Essays in Basketball Analytics

Suraj Keshri

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2019

ABSTRACT

Essays in Basketball Analytics

Suraj Keshri

With the increasing popularity and competition in professional basketball in the past decade, data driven decision has emerged as a big competitive edge. The advent of high frequency player tracking data from SportVU has enabled a rigorous analysis of player abilities and interactions that was not possible before. The tracking data records two-dimensional x-y coordinates of 10 players on the court as well as the x-y-z coordinates of the ball at a resolution of 25 frames per second, yielding over 1 billion space-time observations over the course of a full season. This dissertation offers a collection of spatio-temporal models and player evaluation metrics that provide insight into the player interactions and their performance, hence allowing the teams to make better decisions.

Conventional approaches to simulate matches have ignored that in basketball the dynamics of ball movement is very sensitive to the lineups on the court and unique identities of players on both offense and defense sides. In chapter 2, we propose the simulation infrastructure that can bridge the gap between player identity and team level network. We model the progression of a basketball match using a probabilistic graphical model. We model every touch event in a game as a sequence of transitions between discrete states. We treat the progression of a match as a graph, where each node represents the network structure of players on the court, their actions, events, etc., and edges denote possible moves in the game flow. Our results show that either changes in the team lineup or changes in the opponent team lineup significantly affects the dynamics of a match progression. Evaluation on the match data for the 2013-16 NBA season suggests that the graphical model approach is appropriate for modeling a basketball match.

NBA teams value players who can "stretch" the floor, i.e. create space on the court by drawing their defender(s) closer to themselves. Clearly, this ability to attract defenders varies across players, and furthermore, this effect may also vary by the court location of the offensive player, and whether or not the player is the ball handler. For instance, a ball-handler near the basket attracts a defender more when compared to a non ball-handler at the 3 point line. This has a significant effect on the defensive assignment. This is particularly important because defensive assignment has become the cornerstone of all tracking data based player evaluation models. In chapter 3, we propose a new model to learn player and court location specific offensive attraction. We show that offensive players indeed have varying ability to attract the defender in different parts of the court. Using this metric, teams can evaluate players to construct a roster or lineup which maximizes spacing. We also improve upon the existing defensive matchup inference algorithm for SportVU data.

While the ultimate goal of the offense is to shoot the ball, the strategy lies in creating good shot opportunities. Offensive play event detection has been a topic of research interest. Current research in this area have used a supervised learning approach to detect and classify such events. We took an unsupervised learning approach to detect these events. This has two inherent benefits: first, there is no need for pretagged data to learn identifying these events which is a lobor intensive and error prone task; second, an unsupervised approach allows us to detect events that has not been tagged yet i.e. novel events. We use a HMM based approach to detect these events at any point in the time during a possession by specifying the functional form of the prior distribution on the player movement data. We test our framework on detecting ball screen, post up, and drive. However, it can be easily extended to events like isolation or a new event that has certain distinct defensive matchup or player movement feature compared to a non event. This is the topic for chapter 4.

Accurate estimation of the offensive and the defensive abilities of players in the NBA plays a crucial role in player selection and ranking. A typical approach to estimate players' defensive and offensive abilities is to learn the defensive assignment for each shot and then use a random effects model to estimate the offensive and defensive abilities for each player.

The scalar estimate from the random effects model can then be used to rank player. In this approach, a shot has a binary outcome, either it is made or it is a miss. This approach is not able to take advantage of the "quality" of the shot trajectory. In chapter 5, we propose a new method for ranking players that infers the quality of a shot trajectory using a deep recurrent neural network, and then uses this quality measure in a random effects model to rank players taking defensive matchup into account. We show that the quality information significantly improves the player ranking. We also show that including the quality of shots increases the separation between the learned random effect coefficients, and thus, allows for a better differentiation of player abilities. Further, we show that we are able to infer changes in the player's ability on a game-by-game basis when using a trajectory based model. A shot based model does not have enough information to detect changes in player's ability on a game-by-game basis.

A good defensive player prevents its opponent from making a shot, attempting a good shot, making an easy pass, or scoring events, eventually leading to wasted shot clock time. The salient feature here is that a good defender prevents events. Consequently, event driven metrics, such as box scores, cannot measure defensive abilities. Conventional wisdom in basketball is that "pesky" defenders continuously maintain a close distance to the ball handler. A closely guarded offensive player is less likely to take or make a shot, less likely to pass, and more likely to lose the ball. In chapter 6, we introduce Defensive Efficiency Rating (DER), a new statistic that measures the defensive effectiveness of a player. DER is the effective distance a defender maintains with the ball handler during an interaction where we control for the identity and wingspan of the the defender, the shot efficiency of the ball handler, and the zone on the court. DER allows us to quantify the quality of defensive interaction without being limited by the occurrence of discrete and infrequent events like shots and rebounds. We show that the ranking from this statistic naturally picks out defenders known to perform well in particular zones.

Table of Contents

List of Figures

List of Tables

Acknowledgments

This dissertation summarizes five years of studies and research, a journey that started with conversations around the NBA and the invention of player tracking data with my good friend and ex-roommate Min-hwan Oh in the summer of 2014. Our penchant for number crunching, the popularity of Fantasy Sports, and the desire to make an impact in a relatively new field of analytics in the NBA set us on the track of pursuing research in basketball analytics. I am fortunate to have worked with many talented and generous people in the course of this journey.

First of all, I express my deepest gratitude to Professor Garud Iyengar for guiding me through this journey. His insightful advice has helped me reach an understanding of the machine learning literature, and to build a good intuition of this subject to be able to innovate in a relatively unexplored field of basketball analytics. His continuous encouragement motivated me to explore new ideas and to constantly ask new questions. Sometimes seeking answers to these questions was not easy, yet Professor Iyengar always allowed me to work on my own ideas and to develop as an independent researcher, for which I will always be grateful. In times of need, Professor Iyengar genuinely and patiently supported me, and I thank him for being available at these times regardless of his busy schedule. His scientific curiosity, intellectual breadth and truthfulness have set for me high standards of intellectual inquiry, standards to which I will continue to aspire.

During the earlier stages of my PhD and after my masters degree, I had a great opportunity to work with Professor Liam Paninski in the field of computational neuroscience. At that time, I was relatively new to the field of data science. Assisting him and his postdoctoral students with implementing various machine learning models helped me develop a

solid background in this field and enabled me to pursue my own research later. I am very grateful for the opportunity to work with Ari Pakman, Daniel Soudry, and Uygar Sümbül. My weekly meetings and discussions with them over the course of several years helped me develop a deep interest in machine learning models.

I had a rare opportunity to work with Doug Fearing, the ex-director of analytics at Los Angeles Dodgers, during my summer internship. His deep understanding of the field of baseball analytics and pragmatic approach to solve complex problems using simpler models has deeply benefitted me in my approach towards problems in basketball analytics. I am very grateful for being a part of his extremely talented analytics team at Dodgers during my internship.

I would have never embarked on my research on basketball without the help of my friend and colleague Min-hwan Oh. Watching NBA games and late night discussions around the sport has helped me develop a good understanding of the game to be able to continue my research, and they are some of my fondest memories during my PhD. Chapter 2, 3, and 4 of this dissertation is a result of our collaboration during the first few years of my PhD. I am also very lucky to work with Sheng Zhang, my good friend and collaborator on the topics discussed in Chapter 3 and 4. His help has been tremendously valuable.

Most of the work in this thesis would not have been possible without the help of David Bencs, the director of basketball analytics at Orlando Magic. His practical and deep insights of the game and the analytical problems that a major league basketball team faces has helped me shape my research in a way that emphasizes the practicality of the research outcomes and not just the intellectual fulfillment. I am also very grateful to him for helping us get the SportVU data.

It is said that "a journey of a thousand miles begins with a single step", and my journey began under the guidance of Professor Nomesh Bolia, my undergraduate advisor at IIT Delhi. Although he did not directly participate in this thesis, it would not have been

written had he not guided my first steps into the realm of research. I am deeply indebted to Professor Bolia for igniting my interest in research, and for encouraging me to pursue graduate studies.

This work also benefited from the feedback of Professors Adam Elmachtoub and Mark Broadie. I kindly thank them for participating in my thesis defense proposal presentation and being part of the defense committee, and for helping me to improve both the results developed in this work and setting direction for further research after my dissertation proposal presentation. I also thank Professor Hardeep Johar and Doctor Daniel Guetta for participating in the final thesis defense. I have benefitted greatly from my discussions and friendly conversations with Daniel when I was taking a class on optimization for which he was the TA (one of the best TAs I have ever had).

During my studies at the Columbia University, I had a privilege to interact with its exceptional faculty members - remarkable characters and brilliant thinkers whose lectures deepened my knowledge of optimization, machine learning, statistics, and who set outstanding examples of personal and professional qualities. I thank Professors Martin Haugh, Garud Iyengar, Liam Paninski, Donald Goldfarb, Ward Whitt, Michael Johannes, and Richard Davis for their insightful and engaging courses. I am also sincerely grateful to Garud Iyengar, Martin Haugh, Hardeep Johar, and Donald Goldfarb for giving me the opportunity of being a TA for the courses they have taught, these experiences have greatly helped in my academic and career development. I would also like to thank Professor Martin Haugh for his friendly and professional advice, his feedback on my earlier research, and giving me the opportunity to work with him before joining the PhD program.

I kindly thank Carmen Ng, Jaya Mohanty, Jenny Mak, Kristen Maynor, Lizbeth Morales, Shi Yee Lee, and others in the department and school administration. Your resourcefulness and friendly help with administrative matters during all these years had allowed me to fully focus on my research.

Thanks to my colleagues and friends in the department. I share many cherished memories of Columbia University and New York City with them.

A special thanks goes to my family members - my mother Kiran Keshri for taking care of me, my father Om Prakash Keshri for helping me learn how to learn, my uncle Sanjay Keshri for supporting me in every choice I made, my late grandmother Kaushilya Devi and my late grandfather Anandi Prasad Keshri for their continuing love and encouragement, my brothers and sisters for letting me being a part of their fun, and to everybody else in my family and extended family - thanks for your support.

To My Grandma

Chapter 1

Introduction

New technology and statistics will change the way we understand basketball, even if they also create friction between coaches and front-office personnel trying to integrate new concepts into on-court play. The most important innovation in the NBA in recent years is a camera-tracking system, known as SportVU, that records every movement on the floor and spits it back at its front-office keepers as a byzantine series of geometric coordinates. Fifteen NBA teams have purchased the cameras, which cost about $100,000 per year, from STATS LLC; turning those X-Y coordinates into useful data is the main challenge those teams face.

– [Lowe, 2013]

1.1 Basketball – The Game

Basketball is a team sport in which two teams, of five players each, opposing one another on a rectangular court, compete with the primary objective of shooting a basketball through the defender's hoop (a basket 18 inches in diameter mounted 10 feet high to a backboard at each end of the court) while preventing the opposing team from shooting through their own hoop [1]. Each team takes turn in shooting the ball. A successful attempt ends the possession while an unsuccessful one gives each team a chance to try again (referred to as rebound).

[1] https://en.wikipedia.org/wiki/Basketball

The National Basketball Association (NBA) is a men's professional basketball league in North America; composed of 30 teams divided into eastern and western conference. The NBA was founded in New York City on June 6, 1946. In the NBA, a game is 48 minutes long divided into four quarters each 12 minutes long. The size of the court in the NBA is 94 by 50 feet. Figure 1.1 is a detailed representation of the court in the NBA. An NBA season is divided into regular and post season. In the regular season, each team plays 84 games. 16 teams make it to the playoffs which is an elimitation round which concludes with a final between the best team from the eastern and the western conference.

A field goal in basketball is worth two points, unless made from behind the three-point line, when it is worth three. After a foul, timed play stops and the player fouled or designated to shoot a technical foul is given one or more one-point free throws. The team with the most points at the end of the four quarters wins, but if the score is tied, additional periods of five minutes play (overtime) are mandated until the winner is decided. Each NBA team can have a maximum of 15 players, 13 of which can be active in each game. The lineup of a team is usually divided into starting and bench lineup. The bench lineup usually play against each other. There is no rule about the playing time limit of each player but the starting lineup players, especially the star players, play for most of the game in a tight game situation. Each possession is limited to 24 seconds, counted as the shot clock time. If the offensive team (the team carrying the ball) makes a successful shot, the possession ends. Otherwise (the case of a rebound), the shot clock resets. The defensive team tries to prevent the opponent from making a shot or steal the ball from them.

1.1.1 Player Positions

Players in the NBA position themselves in five court locations in the halfcourt as shown in Figure 1.2. These locations are divided into three broad categories.

Center. Centers are generally the tallest players in the team who position themselves near the basket. On the offensive side, the center's goal is to make high-percentage shots close to

[1]Source: https://4dagl3gexum32g49f6zq3cdy-wpengine.netdna-ssl.com/wp-content/images/Diagrams-of-Basketball-Courts.png

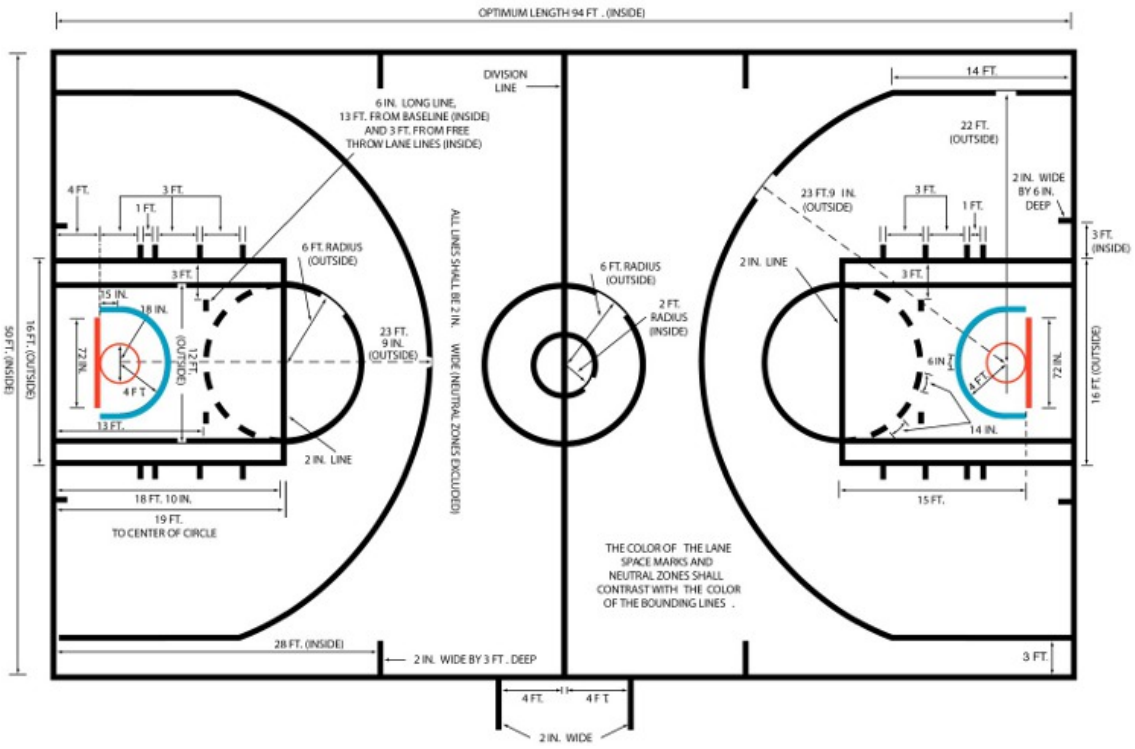


Figure 1.1: A detailed view of the basketball court in the NBA

the basket. They are also required to block defenders to open other players up for driving to the basket. On the defense side, the centre is responsible for keeping the opponent from shooting by blocking shots and passes in the vital area. They are also expected to fight for offesnive and defensive rebounds.

Forward. Forwards are usually the next tallest players in the team positioned inside the three point line. Forwards are responsible to get free for a pass, take outside shots, drive for goals, and rebound. Power forwards have to be able to hit open shots, since they typically aren't the focal point of a defense. They are usually good midrange shooters. The Small forward is usually the shorter of the two forwards on the team but plays the most versatile role both offensively and defensively out of the main five positions.

Guard. These are potentially your shortest players and they should be really good at dribbling fast, seeing the court, and passing. It is their job to bring the ball down the court and set up offensive plays. Dribbling, passing, and setting up offensive plays are a guard's main responsibilities as an offesnive player. They also need to be able to drive to the basket

and to shoot from the perimeter (long-range shots). On defense, a guard is responsible for stealing passes, contesting shots and preventing drives to the hoop.

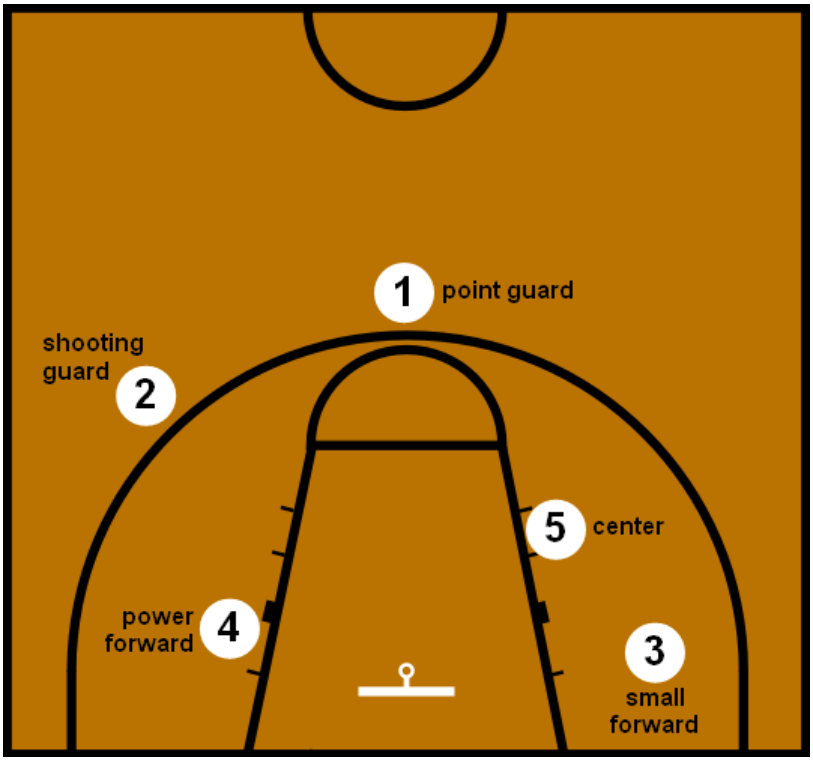


Figure 1.2: Traditional player positions in a basketball game

1.2 Traditional Player Evaluation in the NBA

Box score has been the traditional source of metrics to evaluate players in the NBA. There are the empirical count based statistics for each game. Some of the relevant offesive metrics are:

- Points: Number of points scored by a player. Number of points scored by a player per minute is a good approximation of his point contribution.

[1]Source: https://www.myactivesg.com/Sports/Basketball/How-To-Play/Basketball-Rules/Basketball-Positions-and-Roles

- Field Goal Percentage (FG%): This is the ratio of shots (excluding free throws) made and number of shot attempted by a player. This is a indicator of shot efficiency of a player. This can be further divided into three point percentage (3P%)and two point shot percentage.

- Offensive Rebounds(OREB): Count for the number times the ball was grabbed by a player after a missed shot from a teammate or himself.

- Plus Minus(+/-): The difference between the team's total score versus their opponent's when the player is in the game. This metric captures the overall contribution of the player taking both defense and offense into account.

Defensive metrics:

- Defensive Rebounds (DREB): Count of the number of times the ball was grabbed by a player after a missed shot from an opponent player.

- Steals: Number of times a player stole the ball from the ball handler. (leads to a turnover for the ball handler)

- Blocks: Number of times a player blocked a shot, which resulted in a failed shot attempt.

The biggest shortcomings of box score statistics are not taking all the players on the court into account. Most of the box score statistics are individual players numbers that do not take into accout the identity of other players that affected the event outcome. For instance, FG% do not consider who the defender was or if the shot was an open shot. On the defense side, number of the box score statistics are quite limited. Blocks, steals, and rebounds, along with minutes and what little information offensive numbers yield about defensive performance are all that is available.

After the invention of SportVU data (discussed in the next section), NBA has added some advanced player statistics. On the offensive side, we have the following statistics available on NBA.com:

- Offensive Rating (ORTG): It is a statistic used to measure a player's efficiency at producing points. It is essentially a weighted combination of other box score statistics without a rigorous statistical basis for the weights.
- Offensive Rebound Percentage (OREB%): A metric that normalizes the number of offensive rebounds with the team's offensive rebounds and opponents defensive rebounds. This is a more realistic rebound contribution for a player.
- Effective Field Goal Percentage: This statistic adjusts for the fact that a 3-point field goal is worth one more point than a 2-point field goal.
- True Shooting Percentage: This statistic is an aggregate measure of a players shooting ability taking into account the field goal percentage, free throw percentage, and three-point field goal percentage.

All of these advanced statistics for offense do no take into account individual abilities of either the teammates or the defensive players on the court. Neither do they adjust for the distance between the shooter and the defensive player when the shot was attempted. On the defensive end, we have DREB and DRTG which are the defensive analog of OREB and ORTG respectively. Defensive Win Share (DWS) is another advanced defensive statistics that represents the number of wins attributed to individual players. This is also a very coarse metric which only assigns a positive contribution to a defender when a game is won, which is more of a team effort.

These mainstream player evaluation statistics have largery suffered from the foremen-tioned limitations because of lack of data. Basketball is a dynamic and fast game that involves multiple players interacting is a complex spatiotemporal environment. We need data that captures these interactions to improve these existing statistics. SportVU data has come to the rescue.

1.3 SportVU Data

SportVU is an automated ID and tracking technology that has the ability to collect positioning data of the ball, players and referees during a game. The NBA has partnered with a data provider, STATS LLC, to install the SportVU cameras in the arenas of all 30 teams in the league in 2013. The tracking systems record two-dimensional x-y coordinates of 10 players on the court as well as the x-y-z coordinates of the ball at a resolution of 25 frames per second, yielding over 1 billion space-time observations over the course of a full season. A raw tracking data point has the game clock, shot clock, and locations of the players and the ball. The data used for our research include raw tracking data from year 2012 to 2106. We also have access to the play-by-play data. A play is a macro event that occurs during a possession. We have 23 different plays that include different types of passes, made and missed shots, rebounds, fouls, steals etc. We also have access to the details of possession ending events and the players involved. For instance, in case of a successful shot, we know the shooter and the player who assisted. Finally, we have the on-court data which gives us the identifiers of players on the court at any point in time. Since the advent of the player tracking system, the player tracking data allowed researchers to perform richer analytics than before on player and team evaluation and discover new findings about the aspects of the game itself.

Note that this data is not publicly available. We were able to acquire this data from an NBA team. However, the data used in Chapter 2 was scraped from `NBA.com` in 2014. The exact data may not be available on the website anymore.

1.4 Literature Review

Player ranking and evaluation plays a crucial role in the player acquisition and line up construction in the NBA teams. During the draft season, teams use the player biometric data and the box score data available from the college level basketball to make their player acquisition decisions. Basically there are no other alternative data sets to evaluate these for the major league games. However, when it comes to acquiring free agents or trade major

league players, the teams have the interaction data of the players in the major league environment. Their decision is much more informative because they have observed the player interacting with other major league players. The samples of player's performance are from the actual environment rather than a proxy (college basketball).

Modeling the players' interactions as a dynamic network or as a Markov chain has been of interest in the recent past literature. However, all of these works have suffered from a lack of player interaction information available in the play-by-play or the box score data. [Shirley, 2007] used a possession-based Markov model to model the progression of a basketball match. The model's transition matrix was estimated directly from NBA play-by-play data and indirectly from the teams' summary statistics. Play-by-play data has the information about the start and the end of the possession. For instance, which player got an inbound pass, who finally took the shot and from where, and the shot outcome. The information about the dynamics of ball movement and player interaction during the possesion is missing from the data set. [Fewell *et al.*, 2012] analyzed the movement of ball during a possession as a strategic network, defining players as nodes and ball movements as links. They shed a light on the importance of ball distribution across the team players, especially with the shot specialists and the point guard in the leadership role, on the game outcome. In particular, whether the teams consistently moved the ball towards their shooting specialists, and whether they distributed the ball in a way that reduced predictability emerged as two major offensive strategies used by the NBA teams. However, they do not take the identity of the players into account.

