Using Deep Learning to Understand Patterns of Player Movement in Basketball

by

Akhil Nistala

Submitted to the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

Author: ____
Department of Electrical Engineering and Computer Science
May 24, 2018

Certified by: ____
John Guttag, Thesis Supervisor
May 24, 2018

Accepted by: ____
Katrina LaCurts, Chair, Master of Engineering Thesis Committee
May 24, 2018

Using Deep Learning to Understand Patterns of Player Movement in Basketball

by
Akhil Nistala

Submitted to the Department of Electrical Engineering and Computer Science
on May 24, 2018, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Computer Science and Engineering

Abstract

National Basketball Association (NBA) coaches spend a great deal of time analyzing the effectiveness of various strategies. Typically, this entails countless hours pouring over videos of games, and trying to derive generalizable conclusions from hundreds of thousands of examples. In this thesis, we present a methodology for quantitatively approaching this task. We start from player tracking data that records the position on the court of each player 25 times per second.

We use an unsupervised machine learning pipeline to learn a low-dimensional encoding for each player's movement, over one possession on offense. Each encoding captures the semantics of a single player's movement, such as locations of the endpoints, screen actions, court coverage, and other spatial features. We generate 3 million such *trajectory-embeddings* from 3 seasons of data. These can be clustered to reveal trends in player movement between sets of games.

Our framework can be used to answer such questions as "How did Klay Thompson's movements change between wins and losses during the 2016 NBA Finals?" (18% of his trajectories in wins were movements between the sidelines and corners, compared to 3.5% in losses) and "How much more frequently did Andre Drummond establish position on the right block than the left block during the 2015-2016 regular season?" (Almost 40% of his trajectories from 2015-2016 were right of the basket, compared to less than 15% to the left).

Thesis Supervisor: John Guttag
Title: Professor, Electrical Engineering and Computer Science

Acknowledgments

I would like to thank my thesis advisor, John Guttag, for all of the guidance and patience over the past year and a half.

I would like to thank my lab mate, Joel Brooks, for inspiring me to believe in myself and put forth my best effort.

And lastly, I would like to thank my amazing parents and sister for their love and support.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Motivation

Analytics have taken an increasingly important role in sports over the past decade. Many sports teams went from employing just one statistician to having entire analytics departments. Startups have also emerged as consultants in this space, attempting to help teams analyze data in new and innovative ways, with the hope of improving player performance and ultimately leading to more wins. Although analytics may not have the final say in a team's decisions, they are "always part of the conversation and internal debate" when front offices deliberate on how to assemble a winning roster, according to Celtics GM Danny Ainge [1].

NBA teams have historically relied on film analysts to filter through videos from previous games. In 2011, SportVU fundamentally changed the way that basketball can be analyzed. STATS SportVU utilizes a six-camera system installed in basketball arenas to track the real-time positions of players and the ball. This system has opened the door for a whole new era of data analysis in basketball.

1.2 Problem Description

Our goal was to develop a framework to quantitatively examine players' movements on offense, where a *movement on offense* describes a single player's movement over the course of one offensive possession. This process currently entails a film analyst watching hundreds of hours of game footage, carefully examining each possession and taking notes on certain plays. When a coach wants to look at a new play or movement, the entire process must be repeated. This approach is neither efficient (it takes hundreds of hours) nor comprehensive (film analysts don't retrieve *all* examples of a certain movement).

Some examples of the types of questions we want to be able to answer with our framework are:

- What fraction of Russell Westbrook's total movements from the 2013-2014 season were screen actions above the left wing?
- How did Kevin Love's movements on offense change between the 2013-2014 and 2014-2015 season, when he went from being the #1 option for the Timberwolves to the #3 option on the Cavaliers?
- Who are the players who scored from the left block most frequently during the 2015 playoffs?
- Which player has the most similar patterns of movement to LeBron James?

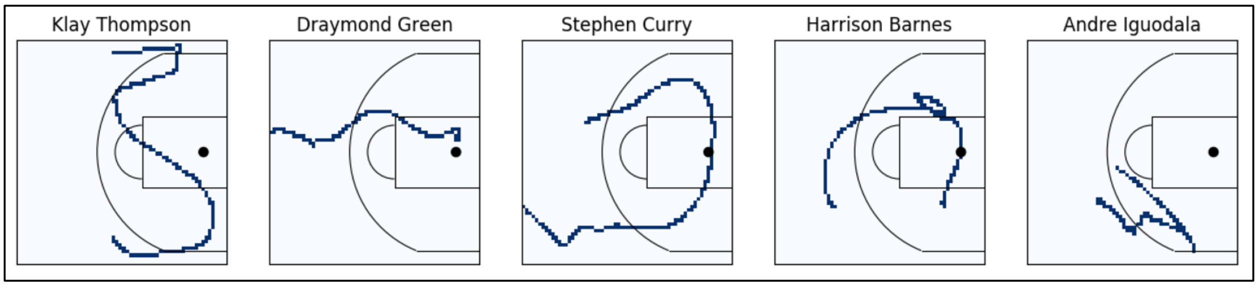
1.3 Proposed Solution

Basketball plays evolve as the offense and defense read and react, leading to a variety of possible movements on offense. However, certain types of players tend to move similarly and occupy the same court positions across many possessions. For example, Pistons' center Andre Drummond often posts up on the right block. As an excellent 3-point shooter, Kyle Korver of the Cleveland Cavaliers is often stationed on the perimeter. More versatile players such as LeBron James and Stephen Curry have a number of different patterns of movement in many areas on the court.

To approach the problem of quantitatively describing player movement, we first developed a way to build an image to capture each player's movement on offense. Each such *trajectory-image* captures a single player's movement on offense over the course of one possession. Figure 1.1 shows the five trajectory-images for the Warrior players for a single Warriors' possession during the 2016 NBA finals. These trajectory-images represent *all movement* for *all players on offense* for a single possession, not just when a player has the ball.

Figure 1.1

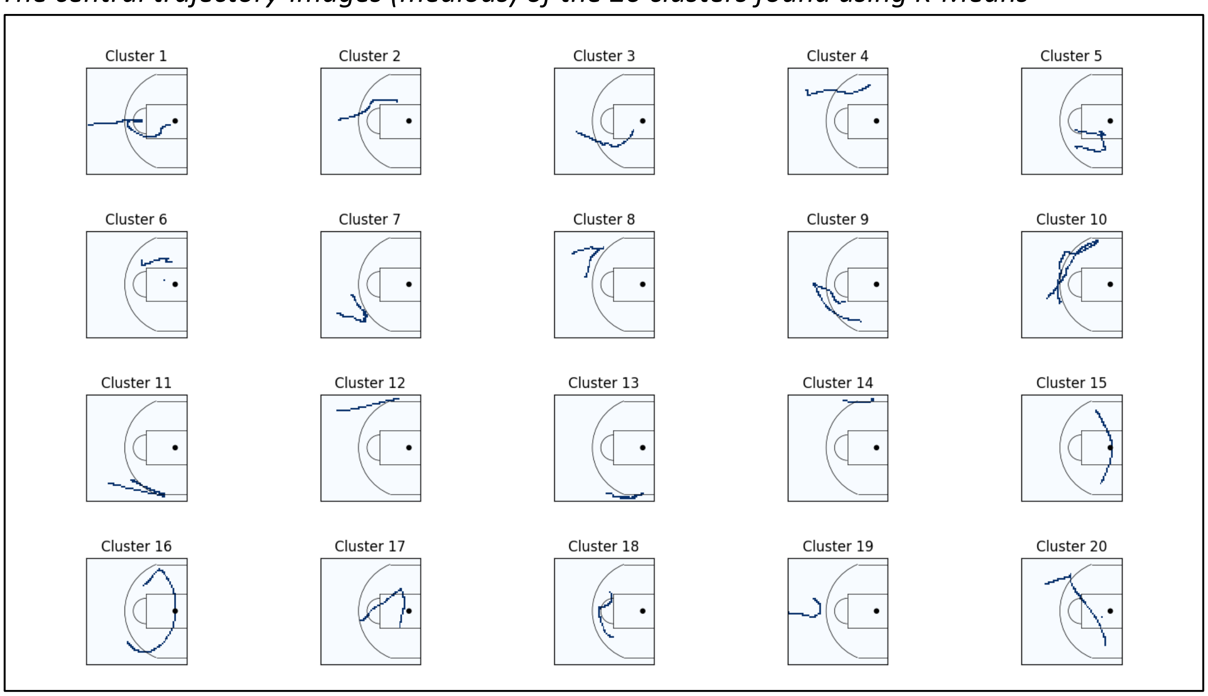
Five trajectory-images for a Warriors' offensive possession during the 2016 NBA Finals



We then used an autoencoder neural network to learn a low-dimensional encoding for each of these images. We ran K-Means clustering on the full set of encodings to identify a subset of trajectory-images that are representative of all movements on offense (Figure 1.2).

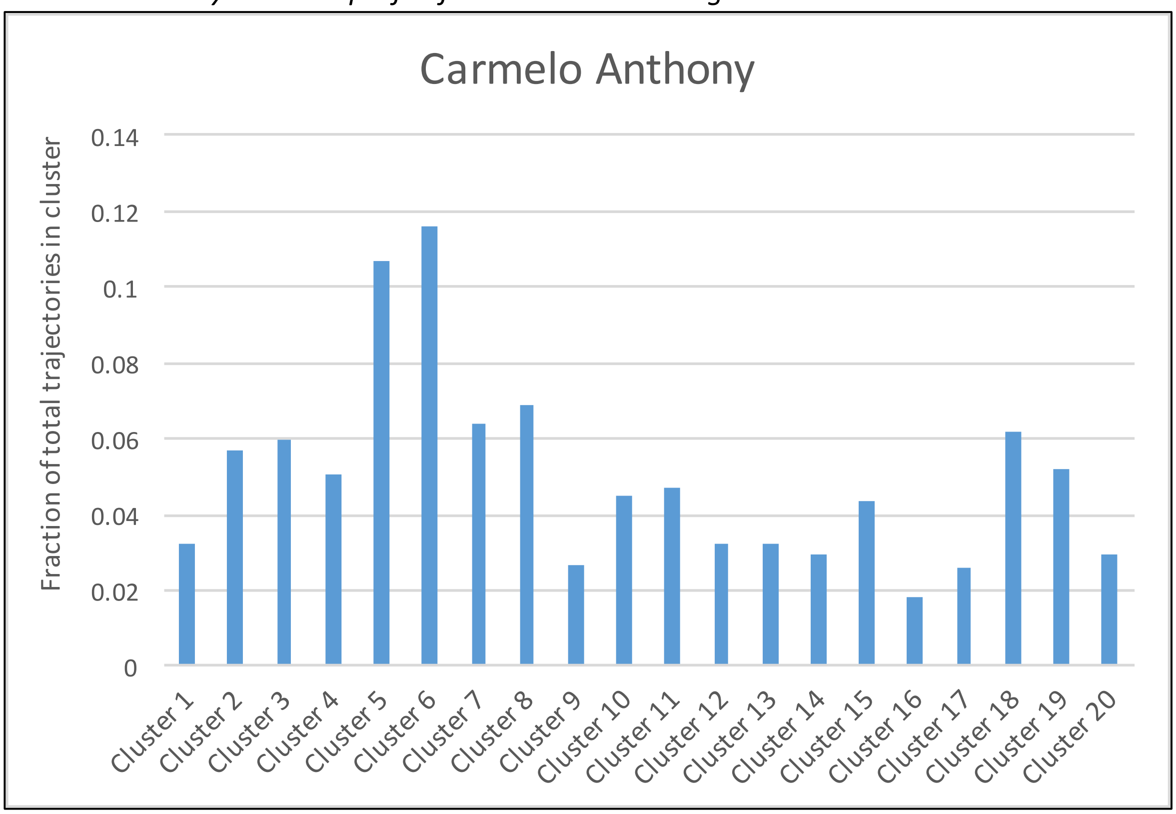
Figure 1.2

The central trajectory-images (mediods) of the 20 clusters found using K-Means



We proceed to construct *cluster-profiles* for each player. A cluster-profile summarizes a player's movements for a given set of possessions. Figure 1.3 shows Carmelo Anthony's cluster-profile for the 2013-2014 season. Each column contains the fraction of Anthony's total movements that follow a certain pattern. For example, clusters 5 and 6 – movements on either side of the paint – account for nearly 23% of Anthony's total movements. Clusters 13 and 14 – movements along either sideline to the 3-point corner, account for less than 7% of Anthony's total movements. Cluster-profiles can be used to answer the types of questions proposed in Section 1.2, and much more.

Figure 1.3
Carmelo Anthony's cluster-profile for the 2013-2014 regular season



1.4 Organization of Thesis

The remainder of this thesis is organized as follows. Chapter 2 introduces the SportVU and Basketball-Reference data. Chapter 3 goes through our methodology in full detail. Chapter 4 presents our experimental results. Chapter 5 contains our analysis and demonstrates why cluster-profiles are a useful tool for analyzing player movement. Chapter 6 concludes with a summary and discussion of future work.

2. Data

This chapter describes the data we were provided, and the techniques we used to filter possessions and build trajectory-images from raw time series data.

2.1 SportVU Data

SportVU's six cameras in each NBA arena record the X and Y coordinate of each player on the court, as well as the X, Y, and Z coordinates of the ball. These positions are recorded 25 times a second. We use data from the 2013-2014, 2014-2015, and 2015-2016 seasons.

We use this position data to build five trajectory-images for each possession – one for each player on offense. This process is described further in Section 3.1. We recognize that there are possessions (such as fast breaks) in which some players don't participate in the play. We therefore skip possessions that are shorter than 4 seconds long or have fewer than five players (indicating that not all players crossed half court).

2.2 Basketball-Reference Data

Some of our experimental results and analysis takes players' listed positions and box-score statistics into account. We got this data from Basketball-Reference.com, a website operated by Sports Reference, LLC.

3. Methodology

3.1 Trajectory-images

We build five images for each possession using the raw player-tracking data from SportVU. Each trajectory-image captures one offensive player's movement in a 64x64 binary image. Building an image for each player-trajectory gives a starting point for comparing player movements across many different possessions [2]. Trajectory-images allow us to represent *any possession* in a common, fixed-size input space. Representing possessions as images also allows us to take advantage of convolutional neural networks and other vision-based techniques. Trajectory-images for ten players (from two possessions) are shown in Figures 3.1 and 3.2.

Figure 3.1
Trajectory-images for a Warriors offensive possession from the 2016 NBA finals

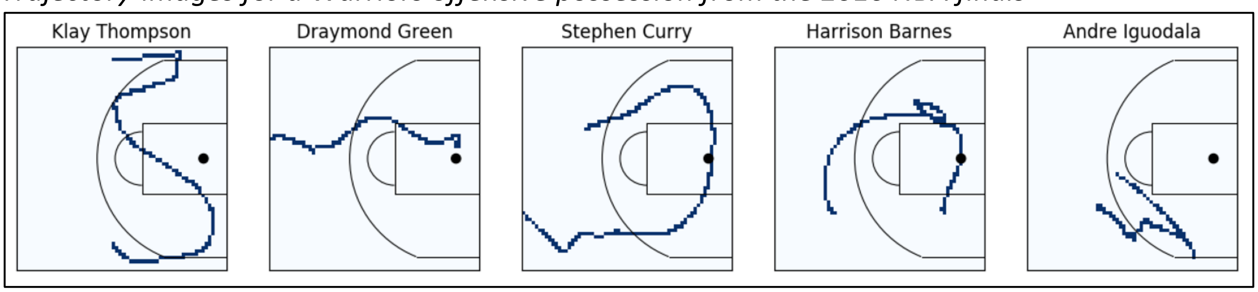
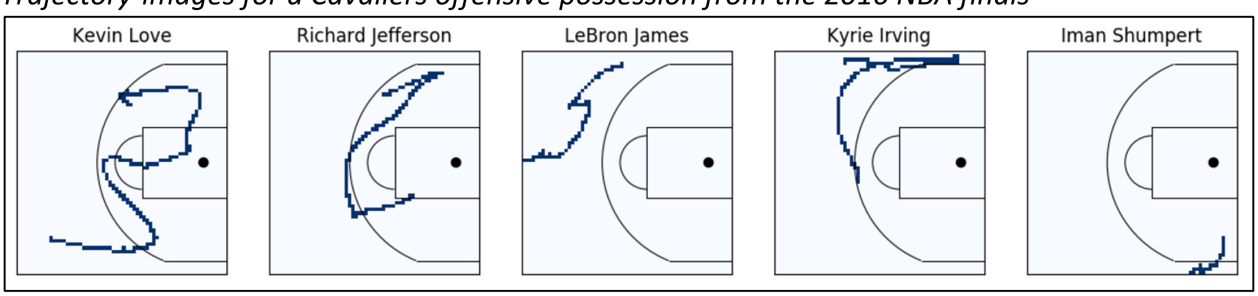


Figure 3.2
Trajectory-images for a Cavaliers offensive possession from the 2016 NBA finals



Each player's (X, Y) time series data for a given possession is pruned to include only coordinates from the moment that all players on offense have crossed half court. Pruning each player's time series ensures that all trajectory-images for a given possession capture the same period of time. Table 3.1 shows the pruned time series data for a Pacers' possession. Figure 3.3 displays a basketball court and its dimensions. The code in Figure 3.4 shows how each trajectory-image is constructed from the time series data.

Table 3.1
SportVU time series data for part of a Pacers' possession, 2013-2014

Gameclock (seconds)	Player	X	Y
715.34	David West	58.64	16.14
715.34	Roy Hibbert	75.86	37.16
715.34	George Hill	47.75	23.55
715.34	Paul George	71.53	6.07
715.34	Lance Stephenson	69.02	43.11
715.30	David West	59.09	16.13
715.30	Roy Hibbert	75.34	37.07
715.30	…	…	…

Figure 3.3
Basketball court with dimensions

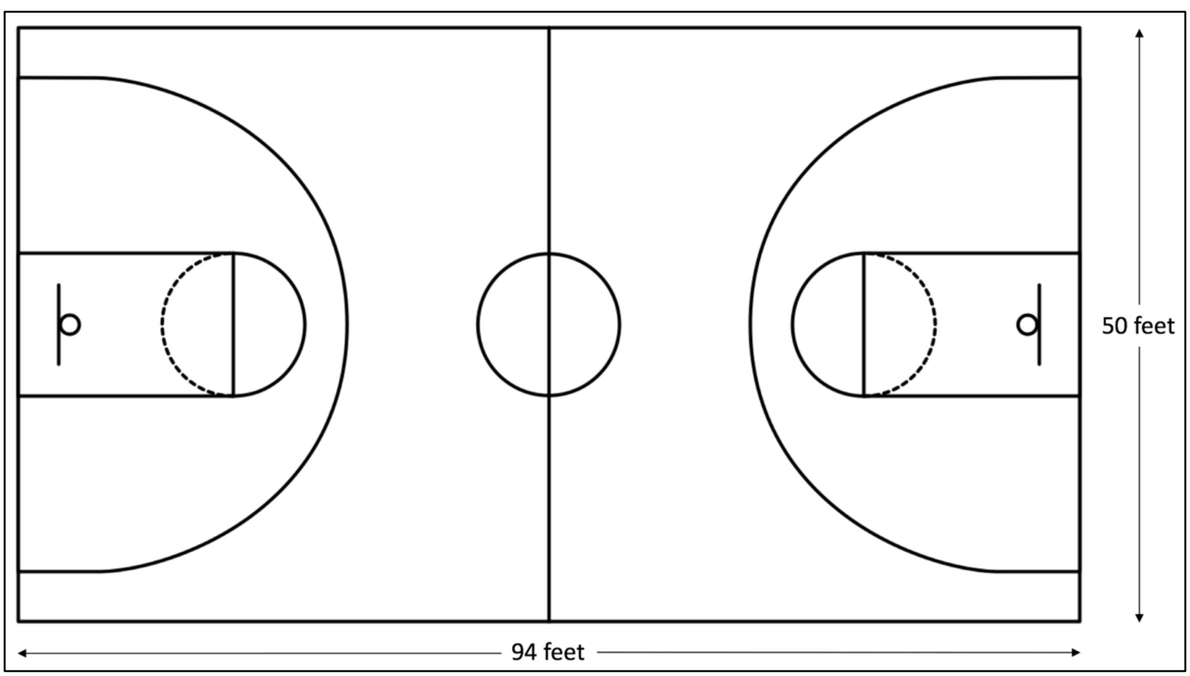


Figure 3.4

Function used to construct trajectory-images from SportVU time series data. coord_to_pixel converts a (X,Y) coordinate on a 94x50 full court to the corresponding pixel in the 64x64 half court trajectory-image

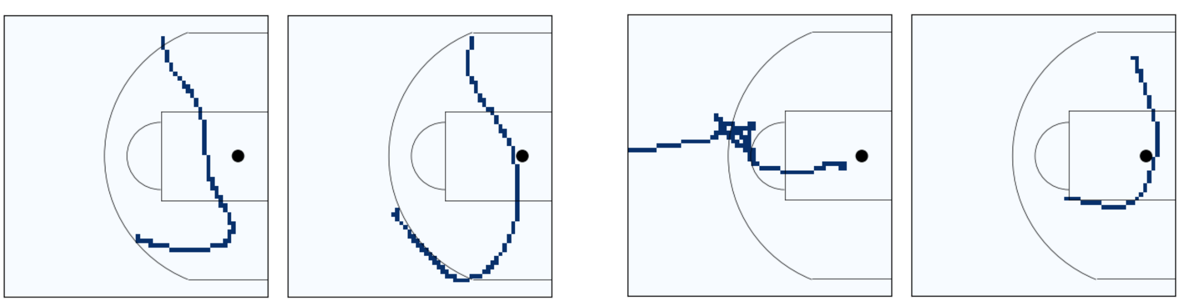
```
def timeseries_to_image(timeseries, width=64, height=64):
    # initialize empty trajectory-image
    img = numpy.zeros((width, height), dtype=numpy.float32)

    for coordinate in timeseries:
        # convert raw (X,Y) coordinate to (row, col) pixel
        row, col = coord_to_pixel(coordinate.X, coordinate.Y, width, height)

        # fill in image pixel
        img[row, col] = 1

    return img
```

3.2 Learning a dense representation

We constructed nearly 3 million trajectory-images from three seasons of data (2013, 2014, 2015). Our goal is to find similar patterns of movement across *all* possessions, but comparing the images directly may not be meaningful. The Euclidean distance between two trajectories that represent similar movements, separated by a few pixels, will be the same as the Euclidean distance between two trajectories that represent vastly different movements. Figure 3.5 depicts two pairs of trajectory-images that have similar Euclidean distances, although the second pair represents vastly different movements. The similarity in Euclidean distance is because of the sparseness of each trajectory-image.

Figure 3.5

Pairs of trajectory-images that have a similar Euclidean distance but different visual similarity



We want our representation for each movement on offense to support operations such as nearest neighbors. The Euclidean distance between a pair of image-representations should correlate with the similarity in the images' movements. We want an encoding for each image that captures the *semantics* of what occurred during the possession.

We recognized that it would be a tedious process to filter through these images and label them with the correct information, such as the locations of screen actions and sharp turns, the endpoints, and so forth. Even developing an algorithm to extract these labels would be difficult since there is no good way to evaluate its accuracy across such a large dataset. We therefore turn to unsupervised learning as a way to gain new insights into trajectory-images.

