strsplit(unlist(names(war)), "[^0-9]+")))))
#FullMarkovDataT0$CumulT[unique(na.omit(as.numeric(unlist(strsplit(unlist(names(war)), "[^0-9]+")))))]<-FullMarkovDataT0$CumulT[unique(na.omit(as.numeric(unlist(strsplit(unlist(names(war)), "[^0-9]+")))))]+rep(x=c(0,0.000001),times=length(unique(na.omit(as.numeric(unlist(strsplit(unlist(names(war)), "[^0-9]+"))))))/2)
#BasicMMod2<-msm(formula=ResT~QUARTER+TIME_SEC,subject=MatchNo,data=FullMarkovDataT0,qmatrix=Basictrans) #not run due to non-convergence (multiple duplicate states???)

#WORKING COVARIATES #HomeRank+AwayRank+H.IN50+A.IN50
#TRY ADDING ONLY DYNAMIC TO THE ABOVE LIST #HomeRank+AwayRank+H.IN50+A.IN50+A.KICK+H.KICK

#Markov model with covariates
TempTrainData<-subset(FullMarkovDataT0,Season==2015)
#TempTestData<-subset(FullMarkovDataT0,MatchNo==23)
# TempTestData$time<-TempTestData$CumulT
# TempTestData$subject<-TempTestData$MatchNo
#WIN/LOSS/DRAW AT TIME T
MModT<-msm(formula=ResT~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans,covariates=covfn,control = list (trace = 2, REPORT = 1,fnscale = 2189252,maxit = 10000,reltol = 1e-08))
#Pred<-PredictMSM(model=MModT,covariates=c("HomeRank","AwayRank","H.IN50","A.IN50"),data=TempTestData,initialprobs=c(0.1,0.3,0.6)) #removed as included in mass test


#WE ALSO NEED MEASURES OF FIT TO COMPARE AND CONTRAST TO THEORY THAT IT IS VERY DIFFICULT TO JUGE FIT FROM
#A STATISTIC DUE TO MODEL STRUCTURE AND THE LIKES, WE CAN HOWEVER PRODUCE PREVALENCE PLOTS AND CHECK
```

```r
#EPOCH ACCURACY AND FINAL RESULT ACCURACY (THIS SHOULD BE ENOUGH)

#WE NEED TO CHANGE THE TESTING FROM LEAVE ONE OUT TO TRAIN ON 2015 AND TEST ON EACH MATCH OF 2017
#HOW WE IMPLEMENT THIS IS A PROBLEM BUT SHOULD BE SOLVED BY THE TIME THE WRITEUP IS DONE


#MASS TESTING (will need to be cleaned for larger application)
TestNames<-NULL
for (i in 24:45){
  assign(x=paste("TestMatch",i,sep=""),subset(FullMarkovDataT0,MatchNo==i)) #Create individual test sets
  TestNames<-c(TestNames,paste("TestMatch",i,sep=""))
}

## Cleaning pt1 (works do far)
# TestList<-lapply(unique(FullMarkovDataT0$MatchNo),function(x) subset(FullMarkovDataT0, MatchNo==x))
# names(TestList)<-paste("TestMatch",unique(FullMarkovDataT0$MatchNo),sep="")

#list(Prediction=ProbRes,PredProbPlot=gg,LineBarPlot=hh,HeatPlot=ii ,MarginPlot=jj, CumulAcc=CumulTAccuracy)

#we need to add in extra information to the above list
#1.cluster matching (result at time t vs actual)
#2.home and away rank
#3.other descriptives for analysis