A possession in a basketball game consists of many macro events, which are different strategic ways in which the players pass or shoot the ball. SportVU data gives a very micro view of events during a possession. It gives the coordinates of all the players on the court but it does not give any information about the alignment or the defensive assignment of the players. Defensive assignment plays a big role in estimating defensive or offensive abilities of players. It allows us to build models that accounts for all the actors (offensive and defensive players) involved in any event we want to analyze. Traditionally, the defensive assignment

data has been available for shot attempts only. Also, the assignment has been based on the physical distance between the players when the shot occurs. The defensive assignment is usually annotated by a human observer. While it is often clear to a human observer who is guarding whom, such information is absent from the data. Annotating the data set is a subjective and labor-intensive task. A straightforward approach for defensive matchup is to make assignment based on physical proximity. However, this approach does not account for the realistic scenario of defensive assignment where the matchup is *sticky* i.e. the assignments do not change frequently. [Franks *et al.*, 2015] is the first paper that accounts for this fact. They uses the tracking data to infer the defensive assignment of players as the hidden state of a Hidden Markov Model (HMM) at each time during a possession. The core, and simplifying, assumption of the model is that a defensive player can only be assigned to one offensive player at a time, which makes defensive matchup a bipartite graph matching problem. The movement of the defensive players are described as a two dimensional gaussian distribution on the court with the mean location as a weighted mean of the location of the assigned offesive player, the hoop, and the ball location. Defensive switches are modeled as an independent multinomial distribution for each defender. We improve upon this model as discussed in the next section.

Offense in a possession is mainly driven by a combination of various play events. Isolation, drive, post up, ball screen etc. are some of the plays that teams use to create shot opportunities. The play-by-play data has been tagged with such events by human annotators. Traditionally, human annotators such as assistant coaches or scouts recognize these plays in real-time by reading players' movements and the signaling gestures on and off the court. Annotating these plays provides useful scouting information, and also helps a team evaluate its own strategy. However, offensive strategies are complex and dynamic, and can be executed with multiple variations as the defense reacts and it becomes a labor-internsive task to tag these events. The task of labeling these events using data mining has been of recent research interest. [McQueen *et al.*, 2014] propose using support vector machines to detect a ball screens using human-labeled data. Their method requires the identity of the on-ball defender, the ball handler, and the screener. They set the defender nearest to the

ball handler as the on-ball defender. We discovered that the nearest-defender assignment is only 76% accurate); thus, we expect that the method is unlikely to be very accurate. Moreover, their work is limited to learning ball screens and requires human tagged data to train the model. [Miller and Bornn, 2017] takes a topic modeling approach to categories movements of players into repeated interpretable structures that would allow efficient search and exploration of player tracking data. The repeatable structures constitue the vocabulory of a basketball possession and the topic could be a play type like corner-three shot. The goal of the paper is to summarize possessions as a collection of topics. However, there is no discussion of detecting events like ball screen and post up. [Wang and Zemel, 2016] classifies different kinds of offensive play calls using neural networks. The input to the network is the player coordinates and the output is an indicator of the event occurrance. The neural network is able to detect patterns in the player's movement that are predictive of the event outcome, similar to an image detection problem. However, their method still depends on the labeled data and is only applicable to detecting team level play calls.

High shot efficiency is the cornerstone of being a great offensive player. Box score and advanced NBA statistics has many statistics to measure the shooting ability of players (Points scored, FG%, ORTG, True Shooting Percentage, to name a few). However, these statistics do not take identity of defensive players into account. [Fearnhead and Taylor, 2011] tries to estimate the offensive and defensive contribution of players taking into account the identity of all the players in the lineup. [Chang *et al.*, 2014] quantifies the quality of shot based on the court location and the average shooting percentage. Then they rank the players based on the quality of shot. However the defense is not taken into account. The lack of defensive matchup data before SportVU has restricted a significant contribution in accurate shooting ability estimation. [Miller *et al.*, 2014], [Franks *et al.*, 2015] are some of the early works after the invention of SportVU data which accounts of defensive matchup to estimate shooting abilities of player. [Cervone *et al.*, 2016] models the point contribution of a player (Expected Point Value) as a time series over the course of a possession (much like a stock market) taking into account other players on the court. Shot quality and defense are two important factors that determine the shooting ability of players. However,

the past literature has not taken both these factors into account. We defines the quality of a shot based on its trajectory and use it to learn the shooting abilities of the players taking defensive matchup into account.

Although defense plays just as important role as offense when it comes to winning a game, conventional metrics like box scores have been designed to summarize offensive play. Apart from defensive rebounds, blocks, and steals, defensive metrics are hard to come by in pre-SportVU literature. Defensive Rating (DRTG) and Defensive Win Share (DWS) are two advanced statistics used by the NBA to capture the defensive contribution of a player. Both these metrics are based on discrete events, and are heavily infuenced by the teammates. [Fearnhead and Taylor, 2011] estimates the defensive and offensive ability of players using a random effects model on the net point differential given the lineups. However, the defensive assignment information has been ignored, probably because of lack of such data. [Franks *et al.*, 2015] uses SportVU data to characterize the defensive abilities of players in preventing shots taking the defensive matchup and player identities into account. All of the past work on defensive rating has been based on events such as shot attempt, shot outcome, point difference etc. Defense is a continuous processes that prevents occurrance of events. For instance, a tight defense may not even allow a ball handler to attempt a shot because of unfavorable shot outcome probability. We will disucss our contribution in this topic in the next section.

1.5 Contribution

With the increasing popularity and competition in professional basketball in the past decade ([Abdul-Jabbar, 2017]) data driven decision has emerged as a big competitive edge. Front offices of NBA teams increasingly rely on quantitative models to construct lineups, acquire players, and win games. The emergence of fantasy sports as a multi-billion dollar market has further motivated the need for prediction tools in the NBA games. The advent of high frequency tracking data from SportVU has enabled a rigorous analysis of player abilities and interactions that was not possible before. There is a need for models that take the

player identity and lineups into account to evaluate individual player abilities. SportVU data has also enabled automating tasks like play event detection which has traditionally been tagged as human annotators. To better exploit the information in the tracking data, we need data driven models that captures the nuances of a basketball game using the traditional machine learning and statistical analysis tools. This need has motivated five research goals addressed in this thesis.

1. Develop a graphical model to simulate the progression of a possession taking into account the identity of the players on the court. This allows use to simulate the effect of a player substitution or even a hypothetical lineup on the outcome, hence help the teams decide a better lineup given the opponent lineup.

2. We improve upon an existing framework to detect defensive assignment as the hidden state of a HMM as the possession progresses. We also quantify the notion of gravity in basketball.

3. We propose an unsupervised learning framework that does not require any manually tagged data to detect play events like post up, screen, and drive that .

4. We propose a new shot efficiency ranking of player based on the quality of shot trajectory rather than the shot outcome.

5. We introduce Defensive Efficiency Rating (DER), a new statistic that measures the defensive effectiveness of a player at a court location accounting for the offensive player they defend.

Conventional approaches to simulate matches have ignored that in basketball the dynamics of ball movement is very sensitive to the lineups on the court and unique identities of players on both offense and defense sides. In the second chapter, we propose the simulation infrastructure that can bridge the gap between player identity and team level network. We model the progression of a basketball match using a probabilistic graphical model. We model every touch event in a game as a sequence of transitions between discrete states.

We treat the progression of a match as a graph, where each node represents the network structure of players on the court, their actions, events, etc., and edges denote possible moves in the game flow. Our results show that either changes in the team lineup or changes in the opponent team lineup significantly affects the dynamics of a match progression. Evaluation on the match data for the 2013-16 NBA season suggests that the graphical model approach is appropriate for modeling a basketball match.

NBA teams value players who can "stretch" the floor, i.e. create space on the court by drawing their defender(s) closer to themselves. Clearly, this ability to attract defenders varies across players, and furthermore, this effect may also vary by the court location of the offensive player, and whether or not the player is the ball handler. For instance, a ball-handler near the basket attracts a defender more when compared to a non ball-handler at the 3 point line. This has a significant effect on the defensive assignment. This is particularly important because defensive assignment has become the cornerstone of all tracking data based player evaluation models. In chapter 3, we propose a new model to learn player and court location specific offensive attraction. We show that offensive players indeed have varying ability to attract the defender in different parts of the court. Using this metric, teams can evaluate players to construct a roster or lineup which maximizes spacing. We also improve upon the existing defensive matchup inference algorithm for SportVU data.

While the ultimate goal of the offense is to shoot the ball, the strategy lies in creating good shot opportunities. Offensive play event detection has been a topic of research interest. Current research in this area have used a supervised learning approach to detect and classify such events. We took an unsupervised learning approach to detect these events. This has two inherent benefits: first, there is no need for pretagged data to learn identifying these events which is a lobor intensive and error prone task; second, an unsupervised approach allows us to detect events that has not been tagged yet i.e. novel events. We use a HMM based approach to detect these events at any point in the time during a possession by specifying the functional form of the prior distribution on the player movement data. We test our framework on detecting ball screen, post up, and drive. However, it can be easily

extended to events like isolation or a new event that has certain distinct defensive matchup or player movement feature compared to a non event. This is the topic for chapter 4.

Accurate estimation of the offensive and the defensive abilities of players in the NBA plays a crucial role in player selection and ranking. A typical approach to estimate players' defensive and offensive abilities is to learn the defensive assignment for each shot and then use a random effects model to estimate the offensive and defensive abilities for each player. The scalar estimate from the random effects model can then be used to rank player. In this approach, a shot has a binary outcome, either it is made or it is a miss. This approach is not able to take advantage of the "quality" of the shot trajectory. In chapter 5, we propose a new method for ranking players that infers the quality of a shot trajectory using a deep recurrent neural network, and then uses this quality measure in a random effects model to rank players taking defensive matchup into account. To penalize the complexity of the neural network model, we use random dropouts [Srivastava *et al.*, 2014]. We show that the quality information significantly improves the player ranking. We also show that including the quality of shots increases the separation between the learned random effect coefficients, and thus, allows for a better differentiation of player abilities. Further, we show that we are able to infer changes in the player's ability on a game-by-game basis when using a trajectory based model. A shot based model does not have enough information to detect changes in player's ability on a game-by-game basis.

A good defensive player prevents its opponent from making a shot, attempting a good shot, making an easy pass, or scoring events, eventually leading to wasted shot clock time. The salient feature here is that a good defender prevents events. Consequently, event driven metrics, such as box scores, cannot measure defensive abilities. Conventional wisdom in basketball is that "pesky" defenders continuously maintain a close distance to the ball handler. A closely guarded offensive player is less likely to take or make a shot, less likely to pass, and more likely to lose the ball. In chapter 6, we introduce Defensive Efficiency Rating (DER), a new statistic that measures the defensive effectiveness of a player. DER is the effective distance a defender maintains with the ball handler during an interaction

where we control for the identity and wingspan of the the defender, the shot efficiency of the ball handler, and the zone on the court. DER allows us to quantify the quality of defensive interaction without being limited by the occurrence of discrete and infrequent events like shots and rebounds. We show that the ranking from this statistic naturally picks out defenders known to perform well in particular zones.

1.6 Software Usage

We relied on open source software for inference in our models. For random effects model, we used the lme4 package in R ([Bates *et al.*,]); for neural network based models, we used the PyTorch package in Python (`pytorch.org`). glm2 ([Marschner, 2011]) package was used to fit generalized linear models in R and for fitting the Non-negative Matrix Factorization (NMF) model, we used the scikit-learn package in Python.

Chapter 2

Graphical Model for Basketball Match Simulation

This chapter is based on the paper "Graphical model for basketball match simulation" [Oh *et al.*, 2015] which is a joint work with Min-hwan Oh and Professor Garud Iyengar.

2.1 Introduction

Predicting the outcomes of professional sports events is one of the most popular practices in the sports media, fan communities and, of course, sport betting related industries. Predictions range from human prediction to statistical analysis of historical data. In recent years with the advent of player tracking data, basketball, specifically the NBA, has received much attention as a domain of analytics. Many new metrics have been introduced to evaluate players and teams. However, there are no studies that fully take advantage of the rich player tracking data to simulate the outcomes of basketball matches. Most of the previous simulation approaches in basketball have focused on win-loss predictions, ignoring the progression of matches. In order to obtain detailed "microsimulation" of a basketball game, [Shirley, 2007] and [Štrumbelj and Vračar, 2012] used a possession-based Markov model to model the progression of a basketball match. However, these studies treat each team as merely a single entity rather than collective union of individual players. These previous studies ignore that in basketball the dynamics of ball movement is very sensitive

to the lineups on the court and unique identities of players on both offense and defense sides.

One can ask, "Is Miami Heat the same team without LeBron James? Or, can Oklahoma City Thunder be an elite team without Kevin Durant and Russell Westbrook?" Taking individual players into account in a simulation process is not just about addressing the issues with trades or changes in the roster in the preseason but also changes in lineups of teams during a season, which appear almost on a day-to-day basis, either in a starting lineup or bench lineup — whether it is due to injuries or strategic reasons. These changes do have an impact on final game results.

[Fewell *et al.*, 2012] used network analysis in which they analyzed ball movement of teams, mapping game progression pass by pass. They assessed differences in team's offensive strategy by their network properties. While their objective was not to simulate matches, they still did not address the unique identities of players, which is prevalent especially in basketball. Questions such as "With Tim Duncan and Tony Parker out tonight, will the Spurs win against the Rockets?" still remained unanswered.

In this paper, we propose the simulation infrastructure that can bridge the gap between player identity and team level network. We model the progression of a basketball match using a probabilistic graphical model. The model shows the ball movement of every play and subsequent game events based on player level pass interaction, shot frequency given teammates and defenders, shot accuracy against the defense, rebound etc. We follow the natural and intuitive flow of a basketball match as shown in Figure 2.1.

2.2 Data

Our model is calibrated using the player tracking data and play-by-play game log data from the matches for the 2013-2016 season[1]. Both these data sets were available on `NBA.com`.

[1]We ignore 2012-2013 season because we have many games missing for that season from our dataset, which makes the data insufficient for a season-by-season analysis.

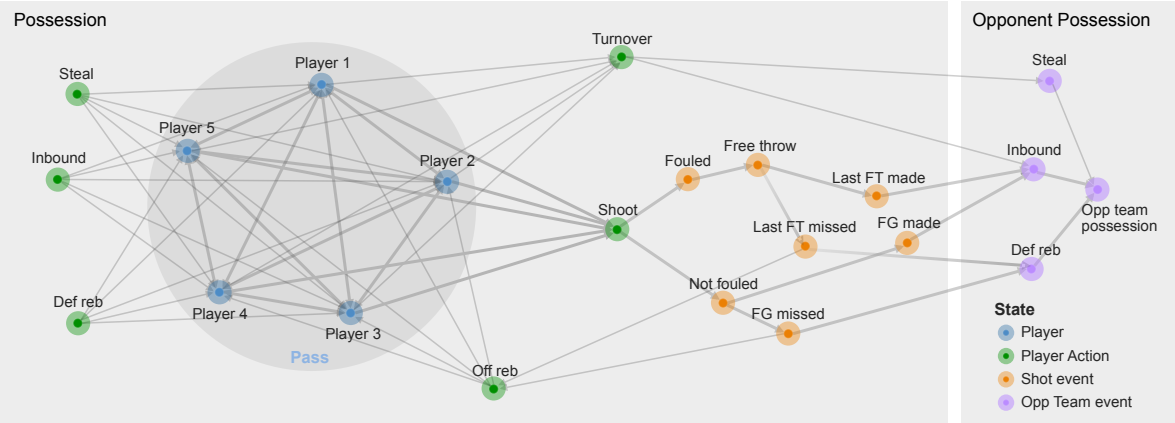


Figure 2.1: Graphical Model for sequence of events in each possession

We also used the lineup data available on `basketball-reference.com`. See Table 2.1 for the specific data fields that we have used for the analysis done in this chapter. Note that we did not have access to the the raw tracking data for the analysis done in this chapter. Many of our modeling choices would have been different if we were using the raw tracking data.

Table 2.1: Data fields used for analysis

xy coordinate of the shot location
Player action and shot events (see Figure 2.1)
Closest defensive assignment for each shot
The lineup for each possession
Number of touches by each player in a possession
Number of passes between each pair of players in a game
Total number of possessions in each game

2.3 Method

We do our analysis on a season-by-season basis. This allows us to compare the result with the actual box score and game outcomes for each season separately. Further, because of players changing teams, team dynamics changes significantly with the change in season.

Training the models separately for each season keeps them in sync with the team dynamics which does not change much over a season. We treat the progression of a match as a graph (see Figure 2.1), where each node represent be the possession of the ball by a player (blue nodes), their actions (green and purple nodes), shot events (orange nodes) etc., and edges denote possible moves in the game flow. We model every touch and event in a game as a sequence of transitions between nodes. We learn the conditional probability of the edges from the data. We simulate ball movements between players, how likely a player is to take a shot, and how defense and teammates affect the dynamics.

Table 2.2: Summary of Notation

Notation	**Meaning**	**Section**
L_o	Offensive team lineup	2.2, 2.4, 2.6
L_d	Defensive team lineup	2.2, 2.6
γ_i	Player i's propensity to take a shot	2.2
$\widetilde{\gamma}_i$	Player i's propensity to take a shot given the defensive lineup	2.2
β_i	Player i's ability to deter shot attempt	2.2
α_{ij}	Tendency of player i to pass to player j	2.4
θ_{id}	Shooting ability of player i at basis d	2.3
ϕ_{id}	Defensive ability of player i to reduce shot accuracy at basis d	2.3
ψ_{id}	Player i's ability to draw a shooting foul at basis d	2.5
ζ_{id}	Player j's foul proneness at basis d	2.5
ρ_i^d	Defensive rebound grabbing ability of player i	2.6
ρ_i^o	Offensive rebound grabbing ability of player i	2.6
τ_a	Average possession time of team a	2.7

2.3.1 Start of Possession

We model the start of a possession by a team as a multinomial distribution between players on the court. If the possession starts with an inbound pass, we sample the starting player

according to distribution of historical backcourt touch data. On the other hand, if the possession starts with a defensive rebound or a steal, then it is trivial since the player who starts the possession has been already decided. Methods to compute rebound probabilities and to sample a steal event are discussed in later sections.

2.3.2 Shot Frequency

We model the probability of a field goal attempt for a given touch as a Bernoulli distribution with probability

$$p\left(S_i = 1 \mid L_o, L_d, \gamma, \beta\right) = \sigma\left(\widetilde{\gamma}_i + \left(\widetilde{\gamma}_i - \frac{1}{4}\sum_{k \in L_o, k \neq i} \widetilde{\gamma}_k\right)\right)$$

$$\widetilde{\gamma}_i = \gamma_i - \sum_{j \in L_d} w_{ij}\beta_j,$$

$$\gamma_i \sim N(0, \sigma_\gamma^2),$$

$$\beta_j \sim N(0, \sigma_\beta^2)$$

with $\sigma(x) = \exp(x)/(1 + \exp(x))$. S_i is an indicator for whether player i attempts a shot given a touch. L_o and L_d represent the lineups of the offensive team and defensive team respectively. γ_i is a parameter which determines how likely a player is to take a shot and β_j is the defensive ability of player j to reduce shot frequency. The weight w_{ij} determines how much player i is affected by the defense of player j which is proportional to the time that player i is guarded by player j[2]. We also have an offset term $\left(\widetilde{\gamma}_i - \frac{1}{4}\sum_{k \in L_o, k \neq i} \widetilde{\gamma}_k\right)$ which is negative (or positive) if the propensity of player j to take a shot is less (or more) than average teammates' propensity. The reasoning behind this model is that an event of a player taking a shot depends not only on his propensity to shoot and his defender but also on the propensity of his teammates. We can take Kevin Durant and Russell Westbrook of the Oklahoma City Thunder as an example: when Kevin Durant is not on the court, Russell Westbrook tends to shoot more.

[2]For simulation purposes, we set the weight w_{ij} propotional to the frequency of an offensive player who players in position a being guarded by a defensive player in position b. We use the closest defensive assignment data when a shot is attempted to learn this weight

This model simplifies into a random effects model with constant factors for each of the independent variables as shown below (note that w is precomputed).

$$p\left(S_i = 1 \mid L_o, L_d, \gamma, \beta\right) = \sigma\left(2\gamma_i - \frac{1}{4}\sum_{k \in L_o, k \neq i} \gamma_k - \sum_{j \in L_d}\left(2w_{ij} - \frac{1}{4}\sum_{k \in L_o, k \neq i} w_{kj}\right)\beta_j\right)$$
$$\gamma_i \sim N(0, \sigma_\gamma^2),$$
$$\beta_j \sim N(0, \sigma_\beta^2)$$

For the dependent variable S_i, we use the number of shot attempts and touches by a player in each possession. Using mean field approximation, we randomly assign 0 or 1 to each touch in a possession by player i such that their sum is equal to the number of shot attempts in that possession. Finally, we fit the model using lme4 package ([Bates *et al.*,]) in R.

2.3.3 Shot Efficiency

To model shot efficiency of a player, our approach is similar to [Franks *et al.*, 2015]. Given that a player attempts a field goal, we model shot efficiency (the probability that the player makes a shot) as a function of the offensive player's skill, the defender at the time of the shot, and the location of the shot on the court.

$$p\left(Y_i = 1 \mid d, \theta, \phi\right) = \sigma\left(\theta_{id} - \phi_{jd}\right)$$
$$\theta_{id} \sim N(0, \sigma_{\theta d}^2), \tag{2.1}$$
$$\phi_{jd} \sim N(0, \sigma_{\phi d}^2)$$

Here, Y_i is an indicator for whether player i made the shot, d represents the basis from which the shot was taken, θ_{id} is the shooting ability of player i at basis d, and ϕ_{jd} is the defensive ability of the closest defensive player j to reduce shot accuracy at basis d. Note that our model is slightly different from [Franks *et al.*, 2015]. In particular, we do not take the distance between the offensive and defensive player into account. The justification is that the ability of the defender is also characterized by how closely he defends the player. Thus, the parameter ϕ_{jd} takes into account how closely is player j able to defend basis d. This assumption is important in our simulation model because we are not modeling the distance between the defender and the shooter while the shooter attempts a shot. We apriori

assign the weighted average of defense depending on the shot basis and the position of the defenders on the court. Thus, while simulating the game, once a player decides to take a shot, we sample the shot location using the basis loadings. Then, the success probability of the shot is given by the shot efficiency model.

2.3.4 Pass Network

We model the passes between players as a network with edge weight parameterized by α_{ij} $(i \neq j)$. The probability that player i passes to player j if player i chooses to pass is given by

$$p\left(i \to j \mid \alpha, L_o\right) = \frac{\alpha_{ij}}{\sum_{k \neq i, k \in L_o} \alpha_{ik}}$$

where we only take into account α's for players on the court. Note that the probability that player i passes to player j depends not only on players i and j but also other teammates on the court. To learn the α matrix, we use EM algorithm ([Bishop, 2006]) on the data of total number of passes between each player and the total number of possession each lineup had in every game. Figure 2.2 shows an example of the α matrix we learned for the San Antonio Spurs, with α_{ij} as the i, j entry of the matrix. This matrix can be used to get the exact pass probabilities among players given the lineup.

To fit the model, we use the data of total number of passes between every pair of players in a game and the total number of passes initiated by each player in a possession. Note that since we do not know the number of passes among the players in each possession, we use EM algorithm to infer that. We treat the number of passes between every pair of players in each possession as a hidden variable. In the E step, given the α parameters, total number of passes between a pair of players in the full game, and the number of passes initiated by a player in a possesion (number of touches - number of shots - number of rebounds - number of turnovers), we sample the number of passes between each pair of players in each possession. In the M step, given the sample of number of passes betweeen each pair of players, we recompute the α parameters.

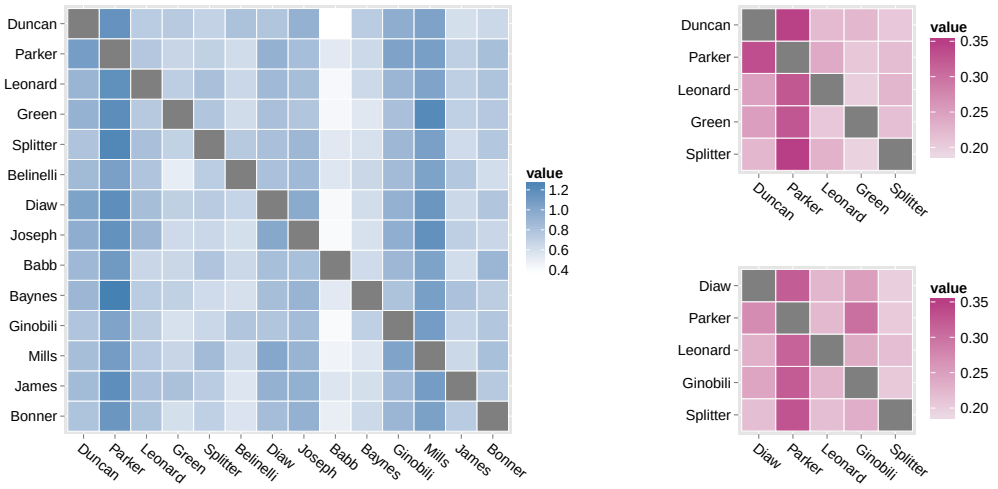


Figure 2.2: Rows are passers and columns are receivers. Note that the diagonal entries are set to zero. The α matrix for the San Antonio Spurs 2013-14 roster (left) shows that Tony Parker and Patty Mills are more likely to receive passes from most of the other players. This was expected due to their position and role as primary ball handler. We create a pass probability matrix for different lineups (right) by extracting a corresponding entry of the α matrix and normalized by each row. We observe that replacing two players in the lineup results in a different pass probability matrix. This allows us to obtain the passing distribution of any arbitrary lineup in a team.