Unsupervised learning has been used to compute low-dimensional, neighborhood preserving embeddings of high dimensional data [3]. Autoencoders are a type of unsupervised learning that have emerged in recent years as an effective way of learning low-dimensional encodings for high-dimensional input vectors [4]. These low-dimensional encodings would facilitate operations such as nearest neighbors and clustering. An autoencoder architecture consists of a series of encoder layers that reduce the input vector to a low-dimensional encoding, followed by a series of decoder layers that attempt to reconstruct the original vector from the encoding.

Convolution and pooling layers have been found to give improved results when training an autoencoder to reconstruct images from the MNIST dataset, which contains binary images of handwritten digits [5]. Although trajectory-images are substantially different than handwritten digits, they do share certain characteristics that encourage us to pursue a similar approach. Figure 3.6 shows some examples from the MNIST dataset.

An MNIST image and a trajectory-image are both binary images with a variety of curves, edges, and angles, and a varying number of endpoints. Most importantly, MNIST images are all different from each other, with slight variations between two images of the same digit. This variation is true of basketball movements as well; e.g. two trajectory-images that contain a movement from the top of the key to the basket will have pixels in slightly different locations.

Figure 3.6
Examples of handwritten digits and their labels from the MNIST dataset

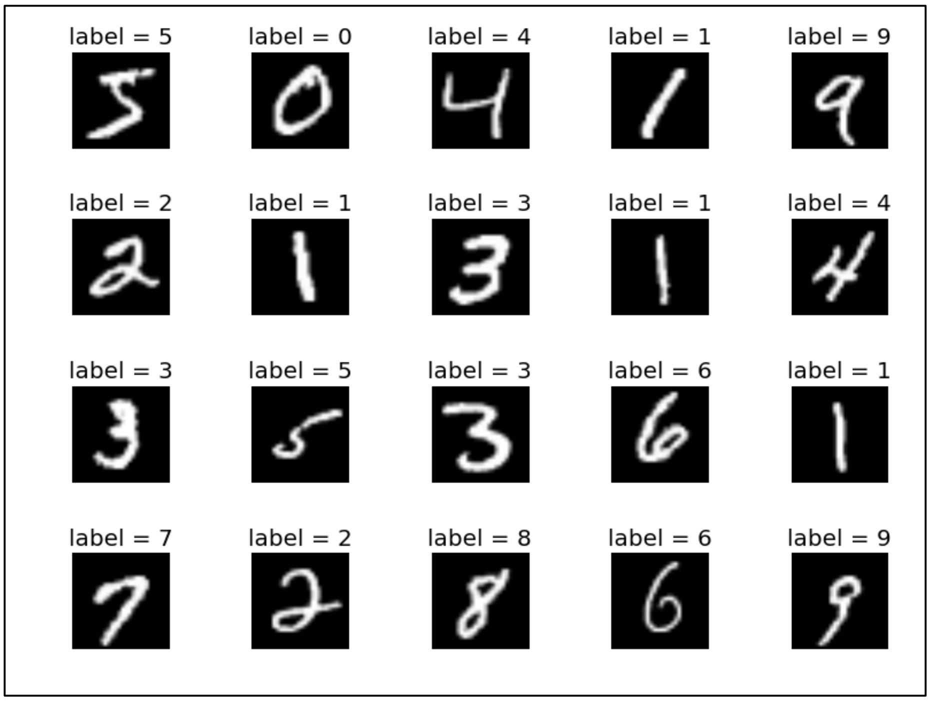


Figure 3.7
Convolutional autoencoder architecture

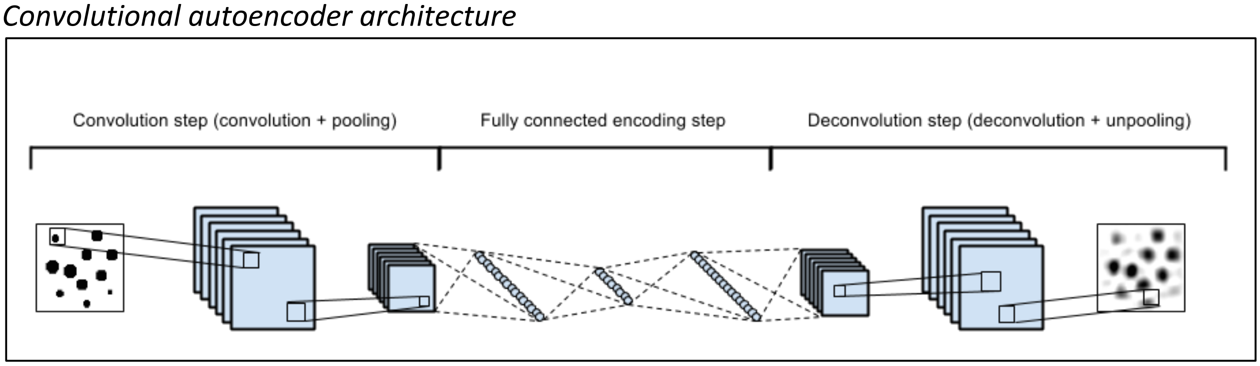


Figure 3.7 displays the architecture we used [6]. It consists of a series of convolution and pooling operations that reduce the input 64x64 trajectory-image to a 4x4x8 volume. This volume is flattened to a 128-dimensional array and connected (with a fully-connected layer) to the 32-dimensional encoding. This encoding is connected to a 128-dimensional array, reshaped to a 4x4x8 volume, and put through a series of deconvolution and upsampling operations to build the reconstructed trajectory-image. Having fully-connected layers in the middle permits the network to learn patterns from different parts of each image, in addition to the spatial patterns it learns from convolution and pooling operations.

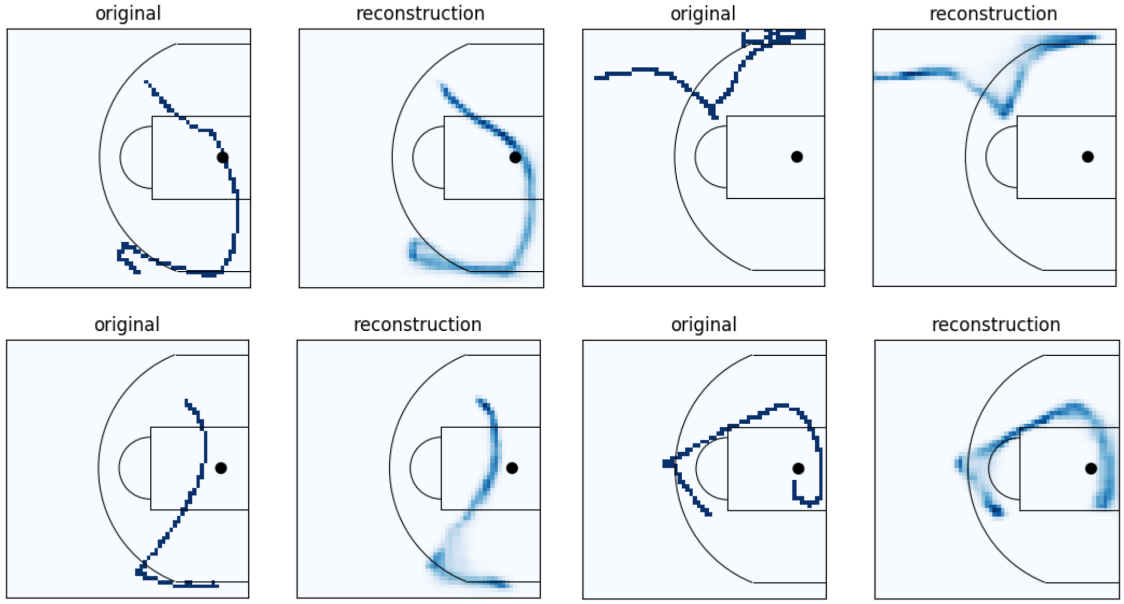
We trained the network to optimize the binary cross entropy loss between the original and reconstructed trajectory-images. We used binary cross entropy as the loss function after empirically finding it to give higher-quality reconstructions than mean squared error.

The network was trained for 500 epochs with a 70/30 split between training and validation data, and achieved its lowest validation loss of 0.0282 on the 409th epoch. We then recovered the 32-dimensional encoding for each of the 3 million trajectory-images. These encodings are referred to as *trajectory-embeddings* for the rest of this thesis. Figure 3.8 displays four trajectory-images and their reconstructions using the optimized model.

Learning a low-dimensional encoding for each trajectory-image lets the network abstract away certain details about each movement while preserving more important ones. Although the reconstructions in Figure 3.8 are blurred, they still capture the overall movement and endpoints of their input trajectory-images.

Figure 3.8

Four trajectory-images and their reconstructions from the autoencoder



3.3 Trajectory-embeddings

We chose to use a 32-dimensional embedding to represent each trajectory-image. This was an empirical decision based on the fact that 32 is the lowest dimension we tested that gave reconstructions that resembled the input trajectory-images. We found that nearby trajectory-embeddings (by Euclidean distance) almost always encode visually similar trajectory-images. Even in instances where nearby trajectory-embeddings encode different movements, the movements tend to occupy the same area of the court. This result demonstrates that our autoencoder network did a good job of capturing spatial patterns in trajectory-images. These experimental results are discussed in more detail in Section 4.1.1.

3.4 K-Means Clustering

Running a clustering algorithm on all 3 million trajectory-embeddings gives us a way to group together similar trajectory-images. We can then look at the original trajectory of each cluster's central element (mediod) to get a sense of what kind of movement each cluster represents.

3.4.1 Choosing K

We used K-Means clustering because it is suitable for large datasets and is not as sensitive to outliers as other clustering techniques. To choose the number of clusters, we ran K-Means for all even numbers between 10 and 60, and looked at the average-cluster-variance for each run of K-Means. Given each trajectory-image T, its embedding E, its cluster C, and its cluster-mediod's embedding M, the average cluster variance is calculated as follows.

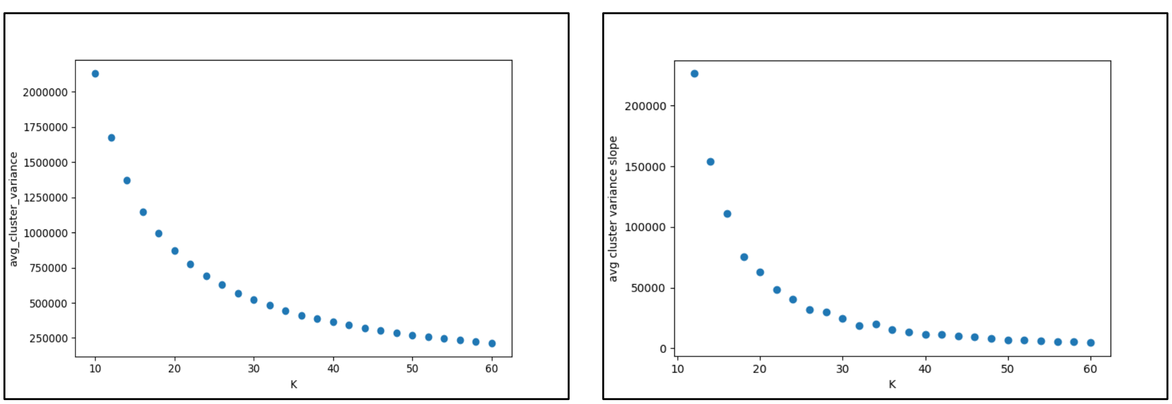
$$\mathrm{variance(T)} = [norm(\mathrm{E} - \mathrm{M})]^2$$

$$\mathrm{in\_cluster\_variance(C)} = \frac{\sum_{\forall P \in C} \mathrm{variance}(T)}{\mathrm{\#points\ in\ cluster\ C}}$$

$$\mathbf{average\_cluster\_variance} = \frac{\sum_{\forall \boldsymbol{C}} \mathbf{in\_cluster\_variance}(\boldsymbol{C})}{\mathbf{\#clusters}}$$

The plot and slope-plot for average cluster variance is in Figure 3.9. The slope graph exhibits an "elbow" between K=20 and K=30, indicating that values of K that fall in this range are reasonable. We chose K=20 as it is the smallest value of K in this "reasonable" range.

Figure 3.9
Average cluster variance (left) and its slope plot (right)

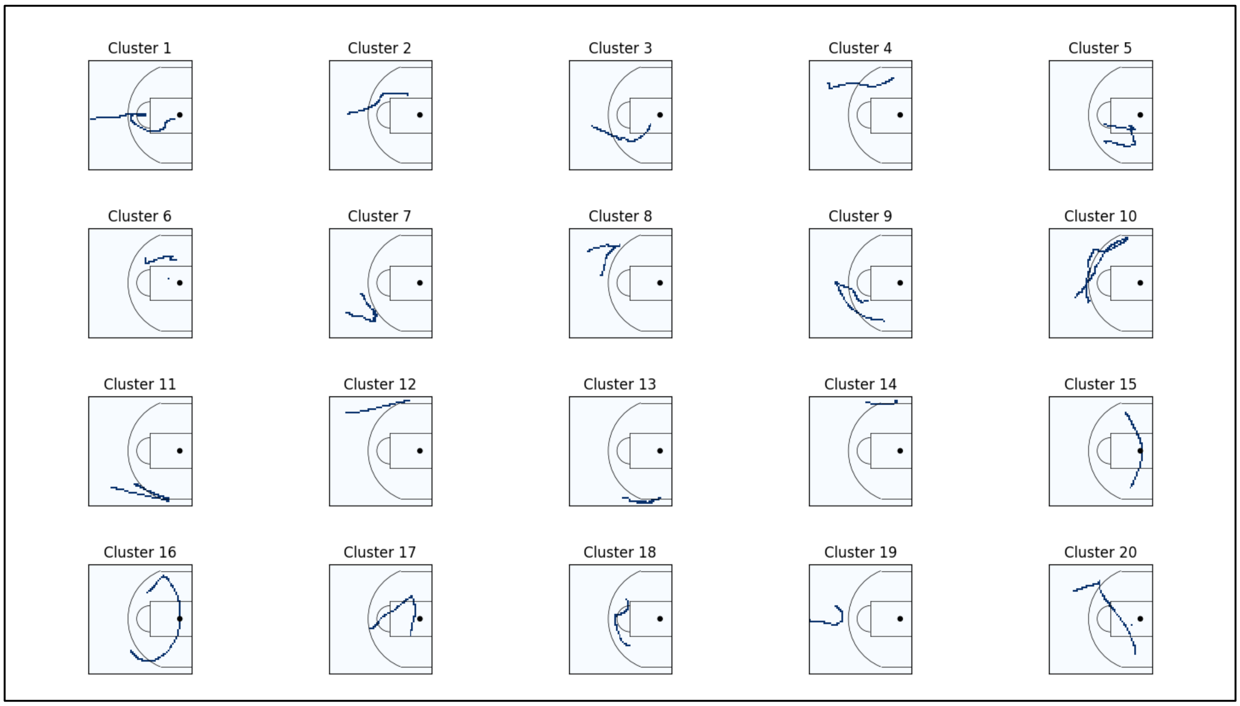


3.4.2 Cluster Descriptions

We computed each cluster's cluster-mean, and found the ten trajectory-embeddings nearest to this cluster mean. The corresponding trajectory-images for these embeddings are given in Appendix A. The nearest embedding to each cluster-mean is denoted as that cluster's cluster-mediod. Each cluster-mediod's trajectory-image is shown in Figure 3.10.

Figure 3.10

Cluster mediods: trajectory-image of the nearest embedding to each cluster-mean



We looked at the trajectory-images of the ten embeddings nearest to each cluster-mean, and assigned a description to each cluster. Seven pairs of clusters contain movements that are symmetrical along an imaginary line that connects both baskets. For example, clusters 1 and 2 are movements to the right/left block from the top of the key. Clusters 7 and 8 are screen actions above the right and left wing.

Five of the remaining clusters have movements that are inherently stand-alone, in that they equally span both the left and right sides of the halfcourt. For example, cluster 15 is a run along the baseline and cluster 18 is a lateral movement in the high post.

Only cluster 20 is neither stand-alone nor is symmetric to another cluster. Table 3.2 has a full list of these cluster-mediod descriptions, with symmetric clusters paired together.

Table 3.2
Cluster-mediod descriptions with near-reflection clusters paired together

Cluster(s)	Descriptions
1, 2	Movement from top of key to right/left block from top of key
3, 4	Movement from top of key/above wings to right/left short corner
5, 6	Movement or screen action, on either side of the paint
7, 8	Screen action above the wings
9, 10	Movement above wings
11, 12	Movement above wings, nearer to the sideline
13, 14	Movement along each sideline to corner
15	Run along baseline
16	Arc-like movement from sideline to sideline
17	Screen action in paint
18	Lateral movement in high post
19	Run past halfcourt, screen action above the key
20	Movement from above left wing to right block

3.4.3 Defining a Cluster Ordering

We order the clusters based on their court positions, cluster symmetry, and observations from the cluster mediods. We found that doing so gives more meaning to cluster-profiles (Section 3.5). Clusters 1 through 14 are the seven pairs of symmetric clusters. The movements of odd clusters (1,3,…,13) fall in the right half of the court, and movements of the even clusters (2,4,…,14) fall into the left half of the court. The clusters are also sorted by court position, with clusters 1 and 2 having movements closest to the basket and clusters 13 and 14 having movements furthest from the basket. Clusters 15 through 19 have the five self-symmetric clusters, also sorted by distance from the basket. Cluster 20 has the only stand-alone cluster of movements.

3.5 Summarizing player movements with cluster-profiles

A player's patterns of movement over the course of many games can be summarized with a cluster-profile. A cluster-profile is a 20-dimensional vector, that represents how frequently a player's trajectory-images over a given span of games fall into each of the 20 clusters from running K-Means. Each element of a cluster-profile for a player represents the fraction of his trajectories that fall into that corresponding cluster.

$$cluster\_profile[P,i] = \frac{\#trajectory\_embeddings\ for\ player\ P\ in\ cluster\ i}{\#total\ trajectory\_embeddings\ for\ player\ P}$$

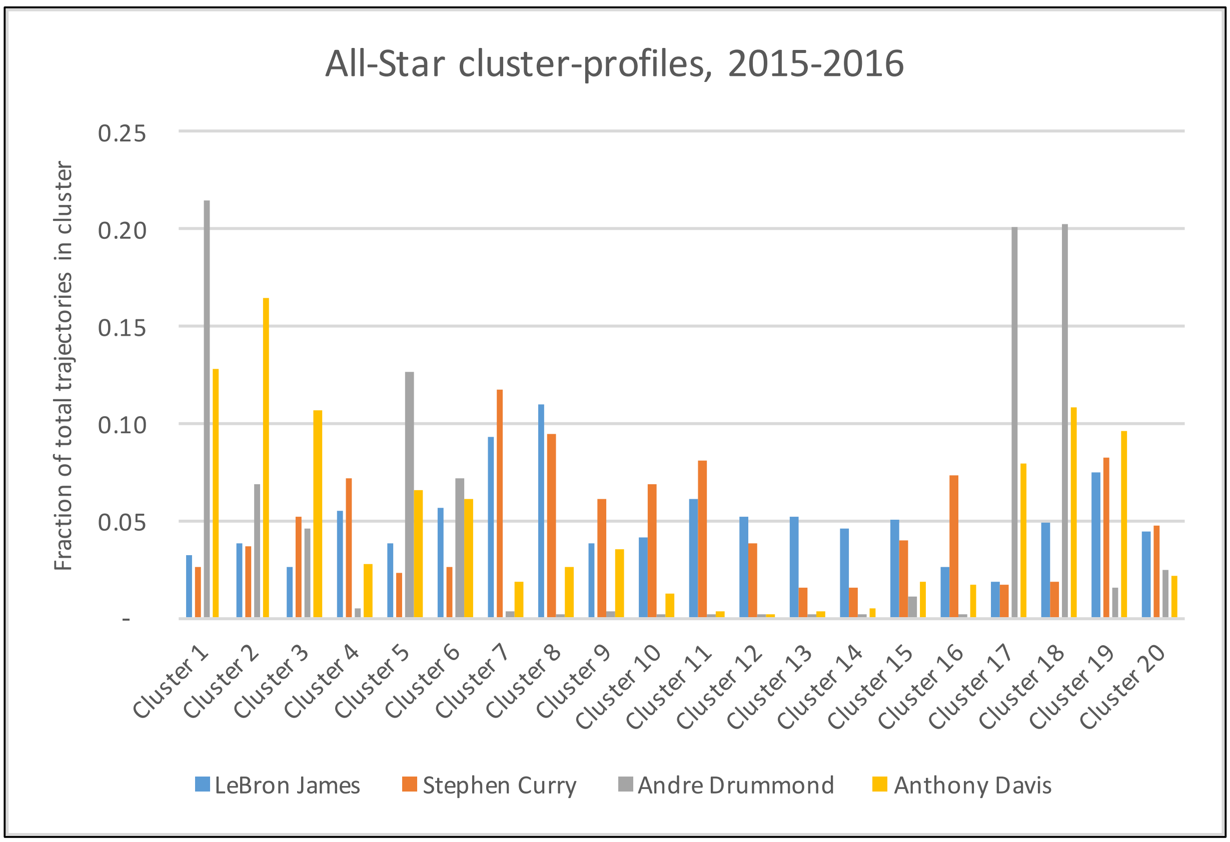
In Figure 3.11, we show the cluster-profiles for four players, computed across all regular season games played in the 2015-2016 season. LeBron James (blue) and Stephen Curry (orange) have similar cluster-profiles, speaking to their versatility on offense. They are both strong passers, and capable of scoring on their own, from both inside and outside. This is in stark contrast to the cluster-profile of Andre Drummond (gray), who, as a pure center, is much more limited in the types of movements he makes on offense. The cluster-profile of Anthony Davis (yellow), a versatile center, is also given.

LeBron James shoots with his right hand but is naturally left-handed. His cluster-profile reveals that more of his movements fall into the left half of the court (clusters 4, 6, 8) than in the right half (clusters 3, 5, 7). The opposite is true for Andre Drummond, who is naturally right-handed and heavily favors the right side of the court. Drummond has more than twice as many movements in clusters 1, 3, and 5 (right-half) than in clusters 2, 4, and 6 (left-half).

James and Curry also have significantly more movements in clusters 9, 10, 11, and 12 than Drummond and Davis. These clusters all contain screen actions and movements above the wings, with clusters 11 and 12 nearer the sidelines. James and Curry are both strong passers and capable shooters, and often engage in screen actions above the wings to create mismatches. Drummond and Davis do more of their damage in the paint, with a large fraction of their movements in clusters 17 and 18.

Figure 3.11

Cluster-profiles for LeBron James, Stephen Curry, Andre Drummond, Anthony Davis



4. Experimental Results

We ran a series of experiments to determine the utility of trajectory-embeddings and cluster-profiles. We ran these experiments on the embeddings and cluster-profiles of the 100 players with the most examples in our dataset – meaning they participated in the most possessions greater than 4 seconds – between 2013 and 2016. Appendix C has a full list of these players. The top 10 players are shown in Table 4.1. Table 4.2 has the number of players per listed position.