StaticRes<-NULL
for (j in TestNames){ #Static initial probabilities
  ObjName<-paste("Pred",j,sep="")
  assign(ObjName,PredictMSM(model=MModT,covariates=c("HomeRank","AwayRank","PastHome","PastAway","Head2Head","Round","Margin","A.BEHI","H.BEHI","A.KICK","H.KICK"),data=get(j),initialprobs=c(0.1,0.3,0.6),lengthThresh=50))
  setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\2018\\Final Markov Models\\Deterministic Probabilities")
  tiff(paste(ObjName,"Plot_1.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  print(get("PredProbPlot",eval(as.symbol(ObjName))))
  graphics.off()
  tiff(paste(ObjName,"Plot_2.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  lay_out(list(get("PredProbPlot",eval(as.symbol(ObjName))),1:3,1:3),list(get("MarginPlot",eval(as.symbol(ObjName))),4,1:3))
  graphics.off()
  tiff(paste(ObjName,"Plot_3.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  lay_out(list(get("LineBarPlot",eval(as.symbol(ObjName))),1:3,1:3),list(get("HeatPlot",eval(as.symbol(ObjName))),4,1:3))
  graphics.off()
  tiff(paste(ObjName,"Plot_4.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  lay_out(list(get("EndBarPlot",eval(as.symbol(ObjName))),1:3,1:3),list(get("ForecastHeat",eval(as.symbol(ObjName))),4,1:3))
  graphics.off()
  tempstat<-cbind(tail(get(ObjName)$PredProbPlot$data,1)[-4][c(3,2,1)],matrix(tail(get(ObjName)$ForeData,1)[-4][c(3,2,1)],nrow=1),get(ObjName)$CumulAcc,get(ObjName)$CumulAccFore,ObjName,length(get(ObjName)$ClusterMatch[,"Vec"]),max(get(ObjName)$
```

```
ClusterMatch[,"End"]-get(ObjName)$ClusterMatch[,"Beg"]),max(get(ObjName)$ClusterMatch[,"End"]),get(ObjName)$ObservedResult,get(ObjName)$FinalPrediction,get(ObjName)$FinalForecast,get(ObjName)$HomeRank,get(ObjName)$AwayRank)
  colnames(tempstat)<-c("WinProb","LossProb","DrawProb","ForeWinProb","ForeLossProb","ForeDrawProb","CumulativeAccuracy","CumulativeAccuracyFore","ObjName","MatchingIntervals","MaxIntSize","FinalMatchingEpoch","ActualResult","PredictedResult","ForecastResult","HomeRank","AwayRank")
  StaticRes<-rbind(StaticRes,tempstat)
}

#write.excel(tail(DynPredTestMatch23$Plot$data,1)[-4][c(3,2,1)],col.names = FALSE)

DynProb<-read.excel()
#DynProb<-as.matrix(DynProb)[,c(3,2,1)]


DynamicRes<-NULL
dynint<-0
for (j in TestNames){ #Dynamic initial probabilities
  dynint<-dynint+1
  ObjName<-paste("DynPred",j,sep="")
  assign(ObjName,PredictMSM(model=MModT,covariates=c("HomeRank","AwayRank","PastHome","PastAway","Head2Head","Round","Margin","A.BEHI","H.BEHI","A.KICK","H.KICK"),data=get(j),initialprobs=as.matrix(DynProb[dynint,]),lengthThresh=50))
  setwd("C:\\Users\\Casey Josman\\Dropbox\\PhD. Research\\2018\\Final Markov Models\\Static Probabilities")
  tiff(paste(ObjName,"Plot_1.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  print(get("PredProbPlot",eval(as.symbol(ObjName))))
  graphics.off()
  tiff(paste(ObjName,"Plot_2.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  lay_out(list(get("PredProbPlot",eval(as.symbol(ObjName))),1:3,1:3),list(get("MarginPlot",eval(as.symbol(ObjName))),4,1:3))
  graphics.off()
  tiff(paste(ObjName,"Plot_3.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  lay_out(list(get("LineBarPlot",eval(as.symbol(ObjName))),1:3,1:3),list(get("HeatPlot",eval(as.symbol(ObjName))),4,1:3))
  graphics.off()
  tiff(paste(ObjName,"Plot_4.tiff",sep=""), height = 12, width = 17, units = 'cm', compression = "lzw", res = 300)
  lay_out(list(get("EndBarPlot",eval(as.symbol(ObjName))),1:3,1:3),list(get("ForecastHeat",eval(as.symbol(ObjName))),4,1:3))
  graphics.off()
  tempdyn<-cbind(tail(get(ObjName)$PredProbPlot$data,1)[-4][c(3,2,1)],matrix(tail(get(ObjName)$ForeData,1)[-4][c(3,2,1)],nrow=1),get(ObjName)$CumulAcc,get(ObjName)$CumulAccFore,ObjName,length(get(ObjName)$ClusterMatch[,"Vec"]),max(get(ObjName)$ClusterMatch[,"End"]-get(ObjName)$ClusterMatch[,"Beg"]),max(get(ObjName)$ClusterMatch[,"End"]),get(ObjName)$ObservedResult,get(ObjName)$FinalPrediction,get(ObjName)$FinalForecast,get(ObjName)$HomeRank,get(ObjName)$AwayRank)
  colnames(tempdyn)<-c("WinProb","LossProb","DrawProb","ForeWinProb","ForeLossProb","ForeDrawProb","CumulativeAccuracy","CumulativeAccuracyFore","ObjName","MatchingIntervals","MaxIntSize","FinalMatchingEpoch","ActualResult","PredictedResult","ForecastResult","HomeRank","AwayRank")
  DynamicRes<-rbind(DynamicRes,tempdyn)
}
```