2.3.5 Shooting Foul & Free Throw

We model shooting fouls as a function of the shooter's ability to draw a foul, defender's foul proneness, and the location (basis) of the shot on the court. The approach is similar to the model for shot efficiency.

$$p\left(SF(i,j)=1 \mid d,\psi,\zeta\right)=\sigma\left(\psi_{id}+\zeta_{jd}\right)$$
$$\psi_{id}\sim N(0,\sigma^2_{\psi d}),$$
$$\zeta_{jd}\sim N(0,\sigma^2_{\zeta d})$$

$SF(i,j)$ is an indicator for whether the player i was fouled by player j while shooting. ψ_{id} is player i's ability to draw a shooting foul at basis d, and ζ_{jd} represents the defender's foul proneness at basis d. Therefore, if a defender is more foul prone, then there is a higher chance of shooting foul. As for the free throws, we use free throw percentage for each player to sample a free throw success event. This is a reasonable approach since a free throw does

not depend on the opponents or teammates.

To fit this model, we use exactly the same approach as the Shot Efficiency model (see 2.3.3). We know the closest defensive assignment for every shot from the possession data. During simulation, we use the defensive assignment based on the historical average (same as the shot efficiency model).

2.3.6 Rebound

We model rebound as a competition between the players on court. Since there is a clear difference in effort required to grab a defensive and an offensive rebound, we assume that each player i has a defensive and an offensive rebound ability represented by ρ_i^d and ρ_i^o respectively. Given the current lineup of offensive and defensive team on the court, the probability of player i grabbing an defensive or an offensive rebound is given by

$$p\left(DR_i = 1 \mid L_d, L_o\right) = \frac{\exp(\rho_i^d)}{\sum_{j \in L_d} \exp(\rho_j^d) + \sum_{k \in L_o} \exp(\rho_k^o)}$$
$$p\left(OR_i = 1 \mid L_d, L_o\right) = \frac{\exp(\rho_i^o)}{\sum_{j \in L_d} \exp(\rho_j^d) + \sum_{k \in L_o} \exp(\rho_k^o)}$$

DR_i and OR_i are indicators for player i grabbing a defensive rebound and an offensive rebound respectively. In this model, the rebound grabbing ability of a team depends on the players of both the teams on court. This model allows us to estimate rebound grabbing probability for arbitrary lineups.

2.3.7 Number of Possessions

In our model, we assume that a possession starts with an inbound pass, a defensive rebound, or a steal. To model number of possessions, we assume that team i on an average takes time τ_i to end a possession. This assumption aligns with the traditional notion of Pace Factor which is a part of Hollinger Team Statistics in the NBA. The Pace Factor of a team is defined as the number of possessions a team uses per game. τ can be interpreted as a factor inversely proportional to the pace factor. Thus, total number of possessions for each

team in a game between team a and team b should be close to $\frac{T_i}{\tau_a+\tau_b}$, where T_i is duration of the game i. We have the data for total number of possessions in a game, η. To learn τ, we minimize the sum of square of error of each game:

$$\min_{\tau} \sum_i \left(\sum_k I_{ki}\tau_k - \frac{T_i}{\eta_i} \right)^2$$

where T_i is duration of game i, η_i is the number of possessions in game i, and $I_{ki} = 1$ if team k plays in game i. [3]

2.3.8 Turnover

In our model, we assume that there are two types of turnover. One type is stolen balls and the other type includes all the other turnovers that results in an inbound pass (offensive foul, out-of-bounds, etc.). We calculate average probability of turnover per touch for each player from the historical data and use that independent of current lineup. We also sample a stolen ball event from turnover event. Given a stolen ball, we assign a steal to a defensive player with probability proportional to his average steal rate compared to average steal rate of his teammates on court. Given a non-stolen ball turnover, we start from an inbound pass.

2.3.9 Simulation

For simulation purposes, lineup is an input parameter to our model. One can try different lineups against an opponent team, modifying the number of possessions given to particular lineups. For fitting and testing our model, we use the actual lineups used in each game. After we learn all the required parameters mentioned above, we compute conditional probability for each edge in the possession graph displayed in Figure 2.1. We draw a sample

[3] One can make a case about using sampling random number of possessions for each game. Although it might be a more practical thing to do, we believe that it will not make a difference in the outcome of the game averaged over large sample. Also, using fixed number of possessions for a given pair of teams allows us to get away with overtime play complexities. The dynamics of games change pretty significantly during overtime plays. The lineups used by teams during overtime are very season and game situation dependent. Handling these edge cases during simulation would not be practical.

Table 2.3: R^2 for True vs. Predicted win percentages

Season	Within sample	Out of sample
2013-14	0.92	0.87
2014-15	0.93	0.86
2015-16	0.91	0.86

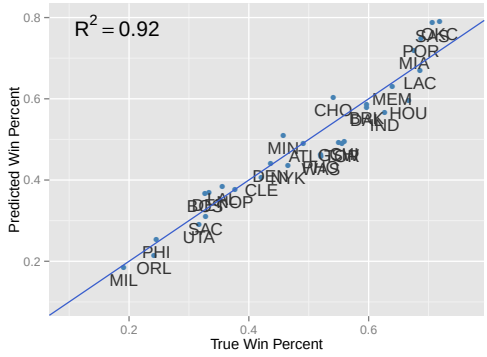
of events using the graphical model for each possession. We repeat this sampling process until we reach the estimated number of possessions. This gives us one sample of single match statistics. We simulate a match multiple times to estimate expected statistics for both players and teams. We then assign a win to a team with more number of wins. We also get the probability that a team will win.

2.4 Result

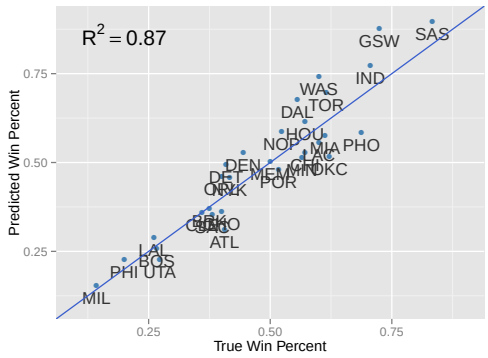
We used our model to simulate the season record for each of the 30 NBA teams for three seasons between 2013-2016. We used 70% of 1230 matches in the regular season as the training set and the remaining 30% as the test set. Figure 2.3 shows that the model provides a good estimate of the teams' actual win percentages with the within-sample R-squared 0.92, and the out-of-sample R-squared 0.87 for season 2013-14. Table 2.3 shows the result for all three seasons.

Our model's performance in predicting average winning percentage is comparable to [Shirley, 2007] and [Štrumbelj and Vračar, 2012]. The predicted per-game season average personal statistics such as points per game (PPG) shows correlation with the actual data (see Table 2.4 and Figure 2.4). For player level statistics, we have higher variance and bias in our prediction, especially for players who score fewer points.

We used our model on the 2014 NBA Finals matchup between the San Antonio Spurs and the Miami Heat. We computed the conditional probabilities for the pass network and player actions of the Spurs' starting lineup against the defense of the Heat's starting lineup, and the graph for the matchup is shown in Figure 2.5a. While fixing the defense, player substitutions — Boris Diaw for Tim Duncan and Manu Ginobili for Danny Green — result in a graph



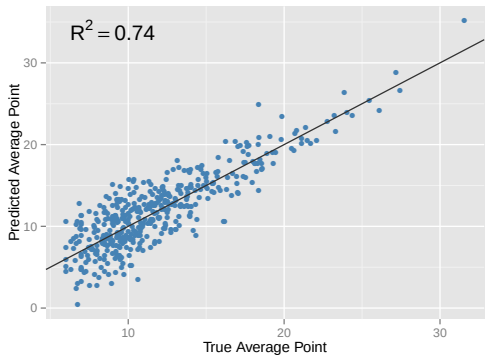
(a) Within-sample result



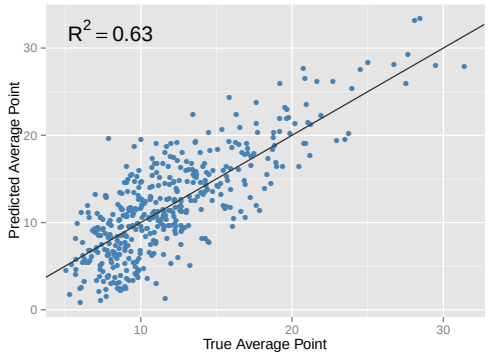
(b) Out-of-sample results

Figure 2.3: True vs. predicted win percentages for the 2013-14 season

with different conditional probabilities of all events (Figure 2.5b). We observe that Ginobili attracts the ball from other players more than Green does. Also with Duncan out, we expect more ball movement between backcourt players, and team shooting also shifts significantly towards guards and small forward. Note that Ginobili and Diaw have the same positions as Green and Duncan respectively. However, the graphs are quite different for the two lineups within the same team. This suggests that defining the possession network of a team only in terms of player positions is not sufficient. For another comparison, we also computed the network of the Spurs' starting lineup against the Portland Trail Blazers' starting lineup, i.e. fixing the team lineup but changing the opponent lineup or team (Figure 2.5c). We observe a clear drop in Duncan's shot attempt probability, compared to the base case against the Heat in Figure 2.5a. This is due to the effect of his defender, Lamarcus Aldridge, who has a higher defensive ability to reduce shot frequency. Subsequently, we observe an increase in



(a) Within-sample result



(b) Out-of-sample result

Figure 2.4: True vs. predicted average point for players for the 2013-14 season

expected shot frequency for Parker. These comparisons suggest that either changes in the team lineup or in the opponent team lineup significantly affects the dynamics of a match progression.

Having established that the ball dynamics is significantly influenced by different sets of players on the court, our model can be used to evaluate the effect of different lineups on game results. For demonstration, we simulated the 2014 NBA Finals with two distinct sets of lineups of the Spurs while keeping the lineups of the Heat fixed (we used the Heat's lineups used in Game 5 of the Finals for both case 1 and case 2). We assigned different weights to each lineup as shown in Figure 2.6 in order to allocate different number of possessions given to lineups in a given game. 101 simulations were performed to determine the probability of win for each team as described in Section 2.9. We applied conventional best-of-seven playoff format to determine the winner of the Finals. The results show that changes in lineups and

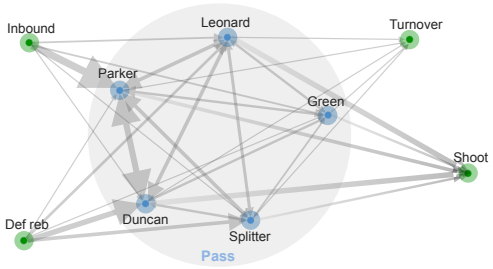
Table 2.4: R^2 for True vs. Predicted PPG for players

Season	Within sample	Out of sample
2013-14	0.74	0.63
2014-15	0.78	0.65
2015-16	0.75	0.65

weights affect the outcome of the series.

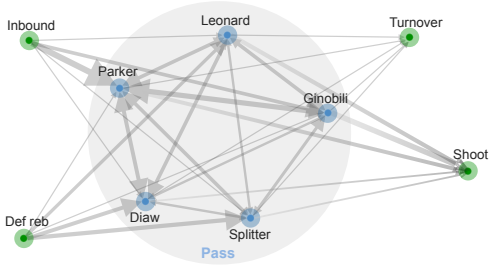
2.5 Conclusion

The simulation model we propose helps answer not only the team level questions, e.g. which team will win a match, or which teams will advance to the playoffs, but also player level questions, such as how well a specific player will perform in a given match or the entire season. The model offers the infrastructure for match simulation that will allow the front office or the coaching staff of the team to evaluate the performance of hypothetical lineups against specific opponents. It also gives insight on minute allocation between players. One of the limitations of our current simulation model is not being able to estimate pass network for hypothetical lineup of players from different teams since we do not have the pass data for players from different teams (also because there are distinct pass structures for different teams as argued by [Fewell *et al.*, 2012]). Also, we do not take assist and block into account in the simulation. Our future endeavor would be to modify our model to overcome these limitations. This will help the teams evaluate the value of future acquisitions in the context of the existing roster. Moreover, the simulation model could also be used to predict performance of Fantasy Basketball teams.



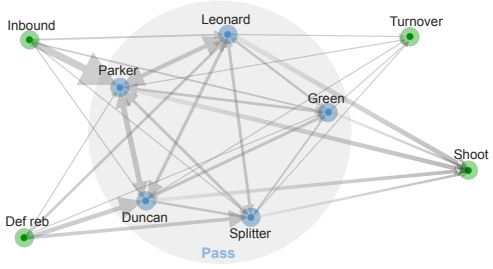
OFF	Duncan	Parker	Leonard	Green	Splitter
DEF	Lewis	Chalmers	James	Wade	Bosh

(a) The San Antonio Spurs starting lineup's offense against the Miami Heat's starting lineup



OFF	Diaw	Parker	Leonard	Ginobili	Splitter
DEF	Lewis	Chalmers	James	Wade	Bosh

(b) Change in the offensive lineup



OFF	Duncan	Parker	Leonard	Green	Splitter
DEF	Aldridge	Lillard	Batum	Matthews	Lopez

(c) Change in the opponent team

Figure 2.5: Graphs of offense with changes in the team lineup or in the opponent lineup/team. (Edge thickness is proportional to the probability of the event)

Case 1

Lineup					Weight
Parker	Green	Leonard	Duncan	Diaw	0.3
Parker	Ginobili	Leonard	Duncan	Splitter	0.15
Mills	Ginobili	Belinelli	Diaw	Splitter	0.1
		⋮			⋮

Results

SAN ANTONIO SPURS **4 : 2** MIAMI HEAT

Case 2

Lineup					Weight
Parker	Ginobili	Leonard	Duncan	Splitter	0.3
Parker	Green	Ginobili	Duncan	Diaw	0.15
Mills	Green	Belinelli	Diaw	Bonner	0.1
		⋮			⋮

Results

SAN ANTONIO SPURS **3 : 4** MIAMI HEAT

Figure 2.6: Simulation results on the 2014 NBA Finals

Chapter 3

Defensive Assignment

This chapter is based on the paper "Automatic event detection in basketball using HMM with energy based defensive assignment" [Keshri *et al.*,] which is a joint work with Minhwan Oh, Sheng Zhang, and Professor Garud Iyengar.

Good basketball always starts with good defense.

–Bobby Knight

3.1 INTRODUCTION

Statistics have always been important in basketball. Points scored, rebounds, assists, steals, etc. have long been recorded and reported on box scores. However, box score statistics provide very coarse information for understanding the ability and characteristics of a player, and strategies of teams. Furthermore, box score statistics mainly describe the offensive performance of players, and the defensive performance goes unrecognized. A straight forward method of choice to learn offensive and defensive abilities of a player for an event (various shot types, for example) has been to model the outcome of the event as a function of the abilities of players involved in that event [Oh *et al.*, 2015; Franks *et al.*, 2015]. While the optical player tracking data provides the location of the players and the ball, it does not provide the defensive assignment of the players. Before we

can assess performance, we need to be able to learn from data which players were defending against which offensive player [1] and which offensive and defensive players were involved in particular events during the matchup. The defensive matchup is a crucial input for estimating player abilities or event detections (such as ball screen, pick and roll etc.). In this work we propose unsupervised learning methods for learning the time evolution of defensive assignment in a possession, which serves as a basis for further analytics on player performance and event detection. NBA teams today value players who can stretch the floor on offense. This attribute, also referred to as *gravity* in basketball (see [Pelton, 2014]), is represented by the ability of an offensive player to draw his defender(s) closer to himself. ("It's impossible to understand the way NBA offenses and defenses operate without understanding gravity" [Pelton, 2014]). Clearly, this attraction effect is not the same for all the players — some players attract their defenders more than others. Furthermore, the ball has attraction that pulls defenders toward itself because of the need to pressure the ball handler and keep him from getting a wide-open shot. This effect varies by the court location and the ball location. For instance, a ball-handler near the basket attracts a defender more compared to a non ball-handler at the 3 point line. The goal of this paper is to propose a model to learn the defensive assignment and attraction of players using the optical player tracking data.

[Franks *et al.*, 2015] introduces a hidden Markov model (HMM) based framework to learn the defensive assignment of players. The hidden states of the HMM refer to the defensive assignment of the five defensive players on the court. However, their method does not take player identity or court location into account. In addition, their defense assignment transition model assumes uniform distribution, i.e. a defender is equally likely to switch to guard any offensive players, and does not capture the interaction of players. This leads to incorrect defensive assignments, especially when multiple players are closely involved in an event. We build upon this framework by allowing spatial variation in the defender behavior among different ball handlers, as well as proposing a more flexible, "bond breaking" model for matchup switches.

[1] We are implicitly assuming one-on-one defense, which is the case in the NBA.

We take an HMM based approach to jointly model the defensive assignment and the attraction coefficients of the players. The hidden states of the HMM refer to the defensive assignment of the five defensive players on the court, similar to [Franks *et al.*, 2015]. The location of a defensive player is modeled as a Gaussian distribution over the court with the mean given by an affine combination of the location of the ball, the hoop, and the location of the offensive player that he is guarding. The parameters of the affine combination defines the attraction coefficients with respect to the ball, the hoop, and the offensive player respectively. A higher attraction coefficient of the offensive player means higher attraction effect i.e. the player pulls the defender closer. We learn a separate attraction coefficient of each player for each point on the court. This allows us to model the gravity of each offensive player. To improve our estimation of the attraction coefficients, we use a Gaussian process (GP) prior over the court with shared mean across the players [Rasmussen, 2004]. Sharing the mean parameter of the GP helps in pooling information across the players. The covariance structure of the GP helps in smoothing the coefficients, hence sharing information across the court locations. The transition in the hidden state refers to the change in defensive assignment. We model the transition distribution using a bond breaking/formation principle. A bond refers to a matchup of a defensive player to an offensive player. Breaking/formation of a bond refers to a change in the defensive assignment. Certain defensive assignments are more stable than others, hence are less likely to change. For example, double team lasts a for short period, and hence is an unstable bond configuration. These bonds are easier to break (or harder to form). This fixes the limitation of [Franks *et al.*, 2015] in which all defensive assignments are equally likely at any time.

Subsection 3.2.1 describes the specification of the location of a defender and the GP prior on the parameters. Subsection 3.2.2 describes the HMM model with our novel bond based transition probability matrix. In Section 3.3, we describe our inference algorithm for the attraction coefficients and the bond parameters. Section 3.4 demonstrates the results and compare the accuracy of our defensive assignment with other benchmarks. We further present the defensive attraction heat maps we learned for selected players as well as average attraction coefficients for selected team rosters. Finally, in Section 3.5, we describe the

usage of our model and next steps.

3.1.1 Data

We use SportVU data from 2015-2016 NBA regular season. We remove players who played in fewer than 700 possessions, leaving a total of around 372 players. The reason why we filter players under this threshold is to be able to get more stable estimates since we pool information across different players. We parse the data from the moment that the ball handler carries that ball to the offensive front court and stops when any possession ending event such as shot, turn over, foul, etc. happens, and we define it as a single possession. Note that we do not train our model on data for more than one season. There are two reasons for this choice. First, we are fitting our model on a very granular data set (close to a billion data points for each season) which is sufficient for a highly accurate estimate of the parameters, and hence, using data from multiple seasons is not likely to have any effect on the accuracy of the parameter estimates. Second, human tagiing on video data in order to test the accuracy of the defensive matchup is a time intensive task. It was not feasible to tag multiple games from different seasons to test our model's performance. Furthermore, the performance of our model would be put to additional tests in the later chapters and we show that the defensive assignments from our model yield significantly superior results as compared to existing models.

3.2 METHODOLOGY

3.2.1 Basic Setting

Let Ω be the space of all possible basketball possessions. For $\omega \in \Omega$, we index each defensive player by $i \in \{1, ..., 5\}$ and each offensive player by $j \in \{1, ..., 5\}$. We define $T(\omega)$ to be the length of the possession ω. Then, for time $0 \leq t \leq T(\omega)$, we have the following locations

- $O_{tj}(\omega)$: location of the offensive player j at t
- $B_t(\omega)$: location of the ball at t,
- H: the location of the hoop

Note that without loss of generality, we transform the space so that all possessions occur in the same half. Hence, H is fixed for all t and ω. Let $Z_{tj} = [O_{tj}, B_t, H]^\top$. We assume that the canonical location for a defender guarding the offensive player j at time t is $Z_{tj}^\top \Gamma$, where $\Gamma = [\gamma_o, \gamma_b, \gamma_h]^\top$ with $\Gamma^\top \mathbf{1} = 1$ i.e. a convex combination of the position of the offender O_{tj}, the current location of the ball B_t, and the location of the hoop H. See Figure 3.1.

Let I_{tij} be an indicator for whether defender i is guarding offender j at time t. Multiple defenders can guard the same offensive player, but each defender can only be guarding one offensive player at any instant. [Franks *et al.*, 2015] assumes that the observed location of a defender i, given that they are guarding offender j, is normally distributed about the mean location

$$D_{ti} | I_{tij} = 1 \quad \sim \quad \mathcal{N}\left(Z_{tj}^\top \Gamma, \sigma_D^2\right)$$

Note that in this setting Γ does not depend on player identity nor location on the court. This is equivalent to arguing that attraction effect is the same for all players and across all court location. However, as stated in the introduction, the heterogeneity in the nature of offensive players regarding defensive attraction suggests that we need to take player identity into consideration. Moreover, for a given offensive player, how much the offensive player draws his defender depends on where the player is located, so Γ is location dependent.

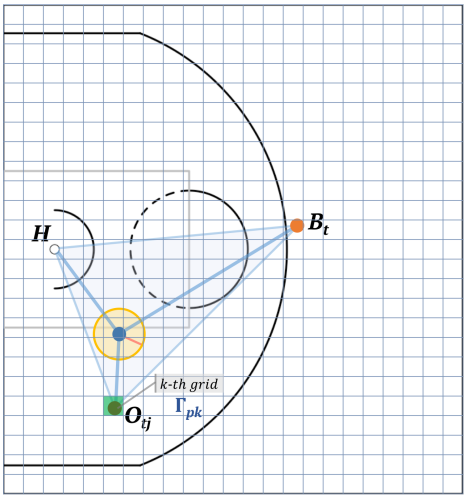


Figure 3.1: Canonical defensive location with player and location dependency on Γ

We index players globally with $p \in P$, where P denotes the set of all players in the league that satisfy the threshold for the number of possessions as stated in Subsection 3.1.1. We divide the offensive half of the court into $2ft \times 2ft$ bins i.e. a total of 575 bins. We index these bins by $k \in \{1, ..., K = 575\}$. We let Γ_{pk} denote the attraction coefficient of player p at location k. When we model defensive assignment and attraction, we pick Γ corresponding to the offensive player and his location. We do so by using a mapping g from time t and offensive player index j to global player index p and the grid index k.

$$g : t \times j \mapsto p \times k$$

The observed location of a defender i at time t, given that they are guarding offender j, is Normally distributed about the mean location $Z_{tj}^{\top}\Gamma_{g(t,j)}$

$$D_{ti}|I_{tij} = 1 \;\; \sim \;\; \mathcal{N}\left(Z_{tj}^{\top}\Gamma_{g(t,j)}, \sigma_D^2\right)$$

In order to allow us to pool information across players while learning Γ_{pk} for all $p \in P$ and $k \in K$, we employ a Gaussian process (GP) prior on the vector Γ_p, where $\Gamma_p = [\Gamma_{p1}, ..., \Gamma_{pK}]^{\top}$ is a stacked vector of Γ_{pk}. We use GP prior with mean μ_Γ and the covariance matrix $\mathcal{K}$ with the structure:

$$\begin{aligned}
\text{cov}(\gamma_{pk}^o, \gamma_{pl}^o) &= k_o(x_k, x_l) = \zeta_o^2 \exp\left(-\phi_o^2 \|x_k - x_j\|^2\right) \\
\text{cov}(\gamma_{pk}^b, \gamma_{pl}^b) &= k_b(x_k, x_l) = \zeta_b^2 \exp\left(-\phi_b^2 \|x_k - x_j\|^2\right) \\
\text{cov}(\gamma_{pk}^h, \gamma_{pl}^h) &= k_h(x_k, x_l) = \zeta_h^2 \exp\left(-\phi_h^2 \|x_k - x_j\|^2\right) \\
\text{cov}(\gamma_{pk}^o, \gamma_{pl}^b) &= cov(\gamma_{pk}^b, \gamma_{pl}^h) = cov(\gamma_{pk}^h, \gamma_{pl}^o) = 0
\end{aligned}$$

where x_k and x_l denote the center of bin k and bin l respectively. These kernels are chosen to induce the spatial smoothness in defensive attraction of players,. Note that we only allow the γ values of same type to be correlated across bins. Furthermore, since we use a GP prior, this yields a normal approximation to the posterior.