Table 4.1

Top 10 players across the 2013, 2014, 2015 NBA seasons by number of examples (trajectory-images)

Player	**Position**	**Number of examples**
Damian Lillard	Point guard	13,477
Monta Ellis	Shooting guard	13,441
Gordon Hayward	Small forward	13,233
Kyle Lowry	Point guard	13,039
John Wall	Point guard	13,039
James Harden	Shooting guard	12,797
Joe Johnson	Shooting guard	12,780
DeMar DeRozan	Shooting guard	12,715
Trevor Ariza	Small forward	12,531
Nicholas Batum	Small forward	12,433

Table 4.2

The number of players for each given position in our analysis.

Position	**Number of players in top 100**
Point guard (pg)	23
Shooting guard (sg)	21
Small forward (sf)	23
Power forward (pf)	20
Center (c)	13

4.1 Trajectory-embedding experiments

4.1.1 Evaluating visual similarity of nearby trajectory-embeddings

We designed an experiment to evaluate the visual similarity of nearby trajectory-embeddings. For 100 randomly chosen trajectory-embeddings, we found the 10 nearest neighbors in embedding-space and retrieved their trajectory-images. We gave each trajectory-embedding a score between 0 and 10, indicating how many of its ten nearby embeddings have a visually similar trajectory-image. We define two trajectory-images to be visually similar if their movements have endpoints in similar locations on the court and span a similar area on the court. Figure 4.1 has the ten nearest-neighbors of a trajectory-image that scored a 10/10. Figure 4.2 has the ten nearest-neighbors of a trajectory-image that did not fare as well, scoring a 5/10.

Figure 4.1
The trajectory-image in the first row is visually similar to its 10 nearest neighbors and scored a 10/10 in our evaluation.

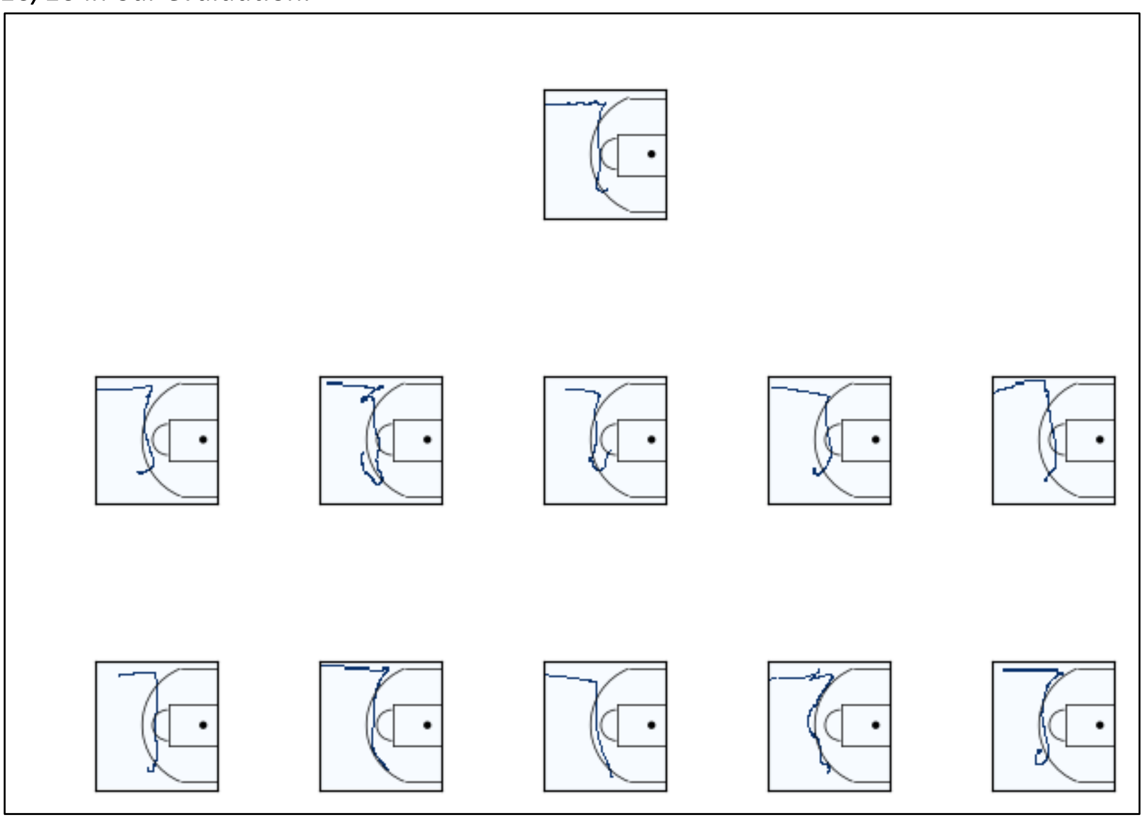
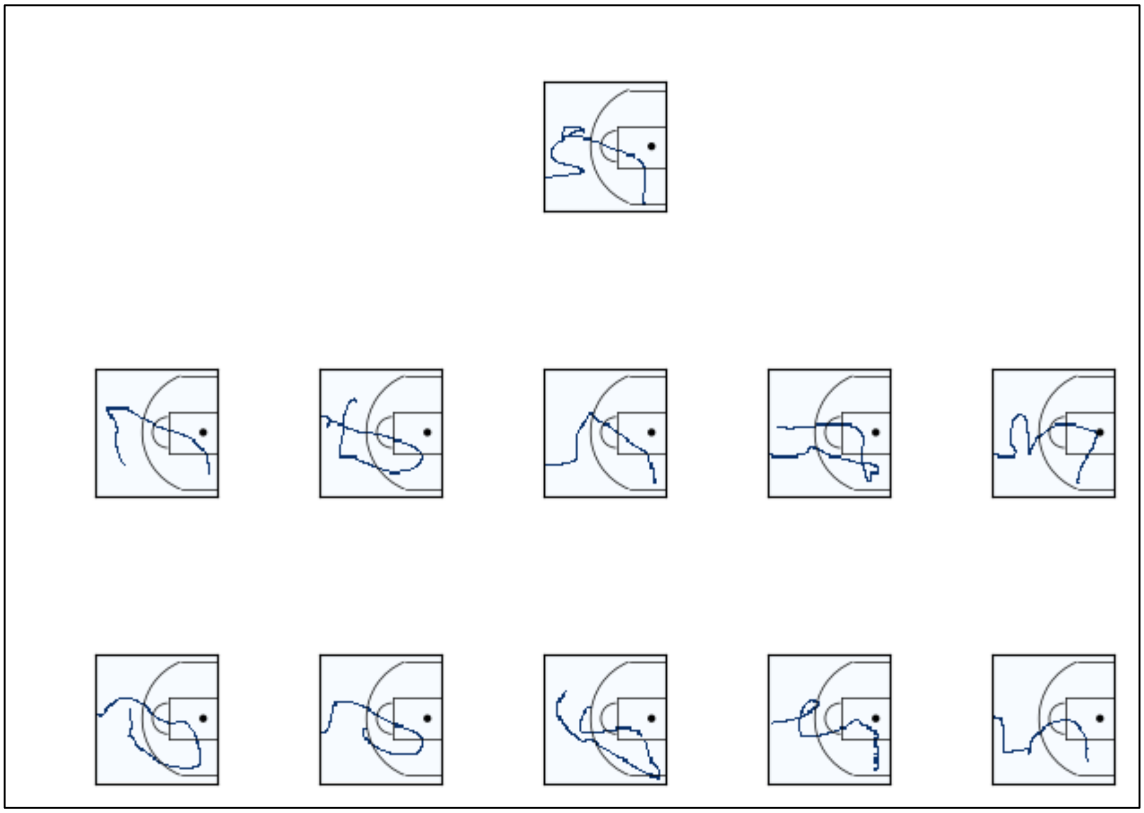


Figure 4.2

The trajectory-image in the first row scored a 5/10 in our evaluation. It is visually similar to its 1st, 3rd, 5th, 9th, and 10th nearest neighbors.



The average score for the 100 randomly chosen trajectory-embeddings was 9.1 (out of a possible 10), demonstrating that nearby embeddings are visually similar. A few observations we made from this experiment are that nearby trajectory-embeddings that are not visually similar still contain movements in the same area of the court (Figure 4.3), and that trajectory-embeddings of shorter movements have less variation in their nearest-neighbors than trajectory-embeddings of longer movements (Figures 4.4, 4.5).

Figure 4.3

Nearby trajectory-embeddings represent movements in the same area of the court. Although the trajectory-images in each row are not the same movement, they are nearby in embedding-space as they contain movements in the same area of the court.

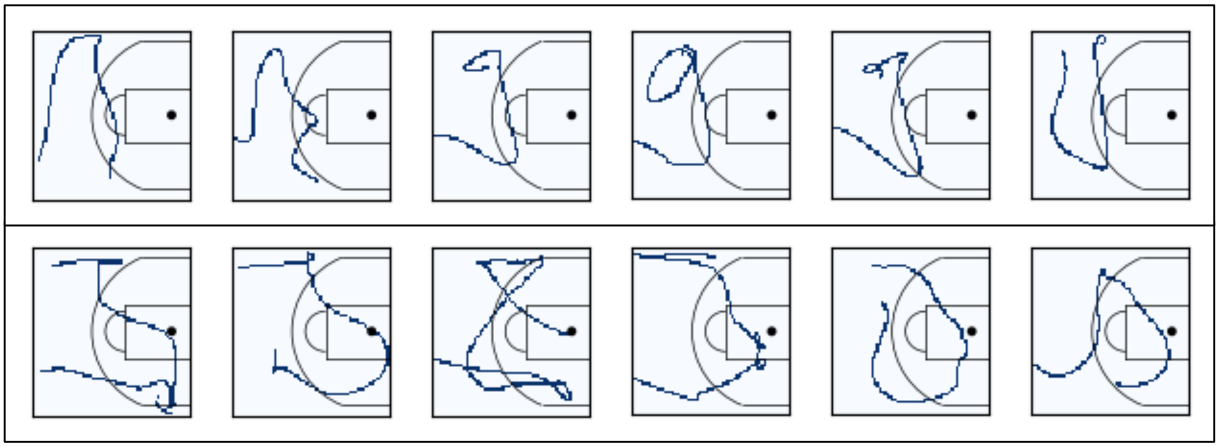


Figure 4.4

Images 2 to 6 are the five nearest-trajectories in embedding-space to Image 1. These trajectories represent shorter movements and have little variation.

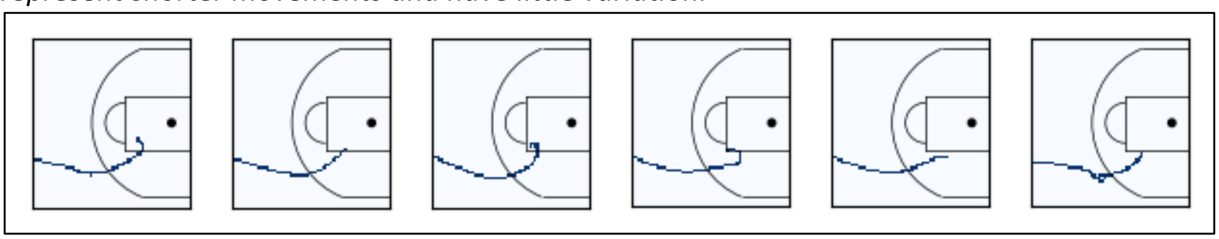
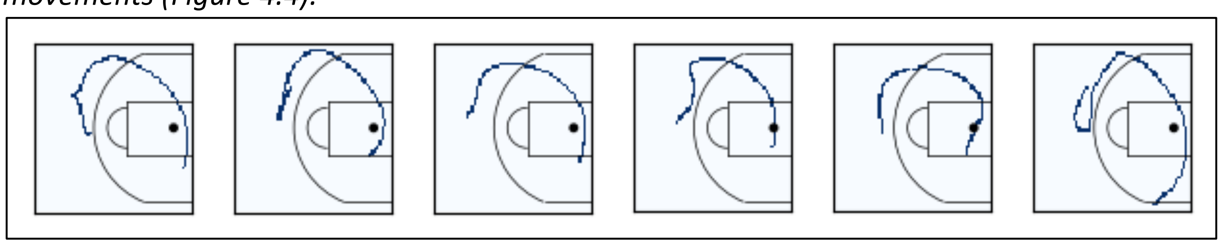


Figure 4.5

Images 2 to 6 are the five nearest-trajectories in embedding-space to Image 1. These trajectories represent longer movements and have more variation than than nearby-trajectories of shorter movements (Figure 4.4).



4.1.2 Cluster mediod distance matrix

We computed the Euclidean distance between all cluster-mediods, and normalized the results so the distance between the furthest cluster-mediods is 1.0. The full cluster-mediod distance matrix can be found in Appendix B.

Table 4.3
Summary of cluster-mediod distance matrix

Cluster	Description	Nearest cluster (distance)	Furthest cluster (distance)
1	Movement from top of key to right block	3 (0.493)	4 (0.936)
2	Movement from top of key to left block	19 (0.456)	11 (0.873)
3	Movement to right short corner	9 (0.427)	4 (0.905)
4	Movement to left short corner	8 (0.474)	11 (0.947)
5	Movement or screen action, right of paint	3 (0.441)	10 (0.972)
6	Movement or screen action, left of paint	17 (0.461)	9 (0.849)
7	Screen action above right wing	11 (0.493)	20 (1.0)
8	Screen action above left wing	12 (0.463)	16 (0.918)
9	Movement along right wing	3 (0.427)	4 (0.921)
10	Movement along left wing	2 (0.545)	5 (0.972)
11	Movement along right wing, closer to sideline	13 (0.434)	4 (0.947)
12	Movement along left wing, closer to sideline	14 (0.459)	1 (0.903)
13	Run along right sideline to corner	11 (0.434)	10 (0.899)
14	Run along left sideline to corner	12 (0.459)	1 (0.807)
15	Run along baseline	16 (0.479)	10 (0.896)
16	Arc-like movement across court	15 (0.479)	19 (0.924)
17	Screen action in the paint	6 (0.461)	11 (0.880)
18	Lateral movement in high post	9 (0.532)	15 (0.899)
19	Run past halfcourt, screen action at top of key	2 (0.456)	16 (0.924)
20	Movement from above left wing to right block	8 (0.594)	7 (1.0)

Table 4.3 summarizes the cluster-mediod distance matrix. It has a description for each cluster and lists each cluster's nearest and further cluster. Nearby clusters have movements in similar areas of the court. For example, clusters 4, 8, and 12 all have movements around the left wing. Clusters 15 and 16 are each other's nearest-cluster and have arc-like movements from sideline to sideline. Distant clusters have movements in different areas of the court. Clusters 16 and 19 are each other's furthest-cluster, an observation supported by their descriptions and

mediod-trajectories. Table 4.4 shows the distance between cluster-mediods of symmetric clusters. We observe that cluster symmetry has little effect on distance between mediods.

Table 4.4

Distance between cluster-mediods of symmetric clusters

Pair of symmetric clusters	Distance between cluster-mediods
1, 2	.662
3, 4	.696
5, 6	.545
7, 8	.814
9, 10	.883
11, 12	.794
13, 14	.565

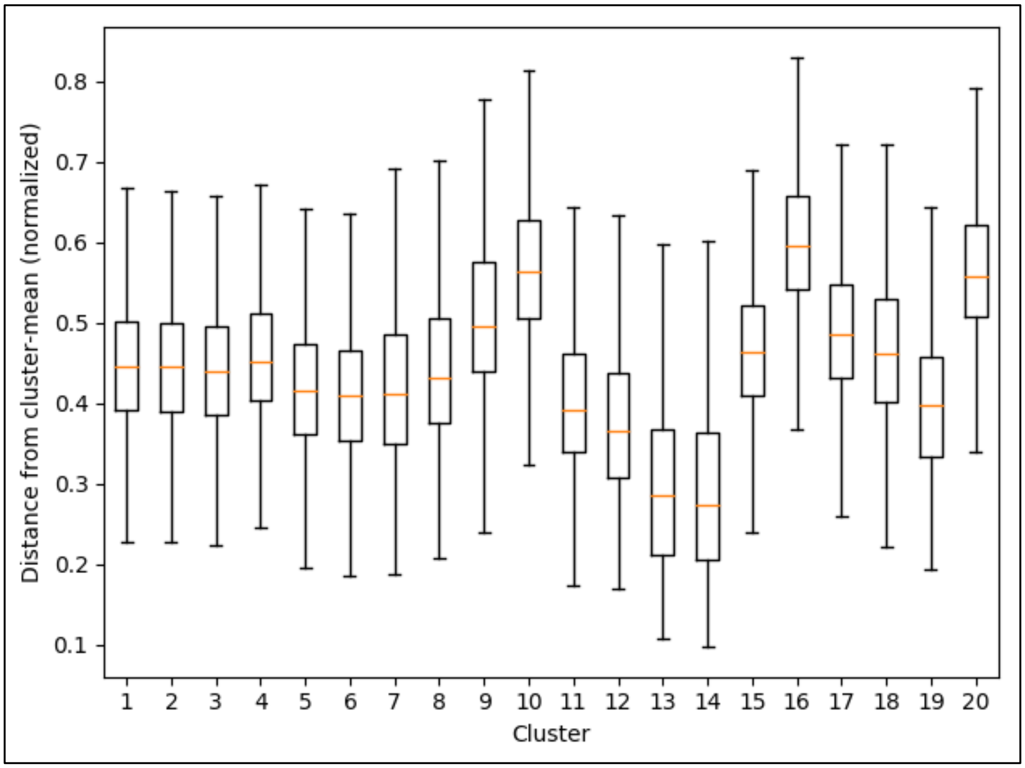
4.1.3 Using box-plots to analyze cluster variance

We want to evaluate how *tight* each cluster is; i.e. the closeness of each cluster's trajectory-embeddings. To do so, we computed the Euclidean distance between each embedding and its cluster's mediod-embedding, and normalized the results so that the furthest such distance is 1.0. We use a box plot (Figure 4.6) to visualize the results for each cluster and conduct further analysis.

Most pairs of symmetric clusters have very similar boxplots. The only exception is cluster 9 and cluster 10. Clusters 13 and 14 are the tightest. These clusters contain movements along the sideline to the 3-point corners. Clusters 16 and 20 are the least-tight clusters. Cluster 16 has arc-like movements across the court and cluster 20 from above the left wing to the right block. Both of these clusters contain movements that span significantly larger areas of the court than other clusters, which explains the larger spread in their boxplots.

Figure 4.6

Boxplot of distances-from-cluster-mean for each cluster



4.1.4 Trajectory-Images of embeddings distant from their cluster-mediod

The trajectory-images of the ten embeddings nearest to each cluster-mean closely resemble reach other (Appendix A). However, this does not imply that all movements within a cluster look similar to each other. Some trajectories-images of embeddings that lie further away from the cluster-mean look significantly different than the cluster-mediod's trajectory-image, and contain patterns of movement similar to those in other clusters as well. Figure 4.7 shows the trajectory-image of each cluster's 50th-percentile embedding, and Figure 4.8 shows the trajectory-image of each cluster's 75th-percentile embedding. Despite being far from the cluster-mean, these trajectories look remarkably similar to the median-trajectories (Figure 3.8), and are consistent with the cluster-mediod descriptions (Table 3.2), indicating that the twenty clusters do an excellent job of capturing a *diverse* and *comprehensive* set of player movements.

Figure 4.7

Trajectory-image of each cluster's 50th-percentile-embedding (median-embedding)

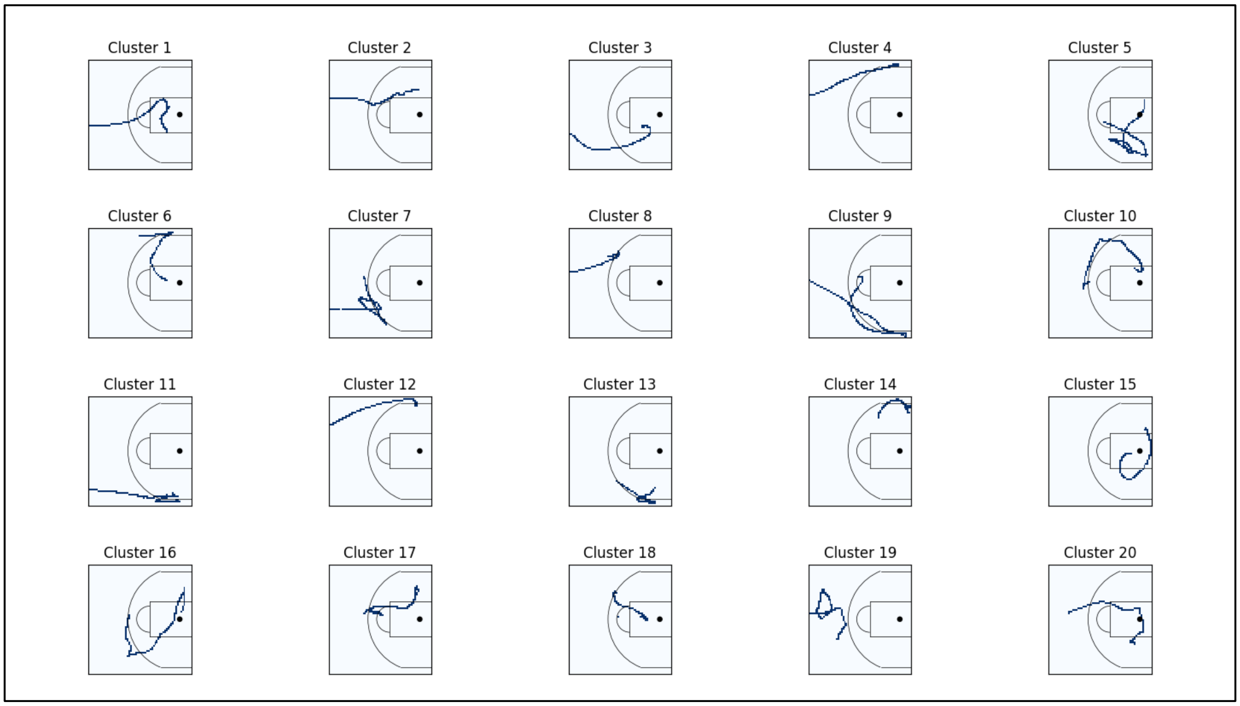
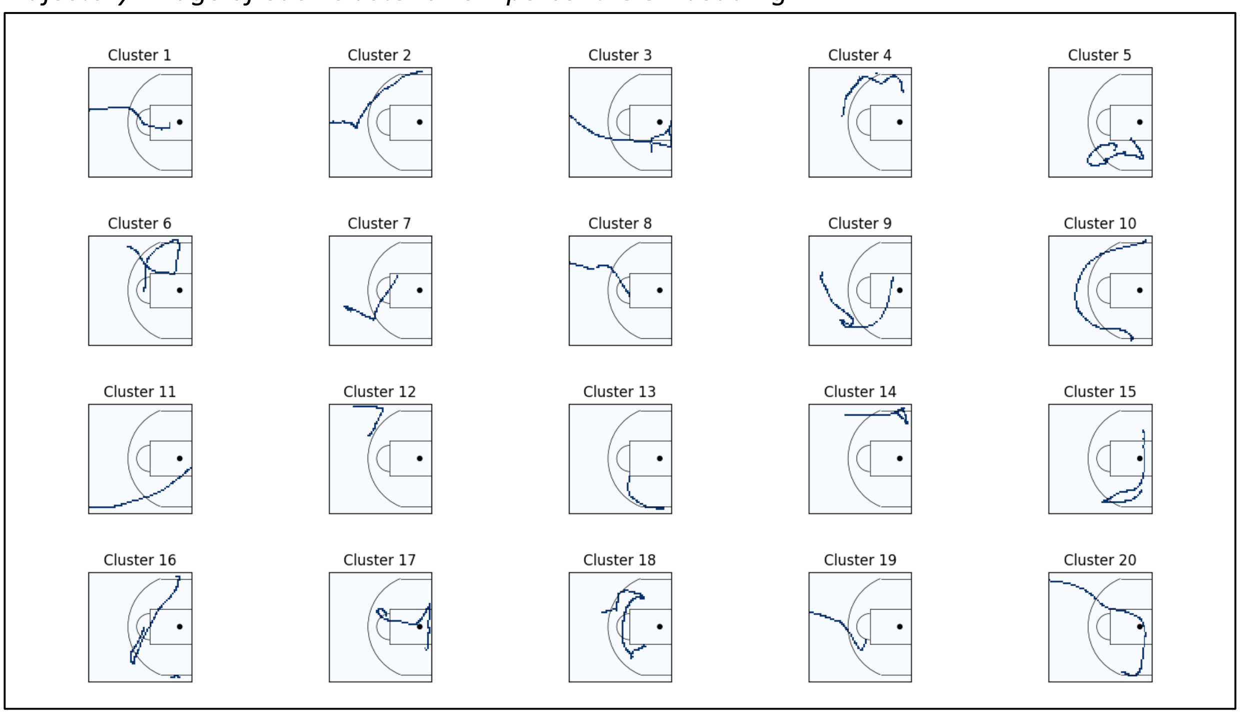


Figure 4.8

Trajectory-image of each cluster's 75th-percentile embedding



4.2 Cluster-profile experiments

A player's cluster-profile is a summary of his aggregate patterns of movement over the course of many games. We calculated three cluster-profiles for each of the 100 players in Appendix C – one per each season of data.