```
#write.excel(tail(DynPredTestMatch23$Plot$data,1)[-4][c(3,2,1)],col.names = FALSE)

## Cleaning pt2 ()
#TestPred<-lapply()

MModStat<-msm(formula=ResT~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans,covariates=~HomeRank+AwayRank+PastHome+PastAway+Head2Head+Round,control = list (trace = 2, REPORT = 1,fnscale = 1094626,maxit = 10000,reltol = 1e-08))
#WIN/LOSS AT END OF MATCH
MModStat2<-msm(formula=Res1~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans2,covariates=~HomeRank+AwayRank+PastHome+PastAway+Head2Head+Round,control = list (trace = 2, REPORT = 1,fnscale = 1094626,maxit = 10000,reltol = 1e-08))

MModDyn<-msm(formula=ResT~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans,covariates=~A.BEHI+A.FRAG+A.HBIN+A.HITO+H.BEHI+H.CLEAR+H.FRFO+H.HITO+H.TACK+A.TACK,control = list (trace = 2, REPORT = 1,fnscale = 1094626,maxit = 10000,reltol = 1e-08))
#WIN/LOSS AT END OF MATCH
MModDyn2<-msm(formula=Res1~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans2,covariates=~A.BEHI+A.FRAG+A.HBIN+A.HITO+H.BEHI+H.CLEAR+H.FRFO+H.HITO+H.TACK+A.TACK,control = list (trace = 2, REPORT = 1,fnscale = 1094626,maxit = 10000,reltol = 1e-08))

MModComb<-msm(formula=ResT~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans,covariates=~HomeRank+AwayRank+PastHome+PastAway+Head2Head+Round+A.BEHI+A.FRAG+A.HBIN+A.HITO+H.BEHI+H.CLEAR+H.FRFO+H.HITO+H.TACK+A.TACK,control = list (trace = 2, REPORT = 1,fnscale = 1094626,maxit = 10000,reltol = 1e-08))
#WIN/LOSS AT END OF MATCH
MModComb2<-msm(formula=Res1~CumulT,subject=MatchNo,data=TempTrainData,qmatrix=Basictrans2,covariates=~HomeRank+AwayRank+PastHome+PastAway+Head2Head+Round+A.BEHI+A.FRAG+A.HBIN+A.HITO+H.BEHI+H.CLEAR+H.FRFO,control = list (trace = 2, REPORT = 1,fnscale = 1094626,maxit = 10000,reltol = 1e-08)) #without A.TACK, H.TACK, H.HITO
```