3.2.2 Hidden Markov Model

In each offensive team possession, defense starts with some initial assignments. These assignments progress over time as players and the ball move around on the court. We

model the progression of defense assignments (as given by the matrix of matchups, $\boldsymbol{I}$) over the course of a possession using a hidden Markov model. The hidden states $\boldsymbol{I_t}$ represent the mapping of defenders to offensive players at any given time t. The complete data likelihood for one possession is the following:

$$\begin{aligned} L(\Gamma, \sigma_D^2) &= P(\boldsymbol{D}, \boldsymbol{I} | \Gamma, \sigma_D^2) \\ &= \prod_{t,i,j} \left[P\left(D_{ti} | I_{tij}, \Gamma, \sigma_D^2\right) P\left(I_{tij} | I_{(t-1)i.}\right) \right]^{I_{tij}} \end{aligned}$$

where $P\left(D_{ti} | I_{tij} = 1, \Gamma, \sigma_D^2\right)$, the emission distribution, is Normally distributed as stated in the basic setting. And, $P\left(I_{tij} | I_{(t-1)i.}\right)$ is the transition distribution, which we discuss in the following section. Hence, the log-likelihood of the data is

$$\begin{aligned} L\left(\boldsymbol{\Gamma}, \sigma_D^2\right) &= \log P(\boldsymbol{D}, \boldsymbol{I} | \Gamma, \sigma_D^2) \\ &= \sum_{t,i,j} \frac{I_{tij}}{\sigma_D^2} \left(D_{ti} - Z_{tj}^T \Gamma_{g(t,j)}\right)^2 + I_{tij} \log P\left(I_{tij} \mid I_{(t-1)i.}\right) \end{aligned}$$

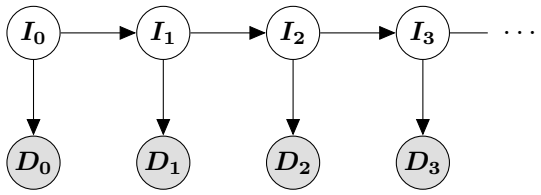


Figure 3.2: Hidden Markov model for each offensive possession: In each offensive team possession, defense starts with some initial assignments. These assignments progress over time as players and the ball move around on the court. We do not explicitly observe the assignments but we do observe locations of defenders. The locations of defenders depend on assignments. Hence, it is sensible to model the sequence of defense as hidden Markov model

3.2.2.1 Transition Probability

A state in our HMM is defined by the defensive matching. There are total of 5^5 ($= 3125$) possible matchings. It is computationally infeasible to learn the transition probability between each pair of states. A possible simplification is to point that each defensive player makes a transition independent of other [Franks *et al.*, 2015]. However, this contradicts the defensive coordination in basketball. For example, it is rare for two players to guard an

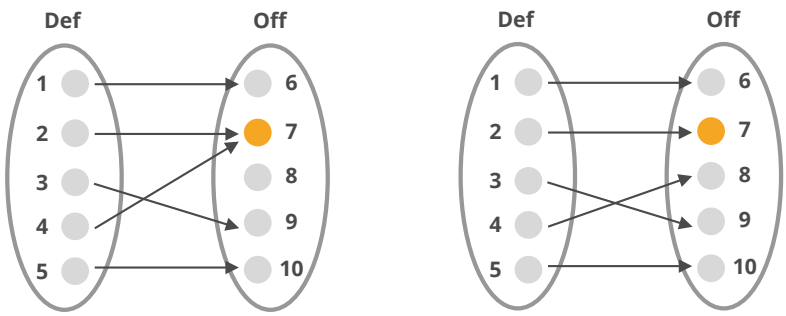


Figure 3.3: Bipartite graph representation of match-up transition: Player 7, highlighted with yellow, is the ball handler. The arrows represent defensive assignments. For example, defenders 2 and 3 are guarding player 7 – a double team state – in the left graph. The right graph shows one-on-one match-up. We assume that the state in the upper graph is a higher energy state (i.e. less stable) than the state in the lower graph. The change in the energy would be $e_2 + \tau - e_3$

off-ball player simultaneously. An independent defense transition model would not penalize such a transition. We propose a transition probability model that builds on a chemical bonding like formalism. Each "bond" corresponds to a defensive matchup and a defensive player switching to guard another player corresponds to breaking and forming a new "bond". There is a bond energy associated with each bond configuration, and there is a cost of going from one configuration to another based on the number of bonds broken and formed. We define four types of bond with the associated energy given by:

- e_1: 1-on-1 on-ball bond
- e_2: 1-on-1 off-ball bond
- e_3: 2-on-1 on-ball bond
- e_4: 2-on-1 off-ball bond

Let B_t denotes the location of the ball at time t. $B_t \in \{0, 1, \dots 5\}$ where $B_t = 0$ means the ball is in the air and $B_t = i$ means the ball is with player $i = 1, 2, \dots 5$. Let k_i denote the number of defensive players defending player i. The defensive matchup at time t is defined by the vector $S_t \in R^4$.

$$S_t = \sum_{i=1}^{5} F_t^i$$

$$\text{where} \quad F_t^i = \begin{cases} [0,0,0,0]^\top, & \text{if } k_i = 0 \\ [1,0,k_i-1,0]^\top, & \text{if } k_i \geq 1 \text{ and } B_t = i \\ [0,1,0,k_i-1]^\top, & \text{if } k_i \geq 1 \text{ and } B_t \neq i \end{cases}$$

We define the energy of a match up as $E_t = S_t^T \mathbf{e}$, where $\mathbf{e} = [e_1, e_2, e_3, e_4]^\top$. Let τ refers to the cost of breaking and forming a new bond i.e. the cost of switching single defensive assignment. Given the ball state at time t and $t+1$, the amount of energy required to change the state from S_t to S_{t+1} is given by $\Delta E = E_{t+1} - E_t + \eta_t \tau$, where η_t is the number of defensive assignment change from S_t to S_{t+1}. We define the probability of changing the state from S_t to S_{t+1} as:

$$P(S_t \to S_{t+1} | B_t, B_{t+1}) \propto e^{-\Delta E} \tag{3.1}$$

Note that to get the exact probability, we need to divide the denominator by sum of the proportionality term for all possible values of S_{t+1}. However, since the number of possible S_{t+1} states are 3125, this makes our optimization algorithm computationally expensive. We make a simplifying assumption that at most one "bond" could be changed at a time. Under this assumption, the number of possible S_{t+1} states becomes 25 which makes our optimization algorithm very tractable. Although this assumption might seem very restrictive, since the temporal resolution of our data is very high ($0.04s$), multiple changes in defense could be captured in consecutive time points and it would have almost same effect as these changes happening simultaneously.

3.3 Inference

Figure 3.4 is the graphical model representation of the defensive assignment model. To sample one possession using the generating process corresponding to the graphical model:

1. For each player and each court location, choose a $\Gamma_{pk} \sim \mathcal{N}(\mu_\Gamma, \mathcal{K})$
2. For each time point, sample defensive assignment I using the Markov chain with the bond energy based transition matrix
3. Sample each defender's location at each time point using $D_{ti}|I_{tij} = 1 \sim \mathcal{N}\left(Z_{tj}^\top \Gamma_{g(t,j)}, \sigma_D^2\right)$

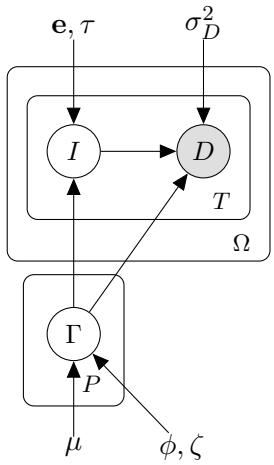


Figure 3.4: Graphical representation of defensive assignment model

We take an MCMC sampling ([Andrieu *et al.*, 2003]) approach to learn the parameters of our model. The output of our model is the defense assignment for each time point in each possession. Our sampling based approach to learn the model parameters is summarized in Algorithm 1. In step 2, given the bond parameters and the normal likelihood of D, sampling $\mathbf{I}$ using forward filtering and backward sampling algorithm is straight forward ([Bishop, 2006]). In step 3, we update the bond parameters given the state transitions $\mathbf{I}$ using the gradient based BFGS algorithm for maximizing the likelihood $P(\mathbf{I}|\tau, \mathbf{e})$ ([Wright and Nocedal, 1999]). Also, leveraging translational invariance, we set the value of $e_1 = 0$ to compute an unique optimal solution to the bond parameters.

For step 4, we index the data with respect to an offensive player and grid. For player p and k-th bin in the grid, we index observation $j \in \{1, ..., N_{pk}\}$, where N_{pk} is the total number of instances that player p is observed in k-th bin. Let $W_k = \sum_{j=1}^{N_{pk}} Z_{kj} Z_{kj}^T$ and

Algorithm 1: Inference for Defensive Assignment

1 Initialize all the fixed parameters. In particular, set $\mathbf{e} = \tau = \mathbf{0}$, $\sigma_D^2 = 5$, $\Gamma = \mu = [0.5, 0.25, 0.25]^\top$, $\phi = 3$, $\zeta^2 = 10$ for all the players at all the court locations. We use notation θ for all the fixed parameters.

2 Sample from $p(I|\Gamma, D, \theta)$ using forward filtering backward sampling algorithm (Bishop 2006)

3 Update $\mathbf{e}$ and τ given the sample of I using 3.1

4 Sample $p(\Gamma|I, D, \theta)$. Note that this step can be done in parallel to step 3

5 Update kernel parameters σ, ϕ, and σ_D given the sample of Γ

6 Repeat steps 2-5 until convergence

$V_k = \sum_{j=1}^{N_{pk}} Z_{kj} D_{kj}$. Define $V = [V_1, ..., V_K]^\top$ and W to be a block diagonal matrix with each block being W_k for $k = 1, ..., K$. Then, we can express the data likelihood as the following:

$$P\left(\boldsymbol{D}|\Gamma_p, \sigma_D^2\right) \propto e^{-\frac{1}{2\sigma^2}\left(\Gamma_p^T W \Gamma_p - 2\Gamma_p^T V\right)}$$

Please refer to the appendix for derivation. Using the GP prior on Γ_p, its posterior distribution is given by

$$\begin{aligned} \mathbb{P}(\Gamma_p \mid \boldsymbol{D}) &\propto \mathbb{P}\left(\boldsymbol{D}|\Gamma_p\right) P(\Gamma_p) \\ &\propto \exp\left(-\frac{1}{2}[\Gamma_p - \mu]^T \Sigma^{-1} [\Gamma_p - \mu]\right) \end{aligned}$$

where $\mu = \left(\frac{W}{\sigma_D^2} + \mathcal{K}^{-1}\right)^{-1}\left(\frac{V}{\sigma_D^2} + \mathcal{K}^{-1}\mu_\Gamma\right)$ and $\Sigma = \left(\frac{W}{\sigma_D^2} + \mathcal{K}^{-1}\right)^{-1}$. Hence, the posterior distribution is

$$\Gamma_p \mid \boldsymbol{D}, \boldsymbol{I}, \sigma_D^2 \sim N(\mu, \Sigma) \quad \text{with } \Gamma_{pk}^T \mathbf{1} = 1$$

Sampling from a normal posterior under a linear constraint can be shown to be equivalent to sampling from a normal distribution under a linear variable transformation(see the appendix). Finally, to update the kernel parameters in step 5, we use moment matching [Hansen, 1982] on the sample covariance of the sampled Γ. We use linear regression to minimize sum of squared errors of the elements of the sample covariance matrix and the kernel function. We use convergence of the data likelihood $p(D|\theta)$ as our convergence criteria.

3.4 RESULTS

We test our model on approximately 10000 randomly selected possessions (sampled from 314 matches) from the optical player tracking data for the 2015-16 NBA regular season. Figure 3.5 shows the convergence of data log-likelihood. Figure 3.6 shows convergence of FISTA algorithm used to fit the bond energy parameters. The estimated energy parameters associated with the bonds are shown in Table 3.1. The order of energy parameters is reasonable: $e_4 > e_3 > e_2 > e_1$. For instance, $e_3 > e_2$ means that a 2-on-1 on-ball defense is less stable than 1-on-1 defense (see Figure 3.3). Also, guarding a player with the ball is a relatively more stable configuration than guarding a non ball-handler. A high value of τ reflects that the change in the defensive assignment is not frequent.

Table 3.1: Estimated Parameters for Defense Assignment Model

Parameters	**Values**
e_1	0
e_2	0.547
e_3	2.055
e_4	2.159
τ	2.652
σ_D^2	6.912

One way to verify how well defensive modeling works is to visually check the sequence of assignments in each possession. Figure 3.7 illustrate snapshots of offensive possessions. The red and blue circles in the figures represent offensive and defensive players respectively. Empirically, our model captures defensive match-ups – which is indicated by lines between offensive and defensive players – very well at any given moment, over different regions of the court. It also accurately infers switches and double teams which appear during the possession shown in the figure.

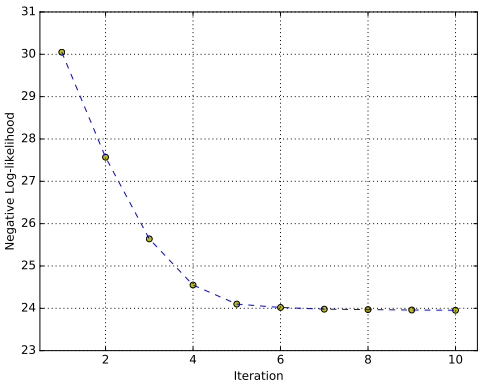


Figure 3.5: Convergence of data log-likelihood

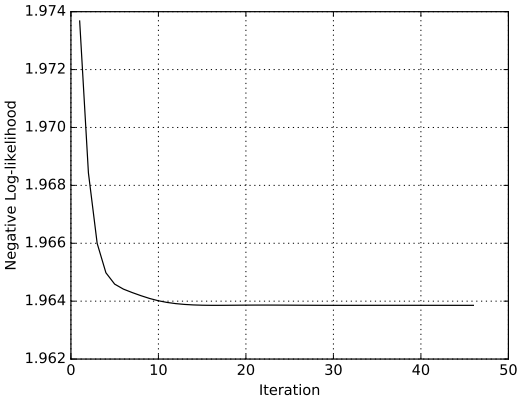


Figure 3.6: Convergence of FISTA algorithm

Note that we do not have access to the true defensive assignment. In order to verify the performance of our model quantitatively and to compare with other benchmarks, we asked independent annotators to create hand-coded labels for defensive assignments by watching the actual videos of Minnesota Timberwolves vs. Toronto Raptors match on February 10th, 2016 and Boston Celtics vs. Atlanta Hawks on April 22nd, 2016. We compared the results from four different defensive assignment criteria – the closest defender, fixed Γ model, and our defense assignment model, Gravity model – against the tagged labels. The closest defender simply assigns the defensive player that is closest to an offensive player. Fixed Γ model is the model suggested in ([Franks *et al.*, 2015]) and does not allow for Γ to be function of the player identity or location. Gravity model is the defense assignment model with player

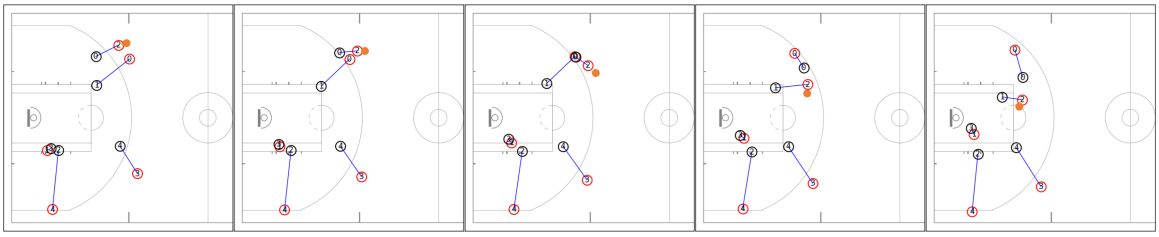


Figure 3.7: The sequence of inferred defensive assignments: The figure illustrates a few snapshots of our match-up modeling results. Empirically, our model captures defensive match-ups very well at any given moment, over different regions of the court. It also accurately infers switches and double teams which appear during the possession shown in the figure.

dependency and location dependency without using bond-based transition probability but with simple uniform transition probability that fixed Γ model uses. Gravity model + BEAT is our full model with bond-based transition probability. We use a matchup at a random time point from each possession of the two games, which gives us a total of 385 data points. Using a random time point from different possessions ensures that the each data points is independent (i.e. unlike matchups from the same possession). If the defensive assignment is exact, we call it a positive outcome, otherwise a negative outcome. Table 3.2 shows that both fixed Γ model and player dependent Gravity models clearly perform better than the baseline closest defender assignment. We use 5000 bootstrap samples to estimate the statistical significance of the accuracy (denoted by σ in Table 3.2). We also compute the p-value for the accuracy of Gravity + BEAT model being greater than other models. For each bootstrap sample, we compute the accuracy of all the models. The p-value for Gravity + BEAT model reported in Table 3.3 is the fraction of samples for which the Gravity+BEAT model had lower accuracy than one of the other models. We observe that our full model, Gravity model with BEAT, performs best among the four models we compare, both in terms of the average accuracy and a very low p-value (see Table 3.3 and Table 3.2). While the improvement in accuracy might appear small in absolute terms, the improvement has a significant impact when it comes to event detection in a complex setting such as ball screen. We discuss this in Chapter 4.

In addition to producing the highest accuracy, our modeling has an advantage over oth-

Table 3.2: Accuracy comparison

Model	**Accuracy (σ)**
Closest Defender	0.7421 (0.0229)
Fixed Γ model	0.9178 (0.0126)
Gravity model	0.9332 (0.0136)
Gravity model + BEAT	0.9584 (0.0103)

Table 3.3: p-value of accuracy of Gravity + BEAT being greater than the accuracy of other models

Model	**p-value**
Closest Defender	0.000
Fixed Γ model	0.006
Gravity model	0.02

ers – we gain more insights about how individual players attracts defenders. Figure 3.8 shows the estimated player attraction coefficients, γ^o, for selected NBA players namely Stephen Curry, DeAndre Jordan, and LeBron James over the court. It shows that different players clearly exhibit different levels of attraction to their defenders, while attraction rate generally increases as players get closer to the hoop. Stephen Curry who is known to be an elite shooter in the NBA with long shooting range, which causes his defenders to guard him closer than they guard other players. The left heat map in Figure 3.8 shows the clear pattern of how closely Curry is guarded over the court. Conversely, we observe that DeAndre Jordan is far less closely guarded by his defender, as shown in the middle heat map. Especially when Jordan is outside the 3-point line, his marginal attraction on defender appears to be very low, which was expected due to his poor shooting ability. One interesting observation is that defensive attraction appears to be higher along the baseline for all the three players (see Figure 3.8), showing longer vertical high values (across the court) than the horizontal (along the court). This may be due to defensive attempts to

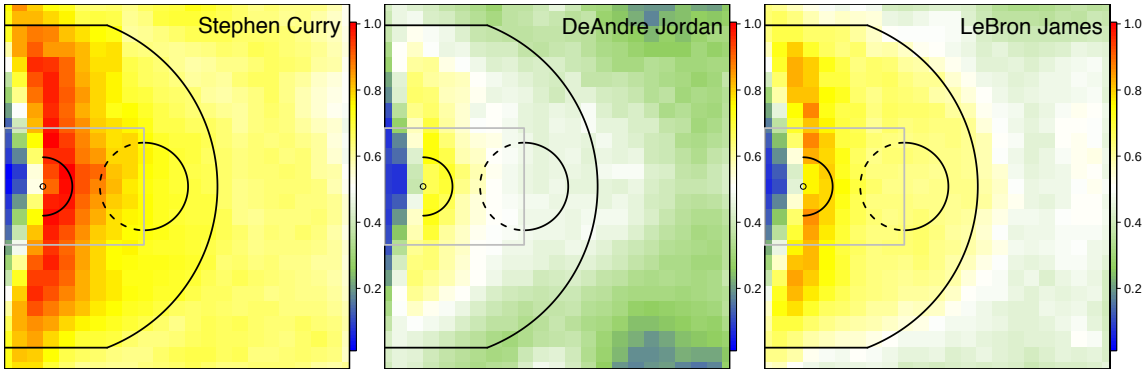
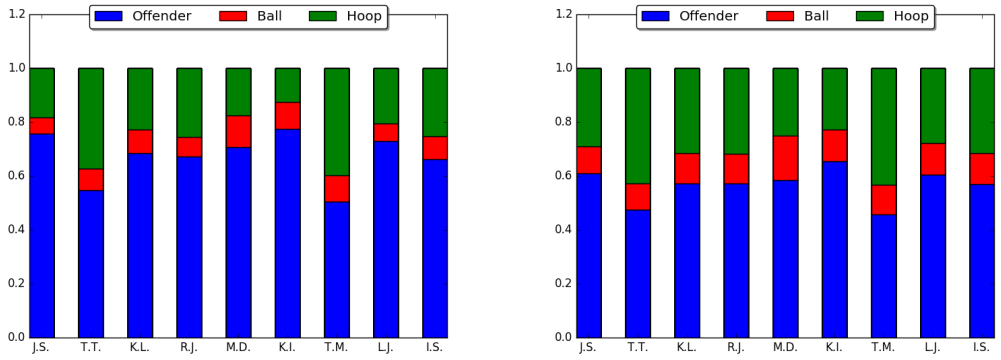


Figure 3.8: Player and location dependency on Γ

prevent easier (and closer) corner 3-pointers.

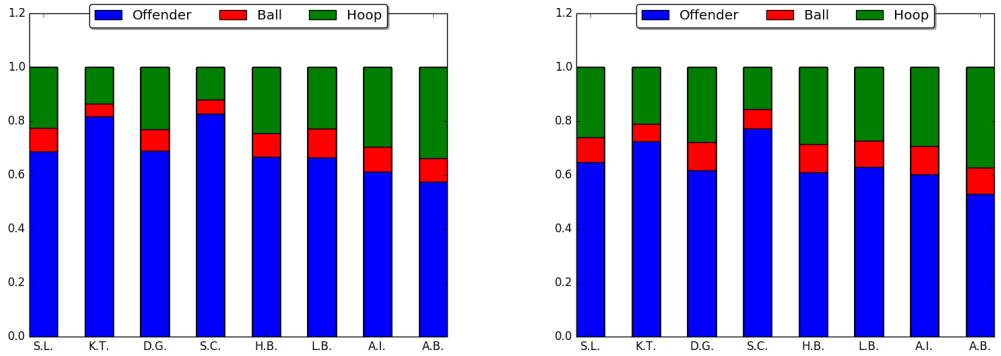


(a) Estimated Γ when players are inside 3-point line (b) Estimated Γ when players are outside 3-point line

Figure 3.9: Estimated Γ vector for Cleveland Cavaliers players: Player names are (from left to right) J.R. Smith, Tristan Thompson, Kevin Love, Richard Jefferson, Matthew Dellavedova, Kyrie Irving, Timofey Mozgov, LeBron James, Iman Shumpert

Figure 3.9 shows Γ vectors inferred from our model for players of the Cleveland Cavaliers. Figure 3.9(a) shows the average Γ over the area inside 3-point line and 3.9(b) shows the average Γ over the court outside 3-point line. $\gamma^o_{p.}, \gamma^b_{p.}, \gamma^h_{p.}$ are represented by blue, red, and green bars respectively.

On both figures, elite shooters, such as J.R. Smith (J.S.), and Kyrie Irving (K.I.), exhibit the highest attraction on defenders. We see that attraction decreases on average as players



(a) Estimated Γ when players are inside 3-point line (b) Estimated Γ when players are outside 3-point line

Figure 3.10: Estimated Γ vector for Golden State Warriors players: Player names are (from left to right) Shaun Livingston, Klay Thompson, Draymond Green, Stephen Curry, Harrison Barnes, Leandro Barbosa, Andre Iguodala, Andrew Bogut

move from inside to outside the 3-point line. We observe by decreased values of $\gamma^o_{p.}$ of all players (decreased blue bars), but players relative difference appear to be consistent, i.e. good shooters still attract defenders more closely than others. This pattern is also shown on the Golden State Warriors in Figure 3.10, where Stephen Curry (S.C.) and Klay Thompson (K.T.) appear to have the highest attraction effect on defenders both outside and inside 3-point line.

3.5 CONCLUSION

Our model serves as a robust tool to compute defensive assignments at any given time with high accuracy. Taking the player identity and court location into account makes the model more realistic. A good defensive assignment model is very crucial to doing any further analysis on player or team evaluation that requires match-up information, such as shooting ability against particular defenders and defensive metrics. Furthermore, it it serves a crucial input to automatically detect certain play events (ball screen, drive, post-up, etc.) whose definition depend on defense assignment. We also introduce a new metric to measure how much a player can stretch the floor in offense, a term referred to as gravity in basketball jargon. The heat map based representation of this metric provides a visual way to analyze

player's defensive attraction. Using this quantization of defensive attraction, teams can try to maximize spacing in their team offense or minimize opponent spacing.