4.2.1 Player movement homogeneity

We calculated the variance of each cluster-profile to get a measure of how diverse a player's movements are. A higher variance indicates that more of a player's trajectory-images are concentrated in a few clusters; a lower variance indicates that a player has trajectory-images in many clusters.

$$cluster\_profile\_homogeneity(P) = \sum_{i=0}^{n\_clusters} (cluster\_profile[P, i])^2$$

Cluster-profile-homogeneity is similar to variance. The differences are we do not subtract the mean from each element before squaring, and we do not divide the final result by the size of the set (*n_clusters*). This results in the maximum possible cluster-profile-homogeneity being 1.0, which would occur if all of a given player's movements fall into one cluster.

The 10 players with the highest and lowest cluster-profile-homogeneities are shown in Tables 4.5 and 4.6. Table 4.7 summarizes the average cluster-profile-homogeneity for each listed position, across all three seasons. Centers have the highest average cluster-profile-homogeneity, and point guards/small forwards have the lowest average cluster-profile-homogeneity.

Table 4.5
10 player-seasons with highest cluster-profile homogeneity

Player, Season	Listed Position	Cluster-profile-homogeneity
Zach Randolph, 2013	pf	0.166
Zach Randolph, 2014	pf	0.160
Andre Drummond, 2015	c	0.156
Robin Lopez, 2014	c	0.156
Zach Randolph, 2015	pf	0.152
Robin Lopez, 2013	c	0.150
Roy Hibbert, 2013	c	0.146
Roy Hibbert, 2014	c	0.140
Marc Gasol, 2014	c	0.140
Robin Lopez, 2015	c	0.136

Table 4.6
10 player-seasons with lowest cluster-profile homogeneity

Player, Season	Listed Position	Cluster-profile-homogeneity
Giannis Antetokounpo, 2015	sf	0.054
Tobias Harris, 2015	pf	0.054
Jeff Green, 2015	sf	0.054
Rudy Gay, 2015	sf	0.056
Rudy Gay, 2014	sf	0.056
Giannis Antetokounpo, 2014	sf	0.056
Matt Barnes, 2015	sf	0.057
LeBron James, 2013	sf	0.057
Dwyane Wade, 2015	sg	0.057
Kevin Love, 2015	pf	0.057

Table 4.7
Average cluster-profile-homogeneity per listed position, per season

	Point guards	Shooting guards	Small forwards	Power forwards	Centers	All players
2013	0.081	0.075	0.074	0.098	0.122	0.087
2014	0.079	0.072	0.076	0.095	0.119	0.085
2015	0.079	0.075	0.073	0.090	0.116	0.084
All seasons	0.080	0.074	0.075	0.095	0.119	0.085

4.2.2 Similarity metric: Hellinger distance

The Hellinger distance is the probabilistic analog of Euclidean distance, and is used to measure the difference between two probability distributions. Although they are not probability distributions, cluster-profiles do have the property that their elements sum to 1, which makes them suitable for such a similarity metric.

$$Hellinger(P, Q) = \frac{1}{\sqrt{2}} * \left\| \sqrt{P} - \sqrt{Q} \right\|_2$$

4.2.3 Cluster-profile similarity

We designed an experiment to evaluate the utility of cluster-profiles. We calculated two values for each player: their self-similarity and their position-similarity-index. If players are significantly more similar to themselves than to other players of the same listed position, then this would indicate that cluster-profiles have some underlying meaning.

For each player, we compute the Hellinger distance between the cluster-profiles for each pair of seasons, i.e. 2013 and 2014, 2013 and 2015, 2014 and 2015. A player's self-similarity is defined as follows.

$$self\_similarity(P) = 1 - \frac{\sum Hellinger\left(cluster\_profile_{P,S1}, cluster\_profile_{P,S2}\right)}{3}$$

$$S1, S2 \in \{2013, 2014, 2015\}$$

For every player, we also compute the Hellinger distance between his cluster-profile and all cluster-profiles for players of the same listed position. The average of these values is a player's position-similarity-index. The maximum possible value for both self-similarity and position-similarity-index is 1.0.

$$position\_similarity\_index(P) = 1 - \frac{\sum Hellinger\left(cluster\_profile_P, cluster\_profile_Q\right)}{\#players\ of\ same\ listed\ position\ as\ P}$$

$$Q \in \{all\ players\ of\ same\ listed\ position\ as\ P\}$$

Table 4.8 summarizes our results with the average self-similarity and position-similarity-index for each listed position. The full table of self-similarity and position-similarity-index values for all players is given in Appendix C. There are only two players, Jose Calderon and Lance Stephenson, who on average are more similar to other players of the same listed position, than they are to themselves. Calderon and Stephenson both played on multiple teams between 2013 and 2016. This could be an influencing factor in their self-similarity being lower than their position-similarity-index.

Table 4.8
Average self-similarity and position-similarity-index by position

Position	Average self-similarity	Average position-similarity-index
pg	0.886	0.829
sg	0.875	0.792
sf	0.875	0.777
pf	0.863	0.730
c	0.880	0.781

5. Analysis

5.1 Cluster-profile Analysis

A cluster-profile is a summary of *a set of trajectory-images*, and is not restricted to one player or just one season of data. In section 3, we looked at cluster-profiles for player-seasons, where each cluster-profile summarized one player's movements over the course of one season. In this section, we compute and analyze cluster-profiles for *varying sets of players* over *different durations of time*.

In section 5.1.1, we compute the cluster-profile for all players' movements and find its nearest neighbors to determine the *average* NBA players, by aggregate movement. In section 5.1.2, we look at cluster-profiles for each listed position, to see how the five traditional positions correlate with player movement. In Section 5.1.3, we look at cluster-profiles for individual players on a per-quarter basis, which reveals some trends in player movements over the course of a game. Section 5.1.4 examines cluster-profiles for individual players on a per-month basis

5.1.1 The most average NBA players by movement

Figure 5.1 shows the cluster-profile for all players' movements. We computed this by dividing the number of embeddings in each cluster by the total number of embeddings (Table 5.1). Each column contains the fraction of total trajectory-images across all players that fall into that cluster. This distribution of trajectories represents the aggregate movement of the entire league.

Figure 5.1

Cluster-profile of all players' movements, 2013-2016

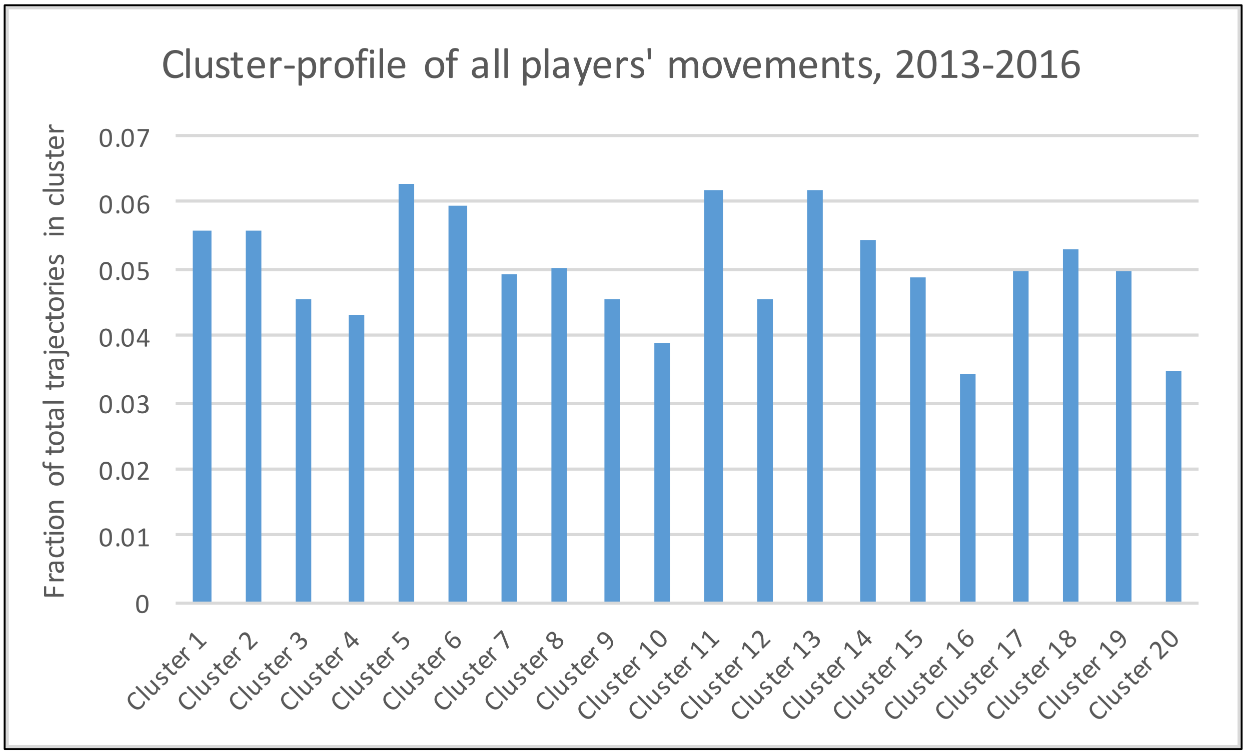


Table 5.1

Size and percentage of total trajectories in each cluster

Cluster	Size	%Total		Cluster	Size	%Total
1	163,763	5.57		11	182,326	6.20
2	163,144	5.55		12	133,335	4.54
3	133,380	4.54		13	181,999	6.19
4	127,343	4.33		14	160,659	5.47
5	184,306	6.27		15	142,661	4.85
6	174,157	5.92		16	100,741	3.43
7	145,582	4.95		17	145,572	4.95
8	148,071	5.04		18	155,132	5.28
9	133,601	4.54		19	146,272	4.98
10	115,411	3.93		20	102,220	3.48
Total: 2,939,675						

Most players have movements distinctive of their role and position and therefore don't have cluster distributions similar to the average cluster-profile. The ten player-seasons with cluster-profiles most similar to the cluster-profile in Figure 5.1 (using Hellinger distance as a similarity metric) are given in Table 5.2.

Table 5.2

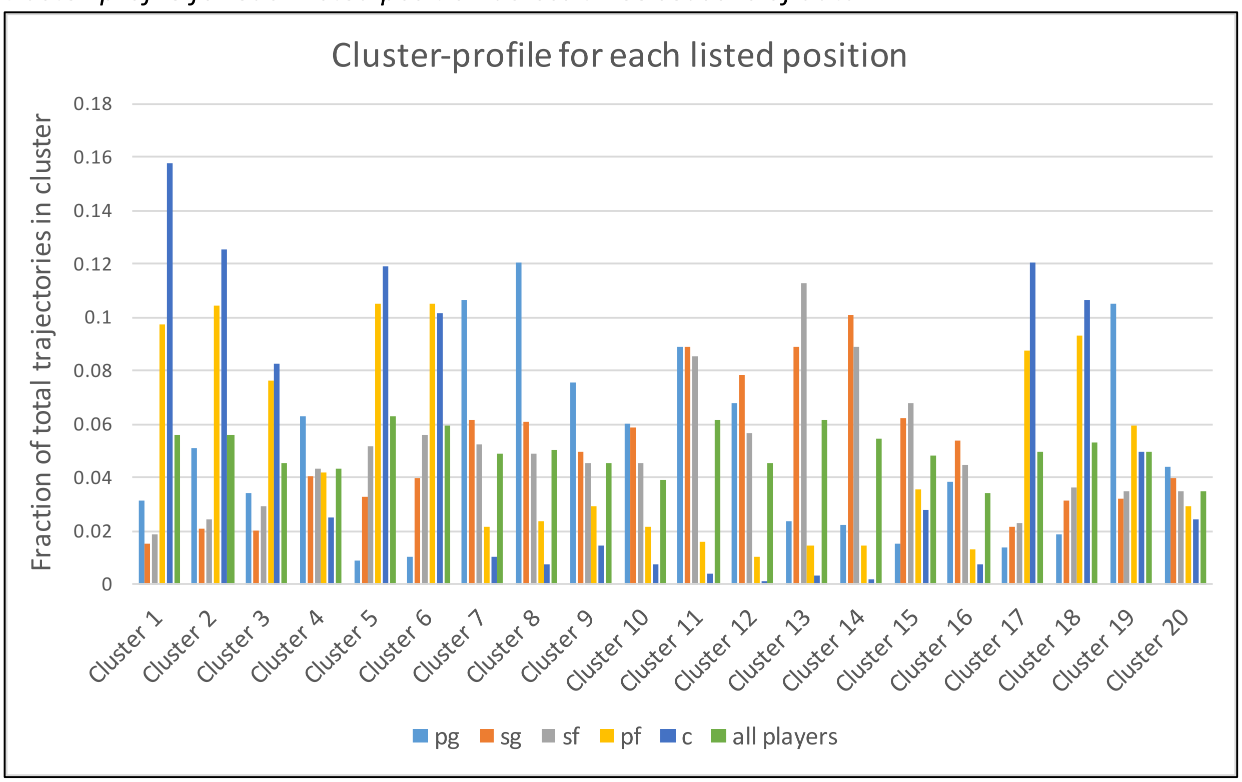
Player-seasons with cluster-profiles most similar to the average cluster-profile

Rank	Player	Similarity		Rank	Player	Similarity
1	Giannis Antetokounmpo, 2015	0.8866		6	LeBron James, 2013	0.856
2	Marvin Williams, 2015	0.8747		7	Rudy Gay, 2015	0.8540
3	Tobias Harris, 2015	0.8747		8	Matt Barnes, 2015	0.8537
4	Kevin Love, 2015	0.8677		9	Draymond Green	0.8518
5	Jeff Green, 2015	0.8614		10	Giannis Antetokounmpo, 2014	0.8494

5.1.2 Cluster-profiles for each listed position

Figure 5.2

Cluster-profile for each listed position across three seasons of data



In this subsection, we look at the aggregate cluster-profile for each listed position to investigate whether the five traditional positions are a good indicator of player movement. Figure 5.2 shows the aggregate cluster-profile for each listed position.

Clusters 1 and 2 contain movements to the right and left block from the top of the key. These clusters make up around 11% of all players' movements but nearly 30% of the movements for centers and 20% for power forwards. These players often get positioned in either block early in possessions, the movement captured by these clusters.

Clusters 5 and 6 make up 12% of all players' movements, but over 20% for both centers and power forwards. These clusters contain trajectories of movements on either side of the paint. Similar to clusters 1 and 2, frontcourt players are often positioned on either side of the paint (to set screens for cutting guards, get offensive rebounds, post up, etc.).

Clusters 7 and 8 contain screen actions above the wings, usually executed by ball-handlers. The cluster-profile data supports this observation, as these clusters only account for 10% of total movements for all players, but over 20% for point guards.

Clusters 9, 10, 11, and 12 contain screen actions and movements above the wings, with clusters 11 and 12 being closer to the sidelines. These movements are usually executed by good ball-handlers and shooters. Supporting this, these clusters account for 29% of movements by point guards, 28% of movements by shooting guards, and 35% of movements by small forwards. 8% of all power forwards' movements and less than 3% of centers' movements fall into these clusters.

Clusters 13 and 14 contain movements along the sidelines to each corner, usually executed by spot-up shooters. Point guards bring the ball up the floor and run plays to pass it to one of these spots. These clusters account for almost 20% of movements by shooting guards and small forwards, compared to just under 11% of total movements. These clusters account for less than 5% of movements by point guards.

Cluster 17 has movements indicative of screen action in the paint, spanning from the high post to the restricted area. A big man usually sets a pick in this area and rolls to the basket. This observation is supported by the corresponding cluster-profile for centers (12% of their movements) and power forwards (9% of their movements). Cluster 18 has short, lateral movements in the high post. It accounts for around 10% of movements for both centers and power forwards, players who set high-screens in that area. Cluster 17 and cluster 18 each account for 5% of total movements across all players.

5.1.3 Trends in individual cluster-profiles throughout the game

We looked at a quarter-by-quarter breakdown of each player's cluster-profile, with the hope of gaining insight into how player movements evolve over the course of a game. Figure 5.3 shows the average cluster-profile for all players, broken down by quarter. There is little variation between quarters in the average cluster-profile, implying that any variation that does occur in an specific player's cluster-profile could be significant. Table 5.3 has the number of trajectory-images in our dataset for each quarter.

Figure 5.3

Cluster-profile of all player's movements, separated by quarter

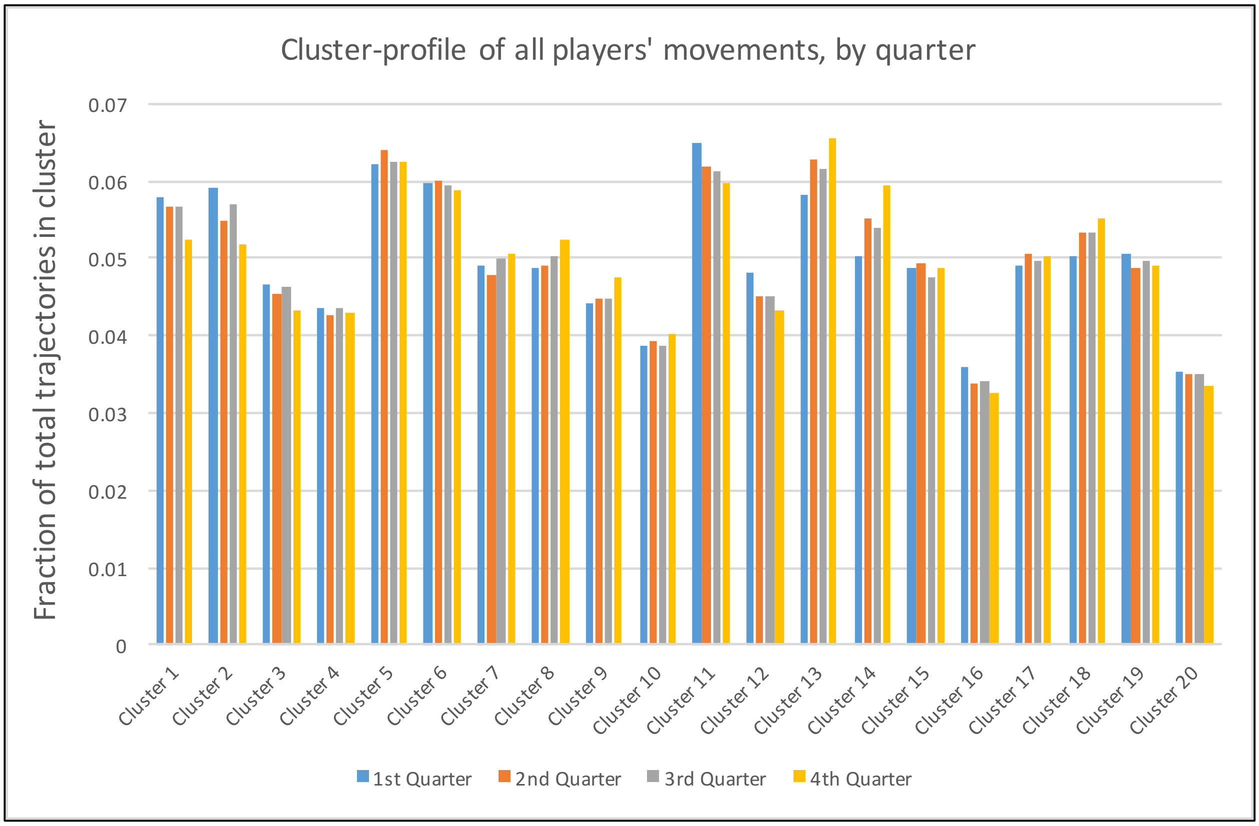


Table 5.3

Number of trajectory-images per quarter in our dataset

Quarter	#Trajectory-images
1	702,640
2	689,935
3	690,005
4	673,515

To get a sense of how similar each player's movements are across the game, we introduce a metric called cross-quarter-homogeneity. We once again use Hellinger distance as the similarity metric between two profiles. Table 5.4 reports the ten players with the highest cross-quarter-homogeneity, and Table 5.5 reports the ten players with the lowest cross-quarter-homogeneity. The per-quarter cluster-profiles for the players in ***bold*** are shown in Appendix D.

$$Hellinger(Q_a, Q_b) = \frac{1}{\sqrt{2}} * \left\| \sqrt{Q_a} - \sqrt{Q_b} \right\|_2$$

$$CQH(P) = Cross\_Quarter\_Homogeneity(P) = 1 - \frac{\sum_{i,j} Hellinger(Q_i, Q_j)}{6}$$

$$i, j \in \{\{1,2\}, \{1,3\}, \{1,4\}, \{2,3\}, \{2,4\}, \{3,4\}\}$$

Table 5.4

Ten players with highest cross-quarter-homogeneity (most homogenous across quarters)

Rank	Player	CQH	Rank	Player	CQH
1	Jeremy Lin	0.9512	6	Enes Kanter	0.9473
2	***Ben McLemore***	***0.9509***	7	Khris Middleton	0.9472
3	Chandler Parsons	0.9503	***8***	***Robin Lopez***	***0.9471***
4	Nicholas Batum	0.9493	9	Wesley Matthews	0.9460
5	Dion Waiters	0.9476	10	Nikola Vucevic	0.9454

Table 5.5

Ten players with lowest cross-quarter-homogeneity (least homogenous across quarters)

Rank	Player	CQH	Rank	Player	CQH
1	Taj Gibson	0.8867	6	Carmelo Anthony	0.9025
2	Harrison Barnes	0.8937	***7***	***Dwyane Wade***	***0.9105***
3	KC Pope	0.8990	8	Jonas Valanciunas	0.9146
4	LeBron James	0.9006	9	DeAndre Jordan	0.9146
5	***Kevin Durant***	***0.9018***	10	PJ Tucker	0.9163

The players with the lowest cross-quarter-homogeneity values have the most variance in these cluster-profiles between quarters. This variance tends to manifest itself in a few clusters rather than across the entire cluster-profile. For example, Taj Gibson's cluster distribution has over an 8% decrease in clusters 1 and 2 (movements from the top of the key to the right/left block) between the first and fourth quarters, and a 6% increase in cluster 5. The remaining clusters remain relatively steady between quarters.