Chapter 4

Event Detection using HMM

This chapter is based on the paper "Automatic event detection in basketball using HMM with energy based defensive assignment" [Keshri *et al.*,] which is a joint work with Minhwan Oh, Sheng Zhang, and Professor Garud Iyengar.

4.1 INTRODUCTION

Outcomes of events such as a *ball screen*, where a player on the offensive team prevents a defender from guarding a teammate by standing in the defender's way, a *drive*, where an offensive player moves faster towards the hoop with the ball, and a *post-up*, where there is an attempt to establish a position near the hoop for a closer shot, etc., are very useful in terms of understanding the characteristics of players and teams. These events provide more context to the ways in which players score. There events are not identified in a traditional box score model or the tracking data. Once these events are identified, one can create statistics on how efficient a player or team is when involved in these events.

[McQueen *et al.*, 2014] propose using support vector machines to detect a ball screens using human-labeled data. Their method requires the identity of the on-ball defender, i.e. the defender who is guarding the ball handler, and they set the defender nearest to the ball handler as on-ball defender. We discovered that the nearest-defender assignment is

only 74% accurate (see Table 3.2); thus, we expect that the method is unlikely to be very accurate. Moreover, their work is limited to learning ball screens. [Wang and Zemel, 2016] classifies different kinds of offensive play calls using neural networks. However, their method still depends on the labeled data and is only applicable to detecting team level play calls.

We approach the task of labeling events in an unsupervised setting in a hierarchical manner. We decided against using supervised data because labeling these events on video clips requires a significant amount of manual labor, and is, therefore, not scalable as we increase the set of events to be labeled – both in numbers of matches and in the number of event types. Furthermore, in our experience, human annotators inevitably make errors that often get amplified in supervised learning methods. Another advantage of using an unsupervised method is that it can be used to detect novel events that have not yet been recognized as a offense strategy in the league. As we will discuss later, our unsupervised method recognizes events using the specified prior distributions of the features that define the event in progress. Thus, if one is able to identify the features and specify the functional form of their distributions when the novel event is in progress, our model can learn to detect that event.

We take a two step approach to detect events of interest. First, we decide the features that are relevant for the events. These features could be position of the players, their defensive assignments, pairwise distances, their velocity etc. The key is to use features that have a different distribution when the events is in progress than when it is not. Once we recognize these features, we contruct a HMM with a binary hidden state. The hidden state represents the progression of the event of interest during a possession. 0 means the event is not in progress and 1 means the event is in progress. We also define the functional form of the emission distributions of the feature variables. The choice of the functional form and the initial parameters of the distribution guides the HMM to detect the occurrance of events as the possession progresses. An event in progress usually lasts for a certain time duration. The transition distribution of the hidden states imposes the stickiness behavior in the hidden state to capture the progression of the event. We learn a separate HMM for

each event (post-up, ball screen, and drive). The rationale for using separate HMMs for each event rather than one HMM for all events is that these actions are not necessarily mutually-exclusive. For example, it is possible to have a ball screen immediately followed by a drive, and therefore, some time epochs can simultaneously be part of two events – this cannot be accommodated in a single HMM. We find that the accuracy of the defensive assignments is crucial of the success of the event detection HMMs.

As we will see later, many of the features that we use to detect specific events can be easily measured from the raw tracking data. The defensive assignment is a very important input to detect many events, e.g. ball screen and post up. As we have discussed in the previous chapter, the defensive assignment is missing from the raw tracking data. We explore the impact of the defensive matchup accuracy in the result section. To fit our models, we first learn the defensive assignment. We learn defensive assignments using Gravity + BEAT model as discussed in Chapter 3. We also use the closest defensive assignment, the vanilla HMM model by [Franks *et al.*, 2015] and the gravity based model assignment to explore the impact of defensive assignment accuracy on event detection. In the next section, we discuss the HMMs that we use to detect the events of interest.

4.2 HIDDEN MARKOV MODEL FOR ACTIONS

In our event detection HMMs, we define the binary hidden state at each time point as an indicator of whether the event is under progress. We leverage the fact that the distribution of some key observables are different when the event is happening as compared to when it is not. We use this prior information to specify the parametric form of the emission distributions of these observables. We expect that the HMM model will learn the exact distribution and clearly distinguish between on and off event state. We model the the fact that when an action occurs, it usually lasts for some period of time by modeling event indicator as a Markov chain. We provide the specific details for each of our actions in the following sections. We will see that the defensive assignment is one the key features used in these HMMs.

4.2.1 Ball Screen

A screen is an attempt to prevent a defender from guarding a teammate by standing in the defender's way. We posit that ball screens can be detected using three observables:

$$\begin{aligned} X_t &= \text{distance between on-ball defender and} \\ &\quad \text{offensive player closest to the ball handler} \\ Y_t &= \text{distance between ball handler and the hoop} \\ C_t &= \text{speed of the offensive player closest} \\ &\quad \text{to the ball handler} \end{aligned}$$

Note that X_t is available as an observable only if we have a model for learning the defensive assignment. It is critical that the defensive assignment is very accurate, since errors in the defensive assignment gets amplified in ball screen HMM.

Let S_t denote the indicator for the ball screen event group. When $S_t = 1$, we expect X_t to be small. ([McQueen *et al.*, 2014]). When $S_t = 0$, X_t may potentially have heavy tails. We model this by setting the postulating the following distribution

$$X_t|S_t = 1 \sim \exp(\lambda_x) \quad X_t|S_t = 0 \sim \log\mathcal{N}(\mu_x, \sigma_x^2).$$

We model the fact that ball screens typically occur near the the 3-point line by setting

$$Y_t|S_t = 1 \sim \log N(\mu_y, \sigma_y^2) \quad Y_t|S_t = 0 \sim \text{Unif}(0, \theta_s),$$

where the baseline distribution is chosen to be uniform to put the least restrictions on the realization of Y_t when $S_t = 0$. During a ball screen event, C_t should be small, i.e. the screener should not moving much once the screen is set [1] We model this by setting

$$C_t|S_t = 1 \sim \exp(\lambda_c) \quad C_t|S_t = 0 \sim \log\mathcal{N}(\mu_c, \sigma_c^2),$$

where the baseline distribution is log-normal.

The hidden states S_t evolve according the transition matrix

	$S_{t+1} = 0$	$S_{t+1} = 1$
$S_t = 0$	ρ_0^s	$1 - \rho_0^s$
$S_t = 1$	$1 - \rho_1^s$	ρ_1^s

(4.1)

[1] The screening player must remain stationary; a moving screen is an offensive foul.

Figure 4.1 is the graphical model representation of this HMM.

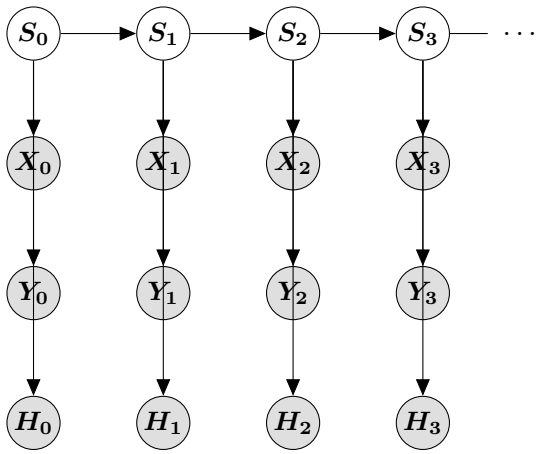


Figure 4.1: Hidden Markov Model for Ball Screen

4.2.2 Drive

A drive is a running movement to the hoop with the ball to take a layup, dunk or pass, etc. The intent of driving is not just to score (although it is probably the most common reason). It may be to draw a foul, draw the defense and hence pass the ball out to a teammate, or simply to create a ball movement.

We posit that a drive event can be recognized using the observables:

$$\begin{aligned} V_t &= \text{velocity towards the hoop,} \\ Y_t &= \text{distance between the ball handler and the hoop,} \end{aligned}$$

where V_t denotes the projection of the velocity vector computed using a single time frame onto the vector between the ball handler and the hoop.

Let G_t denote an indicator for a drive event. When $G_t = 1$, we expect V_t to be large towards the hoop, i.e. fast movement towards the hoop, and the higher the velocity of the ball handler, the more likely that $G_t = 1$. On the other hand, when drive event is not in progress, the velocity can be in any direction. We model this by setting

$$1/V_t^+ | G_t = 1 \sim \exp(\lambda_v) \quad V_t | G_t = 0 \sim \mathcal{N}(\mu_v, \sigma_v^2).$$

As a drive event progresses, the ball handler gets nearer to the hoop. Hence, it is sensible for Y_t to have an exponential distribution. We set

$$Y_t|G_t = 1 \sim \exp(\lambda_y), \quad Y_t|G_t = 0 \sim \text{Unif}(0, \theta_g).$$

The transition probability of Markov chain for the hidden states G_t is given by (4.1).

4.2.3 Post-up

A post-up is to establish a position in the low post, the area near the basket below the foul line. The offensive player usually faces away from the basket, so that his body can protect the ball from the defender.

The postulate that a post-up event can be recognized by focusing the on the observables:

$$\begin{aligned} A_t &= \text{speed of the ball handler} \\ Y_t &= \text{distance of the ball handler and the hoop} \\ R_t &= \text{distance between the ball handler and} \\ &\quad \text{the on-ball defender} \end{aligned}$$

Let U_t be an indicator for a post-up event. Under a post-up event, we expect A_t to be small, i.e. slow movement towards the hoop, and the lower the speed of the ball handler, the more likely it is a post-up event. We model this by setting

$$A_t|U_t = 1 \sim \exp(\lambda_a) \quad A_t|U_t = 0 \sim \log\mathcal{N}(\mu_a, \sigma_a^2)$$

Since post-ups mostly happen in the low post (the area near the basket below the foul line), it makes sense for Y_t to have a log-normal distribution with the mode at the low post. We assume a uniform distribution for the baseline of Y_t. We set

$$Y_t|U_t = 1 \sim \log\mathcal{N}(\mu_y, \sigma_y^2) \quad Y_t|U_t = 0 \sim \text{Unif}(0, \theta_u).$$

During a post-up, R_t should be small, i.e. the on-ball defender should be close to the ball handler once the post-up is set for the duration. We capture this by setting

$$R_t|U_t = 1 \sim \exp(\lambda_r) \quad R_t|U_t = 0 \sim \log\mathcal{N}(\mu_r, \sigma_r^2)$$

We model the time evolution of the hidden state U_t using the transition probability in (4.1).

4.3 INFERENCE

Algorithm 2: Inference for Event Detection

1 Initialize $\hat{P}(h_0), \hat{P}(x|h), \hat{P}(y|h), ...,$ and $\hat{P}(h'|h)$ randomly

2 **E Step**: For each sequence $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, ...,$ compute $\hat{P}(h_0|\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, ...), \hat{P}(h_t, h_{t+1}|\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, ...), \hat{P}(h'|h)$ using forward-backward algorithm

3 **M Step**: Update the model parameters $\hat{P}(h_0), \hat{P}(x|h), \hat{P}(y|h), ...,$ and $\hat{P}(h'|h)$ using MLE

4 Repeat steps 2-3 until convergence

5 Compute most likely sequence of hidden states, $\boldsymbol{h} = (h_0, ..., h_T)$ using Viterbi algorithm

We learn the defense assignment model before training our event detection models. The defense match is an input to our event detection models.

For the inference of event detection, we take a conventional inference approach for HMM using EM, and we use the Viterbi algorithm to compute the most likely sequence of hidden states ([Viterbi, 1967], [Bishop, 2006]). $h_t \in \{S_t, G_t, U_t\}$ is an hidden state of an action, and $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, ...$ are sequences of observed states for an action. Refer to Algorithm 2 for details. We use the HMM package in R ([Himmelmann, 2010]) to fit the model.

4.4 Result

As with defense modeling results, one way to verify how event detection works is a visual verification. Figure 4.2 illustrates a sequence that contains both ball screen and drive events. A ball screen occurs starting in the 3rd shot. Our model correctly detects this ball screen action and identifying the screener and the ball handler, which is indicated by green color. Then, the ball screen is followed by a drive to the basket (starting in the 7th screenshot) and our model also recognizes the action correctly. Figure 4.3 illustrates a sequence that contains a ball screen. Our model appears to correctly detect the event As we observe in both Figures 4.2 and 4.3, our model automatically detects the beginning and the end of actions and identify action types accurately. The demo videos of our methods may be

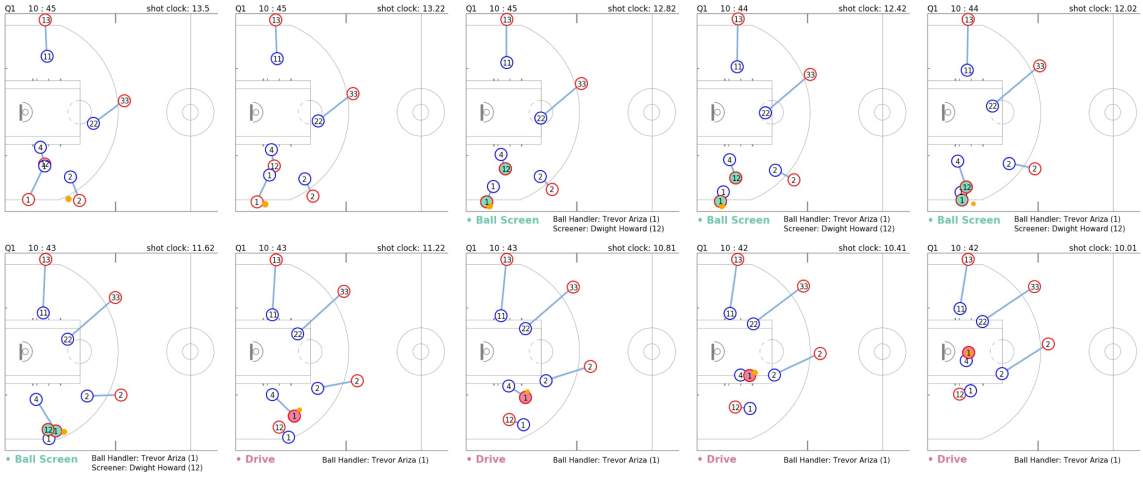


Figure 4.2: The sequence illustrates snapshots of our defense assignment and event detection modeling results from Houston Rocket vs. Washington Wizard match on January 30th, 2016. The red and blue circles are offensive and defensive players respectively.

viewed on the project website: `https://sites.google.com/view/eventdetection/`.

To evaluate the prediction performance of our event detection models, we asked independent human annotators to create hand-coded labels for actions by watching an actual video of Houston Rocket vs. Washington Wizard match on January 30th, 2016 and and Boston Celtics vs. Atlanta Hawks match on April 22nd, 2016. For comparison purposes, the annotators first generated individual player possession segments, during which a ball handler is fixed, i.e. if the ball handler changes, then it will mark a new player possession segment. Since the data contains 25 frames per second, our model predicts on the time scale of 25 frames per second. However, it was not feasible for a human to watch and label actions at the same rate. A human annotator only has access to the game clock to record the start and end time of an event, and hence indicates the time to the closest full second unless there is less than 1 second remaining in a quarter. In order to set up fair comparison between the human annotator and the prediction algorithm, we measured performance on the event segment level. We compared the predicted labels inferred by our model against human-annotated labels. A segment is marked positive in the actual setting if the human annotator found an event in the segment, and zero otherwise. Similarly, a segment was

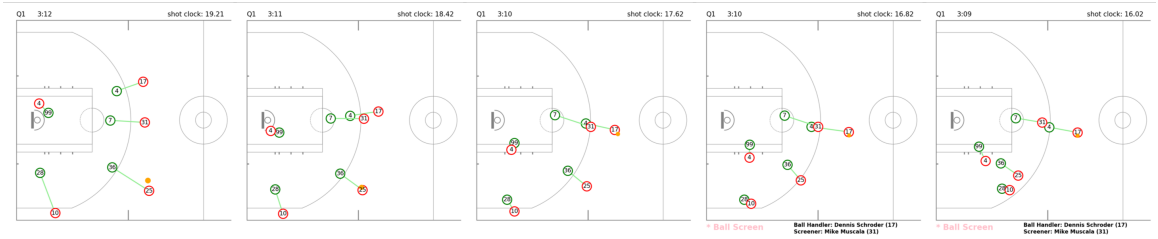


Figure 4.3: The snapshots of a sequence that contains a ball screen event from Boston Celtics vs. Atlanta Hawks match on April 22nd, 2016. The red and blue circles are offensive and defensive players respectively.

marked positive in the predicted setting if our prediction algorithm found an event in the segment, and negative otherwise. The confusion matrices for the three actions are shown in Tables 4.1, 4.2, and 4.3. The results show high accuracy of our action detection models for ball screen, drive, and post-up with accuracy of 0.882, 0.944, and 0.989, respectively for the Gravity + BEAT model. The standard deviation of each element of the confusion matrix was computed using 5000 bootstrap samples of the test data. The statistical significance of improvement achieved by using Gravity + BEAT defensive assignment model reported in Table 4.4 was also computed using the bootstrap samples. For each bootstrap sample, we compute the True Positive (TP) and True Negative (TN) rate of all the models for Ball Screen and Post-up events [2]. Then we compute the percent of bootstrap samples for which the rate of Gravity + BEAT model is greater than or equal to the rate of other models. The p-values of the Gravity + BEAT model shown in Table 4.4 is percent of samples for which the Gravity + BEAT has lower TP and TN rate as compared to the other models.

4.4.1 Discussion on Event Detection Errors

We have a relatively lower accuracy for ball screen event detection compared to the other events. This was somewhat expected due to the complexity of the event and its dependence on pairwise distances of multiple players involved in the event. Note that most of the misclassifications come from false positives, i.e. predicting non-screen events to be screens.

[2]We ignore FP and FN rate because they are additive inverse of TP and TN respectively. We also ignore Drive because all the models will perform equally well on that event

Table 4.1: Ball Screen Detection

	Prediction					
	Using closest defender		**Fixed Γ model**		**Gravity + BEAT**	
Actual	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**
Positive	103(8.736)	60(7.018)	142(9.526)	21(4.427)	155(10.067)	8(2.775)
Negative	84(8.332)	168(10.099)	49(6.627)	203(10.086)	41(6.056)	211(10.253)

Table 4.2: Drive Detection

	Prediction					
	Using closest defender		**Fixed Γ model**		**Gravity + BEAT**	
Actual	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**
Positive	127	7	127	7	127(9.817)	7(2.777)
Negative	7	111	7	111	7(6.165)	111(10.133)

We found that many of these misclassifications come from hand-offs or slip screens (without properly setting a screen) which appear to be very similar to ball screens, especially on coordinate visualization. One example is shown in Figure 4.4, where James Harden (Number 13 on red) comes closer to the on-ball defender but before he sets up a screen, he slides over to the top of the three-point arc. Our model predicts this sequence to be a screen. However, after verifying on the video, we confirmed that it is not a screen. Challenges in distinguishing these subtle difference lie in the limitations of the data, i.e. that fact that the location of the player is represented only as $x-y$ coordinate without any information on player orientation or hand-movements, etc., which is in fact significantly helpful when visually classifying these events. However, despite these limitations, our methods still performs well on events considered.

4.4.2 Accuracy Dependence on Defensive Assignments

We also provide the results using different defensive assignments to show the event detection's dependence on the defensive assignment's accuracy (Tables 4.1, 4.2, and 4.3). First, event detection with closest defender as defense assignments clearly performs poorly on

Table 4.3: Post-up Detection

	Prediction					
	Using closest defender		**Fixed Γ model**		**Gravity + BEAT**	
Actual	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**
Positive	15(3.850)	4 (2.008)	17(4.046)	2(1.397)	18(4.182)	1(0.985)
Negative	109(9.535)	532(10.195)	21(4.544)	620(6.077)	6(2.465)	635(4.975)

Table 4.4: p-value of True Positive (True Negative) percent of Gravity + BEAT model being greater than or equal to other models

Post-up	**TP**	**TN**	**Ball Screen**	**TP**	**TN**
Fixed Gamma	0.000	0.002	Fixed Gamma	0.005	0.001
Closest Defender	0.000	0.000	Closest Defender	0.000	0.000

ball screens and post-up since detecting these events require the identification of defensive players. Using closest defender performs significantly worse in ball screen events since the identification of the on-ball defender can be more challenging in ball screens (due to defender switching, or the defender going under the screen). Note that drive events do not depend on defender identification. Hence, the results are the same for all defensive assignments. Overall, using Gravity + BEAT gives better results than using fixed Γ model due to the higher accuracy in defensive assignments (as reported in Table 3.2).

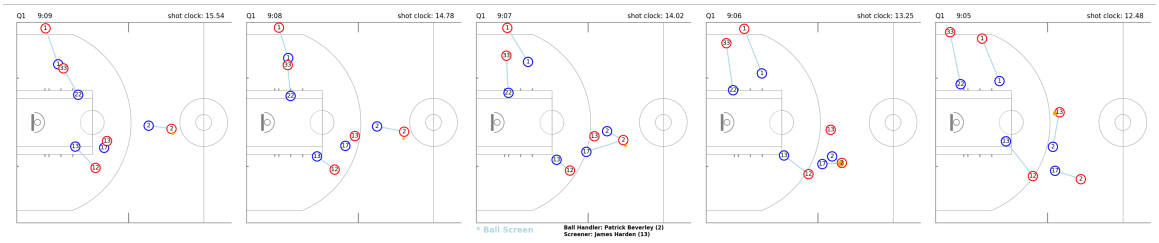


Figure 4.4: The snapshots of a sequence that contains a misclassified ball screen event. The red and blue circles are offensive and defensive players respectively.

4.5 CONCLUSION

Our automatic event detection system provides a new framework to detect events in basketball without any human tagging. We can extend the scope of events to be detected as long as one can describe the events with characteristic conditional distributions. Thus, one can detect or even posit the conditional distributions and check if there is an event that actually occurs significant number of times to fit the specified form with statistical significance. We argue that our work lays a foundation for creating richer analytics both based on event detection and accurate defense assignment information.

Chapter 5

Missed Shots important for Player Ranking?

5.1 Introduction

In basketball, all players compete on both offense and defense, and the core strategies revolve around scoring points on offense and preventing points on defense. Every shot event in a basketball match is the result of a shooter's action under the influence of defense of the opponent team – whether it is defense by a single opponent player or multiple opponent players (or none when a shooter is wide open). While it may not seem difficult to empirically characterize shooting abilities of shooters by simply observing field goal percentages (the percentage of shot made out of total number of shots attempted), it is much more challenging to account for how defenders affect shooting. The outcome of a shot is affected by the ability of both the offensive and defensive players. With the advent of tracking data of players on the court in the NBA, it has become possible to infer the defensive matchup of each shot. Given the defensive player and shooter for each, we can train a random effects model which predicts the shot outcome given the individual offensive and defensive abilities of the players. The random effect coefficients can then be used to rank the players. This gives a more sensible ranking of the players than the rank without taking the defense into account. One can go one step further to rank the players based on different areas on the court or different types of shot. [Miller *et al.*, 2014], [Fearnhead and Taylor, 2011] have used

random effects based models to measure and rank players controlling for the defense.

Previous models have been trained on the binary shot outcome and have completely ignored an interesting piece of data; namely, the trajectory of the ball. We call the likelihood that a trajectory results in a successful shot the **quality** of the trajectory. It is important to note that the trajectory from a shot location is not unique. Some shots use spin while others use the back board (bank shots). This is the reason why players take different types of shot (jump shot, bank shot, hook shot to name a few [myactivesg, 2016]) as function of court location and the defense. A shot outcome is a realization of the random outcome associated with a particular trajectory. A high likelihood, i.e. high quality, shot can be unsuccessful. There are also low quality shots that have very low chance of success. For instance, in this NBA game video [thehelpdefender, 2013] at the 17 sec mark, a player misses a shot but the shot was certainly nowhere close to being made. The shot quality is not part of the tracking data. Our goal in this chapter is two fold. First, we develop a model to infer the quality of the shot using the trajectory of the ball. We then use this quality measure to infer the offensive and defensive abilities of the players using random effects model. We show that this gives a more reliable estimate of player's ability.

5.2 Data

Our data consist of observations from the 2012-2016 NBA regular season. We use about 300,000 shot trajectories to evaluate the players. Our algorithm uses the trajectory of the ball when the shot is in progress. The raw data does not provide the exact time when a shot was attempted by a player (we do not know when the ball left the hand of the shooter). We approximate this time by looking at the ball trajectory and tracing back to the time when the ball leaves the hand of the shooter. We use this time instance as the shot time, and to find the defensive assignment for the shot.

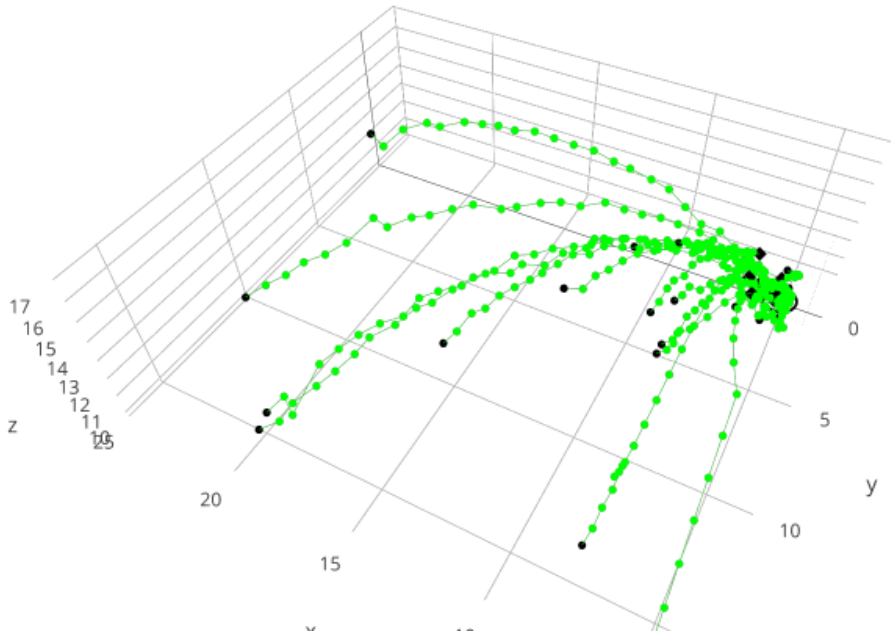


Figure 5.1: Sample trajectories of shots

5.3 Method

Our full model is divided into three parts. First, we estimate the optimality of trajectory of the ball for each shot. Second, we estimate the defensive assignment of the offensive player when a shot was attempted. Finally, we use a random effects based model to estimate the individual offensive and defensive abilities of the players.

5.3.1 Trajectory Optimality

Previous attempts in estimating latent offensive and defensive abilities of the players focused on the shot outcome, i.e. if a shot is missed, zero points are assigned to the player, and full points assigned when the shot is made. The binary treatment of the shot outcome ignores the information in the trajectory of the ball. Players attempt to throw the ball along a trajectory that would make the shot; however, once the ball leaves the players hand, there are many exogenous variables, e.g. the spin of the ball, the backboard, air drag since the ball is not a point-mass, etc., that control the shot outcome. Even a high quality trajectory may

not result in a successful shot. We call these shots as "closely missed shots". On the other hand, there are shots that miss because the quality of the trajectory was very poor. We call these shots "badly missed shots". When evaluating a shot, we do not want to attribute equal weight to the closely missed shot as to badly missed shots. In case of an unsuccessful shot, we would like assign points to the shooter depending on the prior likelihood of the trajectory being successful . Thus, we need a model for assessing the quality of a trajectory, i.e. the likelihood, that the trajectory will result in a successful shot. We have the data for all the shot trajectories and we also know whether a trajectory corresponds to a made shot or a missed shot. A model trained to predict the likelihood of successful shot as a function of shot trajectory should be able to predict the quality of a new unseen trajectory. We begin with a physics based model and then consider neural network based models.

Suppose the ball were a point mass. Then its trajectory would have been a parabola in a plane in the 3D space. The ball could either directly fall into the hoop, in which case, the angle of the parabola with hoop determines success (see Figure 5.2 (b)). Or, the ball can fall into the hoop after bouncing off the backboard, in which the angle of the parabola with the respect to the backboard and the terminal velocity of the ball determine success. The input to the model are six parameters that characterize the trajectory and the output is the shot outcome. These parameters are defined as follows:

1. The angle θ that the first principal component of the trajectory using only the x-y coordinates of the ball makes with the line dividing the court. See Figure 5.2(a) for details.

2. We define the plane of the ball movement as gthe plane defined by adding z-axis to the principal component lineinthe x-y plane. We project each point on the trajectory onto this plane as shown in Figure 5.2(b), and fit a parabola to the projected points. The next three parameters describes the quadratic equation of the parabola.

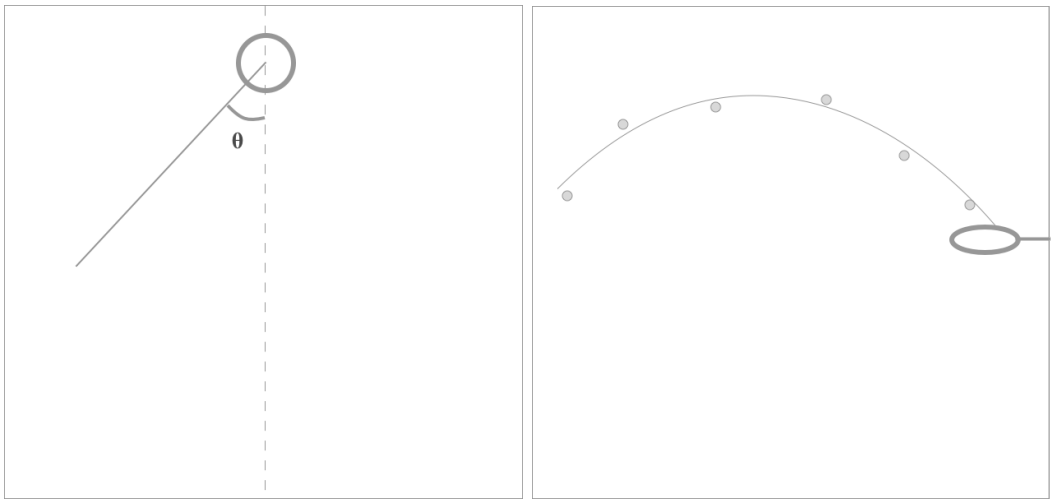
3. The fifth parameter is the velocity of the ball along the principle component plane i.e. the plane of the parabola.

4. The last parameter is the distance of the shot location to the hoop.

We call these six parameters the miscellaneous summary of a trajectory and denote them by the vector **m**. **m** carries complete information of the shot trajectory if the ball is assumed to be a point mass.

This model performed very poorly in predicting the shot outcome. The AUC score was only 0.59 on the validation data. Figure 5.1 shows a few shot trajectories. We see that the trajectory carries a lot of noise in both vertical and lateral direction. It is not a smooth trajectory and it is easy to see kinks and abrupt movements. This is not because the physical behavior of the ball but rather the noise introduced by the data recording instrument. The ball is not a point mass but the instrument treats it as one and only records the location of one point on the ball (this information is not available in the data). This makes the miscellaneous parameters very susceptible to the noise in the trajectory. Even a slight error in the recording of one point on the trajectory could change the angle of the trajectory plane. [Beuoy, 2015] shows that even for free throws, which are the most stable trajectories, the distribution of points where the ball crosses the hoop for a missed vs a made shot is not significantly different. Hence, we conclude that the assumption that the ball is a point mass results in a very noisy estimate of the shot plane and the downstream variables. We, next, discuss more powerful statistical models that are able to overcome the noise.

A model which considers the full trajectory of the ball and is able to filter the noise based on the context could improve the prediction of the shot outcome. Neural networks are a class of machine learning models that are able to learn complex relationship in the data. Using the complete trajectory data is a good use case for such a model. We denote each point on the trajectory by $\mathbf{x}_t$, which is a 4D vector with 3D point and the time stamp. We fit various feed forward neural network models starting with a simple feed forward model which uses the full trajectory of the ball and the summary parameter **m** to predict the shot outcome (see Figures 5.3 and 5.3a). This improved the AUC score to 0.847 and 0.851 respectively. We also tried a Long short-term memory (LSTM) neural network. LSTM is a member of recurrent neural network (RNN) family which is used for modeling temporal data. LSTM performs well in modeling long term temporal dependency in a time series

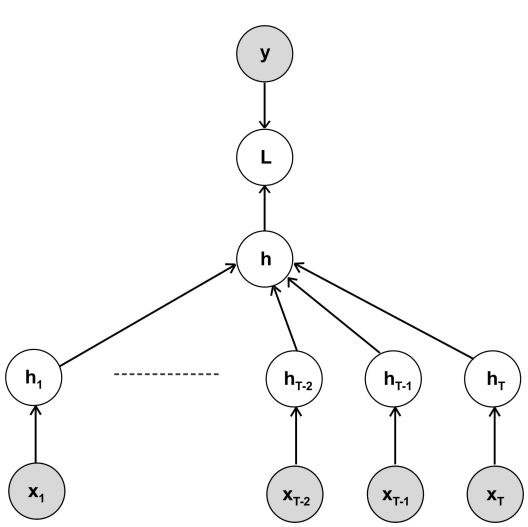


(a) Principal Component plane projection (b) Fitted Parabola to the trajectory

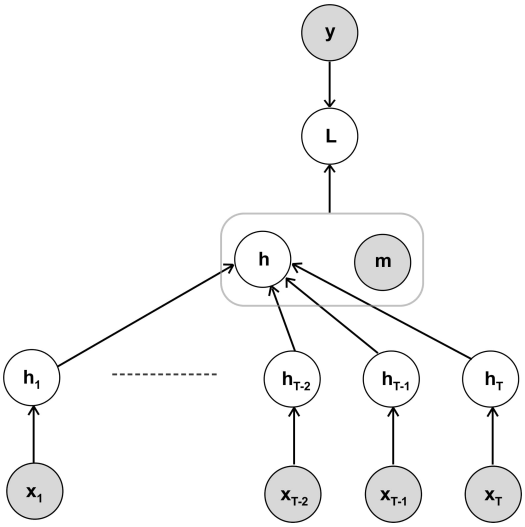
Figure 5.2: Miscellaneous parameters information plots

Table 5.1: BLSTM Test Error at different time points

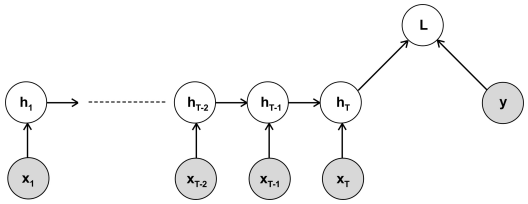
Time Point	Cross Entropy
1	0.3562
2	0.3544
3	0.3521
4	0.3515
5	0.3519
6	0.3520
7	0.3493
8	0.3476
9	0.3486
10	0.3521
11	0.3539
12	0.3550



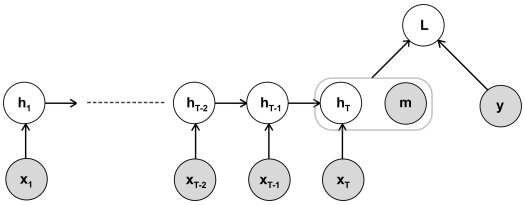
(a) Feed forward Network



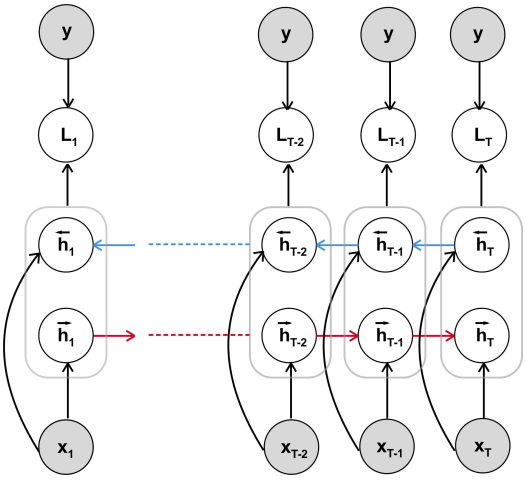
(b) Feed forward Network with Miscellaneous input



(c) LSTM Network



(d) LSTM Network with Miscellaneous input

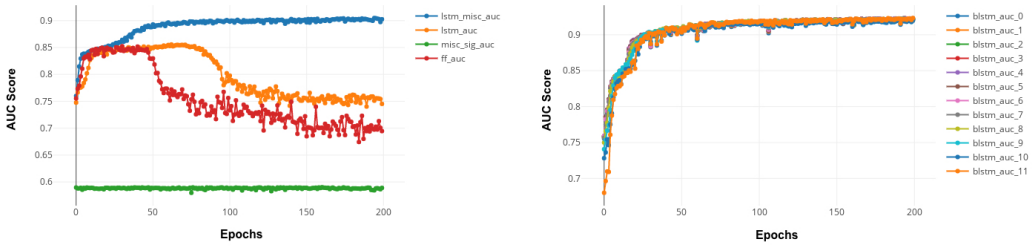


(e) BLSTM Network

Figure 5.3: Neural network based models for predicting shot outcome given the ball trajectory: T is the length of the sequence, y is the shot outcome, L is the loss function, h refer to the hidden states of dimension 32, m is the miscellaneous information of the trajectory. A rounded rectangle refers to the concatenation of the vectors. The final prediction has a sigmoid activation which gives us a loss when predicting the shot outcome

data ([Gers *et al.*, 1999]). Modeling shot trajectory is a perfect use case for a LSTM network. Shot trajectory is naturally a time series data and it also has long term dependency. The outcome of the shot is not only determined by the last few points on the trajectory but also on the ball location when it was released from the hands of the shooter. Shah et. al. [Shah and Romijnders, 2016] uses LSTM based models on three point shots for trajectory simulation purposes. We used two versions of LSTM models, one that only takes the trajectory as input (see Figure 5.3b) and the other one that also uses the miscellaneous trajectory information **m** as shown in Figure 5.3c. The latter model achieved a significant improvement over feed forward model, achieving an AUC score of 0.91.

Finally, we fit a Bi-directional LSTM model shown in Figure 5.3d. Bi-directional LSTM has been very successful at prediction task that depends on the whole context ([Huang *et al.*, 2015], [Zhao *et al.*, 2018]). It takes into account the information flow both forward and backward. Thus, at any point of time, the model has a summary of information from future and the past times. In our case, BLSTM model predicts the shot outcome for each time point on the trajectory. Since the information in this model flows both ways (shown by the red and the blue lines), the full context of the trajectory is available as a summary at each time point. Note that [Zhao *et al.*, 2018] uses the same BLSTM model. The cross entropy at each of the twelve time points on the validation data set is shown in Table 5.1. We see that the error rate goes down until the eighth time point and then starts increasing again. We believe that as the ball gets close to the basket, the error rate improves, but as we get too close the noise in the movement of the ball increases, possibly due to interaction with the backboard. Also, the standard deviation of probability prediction across the time points is 0.0571. We take the average predicted probability across all the time points of BLSTM and achieve an AUC score of 93.1%. We also tried a BLSTM model which also uses the miscellaneous information vector **m** but it did not achieve any significant improvement over the vanilla BLSTM model.



(a) All models except the BLSTM Model

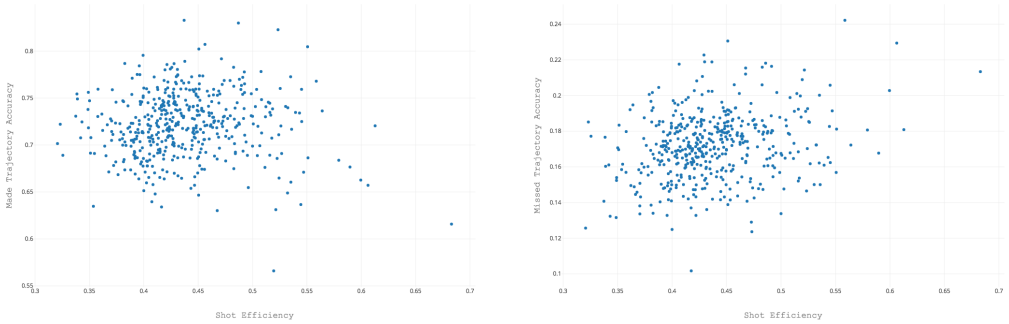
(b) BLSTM for different time points

Figure 5.4: Validation AUC scores of various trajectory models changing with epochs on the validation data

5.3.2 Trajectory Analysis

We visually verified the effectiveness of BLSTM model to distinguish good missed shots from bad missed shots. In Figure 5.7, we show the shots that the player missed and the model predicted the shot to go in with a probability less than 0.05. These are the shots that the model labels as badly missed shots. Indeed, we see these shots are clearly very far from the hoop. On the other hand, in Figure 5.6, we plotted the shots that the players missed and the model predicted the shot to go in with a probability greater than 0.95. In other words, these are the shots the model would predict to go in. We clearly see that these shots are in fact very good shots and didn't miss the hoop by much. We also notice that the missed shots that are predicted to go in with high probability follows a trajectory with high peak (high arc). On the other hand, the missed shots that are predicted to go in with low probability follow a low arc trajectory. This makes sense because a high arc shot is exposed to more surface area of the hoop compared to low arc shot. Similar argument is made for free throw shots by Stephen Curry ([Beuoy, 2015]). Stephen Curry, one of the game's most elite shooters, has high average shot arc for free throws compared to other players.

Our model allows to us to verify if a good shooter's missed shots are on an average more closely missed than an average shooter's missed shots. In Figure 5.5, we plot the average probability of missed shots vs shot efficiency (ratio of made shot and total shot attempted



(a) Average Missed shot probability: Correlation 0.01 (b) Average Made shot probability: Correlation 0.255

Figure 5.5: Scatter plot of average trajectory probability vs shot efficiency of players

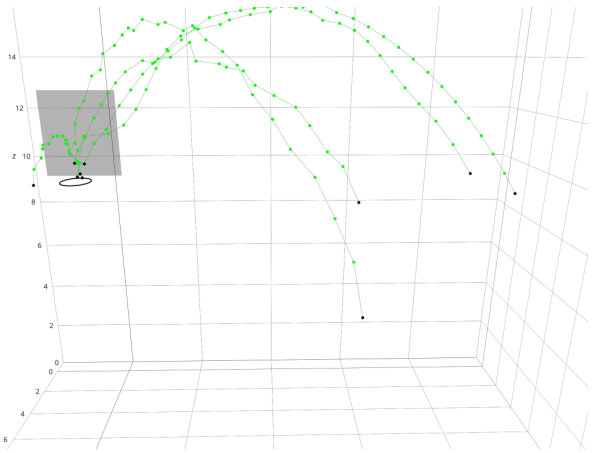


Figure 5.6: Sample missed shots with probability of making the shot more than 0.5

by a player) for each player. Indeed, a player with high shot efficiency (considered to be a better shooter) has higher average probability of missed shot trajectory i.e. their missed shots have more optimal trajectories. On the other hand, we see that the average probability of made shots is not correlated (0.01 correlation) with shot efficiency. One explanation for this observation is that all made shots are equally good i.e. there are no badly made shots. Trajectory quality is more appropriate to distinguish a badly missed shot from a closely missed shot. This leads to the next section of this chapter where we discuss the model we use to infer the shot and defense ability of players.

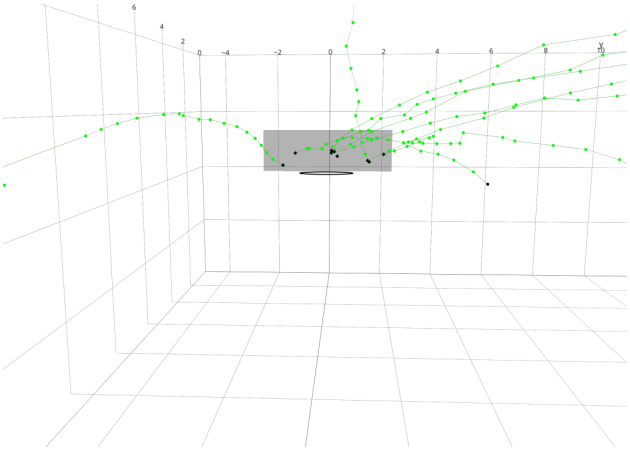


Figure 5.7: Sample missed shots with probability of making the shot less than 0.05

5.3.3 Random Effects

Random effects model is a hierarchical linear model that is used to learn the parameters which vary within a group allowing for the information is pooled across the group. This type of model has recently been used in basketball analytics to learn individual abilities of players [Franks *et al.*, 2015; Oh *et al.*, 2015]. In the past, these models have been trained with shot outcome as the dependent variable. While the shot outcome accurately reflects player's ability when they make a shot, it is not the case when a shot is missed. As we argued in the previous section, missed shots can be a badly missed or a closely missed. We also saw in the previous section that good players on an average have more optimal missed shot trajectories compared to an average or below average shooter. This information can be used to get a better ranking of the players. The BLSTM model gives us the quality of the trajectory for each shot. Our goal is to compare the abilities of players learned using the shot outcome and trajectory quality. The information content in the trajectory data is clearly more than a binary classification of made and missed shot.

$$S_{ijk} \sim \sigma\left(c + \alpha_i - \beta_j + \gamma d_{ijk}\right) \tag{5.1}$$

$$\begin{aligned} \log\left(\frac{P_{ijk}}{1 - P_{ijk}}\right) &\sim c + \alpha_i - \beta_j + \gamma d_{ijk} + \epsilon_{ijk} \\ \epsilon_{ijk} &\sim \mathcal{N}\left(0, \sigma^2\right) \end{aligned} \tag{5.2}$$

Shot-based model in 5.1 is a logistic model where α_i and β_j are the random effects for offensive and defensive abilities of players, c is the intercept, and S_{ijk} is the indicator variable for made vs missed shot. i is the global index of the offensive player, j is the global index of the defensive player, and k refers to the k^{th} shot attempt for these pair of players. Trajectory-based model in 5.2 is based on the trajectory optimality as the dependent variable. P_{ijk} is the probability that the shot S_{ijk} is made given the trajectory of the ball, as predicted by the trajectory model. We transform this probability to a real number using log odds function. This makes the model structurally equivalent to the shot outcome model. In both the models, a prior distribution is specified for the random effects to pool information across the players:

$$\begin{aligned} \alpha_i &\sim \mathcal{N}\left(0, \sigma_\alpha^2\right) \\ \beta_i &\sim \mathcal{N}\left(0, \sigma_\beta^2\right) \end{aligned}$$

In results section, we will compare the ranking improvements and player differentiation advantage we have using the trajectory based model.

In addition to learning players ability over the full season or multiple season, it is often useful to learn player's performance over a few number of games. Also, the abilities of the NBA players change over time. In particular, the players don't perform equally well in every game [Fearnhead and Taylor, 2011]. It will be useful if we can learn how a certain player performed on a game by game basis. We estimate the change in abilities of the players across games using the trajectory model and shot based model. We use a nested random effects model to estimate change in the abilities across games.

$$\begin{aligned} \log\left(\frac{P_{ijk}}{1-P_{ijk}}\right) &\sim c + (\overline{\alpha_i} + \alpha_{ig}) - \left(\overline{\beta_j} + \beta_{jg}\right) + \gamma d_{ijk} + \epsilon_{ijk} \\ \epsilon_{ijk} &\sim \mathcal{N}\left(0, \sigma^2\right) \end{aligned} \tag{5.3}$$

$$S_{ijk} \sim \sigma\left(c + (\overline{\alpha_i} + \alpha_{ig}) - \left(\overline{\beta_j} + \beta_{jg}\right) + \gamma d_{ijk}\right) \tag{5.4}$$

In 5.3 and 5.4, we allow for the possibility that player abilities may change across games (α_{ig}, β_{ig}) as a deviation from average ability ($\overline{\alpha_i}$, $\overline{\beta_j}$) by introducing the prior as shown

below:

$$\begin{aligned} \alpha_{ig} &\sim \mathcal{N}\left(0, \sigma_{\alpha}^{2}\right) \\ \beta_{ig} &\sim \mathcal{N}\left(0, \sigma_{\beta}^{2}\right) \end{aligned}$$

5.4 Results

As the first step, we want to verify whether the trajectory model improves prediction of the shot prediction probability. We use the trajectory model to reclassify a very closely missed counted as a made shot and a very badly made shot counted as a missed shot in the training set. Such a reclassification effectively reduces the noise in the training samples, and we expect this should improve, or at least not reduce, the prediction accuracy of shot outcome. We show later that this reclassification also improves the estimate of player's ability. Specifically, we reclassify a missed shot that has more than 95% probability of being a successful shot based on the trajectory model as a made shot, and we reclassificy a made shot that had less than 0.05% probability of being a made as a missed shot. Note that 5% is the same cutoff that we used to visualize the quality of trajectory in Figure 5.5. Next, we fit the random effects model (5.1) with the reclassified shot outcomes as the independent variable on 85% of the data. We test the model on 15% of the data set. We compute the standard error for prediction accuracy by repeating the experiment 100 times with a random sampled test set. The shot outcome prediction accuracy of the model trained on the reclassified shots using BLSTM trajectory quality was 0.6091 compared to 0.6023 for the original shot outcome; see Table 5.2. The RMSE improved from 0.4876 to 0.4864. While this improvement might appear small, it translates to an improvement of approximately 0.552 points in the final game score (see [Chang *et al.*, 2014] for a similar argument) [1]. We believe that this is not insignificant given that many NBA games are won by one point difference. We also notice that the reclassification using the BLSTM model is statistically superior to all other models. Also, notice that the Misc model does not improve the results over the original shot outcome. This is because we only reclassify close to 0.02%

[1] An average of 200 shot attempts in an NBA game with close to 33% shots being a three point shot translates to an attempt for a total of 466 points. An improvement of 0.001 in shot outcome prediction error corresponds to an improvement of 0.55 points in total points scored in a game.