Many good shooters have a significantly higher fraction of their movements for clusters 13 and 14 (running along either sideline to the corner) in the 4th quarter than in the 3rd quarter, indicating that these players may utilize their shooting prowess more in late-game situations. Table 5.6 shows these cluster-profile-values for a few players.

Table 5.6

- *Columns 1,2, 4, 5 are the fraction of each player's total trajectory-images in clusters 13/14.*
- *Columns 3, 6 are the %increase in a player's movements between the 3rd and 4th quarters.*
- *Columns 7, 8 are the number of total trajectory-images for the 3rd and 4th quarters.*

	1	**2**	**3**		**4**	**5**	**6**		**7**	**8**
Player	Cluster 13: 3rd quarter	Cluster 13: 4th quarter	% increase		Cluster 14: 3rd quarter	Cluster 14: 4th quarter	% increase		nPoss: 3rd quarter	nPoss: 4th quarter
Trevor Ariza	.135	.168	+24.4%		.105	.110	+4.8%		3,593	2,596
Klay Thompson	.120	.133	+10.8%		.094	.130	+38.3%		3,322	2,409
Kyle Korver	.071	.081	+14.1%		.156	.210	+34.6%		3,057	2,533

Cluster 19 contains movements of players running past halfcourt, with a screen action at the top of the key. Many players who are primary ball-handlers for their teams have a significantly higher fraction of their movements in cluster 19 in the 4th quarter than in the 3rd quarter. In the fourth quarter of close games, teams tend to play more in the half-court and rely on these ball-handlers to create plays from the top of the key. Table 5.7 shows the cluster-profile-values for cluster 19 for a few players.

Table 5.7

- *Columns 1,2 are the fraction of each player's total trajectory-images in cluster 19.*
- *Column 3 is the %increase in a player's movements between the 3rd and 4th quarters.*
- *Columns 4,5 are the number of total trajectory-images for the 3rd and 4th quarters.*

	1	**2**	**3**		**4**	**5**
Player	Cluster 19: 3rd quarter	Cluster 19: 4th quarter	% increase		nPoss: 3rd quarter	nPoss: 4th quarter
James Harden	.143	.160	+11.9%		3,947	2,373
LeBron James	.067	.078	+16.4%		3,662	2,361
Damian Lillard	.115	.139	+20.9%		3,834	2,855
John Wall	.131	.162	+23.7%		4, 058	2,580
Victor Oladipo	.085	.106	+24.7%		3,106	2,662
Goran Dragic	.069	.091	+31.9%		3,396	2,405
Stephen Curry	.084	.112	+33.3%		3,716	2,055

Cluster 7 and Cluster 8 contain lateral movements and screen actions above either wing. Point guards and shooting guards have a significant fraction of their movements in these clusters. These players can either make a play with the ball as a triple-threat or can catch-and-shoot after coming off a screen. Several guards have a higher fraction of their movements for these clusters in the 4^{th} quarter than in the 3^{rd} quarter. Table 5.8 shows these cluster-profile-values for a few players.

Table 5.8

- *Columns 1,2,4,5 are the fraction of each player's total trajectory-images in clusters 7/8*
- *Columns 3,6 are the %increase in a player's movements between the 3^{rd} and 4^{th} quarters.*
- *Columns 7,8 are the number of total trajectory-images for the 3^{rd} and 4^{th} quarters*

	1	**2**	**3**		**4**	**5**	**6**		**7**	**8**
Player	Cluster 7: 3^{rd} quarter	Cluster 7: 4^{th} quarter	% increase		Cluster 8: 3^{rd} quarter	Cluster 8: 4^{th} quarter	% increase		nPoss: 3^{rd} quarter	nPoss: 4^{th} quarter
JR Smith	.077	.082	+6.49%		.088	.109	+23.9%		3,026	2,159
Mike Conley	.097	.107	+10.3%		.116	.146	+25.9%		3,064	2,113
Stephen Curry	.117	.151	+29.1%		.104	.117	+12.5%		3,716	2,055
Russell Westbrook	.134	.151	+12.7%		.113	.158	+39.8%		2,706	1,745

5.1.4 Trends in individual cluster-profiles throughout the season

We computed cluster-profiles for each of the top 100 players on a monthly basis, for the months November through March (the months with regular season basketball from start to end). We did not find any general trends as we did in section 5.1.3. A single player's movement over the course of a season is influenced by many factors – working back into form from injury; being asked to fill in different roles due to roster and coaching personnel changes, and so on. Because of this, it is difficult to find general trends across players of the same position or the same team, and to pinpoint any one factor as being the sole reason why a player's cluster-profile differs from month to month.

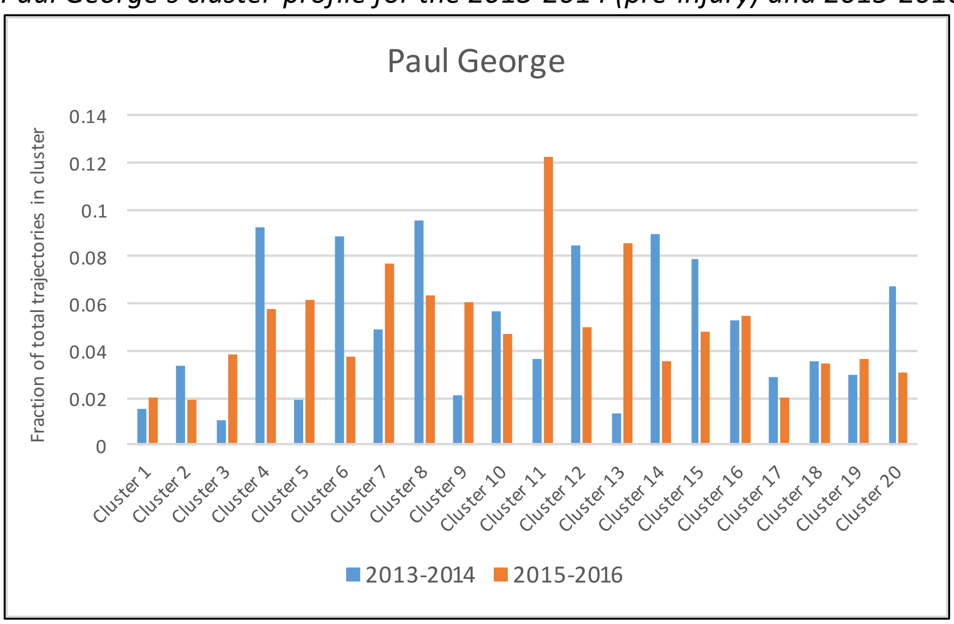
5.2 Case studies into effects of injures and roster/coaching changes on player movement

In this section, we look at a few specific cluster-profiles to study the effects of injures, trades, and roster changes on individual players' patterns of movement. In doing so, we hope to demonstrate the efficacy of cluster-profiles as an effective tool for analyzing player movement.

5.2.1 Paul George's 2014 injury

During a 2014 off-season scrimmage with the US national basketball team, Paul George landed awkwardly at the base of a basket stanchion and suffered a compound fracture of both bones in his lower right leg. He missed most of the 2014-2015 season, only playing six games and averaging 15 minutes. He made a full recovery for the 2015-2016 season, playing in 81 games and averaging career highs in points and assists. Figure 5.4 shows George's cluster-profile for the 2013-2014 and 2015-2016 seasons.

Figure 5.4
Paul George's cluster-profile for the 2013-2014 (pre-injury) and 2015-2016 (post-injury) seasons



We observe that Paul George made significantly more plays in the right half of the court during the 2015-2016 season than the 2013-2014 season. Aggregate movement in the right half of the court – clusters 1, 3, 5, 7, 9, 11, and 13 – increased from 16.5% to 46.6%, whereas movements in the left half of the court – clusters 2, 4, 6, 8, 10, 12, and 14 – decreased from 54.1% to 31.0%.

5.2.2 2014-2015 Cleveland Cavaliers

Following LeBron James's 2014 decision to return to the Cleveland Cavaliers after a four-year stint with the Miami Heat, the Cavaliers underwent a roster and coaching personnel overhaul. Anthony Bennett and Andrew Wiggins (the Cavaliers' 2014 draft pick) were traded for Kevin Love. Brendan Haywood, James Jones, Shawn Marion, Mike Miller, Timofey Mozgov, Kendrick Perkins, Iman Shumpert, and JR Smith were all new additions through free agency or trade.

In this subsection, we look at the cluster-profiles of the Cavaliers' All-Star core – LeBron James, Kyrie Irving, Kevin Love – for the 2013-2014, 2014-2015, and 2015-2016 seasons, and make observations on significant changes.

The Cavaliers had three coaches over these three seasons. They fired Mike Brown towards the end of the 2013-2014 season and hired Mike Blatt. Mike Blatt was fired midway through the 2015-2016 season and replaced by Tyronn Lue, who was the Cavaliers' associate head coach from the start of the 2014-2015 season.

LeBron James is one of the greatest scorers and playmakers the game has seen and has been the focal point of his team's offense ever since he entered the league, often singlehandedly controlling the pace and flow of a game. His cluster-profile (Figure 5.5) changed the least amongst the Cavaliers' All-Star core.

Clusters 5 and 6 – movements on either side of the paint – went down from 11.9% to 6.6% of LeBron's trajectory-images between his last season on the Heat and first season back with the Cavaliers. James played the power forward position frequently for the Miami Heat in a small-ball lineup featuring Chris Bosh at center. With a frontcourt featuring Kevin Love and Tristan Thompson, James did not play this position as much in Cleveland.

Figure 5.5
LeBron James's cluster-profiles for 2013-2014, 2014-2015, and 2015-2016

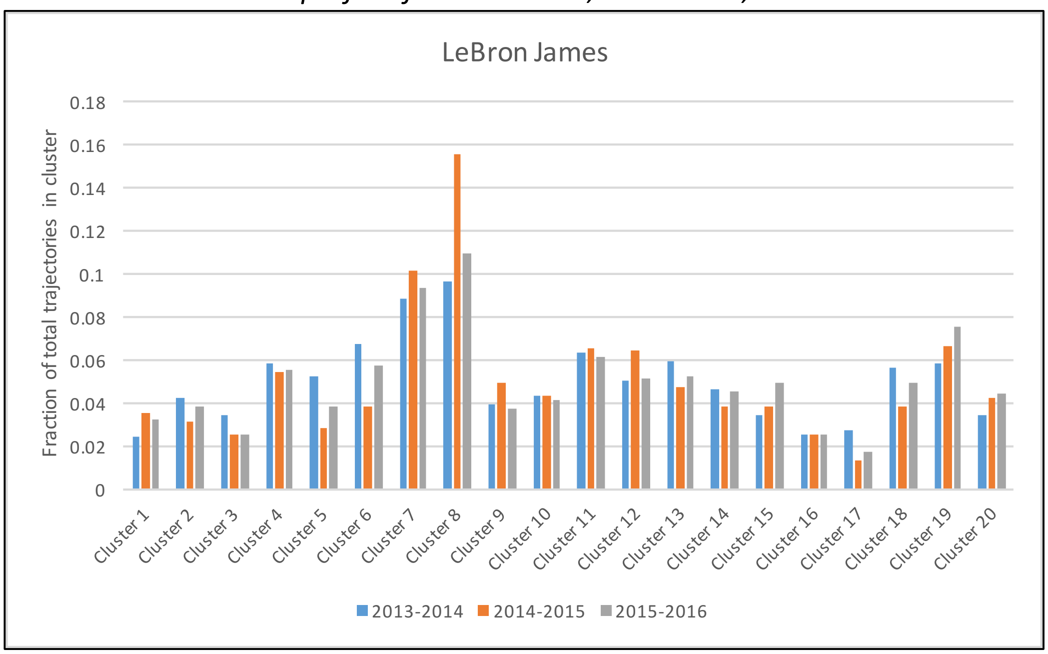


Table 5.9
James per 36 minutes, 2013 to 2016

	PTS	REB	AST	FG%
2013-2014	25.9	6.6	6.1	.567
2014-2015	25.5	6.0	7.4	.488
2015-2016	25.5	7.5	6.8	.520

Clusters 7 and 8 – screen actions above either wing – went up from 18.4% to 25.7% of James' total movements during his first season in Cleveland, then decreased to 20.3% the following season.

Many clusters – 1, 2, 5, 6, 9, 12, 13, 14 – experienced a small decrease or increase between 2013-2014 and 2014-2015 but returned to their previous value during 2015-2016. Potential contributing factors to this phenomenon are 1) James adjusted his game more during his first season in Cleveland as he fit in with co-stars Irving and Love, and 2) Tyronn Lue adapted his offense more to LeBron's game than David Blatt did the previous season.

Figure 5.6

Kyrie Irving's cluster-profiles for 2013-2014, 2014-2015, and 2015-2016

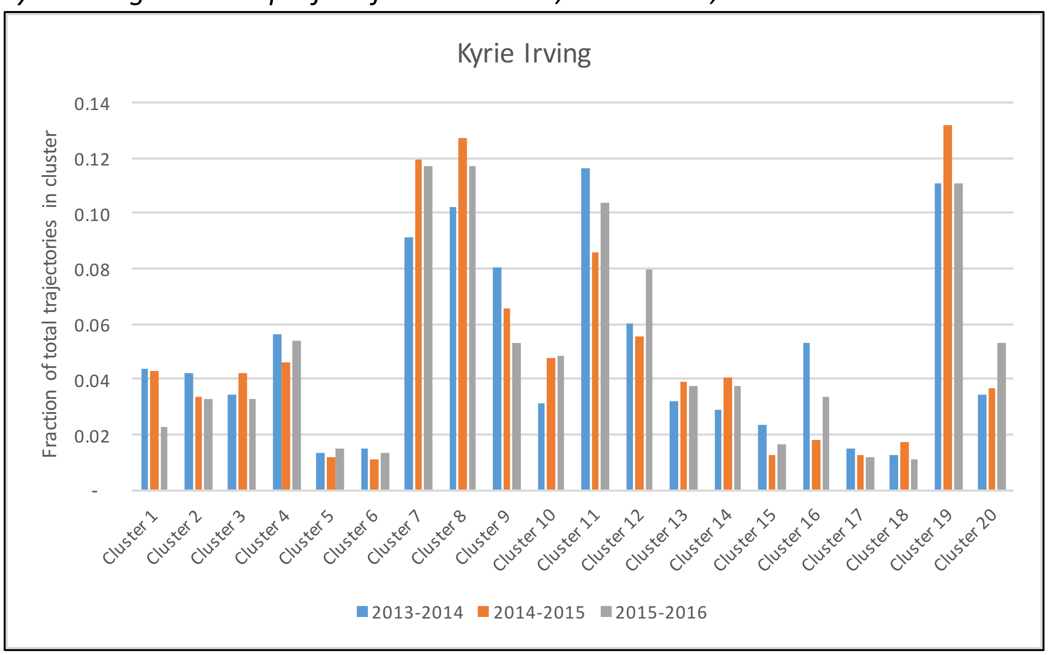


Table 5.10

Irving per 36 minutes, 2013 to 2016

	PTS	REB	AST	FG%
2013-2014	21.3	3.7	6.2	.430
2014-2015	21.5	3.1	5.1	.468
2015-2016	22.5	3.4	5.4	.448

Apart from a few clusters, Irving's cluster-profile (Figure 5.6) did not change much over these three seasons. He was the focal point of the Cavaliers' offense for the first three seasons of his career, and remained a primary option until he was traded in 2017.

Clusters 7 and 8 – movements of a screen action above either wing – went up from 19.4% to 24.7% of Irving's movements between 2013-2014 and 2014-2015. These clusters also increased for LeBron James and Kevin Love. The Cavaliers ran more screen actions above the wings once they acquired two All-Stars who are both proficient shooters and passers.

Figure 5.7

Kevin Love's cluster-profiles for 2013-2014, 2014-2015, and 2015-2016

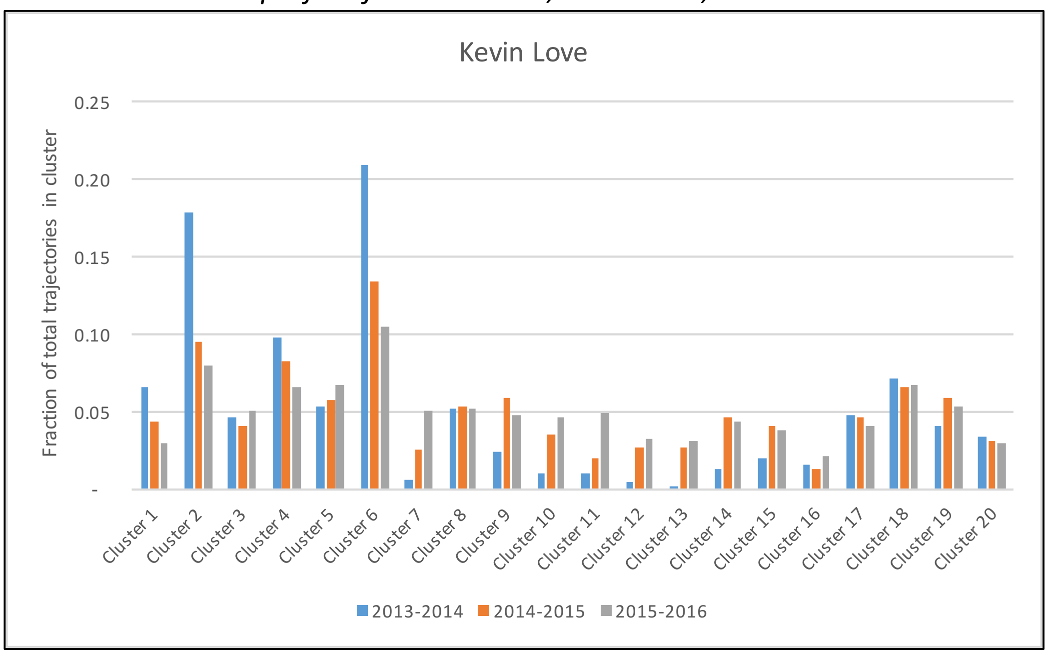


Table 5.11

Love per 36 minutes, 2013 to 2016

	PTS	REB	AST	FG%
2013-2014	25.9	12.4	4.4	.457
2014-2015	17.5	10.4	2.4	.434
2015-2016	18.3	11.3	2.8	.419

Kevin Love was the focal point of the Timberwolves' offense from 2010 to 2014. His role with the Cavaliers changed drastically as he became the third option on offense after LeBron

James and Kyrie Irving. This manifests itself in his box-score statistics (Table 5.11) as well as his cluster-profile (Figure 5.7).

Love liked to post up on the left block in Minnesota, a place he did not occupy as much in Cleveland. Cluster 2 (run to the left block from the top of the key) decreased from 17.9% to 9.4% of Love's movements during his first season in Cleveland. Cluster 6 (movement left of the paint) also decreased from 20.9% to 13.3% of Love's movements during this time span.

Clusters 9 through 12 are perhaps the biggest indicator of the vastly different role Love had for the Cavaliers than he did for the Timberwolves. Clusters 9 and 10 contain movements along either wing; clusters 11 and 12 also contain movements along the wings, closer to the sidelines. These four clusters accounted for just 4.8% of Love's trajectory-images from 2013-2014, a number that increased to 14.0% in 2014-2015 and 17.5% in 2015-2016. Clusters 13 and 14 (movements along each sideline to the corner) also increased from 1.5% to 7.3% during this same time span. Love accepted a new role in the Cavaliers offense and became more of a perimeter player, veering away from his favorite spot near the left block.

5.2.3 2014-2015 Golden State Warriors

Mark Jackson coached the 2013-2014 Golden State Warriors to a 51-31 record. However, after losing in the first round to the Los Angeles Clippers, Jackson was fired and replaced with Steve Kerr. Kerr ran a fast, free-flowing offense, taking advantage of Draymond Green's court vision and Steph Curry's and Klay Thompson's shooting range, which extends back to 30 feet. He led the Warriors to a 65-17 record and an NBA championship in his first season as head coach. The Warriors won an NBA-record 73 games in 2015-2016, but lost to the Cleveland Cavaliers in a tightly contested 7-game NBA finals. In this subsection, we look at the regular season cluster-profiles of the Warriors' All-Star core – Stephen Curry, Klay Thompson, Draymond Green – and make observations on significant differences.

Stephen Curry won back-to-back league MVP (Most Valuable Player) awards during Steve Kerr's first two seasons as head coach. His second MVP was the first unanimous selection in league history, after a record-shattering season that saw Curry make over 400 3-pointers and lead the league in many statistical categories on offense.

Figure 5.8

Stephen Curry's cluster-profiles for 2013-2014, 2014-2015, and 2015-2016

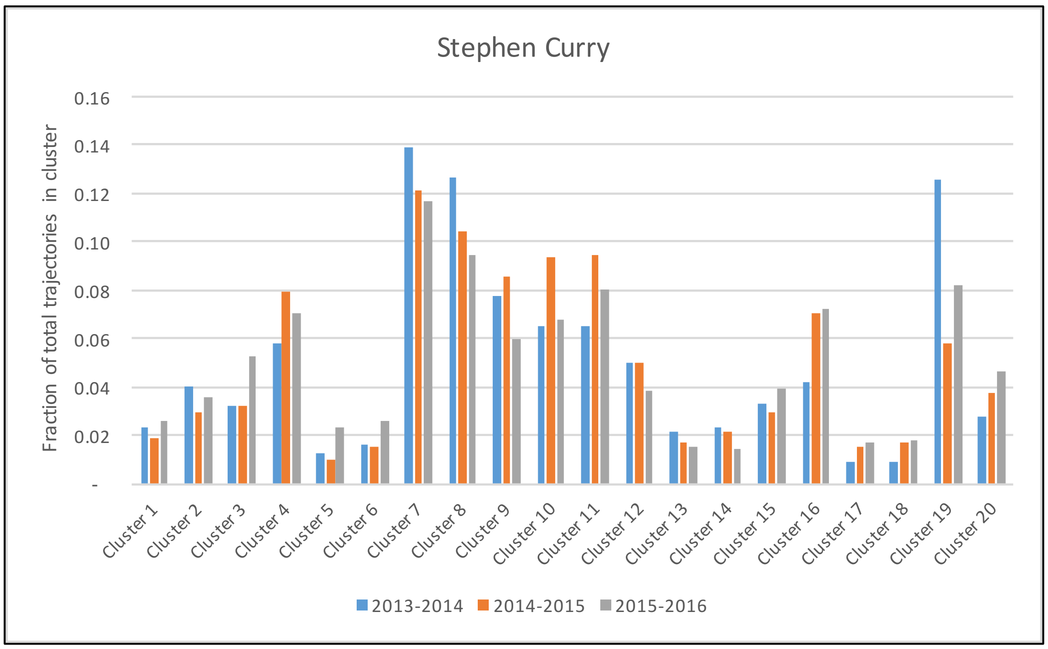


Figure 5.8 displays Curry's cluster-profile for the three seasons in our analysis. Curry's movements in cluster 19 (high-screen at the top of the key) more than halved from 12.6% in 2013-2014 to 5.8% in 2014-2015, as Draymond Green picked up more ball-handling duties in Kerr's offense. Curry was involved in more off-the-ball screens and movements in Kerr's offense. Cluster 16 (movement along the baseline) increased from 4.2% to 7.1% during 2014-2015, and remained steady the next season.