Table 5.2: Shot outcome prediction results for model trained on reclassified shots using trajectory models vs original shots data

Model	Accuracy(s.e.)	RMSE(s.e.)
Original shot model	0.6023(0.0007)	0.4876(0.0002)
Misc Model	0.6027(0.0006)	0.4875(0.0002)
LSTM+Misc Model	0.6049(0.0007)	0.4872(0.0002)
BLSTM Model	0.6091(0.0008)	0.4864(0.0002)

Table 5.3: Standard Deviation of Random Effects (and their standard error) corresponding to 5.1 and 5.2

Model	$\sigma_\alpha(s.e.)$	$\sigma_\beta(s.e.)$
Shot Model	0.1143(0.009)	0.0267(0.004)
Misc Model	0.0120(0.0007)	0.0052(0.0003)
LSTM+Misc Model	0.1642(0.0122)	0.0696(0.004)
BLSTM Model	0.1728(0.0125)	0.0746(0.005)

of the original shot outcomes (compared to 0.4% for the BLSTM model) and that does not significantly affect the prediction in the test set.

Next, we use the models defined in 5.2 and 5.2 to learn the offensive and defensive abilities of players. First, we compare the averaged BLSTM model prediction against the shot based model. In 5.8, we plot the random effect for the two models against each other. We observe that the random effects of these models are highly correlated. In particular, we see that the random effects that are positive (negative) in the shot-based model becomes more positive (negative) in trajectory-based model. This shows that on an average a good shooter is even better optimal trajectory shooter. This helps in better ranking of the players. We also notice an increase in posterior standard deviation of the random effects in BLSTM based model for both offensive and defensive abilities as shown in Table 5.3. This also points to the fact that the BLSTM based model is better at differentiating the players (when the standard deviation is zero, all players are equally good). For sanity check, we also

Table 5.4: Standard Deviation of Nested Random Effects and Random Effects corresponding to 5.3 and 5.4

Model	σ_α	σ_β	$\sigma_{\overline{\alpha}}$	$\sigma_{\overline{\beta}}$
Nested BLSTM Model	0.1826	0.0758	0.1146	0.1367
Nested Shot Model	0.1143	0.0267	0	0

Table 5.5: Top Shooter Comparison

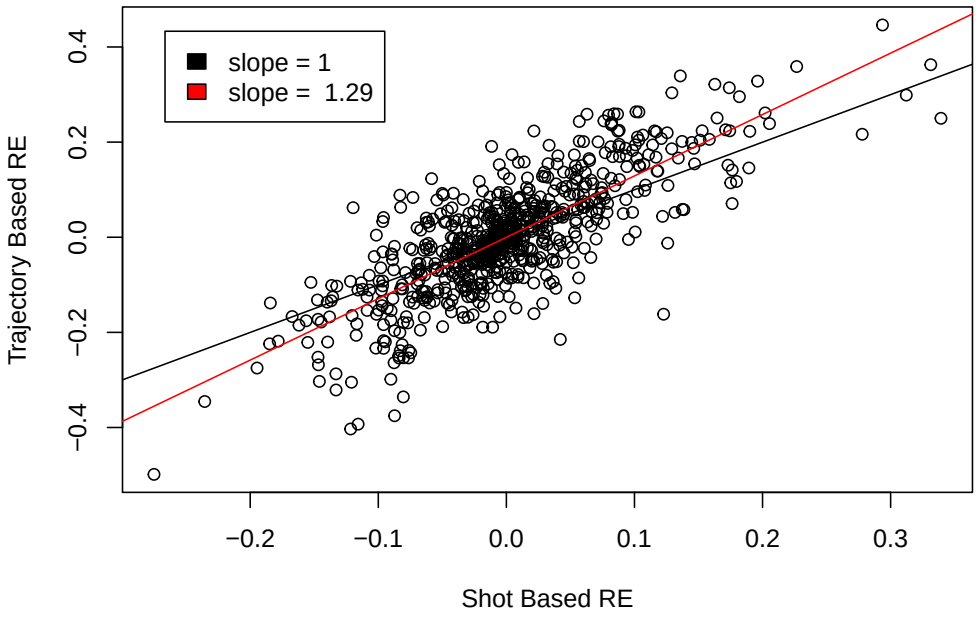
Shot Model	BLSTM Model	LSTM+Misc Model	Misc Model
Jared Dudley	Kyle Korver	Kyle Korver	Klay Thompson
Stephen Curry	Stephen Curry	Kevin Durant	Jeremy Evans
Chris Paul	J. J. Redick	J.J. Redick	Shaun Livingston
Kyle Korver	Kevin Durant	C.J. McCollum	Ed Davis
Jose Calderon	Patty Mills	Mo Williams	Shawne Williams
J.J. Redick	C. J. McCollum	Meyers Leonard	Miles Plumlee
Dirk Nowitzki	Klay Thompson	Patty Mills	Ray Allen
Anthony Tolliver	Steve Novak	Chris Paul	Bojan Bogdanovic
Patty Mills	Chris Paul	Wesley Matthews	Nikola Mirotic
Jason Smith	Wesley Matthews	Steve Novak	Thabo Sefolosha

Table 5.6: Top Shot Defender Comparison

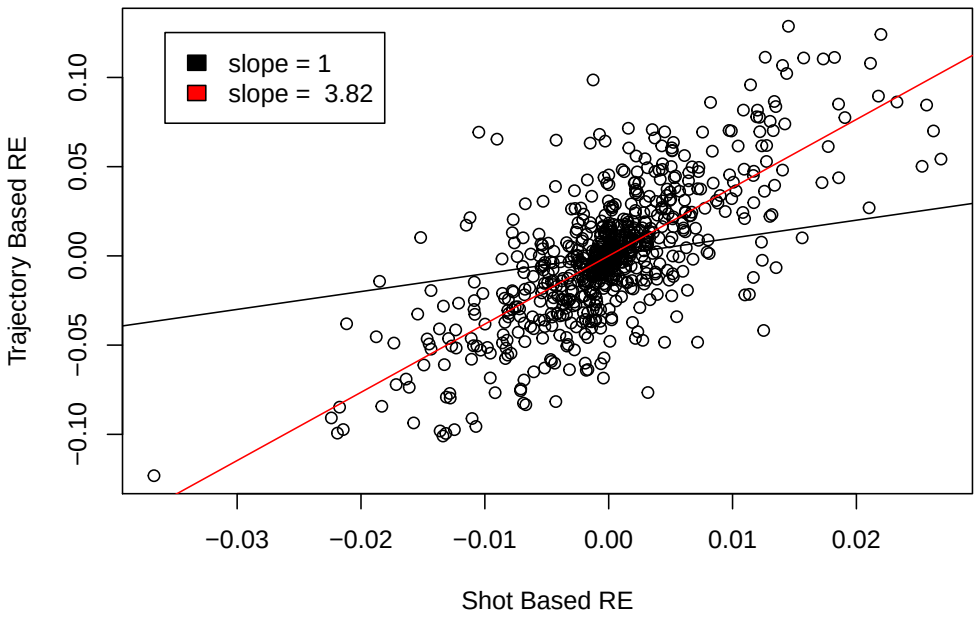
Shot Model	BLSTM Model	LSTM+Misc Model	Misc Model
Timofey Mozgov	Tiago Splitter	David Lee	Draymond Green
Paul George	Luc Mbah a Moute	P.J. Tucker	Frank Kaminsky
Stephen Curry	Alan Anderson	Chris Paul	Russell Westbrook
Tristan Thompson	Kawhi Leonard	Tiago Splitter	Lance Stephenson
Robin Lopez	Draymond Green	Draymond Green	Nikola Vucevic
Luc Richard Mbah a Moute	David Lee	Klay Thompson	Kawhi Leonard
Kendrick Perkins	Kosta Koufos	Luc Mbah a Moute	Jrue Holiday
Kosta Koufos	Gerald Green	Kawhi Leonard	Markel Brown
Kyle Korver	P.J. Tucker	Stephen Curry	Chris Johnson
Brook Lopez	DeAndre Jordan	Danny Green	Derrick Favors

compare the second best trajectory model (LSTM+Misc) and the most basic model (just using miscellaneous data) to the averaged BLSTM model. We notice that the posterior standard deviation decrease as the predictability of the trajectory models decrease. This shows that the BLSTM model, which contains the most accurate information about the trajectories, gives the best posterior standard deviation.

In Table 5.5, we compare the offensive ranking of the players we get using different models. BLSTM based model ranks more shooters who are considered to be elite shooters in the NBA in the top 10 e.g. Kyle Korver, Steph Curry, Kevin Durant, Klay Thompson as compared to what the shot-based model suggests. The shot-based model ranks players such as Jared Dudley, Jason Smith, Jose Calderon, and Anthony Tolliver are among the top 10 shooters in the league. While these players may be good shooters, they are not necessarily considered to be top tier shooters by many experts, i.e. ranking them as top shooters may be controversial. That said, it is clearly an error to choose them over players like Stephen Curry and Klay Thompson. LSTM+Misc model has many players that appear in BLSTM model as well but the disappearance of elite shooter like Stephen Curry from



(a) Offensive Random effects



(b) Defensive Random effects

Figure 5.8: Comparison of random effects learned using the two models. The x-axis and the y-axis corresponds to the shot-based model and trajectory-based model respectively

top ten indicates that the BLSTM model results in a better ranking. Also, Misc Model ranking is clearly less plausible, compared to all other models.

Regarding learning defensive ability of players, the results from both the shot-based model and BLSTM model are similar(see Table 5.6). That said, the shot-based model ranking Stephen Curry among the top defenders is a conclusion that many experts would dispute. Also, only the trajectory-based models include the elite defender Draymond Green within top 10 shot defenders. One thing to note is that we have a mix of good rim protectors, such as DeAndre Jordan and good perimeter defenders, such as Kawhi Leonard in BLSTM Model. Alan Anderson, who is also known for his defensive ability, appears in BLSTM list of top defenders.

Another thing to note is that the results from both the shot-based and BLSTM model do include elite defenders such as LeBron James. Note that this analysis is on the effect of defenders on shot outcomes of attempted field goals. We need to mention that offensive players may choose not to shoot when defended by elite defenders or good shot blockers. Evaluating this defensive ability requires a different modeling approach.

Finally, we analyze the effect of nested random effects model. We find that for the shot based model (5.4), the posterior standard deviation of the game level effects are zero. This means that the shots do not have enough information to capture the change in the player's ability across games. On the other hand, for the BLSTM model, we find that the game level deviation in player abilities captured by α_{ig} and β_{jg} have significant posterior standard deviation (see Table 5.4). We also notice an increase in the posterior deviation of the player level posterior standard deviation (0.1728 to 0.1826 for offensive and 0.0746 to 0.0758 for defensive abilities) when using a nested random effects model (5.3) vs a simple random effects model (5.2).

5.5 Conclusion

We proposed a novel way to evaluate players in NBA using the trajectory data of shot. Previous research in this area have ignored this extra information of ball trajectory. We show that using the ball trajectory data, not only are we able to get a better ranking of the players but also get a clear distinction of player's abilities. We are also able to measure player's ability on a game-by-game basis, something that a shot based model does not have enough information to do.

Chapter 6

Defensive Effort Ranking

"The players are just so quick in the NBA," Sterner says. "One or two feet can make a huge difference."

– Sterner, a Raptors assistant and something of a tech guru.5

6.1 Introduction

The "pesky" defense is an NBA jargon for describing a relentless defender. It is about being all over the ball handler, pressuring him and reaching in and slapping the ball, and contesting the shots. The effort of this sort is not captured in any contemporary basketball statistics. A good defensive player prevents opponents from making a shot, attempting a good shot, making an easy pass, or other scoring events, eventually leading to wasted shot clock time. The salient feature here is that a good defender prevents events. Consequently, event driven metrics, such as box scores, shot outcomes, points scored etc. cannot measure defensive abilities effectively. Defensive Rating (DRTG) and Defensive Win Share (DWS) are two advanced statistics used by the NBA to capture the defensive contribution of a player. Both these metrics are based on discrete events, and are heavily infuenced by the teammates. They are averaged over large number of games and do not give any statistically significant insight on a player's defensive ability on a game by game basis. There are other more advanced recent metrics to estimate defensive ability of players ([Franks *et al.*, 2015],

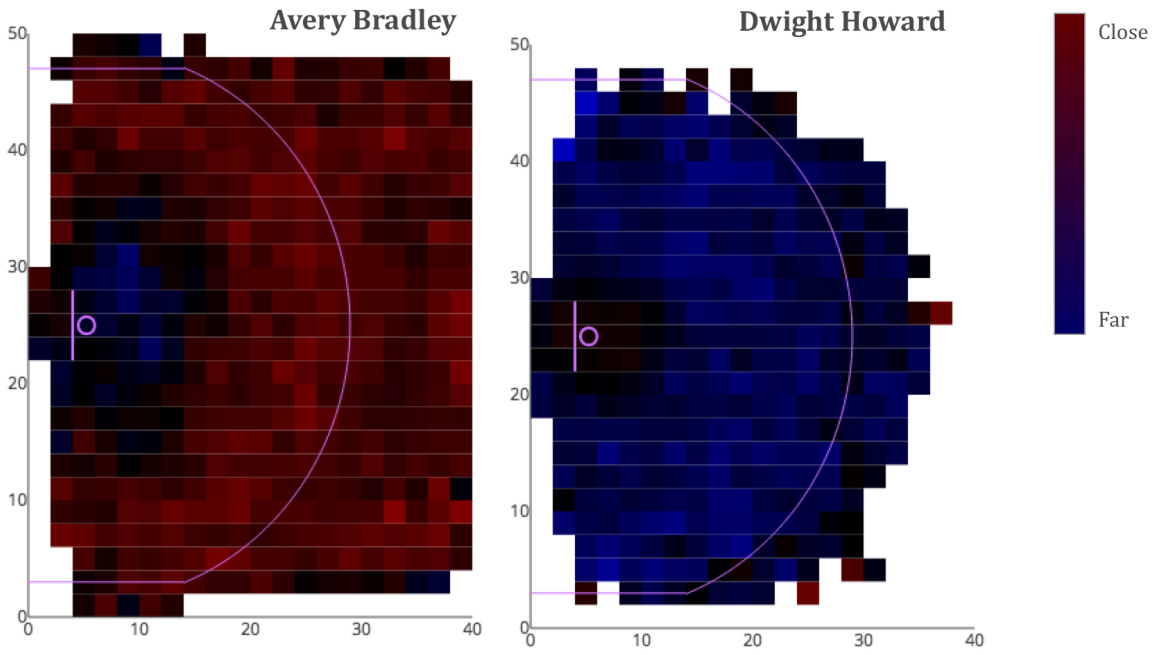


Figure 6.1: The heat map of average distance maintained by Avery Bradley and Dwight Howard with their defensive assignment as an on-ball defender across the court.

[Oh *et al.*, 2015]); however these are all based on event driven data.

Defense is a continuous process that cannot be summarized by discrete events. Conventional wisdom in basketball is that "pesky" defenders continuously maintain a close distance to the ball handler. A closely guarded offensive player is less likely to take or make a shot, less likely to pass, and more likely to lose the ball; hence, offensive players look for open spaces. [Franks *et al.*, 2015] show that the effect of defender's distance on the log odds of making a shot varies linearly between 0-6 ft for all the court locations. However, the analysis of the distance maintained by a defender with the ball handler is missing in recent literature. This paper aims to fill this gap. We introduce Defensive Effectiveness Rating (DER), to measure the effective distance a defender maintains with the ball handler, i.e. the offensive player with the ball. Consistently maintaining a close distance leads to an effective defense. In Figure 6.1, we display the average distance from the ball handler maintained by Avery Bradley, one of the best defenders in basketball, and Dwight Howard,

an average defender. Avery Bradley maintains a much closer distance with the ball handler. We also notice that the average distance changes over the court locations. Avery Bradley is a guard and he defends the ball handler very closely for almost all court locations except the center. Dwight Howard, who is a center/forward, maintains a relatively closer distance near the center compared to rest of the court. Clearly, defenders vary in how closely they guard players closely in different zones of the court. Further, players with longer wingspan are defensively more effective. Our proposed DER metric corrects for the defensive player's position, wingspan, and the ball handler's attribute.

6.2 Data

Our data consist of SportVU data from the 2012-2016 NBA regular and postseason. For the purpose of our analysis, we only consider the part of on-ball defense that happens in 50-by-40 feet part of the half court. Further, we only consider the data points when all the offensive players are in the half court. This makes sure that we only use the data after offense has been setup.

6.3 A Benchmark Model

We introduce a benchmark ranking that captures the overall defensive contribution of a player. This ranking will help us evaluate the defensive effort rankings we get using various models. We use a model which accounts for the overall contribution of each player in a possession using an additive linear model of players' abilities [Fearnhead and Taylor, 2011]. In particular, we define

$$\overline{P}_{L_i,L_j} = \sum_{i \in L_i} \gamma_i - \sum_{j \in L_j} \beta_j + \epsilon \tag{6.1}$$

Here, $P_{L_i,L_j}$is the average point per possession scored by lineup L_i when playing against lineup L_j. γ_i is the overall offensive contribution rating (OCR) of a player, β_j is his overall defensive contribution rating (ODR), and ϵ is the error term. In this model, we do not take into account any player level detail. For instance, we do not consider the ball handler, the shooter, the defender etc. This model gives equal weight to all the players in the lineup for

Table 6.1: Players with top 10 Overall Offensive/Defensive Rating

OOR	ODR
LeBron James	Kevin Garnett
Russell Westbrook	Tiago Splitter
Klay Thompson	Draymond Green
Stephen Curry	Mario Chalmers
Chris Paul	Tony Snell
James Harden	Tony Allen
Kevin Durant	Alex Len
Kevin Love	Nick Collison
Tristan Thompson	Danny Green
J.J. Redick	Andre Iguodala

increasing or reducing the point differential against the opponent lineup. A player with a high β_j reduces the average number of points scored by the opponent lineup per possession. On the other hand, a player with a high γ_i increases the average number of points scored by the lineup he is playing in. We benchmark the rankings we get from our defensive effort models against ODR. Table 6.1 shows top ten players based on OCR and ODR. We see many elite offensive and defensive players in the list. Thus, ODR is a good benchmark for our later results.

6.4 Model

DER is a metric based on the distance a defender maintains with the ball handler. Note that the metric can be computed for an off-ball defense as well, but the defenders are typically focused on putting their best effort to guard a ball handler. The raw tracking data gives us the location of all the players and the ball. The ball handler is the player in the offensive lineup who is closest to the ball. To compute this distance between the defender and the ball handler, we need to first learn the defensive assignment for a possession in the game. We use the Gravity + BEAT model introduced in Chapter 3. Once we know the defensive

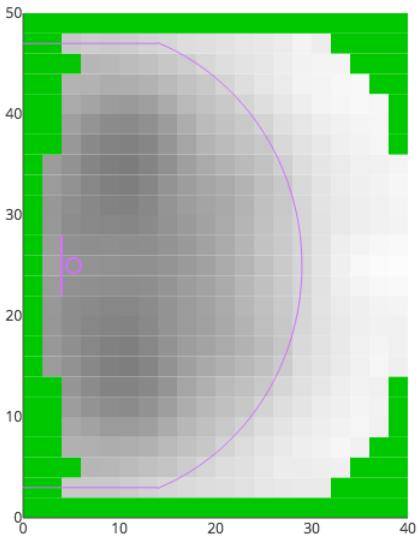


Figure 6.2: The heatmap of average distance maintained by a onball defender with the ball handler.

assignment for a possession, we can compute the distance between the ball handler and the defender at any moment. For a spatial analysis, we divide the court into $2ft \times 2ft$ square bins. Let d_{pk} be the distance between the on-ball defender p and the ball handler when the ball handler is in bin k. Let $\mathrm{E}[d_{pk}]$ be the average distance maintained by defender p with any ball handler when the ball handler is in bin k. To compute an empirical estimate of $\mathrm{E}[d_{pk}]$, we simply average over the distance at all the time points when the defender p is an on-ball defender and the ball handler is in bin k. Then, the mean value of $\mathrm{E}[\bar{d}_p]$ over k, is a simple estimate of defensive effort of defender p. However, this metric is flawed as a basis of comparison between players. To understand this, we visualize the variation of $\mathrm{E}[d_{pk}]$ over the court. We compute $\mathrm{E}[\bar{d}_k]$, which is the average distance maintained by any defender when the ball handler is in bin k. Figure 6.2 shows the value of $\mathrm{E}[\bar{d}_k]$ over the court. We see that the average distance maintained by a defender decreases as we approach the basket. This observation is in keeping with out intuitive understanding of defense: as the distance to the basket goes down, the defender is likely to guard the ball handler closely because the

probability of making a shot is higher. Thus, if we use $\mathrm{E}[\bar{d}_p]$ to compare players, it would favor the defenders who mostly defend closer to the basket since defenders closer to the basket naturally defend the ball handler closely. This does not reflect any "extra effort" put up a player. A more intuitive basis for comparison would be to define a metric that normalizes for the court location variation and compares all position players fairly.

6.4.1 Classifying defensive players

We want to compare the defensive players based on their location. It would make little sense to compare the defensive abilities of a center player with a perimeter defender. Also, comparing defenders based on the position that they play would allow the teams to use the ranking in a more effective manner. For instance, a team might choose to play one center player over another center player, but they would not choose to play a center at a three point location even though they have a slightly higher ranking at that location compared to a shooting guard. We develop a data driven approach to classify the defenders. To do this, we consider the distribution of the location of the ball handler when a given defender is defending him. Note that we do not use the location of the defender himself for classification. We classify the defenders based on the location of the on-ball handlers because if two defenders usually guard ball handlers in a certain part of the court, the team has a choice to choose one over the other based on the performance.

Let P denote the total number of players and K denote the number of square bins . Let F be a $P \times K$ matrix with the value at (p, k) index denoting the total number of times the defender p defended a ball handler in bin k. We use Non-Negative Matrix Factorization [Lee and Seung, 2001] to decompose the matrix F as follows:

$$F = WH, \tag{6.2}$$

where W is a $P \times M$ matrix and H is a $M \times K$ matrix. We choose $M = 3$ based on the error rate curve as shown in Figure 6.3. We normalize each column of H to sum up to 1. We interpret each row of H as a basis or a "zone" of the court, with each element of the row as the loading of each bin on the court in the given basis. We rescale each row of

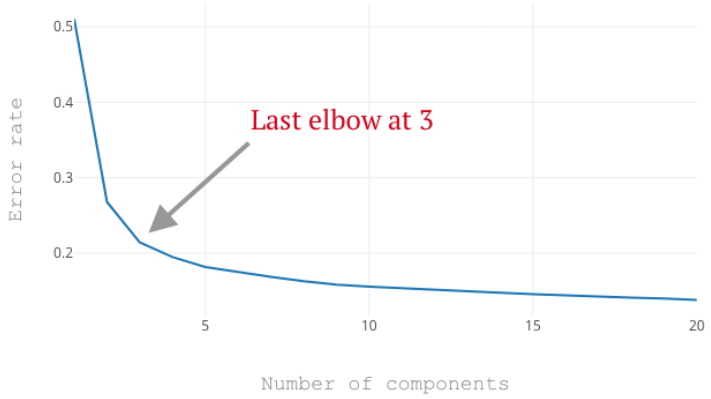


Figure 6.3: The RMSE vs M for the NMF algorithm in equation 6.2

Table 6.2: Basis Statistics of d

Basis	$\bar{d}_b$	$\sigma^b_{\bar{d}}$
Near Hoop	3.275	0.135
Midrange	4.588	0.401
Three Pointers	5.380	0.311

estimated W by sum of corresponding column of estimated H. Each row of W is then the count assigned to each basis for each player. The top row of Figure 6.4 shows the actual basis weights corresponding to the rows of H matrix. These bases align very well with our understanding of basketball positions: Center, Forward, and Guard. Next, we assign each bin to the basis with the highest weight to obtain a "binary" version of the bases. The bottom row of Figure 6.4 displays the binary basis. Thus, we have divided the court into three different "zones" that corresponds to the defense tendency in the NBA.