Table 5.12
Curry per 36 minutes, 2013 to 2016

	PTS	REB	AST	FG%
2013-2014	23.7	4.2	8.4	.471
2014-2015	26.2	4.7	8.5	.487
2015-2016	31.7	5.7	7.0	.504

Klay Thompson, the Warriors second All-Star guard, is one of the best 3-point shooters in NBA history. He is also a capable defender, often picking up the opposing team's toughest guard assignment. His points per 36 minutes increased the most amongst the Warriors' core, as Steve Kerr incorporated Thompson's sharpshooting abilities in his offense. Figure 5.9 displays Thompson's cluster-profiles for 2013 to 2016.

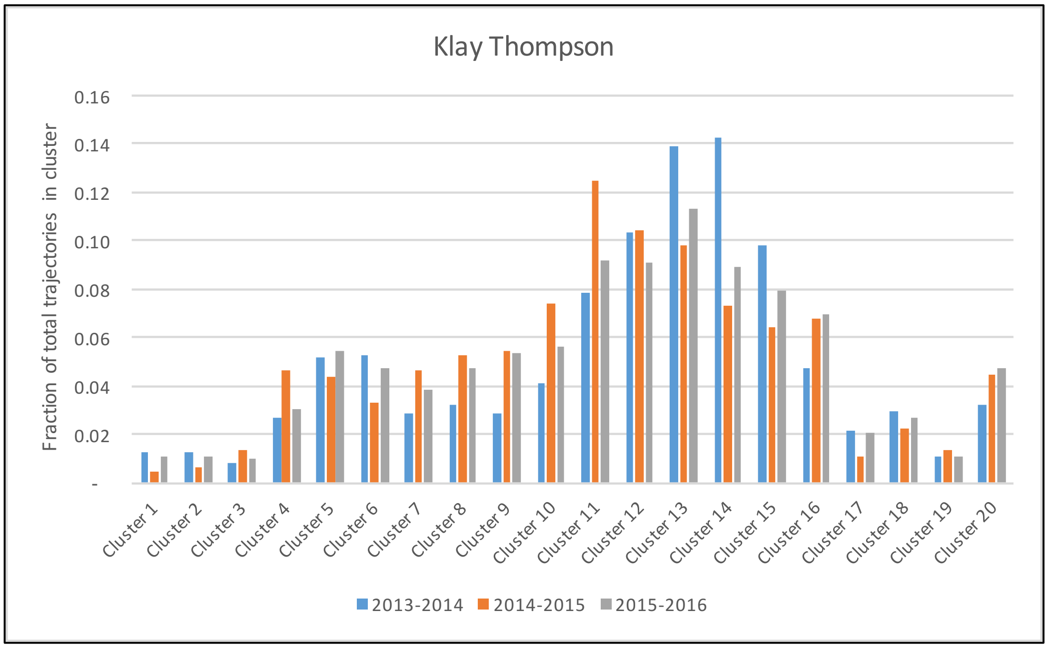
Table 5.13
Thompson per 36 minutes, 2013 to 2016

	PTS	REB	AST	FG%
2013-2014	18.7	3.1	2.3	.444
2014-2015	24.5	3.6	3.3	.463
2015-2016	23.9	4.1	2.2	.470

Clusters 13 and 14 – movements along either sideline to the 3-point corners – accounted for over 28% of Thompson's trajectory-images in 2013-2014. This number reduced to 17.2% during Kerr's first season as coach. Kerr had a more free-flowing offense and employed more off-ball screens and movements to free up his 3-point shooters. This is also supported by cluster 16 – an arc-like movement between sidelines – which for Thompson increased from 4.7% to nearly 7% during Kerr's first two seasons.

Figure 5.9

Klay Thompson's cluster-profiles for 2013-2014, 2014-2015, and 2015-2016



Thompson was involved in 14.3% more screen actions and movements above the wings during 2014-2015 than the previous season. These movements are captured by clusters 7 and 8 (screen actions above the wings), clusters 9 and 10 (movements above the wings), and clusters 11 and 12 (movements above the wings, closer to the sidelines). Kerr's decision to use Thompson less as a spot-up shooter and more as an off-ball threat led to a huge boost for the Warriors' offense.

Draymond Green is one of the most versatile players in the league. He can pass the ball, set screens, and fill a variety of other roles on offense. He can also defend all five positions and was the league's Defensive Player of the Year in 2017. This versatility is reflected in his cluster-profile (Figure 5.10), with movements spread across many clusters rather than being concentrated in a few. This resulted in many clusters increasing or decreasing slightly during the three seasons in our analysis.

Figure 5.10

Draymond Green's cluster-profiles for 2013-2014, 2014-2015, and 2015-2016

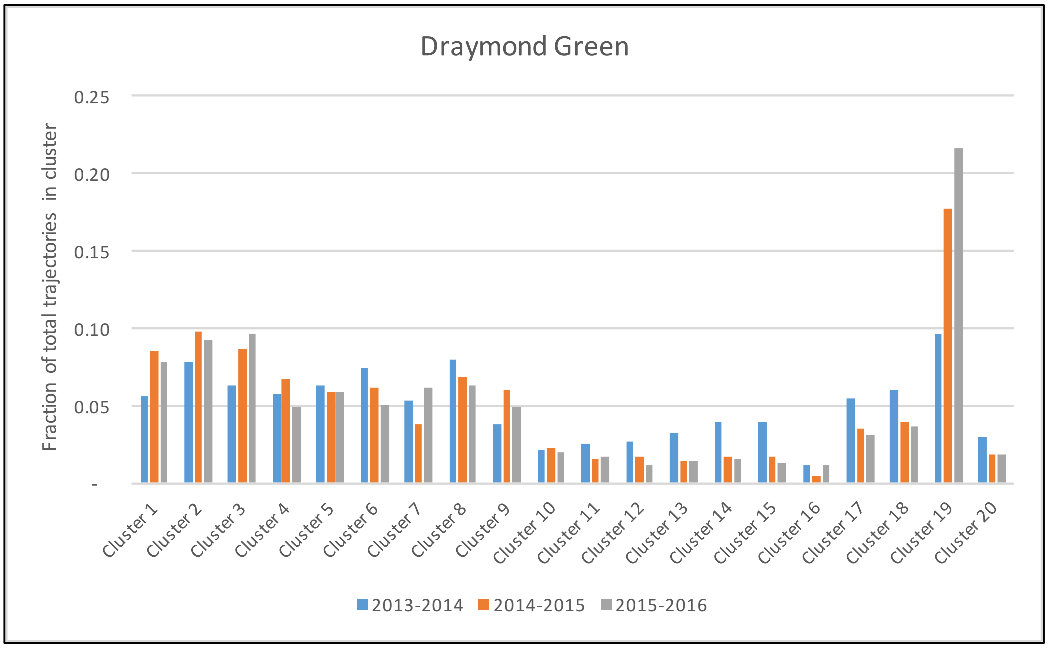


Table 5.14

Green per 36 minutes, 2013 to 2016

	PTS	REB	AST	FG%
2013-2014	10.2	8.2	3.0	.407
2014-2015	13.3	9.4	4.2	.443
2015-2016	14.5	9.9	7.7	.490

Cluster 19 – movements of a high screen at the top of the key – increased from 9.7% of Green's trajectory-images in 2013-2014 to 17.7% and 21.6% during Kerr's first two years as coach. Green brought the ball up the floor more frequently in Kerr's offense. He also had more overall ball-handling responsibilities, which is reflected in his assist numbers across these seasons (Table 5.14).

Clusters 13 and 14 – movements along the sidelines to the 3-point corners – decreased from 7.1% of Green's trajectory-images in 2013-2014 to 3% in subsequent seasons. Green is not known for his spot-up or catch-and-shoot abilities.

5.3 Case studies into changes in player movement during playoff comebacks

This section examines how player movement changes between playoff wins and playoff losses. The two playoff series we look at are the 2016 Western Conference Finals (Section 5.3.1) and the 2016 NBA Finals (Section 5.3.2). Both series featured a comeback by the winning team after being down 3 games to 1. The Warriors won three straight games (two home and one away) against the Thunder to advance to the finals, and the Cavaliers also won three straight games (one home and two away) against the Warriors to win the championship.

For each series, we present two cluster-profiles, for wins and losses, for the top three performers from each team. Figures 5.11 and 5.12 show these cluster-profiles for the 2016 Western Conference Finals. Figures 5.13 and 5.14 show these cluster-profiles for the 2016 NBA Finals.

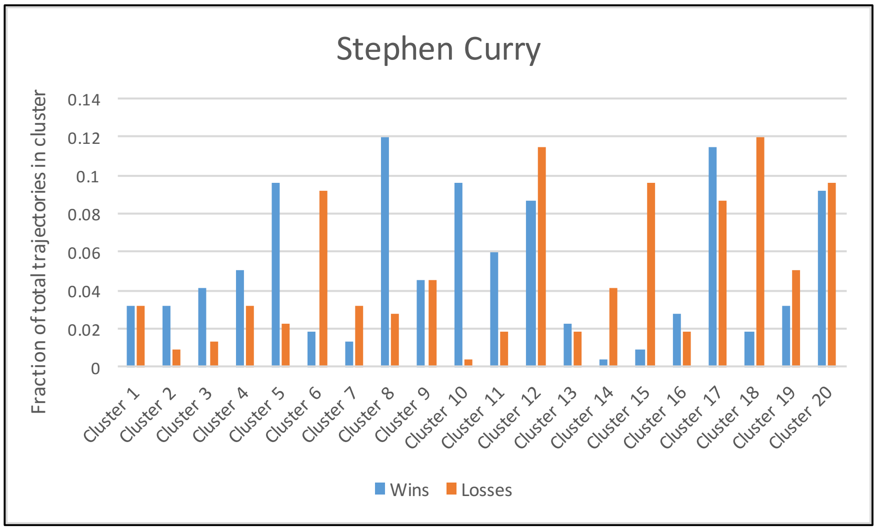
We notice that in both of these series, Klay Thompson has a much higher fraction of his trajectories in clusters 13 and 14 (movements along the sidelines to the corners) in wins than in losses. In the Western Conference Finals, nearly 20% of his trajectories fell into these clusters in the Warriors' 4 wins, as compared to only 4% in losses. In the NBA Finals, 18% of his trajectories fell into these clusters in the Warriors' 3 wins, as compared to only 3.5% in losses.

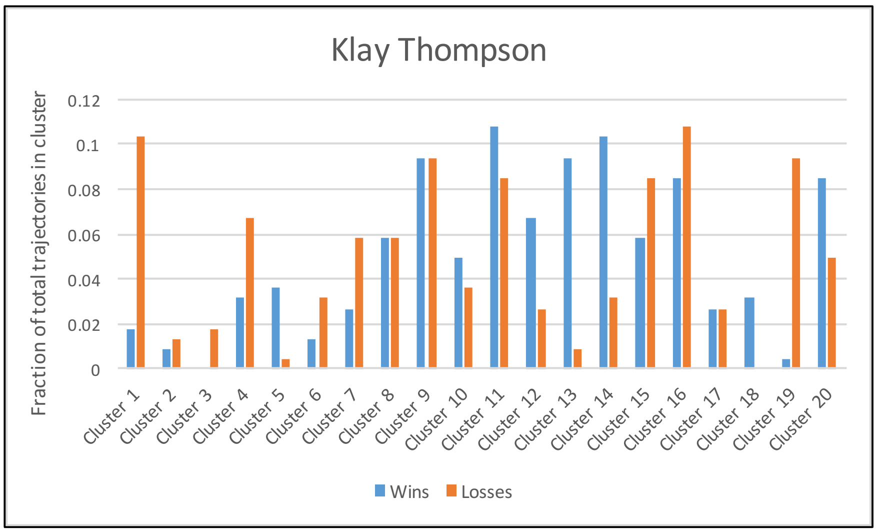
We leave any further observations and analysis up to the reader and refer them to Figure 3.7, which contains has each cluster-mediod's trajectory-image and Table 3.2, which has descriptions for each cluster-mediod.

5.3.1 2016 Western Conference Finals: Warriors def. Thunder

Figure 5.11

Cluster-profiles for Curry, Thompson, and Green for the 2016 WCF; separated by wins and losses





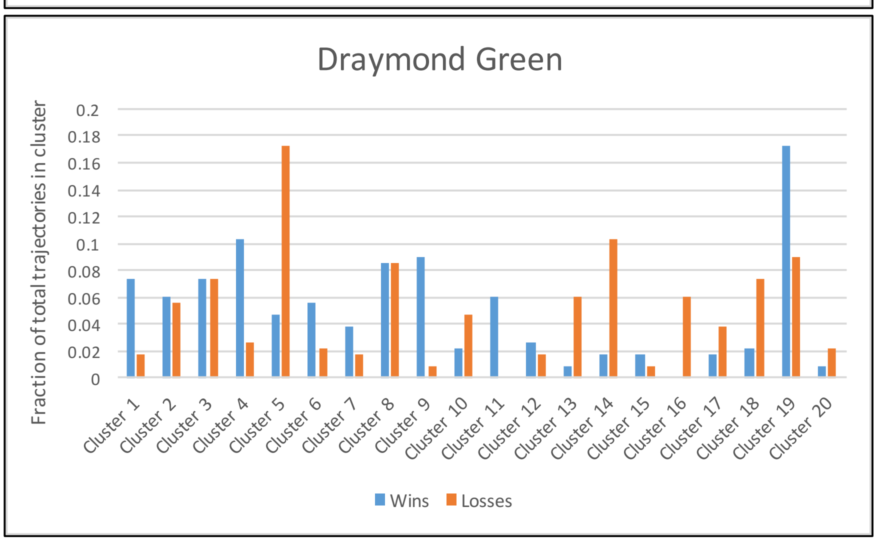
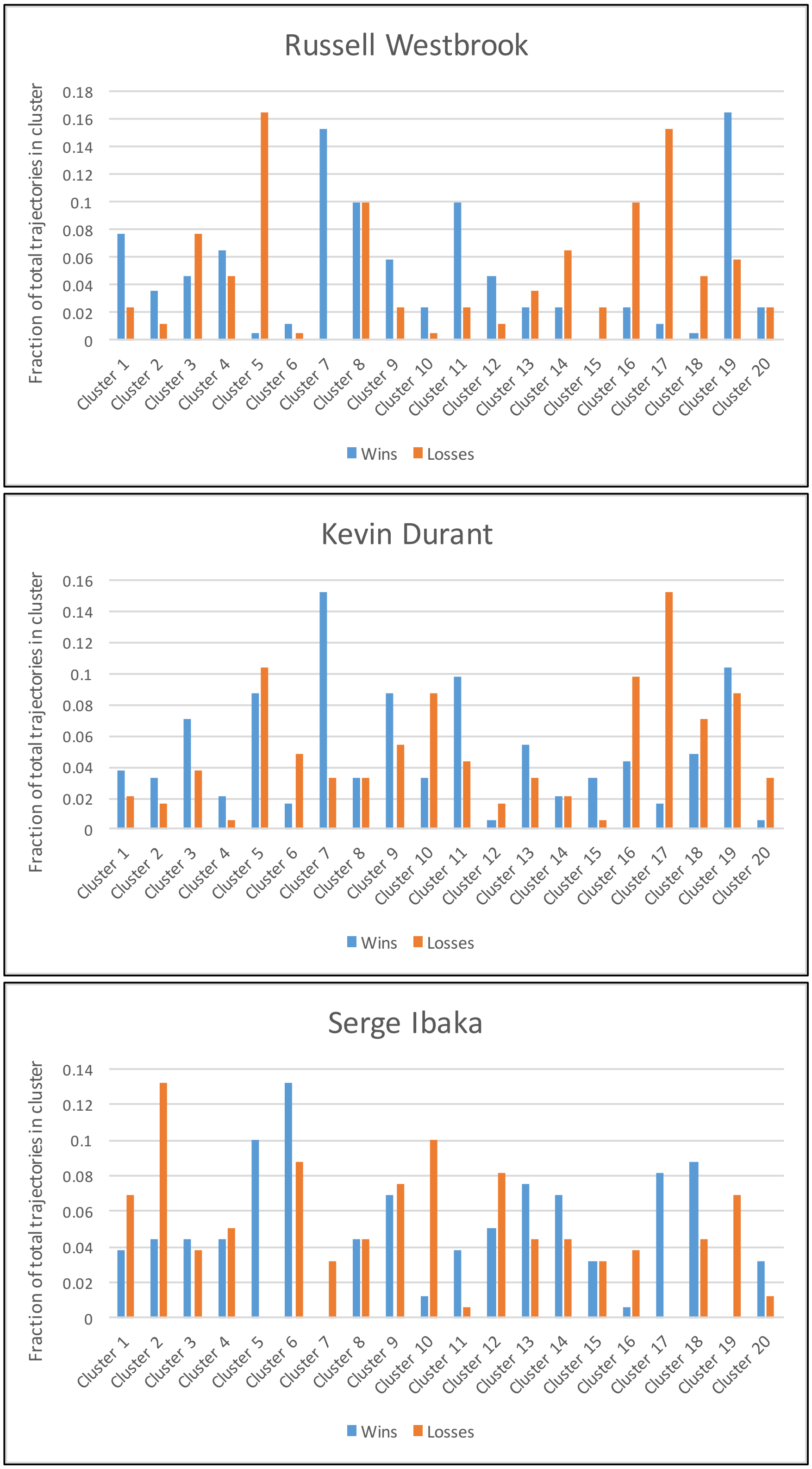


Figure 5.12

Cluster-profiles for Westbrook, Durant, and Ibaka for the 2016 WCF; separated by wins and losses



5.3.2 2016 NBA Finals: Cavaliers def. Warriors

Figure 5.13

Cluster-profiles for Curry, Thompson, and Green for the 2016 Finals; separated by wins and losses

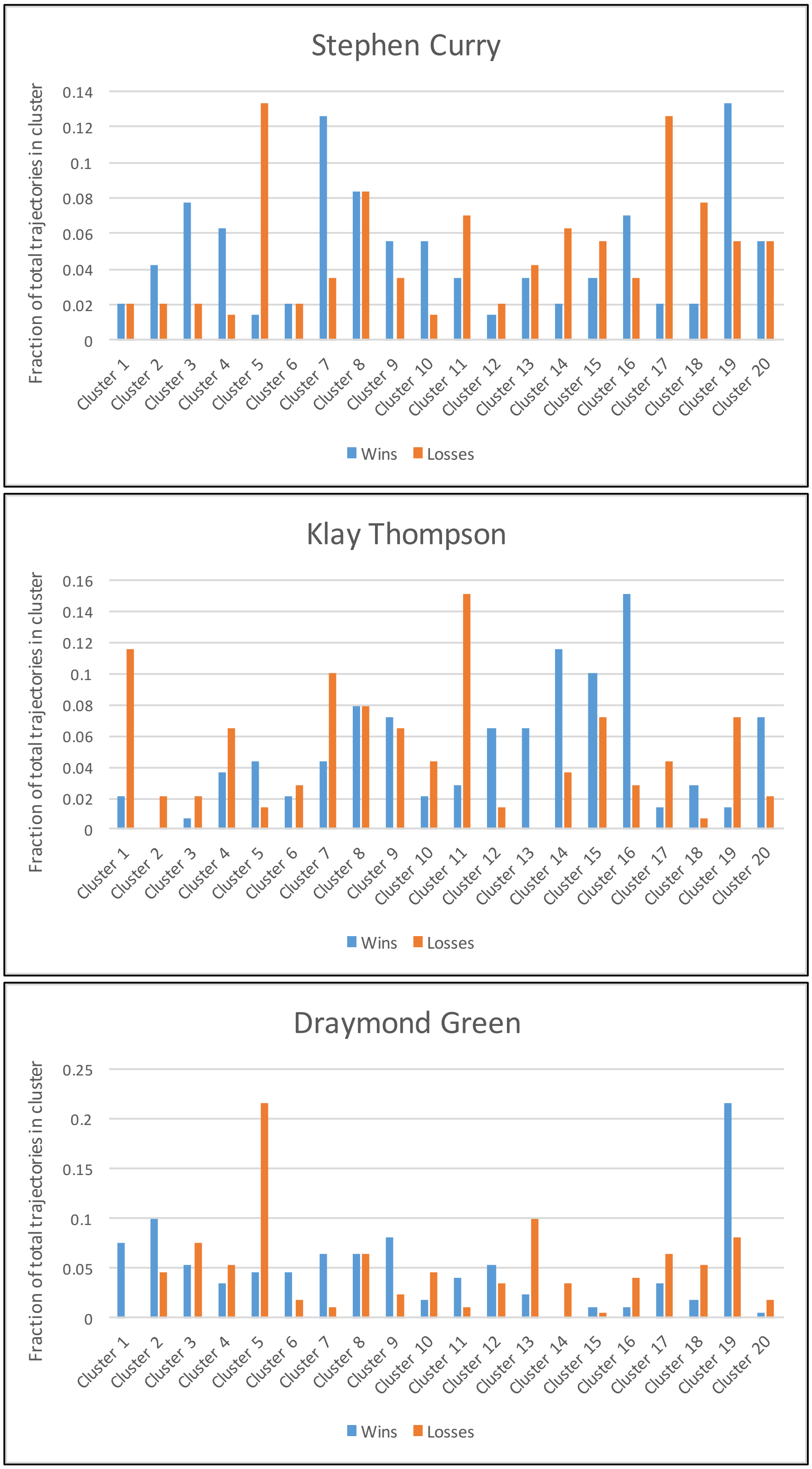
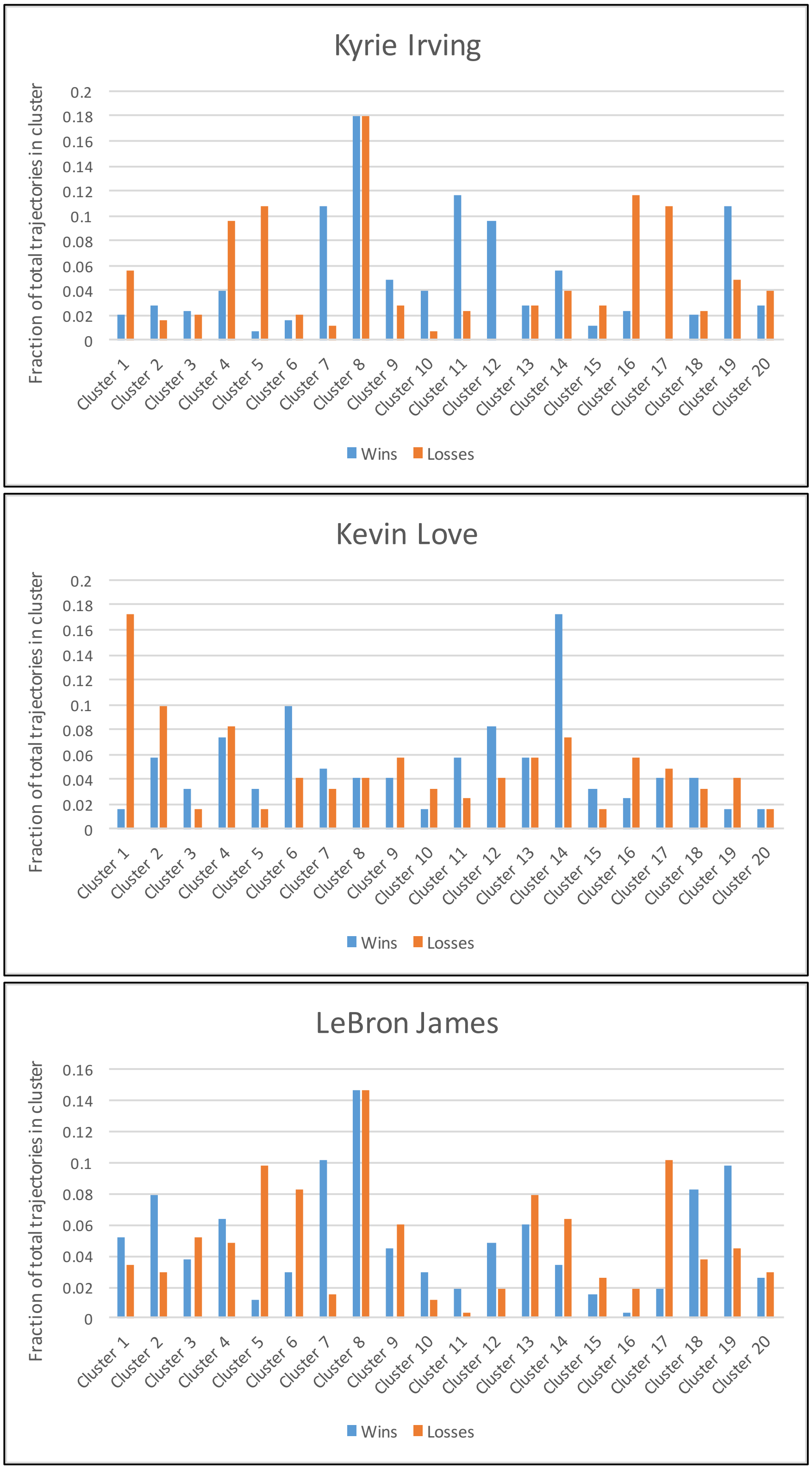


Figure 5.14

Cluster-profiles for Irving, Love, and James for the 2016 Finals; separated by wins and losses



5.4 Comparing Kobe Bryant's movements between his last season and final game

Figure 5.15

Kobe Bryant's cluster-profile for his final game (blue) and the rest of the 2015-2016 season (red)

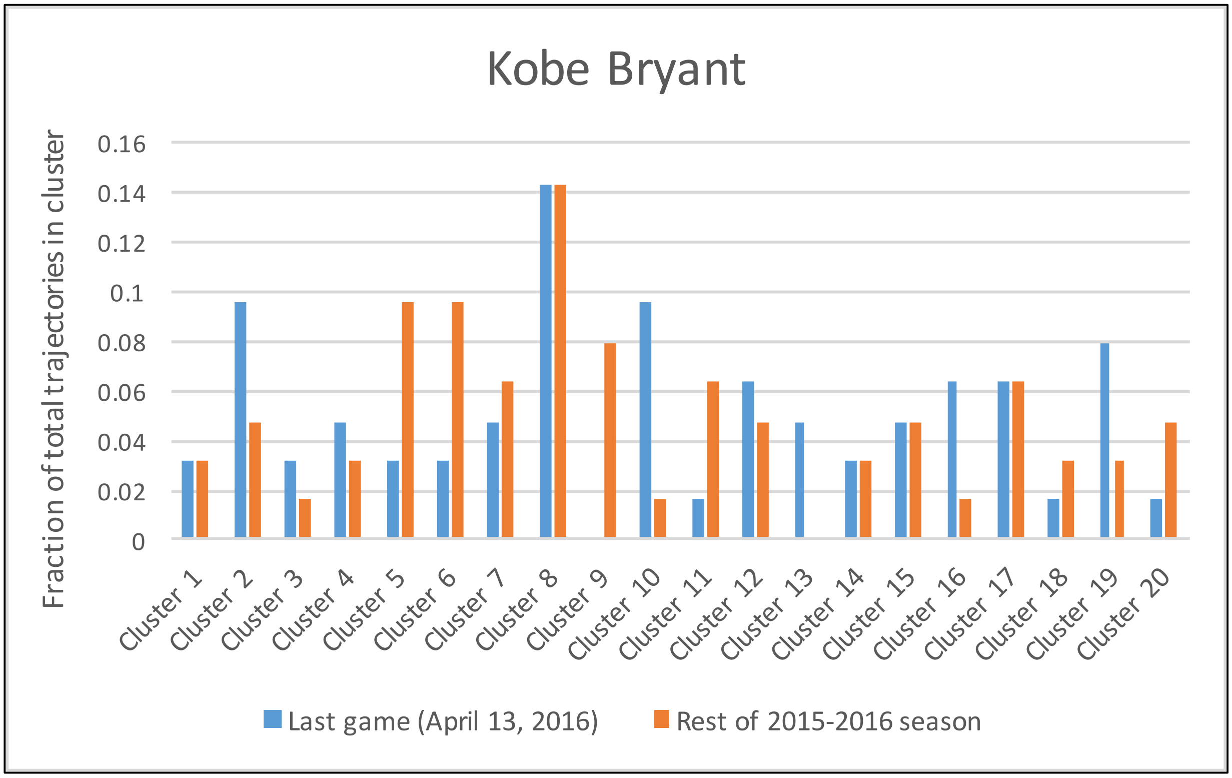


Table 5.15

Number of trajectory-images used to generate each cluster-profile in Figure 5.15

Time Period	**# Trajectory-images**
Final game (April 13, 2016)	63
Rest of 2015-2016 season	2751

Kobe Bryant is considered one of the greatest players in NBA history. He played his entire 20-season career for the Los Angeles Lakers, winning 5 championships, 2 Finals MVP awards, and one regular season MVP award. After rupturing his Achilles tendon in a game against the Warriors on April 13, 2013, Bryant struggled through the next two seasons, playing only 41 out of a possible 164 games. He managed to play 66 games during 2015-2016 after announcing in November 2015 that he would retire at the end of the season.

In this section, we compare Bryant's cluster-profile for his final game, during which he scored 60 points on a career-high 50 field goal attempts, to his cluster-profile for the rest of the 2015-2016 season. These cluster-profiles can be seen in Figure 5.15. Table 5.15 shows the number of trajectory-images used to generate cluster-profiles for these games. These are the number of possessions involving Bryant that were longer than 4 seconds long.

Clusters 5 and 6 – movements on either side of the paint – accounted for nearly 20% of Bryant's movements during the 2015-2016 season but only 6% of his movements for his final game. Cluster 19 - movements of a high screen at the top of the key – increased from 3% of Bryant's movements during the season to 8% of his movements during his final game. Due to nagging injuries and tired legs, Bryant was forced to be more of a post-up threat in his last season. These clusters suggest that he abandoned this approach during his final game to be a primary ball-handler and perimeter threat one last time.

Clusters 9 and 11 – movements above the left wing – decreased from 14.2% of Bryant's movements during the season to only 1.5% of his movements in his final game. Clusters 10 and 12 – movements above the right wing – went up from 6.3% for the season to 15.9% for Bryant's final game. These clusters suggest that Bryant played above the left wing for most of the season but played above the right wing – his preferred spot – during his final game.

6. Summary and Conclusion

We review our major concepts and frameworks section 6.1, and discuss future work and conclude in section 6.2.

6.1 Summary

We learn a low-dimensional representation of a single player's movement over the course of one possession on offense. For each possession, we use each of the five offensive players' raw time series (X,Y) coordinates to build a *trajectory-image* depicting how that player moved. These images represent let us represent all movements on offense in a common, fixed-sized input space. We construct 3 million images from three seasons of NBA player tracking data.

Comparing these images to find similar patterns of movement still remains a difficult task, which motivates a deep learning solution. We put all 3 million trajectory-images through a convolutional neural network to learn a low-dimensional encoding for each image. Each *trajectory-embedding* captures the spatial patterns in a trajectory-image, such as the start and end point, screen actions, and court coverage. Training a single model for all images lets us learn a low-dimensional encoding for each movement, in the context of all movements.

We then run K-Means over all trajectory-embeddings to find clusters of visually similar movements. Our framework can be used to compare movements across players and teams, analyze changes in player movement over time, efficiently search a database of possessions for a certain type of movement, and so forth.

6.2 Future Work

Understanding Patterns of Movement on Defense

This research has demonstrated that deep learning can be used to learn patterns of basketball movement on offense. We are very interested in applying the same methodology to defensive movements.

Database of Basketball Movements

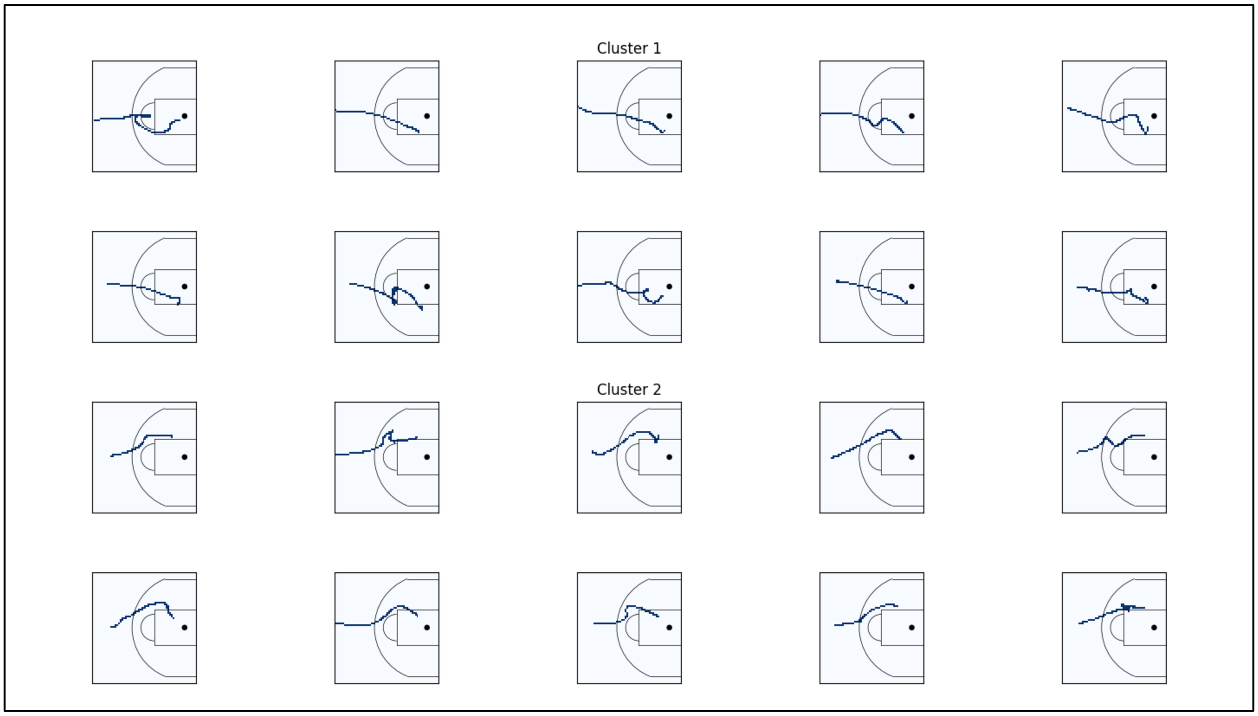
We plan on using trajectory-embeddings to build a database of basketball movements that is easily scalable and efficient to search. The trajectory-embeddings can be used in conjunction with SportVU event data and Basketball Reference data to answer many types of questions. Some examples of queries we envision this database supporting are:

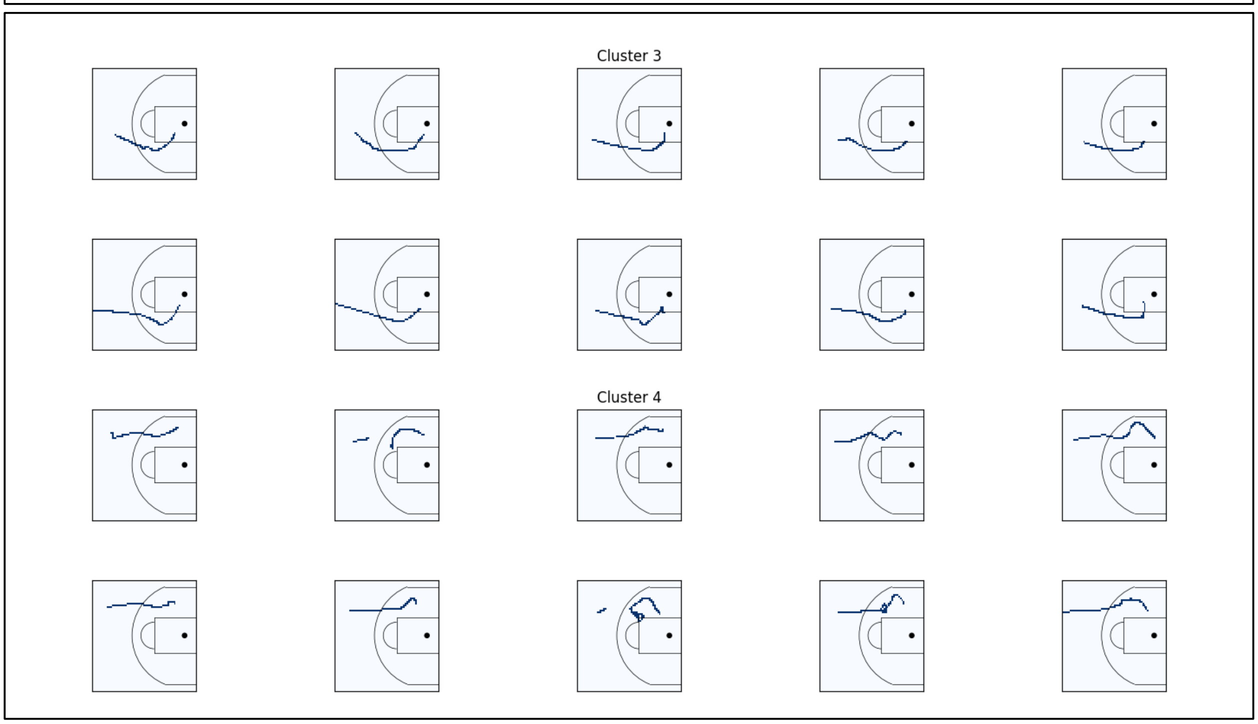
- Find all games in which movements to the basket accounted for over 25% of LeBron James's trajectory-images
- Find all possessions in which James Harden drove to the basket and passed it to the right corner for an assist
- Find all possessions in which movements are most similar to <set of 1 to 5 trajectory-images>

Appendix

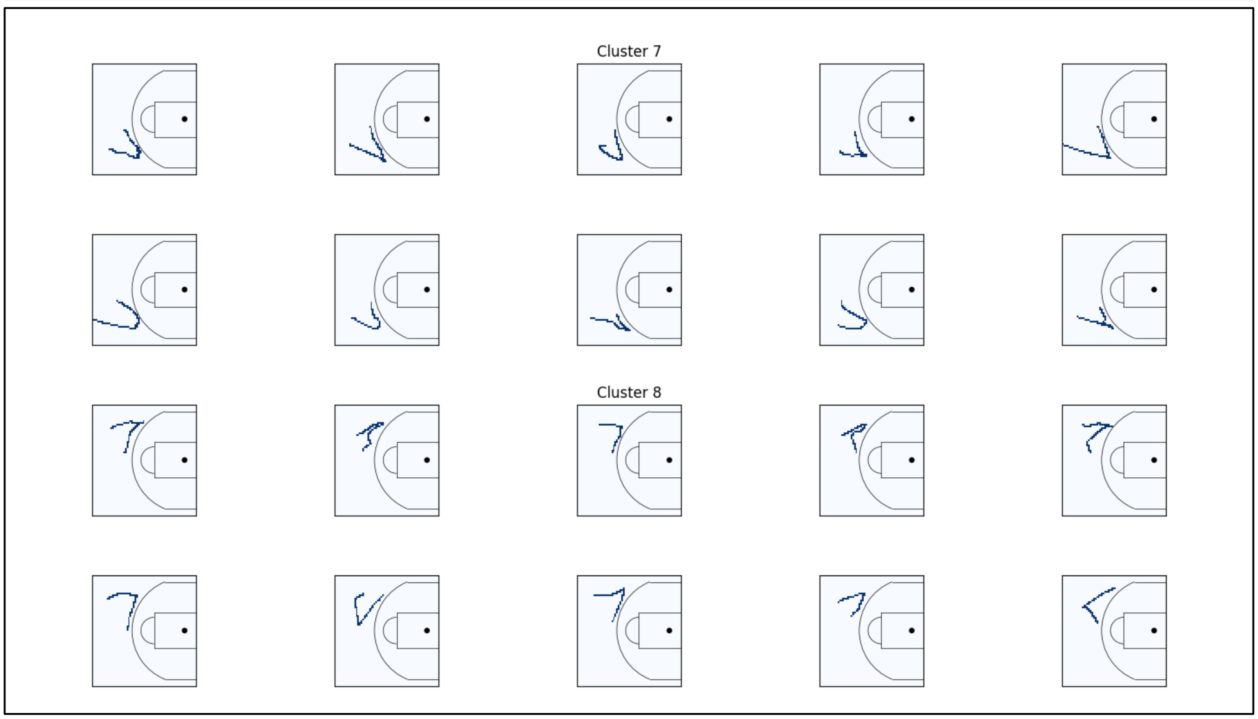
Appendix A: Trajectory-images nearest to each cluster-mean

Ten trajectory-images nearest to each cluster-mean.