6.4.2 Defensive Effort

Our analysis lends itself well to evaluate players defense in different zones identified in Figure 6.4. To evaluate the effort of a player in a zone, we calculate the average distance the player maintains with the assigned ball handler when the ball handler is in the zone. This allows

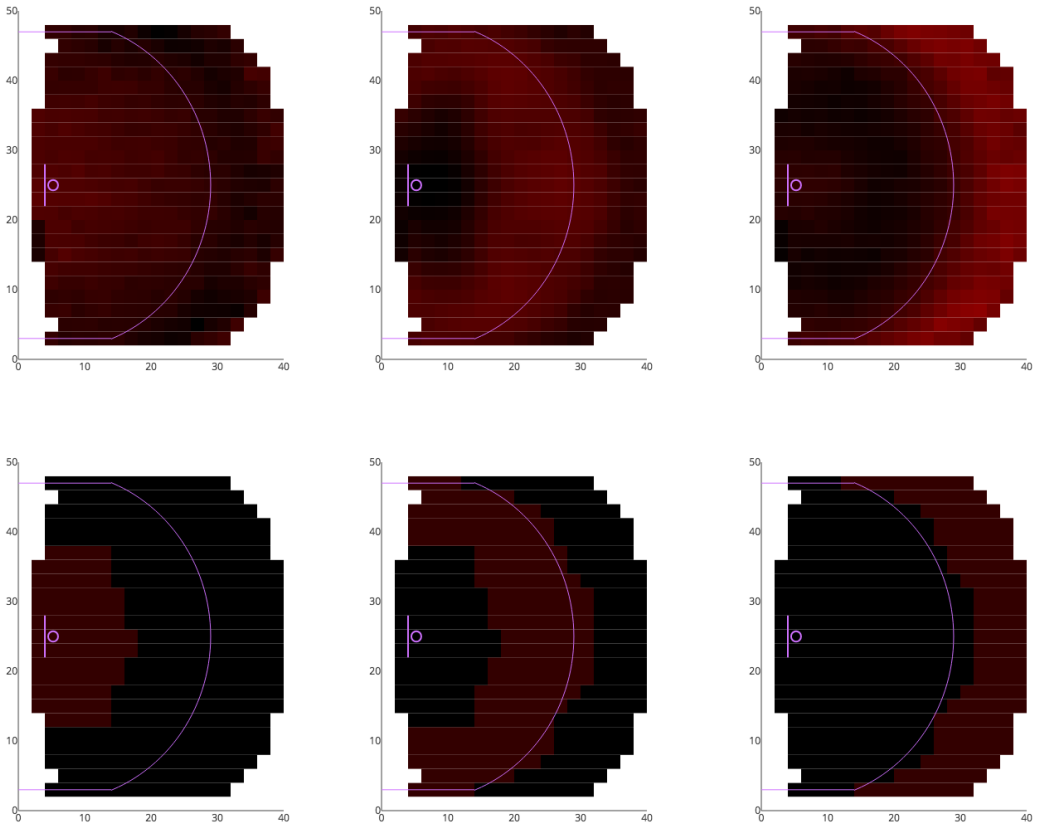


Figure 6.4: Basis loading corresponding to three basis selected by NMF algorithm: Top Left: Center basis, Top Middle: Mid range basis, Top Right: Three point basis. The bottom row shows the binary assignment of a bin to a basis with the highest weight.

us to evaluate the defense effort of each player in each zone separately and compare their ranking in the respective zones. We define d_{pb} as the defensive effort of player p in zone b using the weighted average of $\bar{d}_{pk}$, the average distance between the defender p and the ball handler when the ball handler is in bin k . The estimate of d_{pb}, denoted by $\bar{d}_{pb}$, is computed using:

$$\bar{d}_{pb} = \frac{\sum_k \bar{d}_{pk} N_{pk} I_{kb}}{\sum_k N_{pk} I_{kb}} \tag{6.3}$$

where I_{kb} is the indicator variable for basis k belonging to zone b and N_{pk} is the number of data points observed for defender p guarding the ball handler in bin k.

In Table 6.2, we display $\bar{d}_b$, the average of $\bar{d}_{pb}$ across all players for each basis, and their standard deviation $\sigma^b_{\bar{d}}$. We see that the average distance and its standard deviation varies significantly by the basis. To rank the players, we choose measure $\Delta\bar{d}_{pb}$ shown below, which normalizes for the basis average and standard deviation. It is also independent of the chosen unit of measurement.

$$\Delta\bar{d}_{pb} = \frac{\bar{d}_{pb} - \bar{d}_b}{\sigma^b_{\bar{d}}} \tag{6.4}$$

We call this statistic DEPM (Defense Effort Plus Minus). A positive value represents below average defensive effort since the player maintains a larger than average distance maintained by players in that zone. On the other hand, a negative value represents an above average defensive effort. To find out how well the DEPM rankings align with the ODR, we took the weighted average DEPM across the three zones for each player computed using:

$$\Delta\overline{d}_p = \sum_b W_{pb} \Delta\overline{d}_{pb} \tag{6.5}$$

where W_{pb} is the basis weight assigned to player p for basis b using the estimated W in 6.2. We compute both Pearson and Spearman correlation of $\Delta\overline{d}_p$ with ODR. Spearman correlation gives us the rank correlation of the two rankings whereas Pearson correlation is a linear correlation. We found a low correlation value of 0.1 and 0.08 for Pearson and

Spearman respectively. When we ranked the players using this metric, we observed that elite defenders, like Avery Bradley, Robert Covington, Jimmy Butler, and Andre Roberson, show up in top rankings. However, we also found that players like Isaiah Cannan, Toney Douglas, Ish Smith and C.J. McCollum show up high in the ranking. These players might put a good defensive effort, but they are not necessarily considered elite defenders. After delving deeper, we found out that these players have much smaller wingspans compared to the average wingspan in the NBA. While guarding closely reflects contribution towards reducing the shot efficiency of players, the wingspan plays a direct role in the effectiveness of the defense. A player who has a relatively shorter wingspan, while guarding very closely, may not be as effective.

6.4.3 Wingspan Effect

Wingspan is the term used to describe the length of a basketball player's arms and hands. It is a conventional wisdom in the NBA that players with large wingspans are very effective in defense. A large wingspan is particularly helpful in blocking shots, rebounding, reaching into passing lanes for steals etc. Fortunately, the wingspan data of players is available from `NBA.com`. We create a new statistics called EDEPM (Effective Defensive Effort Plus Minus) which is the rescaled version of DEPM (see 6.4) taking the wingspan of player into account.

$$\bar{E}_{pk} = \frac{\bar{d}_{pk}}{w_p}$$
$$\Delta\bar{E}_{pb} = \frac{\Delta\bar{d}_{pb}}{w_p}$$

This rescaling allow a defender with large wingspan to have a smaller effective distance with the ball handler. We compute the correlations for this model with ODR using the weighted EDEPM across all basis (see 6.5). We observe a significant improvement in both Pearson correlation (0.158) and Spearman correlation (0.147). We also observe some very positive change in the rankings over DEPM model as shown in Table 6.3. Elite defenders like Kawhi Loenard, Anthony Davis, and Nene improve their ranking significantly. We also observe that the rankings of players with shorter wingspan like Toney Douglas, C.J. McCollum, and

Table 6.3: Notable Change in Rankings in EDEPM over DEPM

Player Name	Position	DEPM Rank	EDEPM Rank
Anthony Davis	Near-hoop	39	2
Luc Richard Mbah a Moute	Near-hoop	8	3
Paul George	Far-three	28	16
Kawhi Leonard	Far-three	44	9
Nene	Near-hoop	37	4
Larry Sanders	Near-hoop	83	9
Trevor Ariza	Midrange	115	25
C.J. McCollum*	Far-three	6	17
Ish Smith*	Far-three	4	28
Toney Douglas*	Far-three	10	42

Ish Smith, who were in top rankings in DEPM model, dropped to below ten.

6.4.4 Accouting for ball handlers

EDEPM model has a limitation. It does not account the identity of the ball handler. Players tend to guard good shooters more closely than a bad shooter. If a ball handler shoots very well from a basis, it is necessary to defend him closely on that basis to reduce the chances of the shot going in. On the other hand, if a ball handler does not shoot well from a basis, it makes more sense to guard him from a larger distance to reduce the chances of him taking a shot from a point closer to the hoop. To control for the ball handler's shooting ability, we use a random effects based model (see [Bates *et al.*, 2014]):

$$\begin{aligned} \bar{E}_{pqk} &= c_b + \alpha_{pb} + a_b\beta_{qb} + \epsilon_{pqk} \\ \alpha_{pb} &\sim \mathcal{N}\left(0, \sigma_\alpha^2\right) \\ \epsilon_{pqk} &\sim \mathcal{N}\left(0, \sigma^2\right) \end{aligned} \tag{6.6}$$

where $\bar{E}_{pqk}$ is the wingspan normalized average distance that player p maintains when defending ball handler q at bin k, α_{pb} is the ability of defender p to maintain a close distance

with a ball handler in basis b, β_{qb} is the estimate of the shooting efficiency of the ball handler q in basis b, a_b is the fixed effect term which measure the effect of shot efficiency of the ball handler on the effective distance that the defender maintains. A negative estimate of a_b would point to the effect of a good shooter guarded more closely in basis b. We call $\alpha_{pb}/\sigma_\alpha$ the Defensive Efficiency Rating (DER) of a player p in basis b. We estimate β_{qb} using the following model for basis based shot efficiency model [1]

$$\begin{aligned} S_{ijb} &\sim \sigma\left(c_b + x_{ib} - y_{jb}\right) \\ x_{ib} &\sim \mathcal{N}\left(0, \sigma_x^2\right) \\ y_{ib} &\sim \mathcal{N}\left(0, \sigma_y^2\right) \end{aligned} \tag{6.7}$$

S_{ijb} is an indicator of the shot outcome when it is attempted from a point on basis b determined above, x_{ib} is the random effect corresponding to the ability of the offensive player i to make a shot when attempting it from basis b, y_{jb} is the ability of the defensive player j to prevent a positive shot outcome when the ball handlers he is guarding attempts a shot from basis b. We set β_{qb} to be $\exp(\hat{x}_{qb})$ where $\hat{x}_{qb}$ is the estimated value of x_{qb} to make it positive, a more intuitive input for our model. The estimates of the a_b parameter (and its standard deviation) in the model described in (6.6) are shown in Table 6.4. We notice that the players with higher shot efficiency are guarded closely in every basis. The effect is more pronounced as we move farther from the basket. This explains the fact the near the basket, since the probability of making a shot is high across all the players, the players are guarded closely regardless of their shot efficiency. Good shooters are guarded much more closely compared to an average shooter when they are far from the basket. This is explained by higher negative value of a_b for midrange and three point basis.

6.5 Inference

We use

[1] We choose to use the shot efficiency model instead of a trajectory based model. The reason is based on the result from Figure 7 in [Franks *et al.*, 2015], which shows that the shot efficiency varies nearly linearly as a function of the distance of the guarding defender from the shooter. This aligns with our formulation in 6.6 which assumes a linear relationship between shot efficiency and $\bar{E}_{pqk}$.

Table 6.4: Effect of ball handler's shot efficiency on EDEPM

Basis	a_b	Error
Near Hoop	-0.010	0.005
Midrange	-0.132	0.004
Three Pointers	-0.166	0.006

6.6 Results

Let us summarize the models we have discussed and how they compare with the result of the DER model. To estimate the defensive effectiveness of an on-ball defender, we started with simply computing the average distance the defender maintains with a ball handler over the court. As shown in Figure 6.2, we found that this distance varies over the court. A defender who frequently defends a ball handler close to the hoop naturally maintains a close distance as shown by the statistics in Table 6.2. This motivated the decision of dividing the court into different zones so that we can compare the distance maintained by defenders in each zone separately. We use NMF to detect three zones in the court as shown in Figure 6.4. Next, we computed the DEPM, which gives us the ranking of defenders based of the deviation from the average distance and normalized by the standard deviation. However, we noticed in the rankings that a few players who are not necessarily considered great defenders show up high in the rankings. We identified that the wingspan of a defender plays a big role in his defensive effectiveness. This helped us improve our model by taking wingspan into account as shown in Section 6.4.3. We notice that this led to a significant improvement in rankings across different basis as shown by Table 6.3. Next, in Section 6.4.4, we argued the importance of the shot efficiency of a ball handler in the distance that a defender maintains with him. We improved our model by taking the the shot efficiency of the ball handler in different basis into account and defining a random effects based model to compute the Defensive Effectiveness Rating (DER) metric of a player.

Table 6.5 shows the improvement in correlation of the overall ranking we get from various models with the ODR ranking. We see a remarkable improvement in the correlation of the

Table 6.5: Correlation with ODR

Model	Pearson Correlation (value, p-value)	Spearman Correlation (value, p-value)
DEPM	(0.100,0.080)	(0.080,0.120)
EDEPM	(0.159,0.020)	(0.147,0.030)
DER	(0.230, 0.001)	(0.197, 0.003)

Table 6.6: DER rankings for different basis

Near center basis	Midrange basis	Three pointer basis
Jimmy Butler	Andre Roberson	Robert Covington
Luc Richard Mbah a Moute	Avery Bradley	Avery Bradley
Anthony Davis	Robert Covington	K.J. McDaniels
Tristan Thompson	Kawhi Leonard	Tony Snell
Nene	Al-Farouq Aminu	Markel Brown
Draymond Green	Tony Snell	Marcus Smart
Larry Sanders	Markel Brown	Andre Roberson
Timofey Mozgov	Tony Allen	Kawhi Leonard
Anthony Tolliver	Nene	Tyler Ennis
Derrick Favors	Kentavious Cladwell-Pope	DeMarre Carroll

DER ranking from 0.158 to 0.230 for Pearson and 0.147 to 0.197 for Spearman. This is a significant validation for our model: DER does not assume anything about the discrete events like shot outcome, blocks, assists etc. that have been used to define the defensive rankings. Still we get a very significant correlation (with p value 0.001 and 0.003) with ODR which is a point score based defensive rating metric.

Table 6.6 shows the rankings of top defenders in different basis based on DER. In all the zones, we see some elite defenders. Top defenders in the near center basis are mostly different from the Midrange and Three pointer basis. Defenders like Draymond Green,

Nene, Larry Sanders, Anthony Davis, Moute, Jimmy Butler are considered as some of the best shot blockers near the hoop. Other defenders are also considered very good. We do not see a defender that someone can reject as a good defender. Same holds true for the midrange and three pointer basis. We also see an overlap in the top defenders in the midrange and three pointer basis. This makes sense since the point guards do defend players in the both zones. For instance, the heatmap of Avery Bradley in figure 1 shows a close distance maintained by him in both midrange and three pointers basis, but not in the near hoop basis. Defenders like Andre Roberson, Avery Bradley, Robert Covington and Kawhi Leonard are considered some of the peskiest defenders in the NBA. They are also considered outstanding perimeter defenders.

6.7 Conclusion

We see that the DER allows us to quantify the quality of defensive interaction without being limited by the occurrence of discrete and infrequent events. Thus, it allows us to measure the defensive effectiveness of a player in a game, which was previously not possible due to limited number of discrete events. A significant correlation with ODR ranking proves that the DER model's rankings do have significant similarity with the point based rankings. DER could help teams decide a good defender for each zones and help in their defensive player acquisitions. While we agree that DER is not a rating that would replace existing defensive ratings, we also believe that it is an important piece that deserves its place in evaluating the defensive contribution of the players in the NBA.

Bibliography

[Abdul-Jabbar, 2017] Kareem Abdul-Jabbar. The nba, and not the nfl, is the league of america's future, 2017. [Online; accessed March 1, 2019].

[Andrieu *et al.*, 2003] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An introduction to mcmc for machine learning. *Machine learning*, 50(1-2):5–43, 2003.

[Bates *et al.*,] Douglas Bates, Martin Maechler, Ben Bolker, Steven Walker, Rune Haubo Bojesen Christensen, Henrik Singmann, Bin Dai, Fabian Scheipl, and Gabor Grothendieck. Package 'lme4'.

[Bates *et al.*, 2014] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. Fitting linear mixed-effects models using lme4. *arXiv preprint arXiv:1406.5823*, 2014.

[Beuoy, 2015] Michael Beuoy. Introducing sharc: Shot arc analysis., 2015. [Online; accessed 12-May-2018].

[Bishop, 2006] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[Cervone *et al.*, 2016] Daniel Cervone, Alex D'Amour, Luke Bornn, and Kirk Goldsberry. A multiresolution stochastic process model for predicting basketball possession outcomes. *Journal of the American Statistical Association*, 111(514):585–599, 2016.

[Chang *et al.*, 2014] Yu-Han Chang, Rajiv Maheswaran, Jeff Su, Sheldon Kwok, Tal Levy, Adam Wexler, and Kevin Squire. Quantifying shot quality in the nba. In *Proceedings of the 8th Annual MIT Sloan Sports Analytics Conference. MIT, Boston, MA*, 2014.

[Fearnhead and Taylor, 2011] Paul Fearnhead and Benjamin Matthew Taylor. On estimating the ability of nba players. *Journal of quantitative analysis in sports*, 7(3), 2011.

[Fewell *et al.*, 2012] Jennifer H Fewell, Dieter Armbruster, John Ingraham, Alexander Petersen, and James S Waters. Basketball teams as strategic networks. *PloS one*, 7(11):e47445, 2012.

[Franks *et al.*, 2015] Alexander Franks, Andrew Miller, Luke Bornn, Kirk Goldsberry, et al. Characterizing the spatial structure of defensive skill in professional basketball. *The Annals of Applied Statistics*, 9(1):94–121, 2015.

[Gers *et al.*, 1999] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. 1999.

[Hansen, 1982] Lars Peter Hansen. Large sample properties of generalized method of moments estimators. *Econometrica: Journal of the Econometric Society*, pages 1029–1054, 1982.

[Himmelmann, 2010] Maintainer Lin Himmelmann. Package 'hmm'. 2010.

[Huang *et al.*, 2015] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[Keshri *et al.*,] Suraj Keshri, Min-hwan Oh, Sheng Zhang, and Garud Iyengar. Automatic event detection in basketball using hmm with energy based defensive assignment. *Journal of Quantitative Analysis in Sports*.

[Lee and Seung, 2001] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

[Lowe, 2013] Zach Lowe. Lights, cameras, revolution, 2013. [Online; accessed March 1, 2019].

[Marschner, 2011] Ian C. Marschner. glm2: Fitting generalized linear models with convergence problems. *The R Journal*, 3:12–15, 2011.

[McQueen *et al.*, 2014] Armand McQueen, Jenna Wiens, and John Guttag. Automatically recognizing on-ball screens. In *2014 MIT Sloan Sports Analytics Conference*, 2014.

[Miller and Bornn, 2017] Andrew C Miller and Luke Bornn. Possession sketches: Mapping nba strategies. In *MIT Sloan Sports Analytics Conference*, 2017.

[Miller *et al.*, 2014] Andrew Miller, Luke Bornn, Ryan Adams, and Kirk Goldsberry. Factorized point process intensities: A spatial analysis of professional basketball. In *International Conference on Machine Learning*, pages 235–243, 2014.

[myactivesg, 2016] myactivesg. The different types of basketball scoring shots and how to execute them. `"https://www.myactivesg.com/Sports/Basketball/Training-Methods/Basketball-for-Beginners/The-Different-Types-of-Basketball-Scoring-Shots"`, 2016. [Online; accessed February 10, 2018].

[Oh *et al.*, 2015] Min-hwan Oh, Suraj Keshri, and Garud Iyengar. Graphical model for baskeball match simulation. In *Proceddings of the 2015 MIT Sloan Sports Analytics Conference, Boston, MA, USA*, volume 2728, 2015.

[Pelton, 2014] Kevin Pelton. Explaining gravity in basketball, 2014. [Online; accessed 25-Aug-2018].

[Rasmussen, 2004] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

[Shah and Romijnders, 2016] Rajiv Shah and Rob Romijnders. Applying deep learning to basketball trajectories. *arXiv preprint arXiv:1608.03793*, 2016.

[Shirley, 2007] Kenny Shirley. A markov model for basketball. In *New England Symposium for Statistics in Sports*, 2007.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[Štrumbelj and Vračar, 2012] Erik Štrumbelj and Petar Vračar. Simulating a basketball match with a homogeneous markov model and forecasting the outcome. *International Journal of Forecasting*, 28(2):532–542, 2012.

[thehelpdefender, 2013] thehelpdefender. A video on youtube looking at the defensive work of...alananderson.(2013). `"https://www.youtube.com/watch?v=KBf_1LJFxCQ&t=15s"`, 2013. [Online; accessed February 10, 2018].

[Viterbi, 1967] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269, 1967.

[Wang and Zemel, 2016] Kuan-Chieh Wang and Richard Zemel. Classifying nba offensive plays using neural networks. In *Proceedings of MIT Sloan Sports Analytics Conference*, pages 1–9, 2016.

[Wright and Nocedal, 1999] Stephen Wright and Jorge Nocedal. Numerical optimization. *Springer Science*, 35(67-68):7, 1999.

[Zhao *et al.*, 2018] Yu Zhao, Rennong Yang, Guillaume Chevalier, Rajiv C Shah, and Rob Romijnders. Applying deep bidirectional lstm and mixture density network for basketball trajectory prediction. *Optik*, 158:266–272, 2018.

Appendices

.1 DERIVATION OF POSTERIOR DISTRIBUTION OF Γ

The data likelihood is

$$\begin{aligned}
&P\left(\boldsymbol{D}|\Gamma_p, \sigma_D^2\right) \\
&= \prod_{t,j,k} \left[P\left(D_{ti}|I_{tij}, \Gamma, \sigma_D^2\right) P\left(I_{tij}|I_{(t-1)i.}\right)\right]^{I_{tij}} \\
&\propto \prod_{k=1}^{K} \prod_{j=1}^{N_{pk}} e^{-\frac{1}{2\sigma^2}\left[Z_{kj}^T \Gamma_{pk} - D_{kj}\right]^T \left[Z_{kj}^T \Gamma_{pk} - D_{kj}\right]} \\
&\propto e^{-\frac{1}{2\sigma^2}\left(\sum_k \Gamma_{pk}^T W_k \Gamma_{pk} - 2\sum_k \Gamma_{pk}^T V_k\right)}
\end{aligned}$$

where $W_k = \sum_{j=1}^{N_{pk}} Z_{kj} Z_{kj}^T$ and $V_k = \sum_{j=1}^{N_{pk}} Z_{kj} D_{kj}$. Note that we no longer write the likelihood with respect to time. Only thing that time tells us is the assignment of defensive players to offensive players. Once we have estimated the sequence I, then we no longer have time dependence in our data. From the estimation perspective, it is more efficient to write the likelihood in terms of events.

Let $V = [V_1, ..., V_K]^\top$ and $\Gamma_p = [\Gamma_{p1}, ..., \Gamma_{pK}]^\top$. Also, we define W to be a block diagonal matrix with each block being W_k for $k = 1, ..., K$. Then, we can express the data likelihood as the following:

$$P\left(\boldsymbol{D}|\Gamma_p, \sigma_D^2\right) \propto e^{-\frac{1}{2\sigma^2}\left(\Gamma_p^T W \Gamma_p - 2\Gamma_p^T V\right)}$$

Using the GP prior on Γ_p mentioned above

$$\Gamma_p \sim GP(\mu_\Gamma, \mathcal{K}),$$

we can compute the posterior distribution

$$\begin{aligned}
&P(\Gamma_p \mid \boldsymbol{D}) \\
&\propto P\left(\boldsymbol{D}|\Gamma_p\right) P(\Gamma_p) \\
&\propto e^{-\frac{1}{2\sigma_D^2}\left(\Gamma_p^T W \Gamma_p - 2\Gamma_p^T V\right)} \cdot e^{-\frac{1}{2}(\Gamma_p - \mu_\Gamma)^T \mathcal{K}^{-1} (\Gamma_p - \mu_\Gamma)} \\
&\propto \exp\left(-\frac{1}{2}[\Gamma_p - \mu]^T \Sigma^{-1} [\Gamma_p - \mu]\right)
\end{aligned}$$

where $\mu = \left(\frac{W}{\sigma_D^2} + \mathcal{K}^{-1}\right)^{-1} \left(\frac{V}{\sigma_D^2} + \mathcal{K}^{-1}\mu_\Gamma\right)$ and $\Sigma = \left(\frac{W}{\sigma_D^2} + \mathcal{K}^{-1}\right)^{-1}$. Hence, the posterior

distribution is

$$\Gamma_p \mid \boldsymbol{D}, \boldsymbol{I}, \sigma_D^2 \sim N(\mu, \Sigma) \quad \text{with } \Gamma_{pk}^T \mathbf{1} = 1$$

.2 SAMPLING OF MULTIVARIATE GAUSSIAN DISTRIBUTION WITH LINEAR CONSTRAINTS

We want to simulate $z \sim N(\mu, \Sigma)$ conditioned on $Fz = v$. Without loss of generality, assume that $F \in \mathbb{R}^{m \times n}$ has full row rank. Define:

- $P = F^\top (FF^\top)^{-1} F$: projection onto $\mathcal{F} = \{F^\top v : v \in \mathbb{R}^m\}$
- $P^\perp = I - P$: projection onto the linear space orthogonal to $\mathcal{F}$

Then, the distribution of z conditioned on $Fz = v$ is given by

$$z \sim N\left(\bar{v} + P^\perp \mu, P^\perp \Sigma (P^\perp)^\top\right)$$

where $\bar{v} = F^\top (FF^\top)^{-1} v$