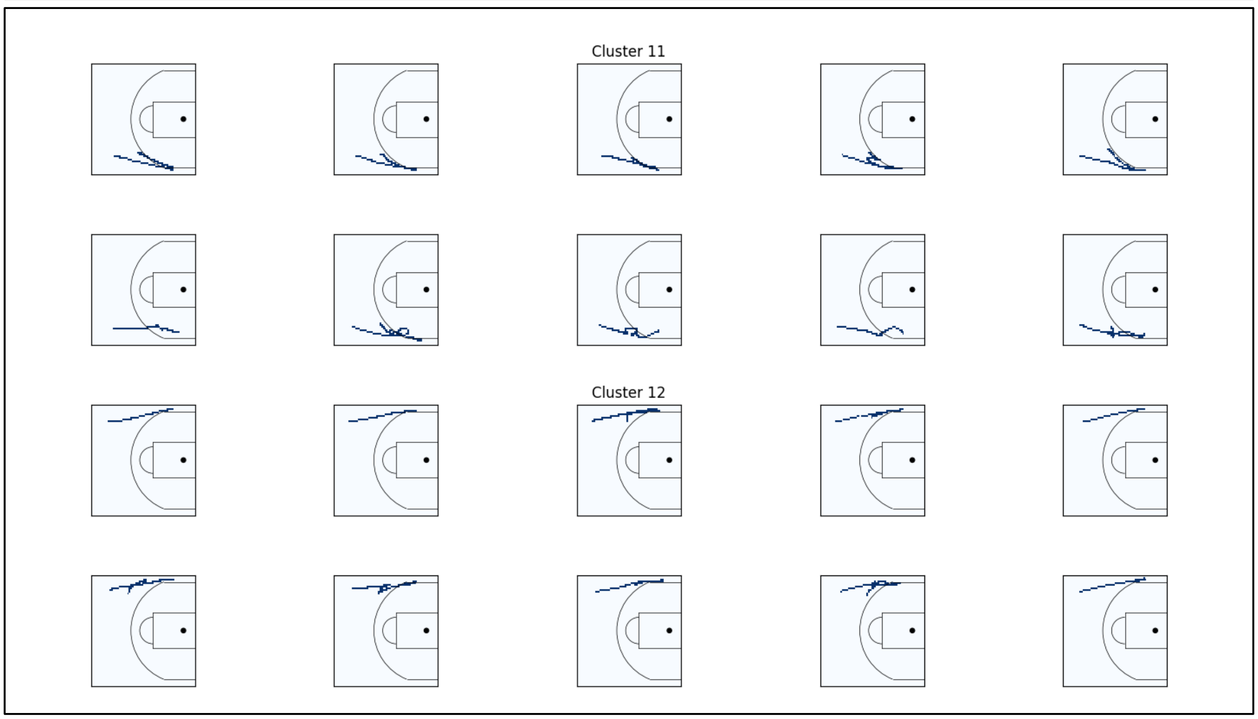





Cluster 5
Cluster 6

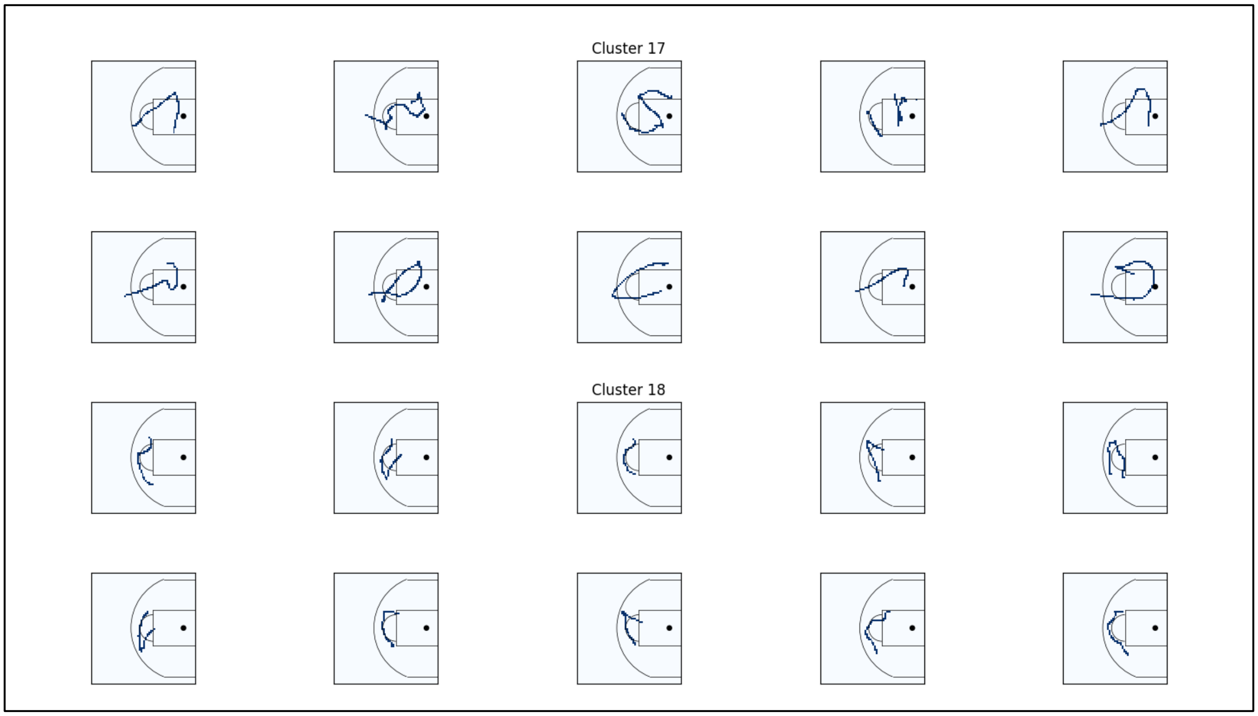

Cluster 7
Cluster 8

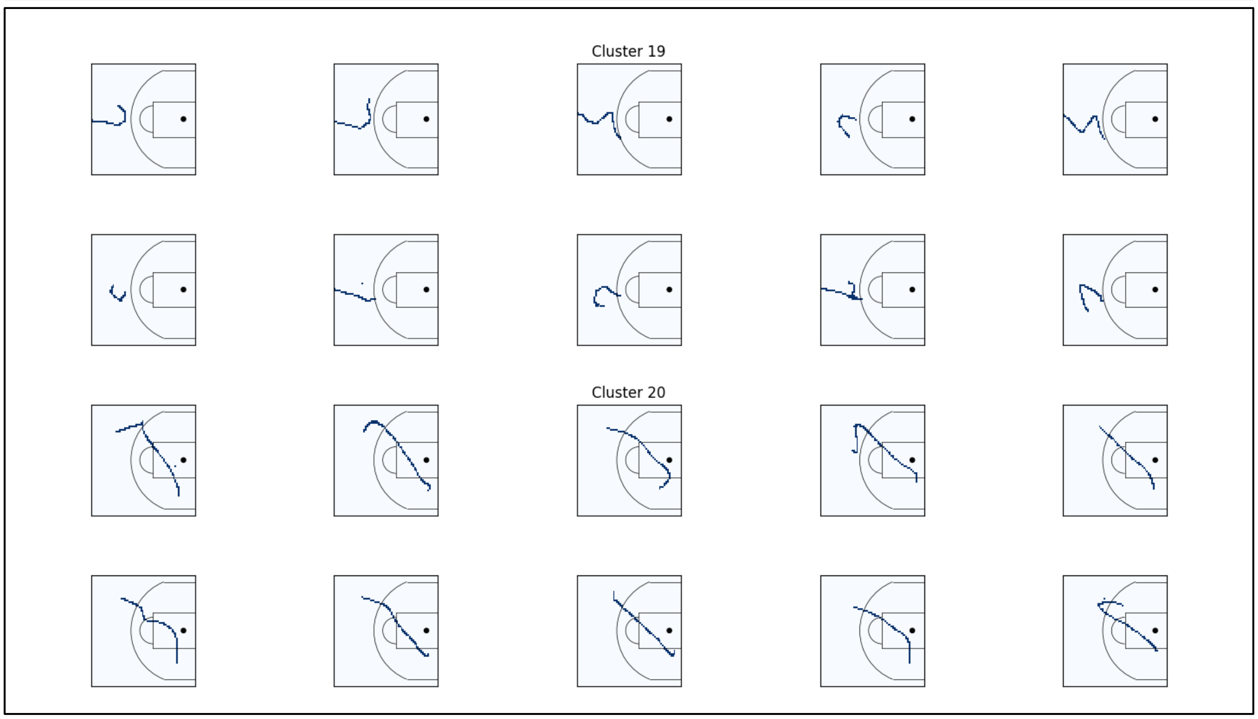

Cluster 9
Cluster 10


Cluster 11
Cluster 12


Cluster 13
Cluster 14


Cluster 15
Cluster 16


Cluster 17
Cluster 18


Cluster 19
Cluster 20

Appendix B: Cluster-mediod distance matrix

Cluster-mediod distance matrix. Column K lists the Euclidean distances between cluster K's mediod and the remaining clusters' mediods. The nearest cluster is highlighted in green and the furthest cluster in red.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Cluster 1	0.000	0.662	0.639	0.712	0.843
Cluster 2	0.662	0.000	0.697	0.607	0.899
Cluster 3	0.639	0.697	0.000	0.696	0.636
Cluster 4	0.712	0.607	0.696	0.000	0.644
Cluster 5	0.843	0.899	0.636	0.644	0.000
Cluster 6	0.781	0.746	0.574	0.666	0.545
Cluster 7	0.627	0.757	0.549	0.690	0.753
Cluster 8	0.815	0.768	0.741	0.671	0.677
Cluster 9	0.625	0.707	0.696	0.673	0.681
Cluster 10	0.804	0.767	0.782	0.794	0.881
Cluster 11	0.736	0.645	0.806	0.672	0.820
Cluster 12	0.905	0.814	0.798	0.598	0.655
Cluster 13	0.427	0.621	0.532	0.777	0.760
Cluster 14	0.493	0.834	0.578	0.807	0.799
Cluster 15	0.571	0.663	0.674	0.727	0.767
Cluster 16	0.650	0.434	0.744	0.778	0.936
Cluster 17	0.790	0.715	0.823	0.459	0.731
Cluster 18	0.790	0.666	0.650	0.522	0.690
Cluster 19	0.441	0.575	0.642	0.734	0.972
Cluster 20	0.739	0.625	0.889	0.589	0.896
Average distance to other clusters	0.687	0.695	0.692	0.675	0.768
Max distance to another clusters	0.905	0.899	0.889	0.807	0.972
Min distance to another cluster	0.427	0.434	0.532	0.459	0.545

	Cluster 6	Cluster 7	Cluster 8	Cluster 9	Cluster 10
Cluster 1	0.781	0.627	0.815	0.625	0.804
Cluster 2	0.746	0.757	0.768	0.707	0.767
Cluster 3	0.574	0.549	0.741	0.696	0.782
Cluster 4	0.666	0.690	0.671	0.673	0.794
Cluster 5	0.545	0.753	0.677	0.681	0.881
Cluster 6	0.000	0.577	0.649	0.456	0.806
Cluster 7	0.577	0.000	0.814	0.727	0.735
Cluster 8	0.649	0.814	0.000	0.636	0.594
Cluster 9	0.456	0.727	0.636	0.000	0.883
Cluster 10	0.806	0.735	0.594	0.883	0.000
Cluster 11	0.821	0.689	0.918	0.924	0.798
Cluster 12	0.631	0.750	0.474	0.770	0.653
Cluster 13	0.711	0.723	0.804	0.588	0.898
Cluster 14	0.627	0.537	0.810	0.543	0.798
Cluster 15	0.753	0.793	0.841	0.672	1.000
Cluster 16	0.873	0.880	0.911	0.837	0.922
Cluster 17	0.801	0.819	0.463	0.779	0.663
Cluster 18	0.497	0.461	0.734	0.748	0.734
Cluster 19	0.867	0.632	0.855	0.807	0.658
Cluster 20	0.767	0.596	0.878	0.823	0.767
Average distance to other clusters	0.692	0.690	0.740	0.714	0.786
Max distance to another clusters	0.873	0.880	0.918	0.924	1.000
Min distance to another cluster	0.456	0.461	0.463	0.456	0.594

	Cluster 11	Cluster 12	Cluster 13	Cluster 14	Cluster 15
Cluster 1	0.736	0.905	0.427	0.493	0.571
Cluster 2	0.645	0.814	0.621	0.834	0.663
Cluster 3	0.806	0.798	0.532	0.578	0.674
Cluster 4	0.672	0.598	0.777	0.807	0.727
Cluster 5	0.820	0.655	0.760	0.799	0.767
Cluster 6	0.821	0.631	0.711	0.627	0.753
Cluster 7	0.689	0.750	0.723	0.537	0.793
Cluster 8	0.918	0.474	0.804	0.810	0.841
Cluster 9	0.924	0.770	0.588	0.543	0.672
Cluster 10	0.798	0.653	0.898	0.798	1.000
Cluster 11	0.000	0.794	0.766	0.893	0.806
Cluster 12	0.794	0.000	0.921	0.936	0.903
Cluster 13	0.766	0.921	0.000	0.565	0.528
Cluster 14	0.893	0.936	0.565	0.000	0.783
Cluster 15	0.806	0.903	0.528	0.783	0.000
Cluster 16	0.683	0.947	0.555	0.920	0.493
Cluster 17	0.808	0.477	0.859	0.903	0.841
Cluster 18	0.621	0.547	0.849	0.796	0.809
Cluster 19	0.684	0.935	0.585	0.610	0.773
Cluster 20	0.479	0.737	0.862	0.850	0.895
Average distance to other clusters	0.756	0.750	0.702	0.741	0.752
Max distance to another clusters	0.924	0.947	0.921	0.936	1.000
Min distance to another cluster	0.479	0.474	0.427	0.493	0.493

	Cluster 16	Cluster 17	Cluster 18	Cluster 19	Cluster 20
Cluster 1	0.650	0.790	0.790	0.441	0.739
Cluster 2	0.434	0.715	0.666	0.575	0.625
Cluster 3	0.744	0.823	0.650	0.642	0.889
Cluster 4	0.778	0.459	0.522	0.734	0.589
Cluster 5	0.936	0.731	0.690	0.972	0.896
Cluster 6	0.873	0.801	0.497	0.867	0.767
Cluster 7	0.880	0.819	0.461	0.632	0.596
Cluster 8	0.911	0.463	0.734	0.855	0.878
Cluster 9	0.837	0.779	0.748	0.807	0.823
Cluster 10	0.922	0.663	0.734	0.658	0.767
Cluster 11	0.683	0.808	0.621	0.684	0.479
Cluster 12	0.947	0.477	0.547	0.935	0.737
Cluster 13	0.555	0.859	0.849	0.585	0.862
Cluster 14	0.920	0.903	0.796	0.610	0.850
Cluster 15	0.493	0.841	0.809	0.773	0.895
Cluster 16	0.000	0.871	0.836	0.678	0.810
Cluster 17	0.871	0.000	0.697	0.800	0.724
Cluster 18	0.836	0.697	0.000	0.792	0.525
Cluster 19	0.678	0.800	0.792	0.000	0.669
Cluster 20	0.810	0.724	0.525	0.669	0.000
Average distance to other clusters	0.777	0.738	0.682	0.722	0.743
Max distance to another clusters	0.947	0.903	0.849	0.972	0.896
Min distance to another cluster	0.434	0.459	0.461	0.441	0.479

Appendix C: Top 100 players by number of examples

- *Top 100 players by number of examples in our dataset. These are the 100 players who appeared in the most possessions > 4 seconds long between 2013 and 2016.*
- *Players separated by listed position and sorted by self-similarity and position-similarity-index (Section 4.2.3)*

Point Guards

Player	self-similarity	position-similarity-index
Tony Parker	0.944	0.767
Jeff Teague	0.923	0.825
John Wall	0.922	0.837
Damian Lillard	0.922	0.839
Mike Conley	0.918	0.847
Kyle Lowry	0.910	0.840
Kyrie Irving	0.904	0.848
Chris Paul	0.904	0.819
Deron Williams	0.897	0.857
Russell Westbrook	0.895	0.811
Michael Carter-Williams	0.892	0.833
George Hill	0.891	0.840
Stephen Curry	0.888	0.834
Kemba Walker	0.886	0.834
Isaiah Thomas	0.869	0.824
Mario Chalmers	0.868	0.831
Ty Lawson	0.868	0.842
Goran Dragic	0.865	0.828
Trey Burke	0.864	0.840
Brandon Knight	0.864	0.847
Jeremy Lin	0.848	0.806
Reggie Jackson	0.827	0.809
Jose Calderon	0.801	0.804
Average	**0.886**	**0.829**

Shooting Guards

Player	self-similarity	position-similarity-index
Danny Green	0.935	0.820
Kyle Korver	0.919	0.806
Courtney Lee	0.919	0.815
James Harden	0.916	0.665
Joe Johnson	0.916	0.824
Bradley Beal	0.915	0.822
Ben McLemore	0.912	0.795
Avery Bradley	0.898	0.790
Dwyane Wade	0.893	0.804
Klay Thompson	0.885	0.818
Jimmy Butler	0.878	0.789
Dion Waiters	0.877	0.819
DeMar DeRozan	0.867	0.808
J.R. Smith	0.867	0.817
Monta Ellis	0.862	0.763
Evan Turner	0.852	0.797
Kentavious Caldwell-Pope	0.852	0.777
Arron Afflalo	0.834	0.810
Gerald Henderson	0.834	0.782
Victor Oladipo	0.822	0.756
Lance Stephenson	0.726	0.747
Average	**0.875**	**0.792**

Small Forwards

Player	self-similarity	position-similarity-index
Terrence Ross	0.930	0.749
Kawhi Leonard	0.922	0.803
Chandler Parsons	0.918	0.801
LeBron James	0.916	0.756
Trevor Ariza	0.913	0.805
Rudy Gay	0.907	0.802
P.J. Tucker	0.904	0.767
Jeff Green	0.900	0.825
Wesley Matthews	0.891	0.799
Wesley Johnson	0.889	0.802
Khris Middleton	0.881	0.816
Gordon Hayward	0.878	0.792
Nicolas Batum	0.876	0.793
Corey Brewer	0.871	0.767
Carmelo Anthony	0.869	0.718
Harrison Barnes	0.866	0.797
Draymond Green	0.866	0.616
Luol Deng	0.855	0.802
Kevin Durant	0.850	0.723
Giannis Antetokounmpo	0.822	0.803
Matt Barnes	0.821	0.796
Marcus Morris	0.810	0.782
Paul George	0.768	0.747
Average	**0.875**	**0.777**

Power Forwards

Player	**self-similarity**	**position-similarity-index**
Paul Millsap	0.936	0.770
Dirk Nowitzki	0.916	0.738
Greg Monroe	0.913	0.743
Zach Randolph	0.912	0.662
Amir Johnson	0.900	0.776
Derrick Favors	0.888	0.736
Kenneth Faried	0.879	0.723
Markieff Morris	0.878	0.770
Taj Gibson	0.874	0.746
Blake Griffin	0.871	0.760
David West	0.865	0.734
Thaddeus Young	0.863	0.784
Tristan Thompson	0.858	0.735
Tobias Harris	0.835	0.608
Patrick Patterson	0.829	0.680
Serge Ibaka	0.817	0.749
Anthony Davis	0.815	0.754
Kevin Love	0.808	0.723
Marvin Williams	0.806	0.698
LaMarcus Aldridge	0.786	0.714
Average	**0.863**	**0.730**

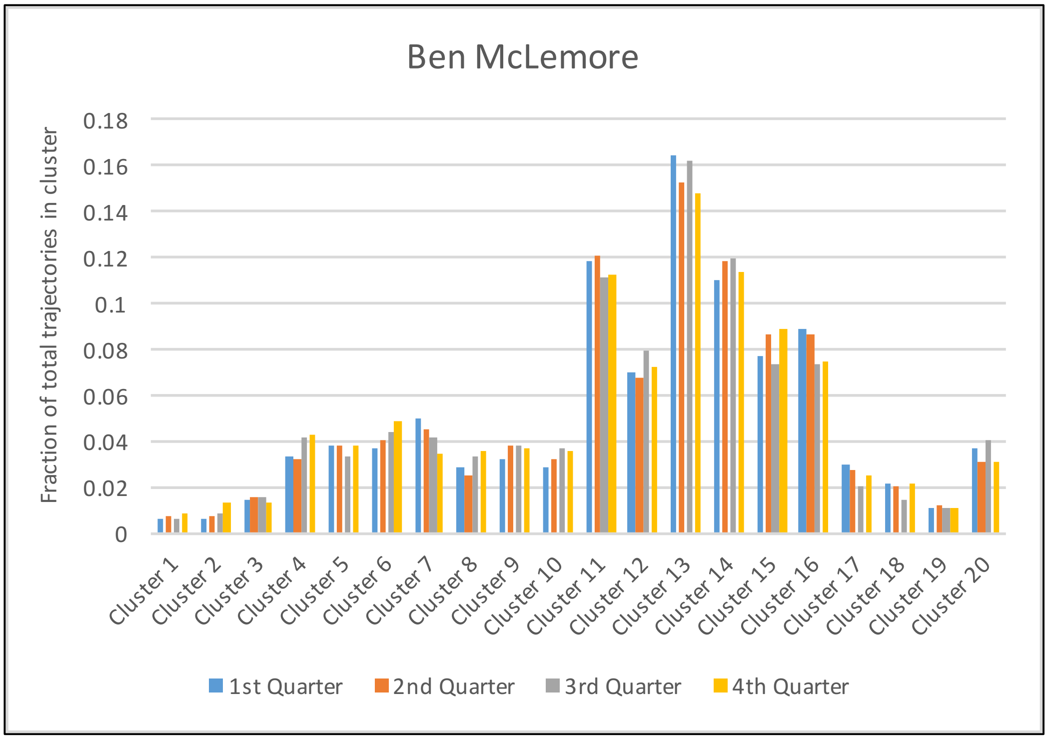
Centers

Player	self-similarity	position-similarity-index
Marc Gasol	0.939	0.727
Jonas Valanciunas	0.930	0.817
Marcin Gortat	0.928	0.822
Tim Duncan	0.923	0.810
Al Horford	0.902	0.805
DeAndre Jordan	0.899	0.811
DeMarcus Cousins	0.890	0.719
Nikola Vucevic	0.890	0.815
Pau Gasol	0.852	0.779
Roy Hibbert	0.847	0.769
Andre Drummond	0.842	0.768
Enes Kanter	0.810	0.780
Robin Lopez	0.782	0.732
Average	**0.880**	**0.781**

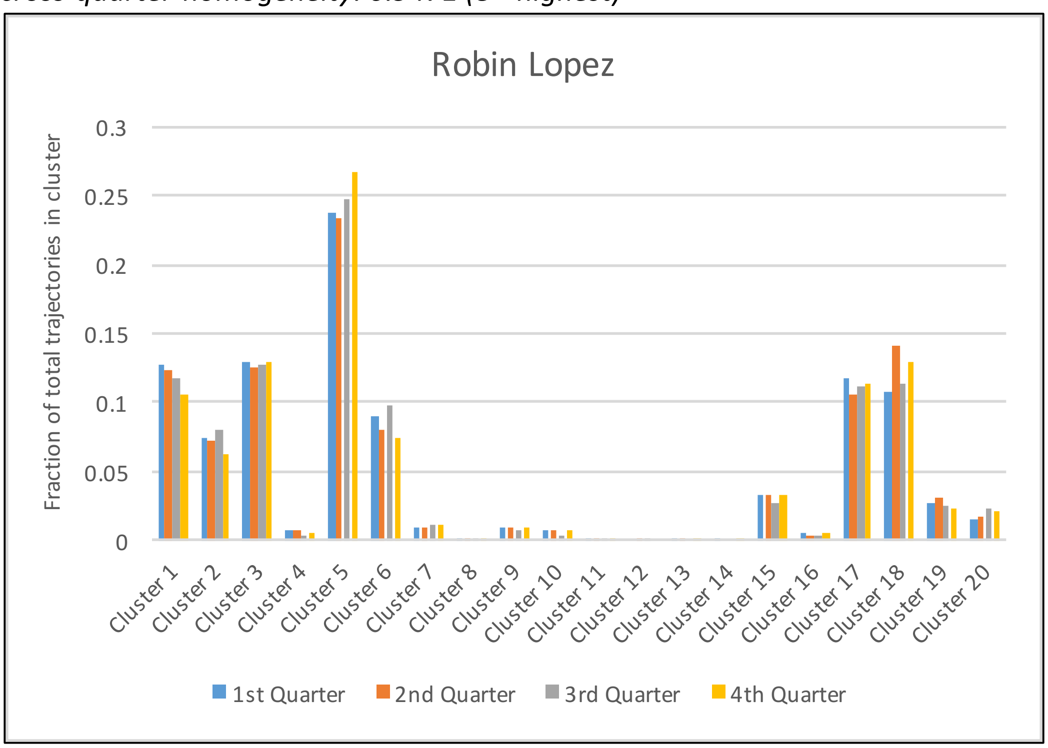
Appendix D: Example cluster-profiles, split by quarter

Cluster-profiles of players with lowest and highest cross-quarter-homogeneity (Section 5.2.3)

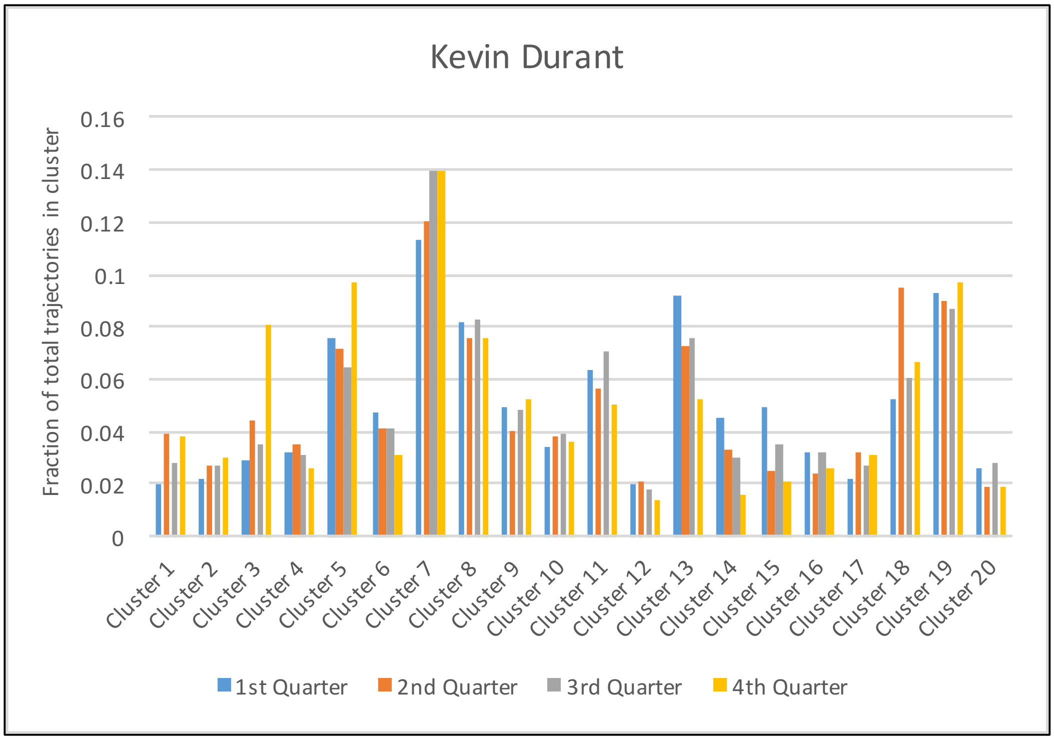
cross-quarter-homogeneity: 0.9509 (2nd highest)



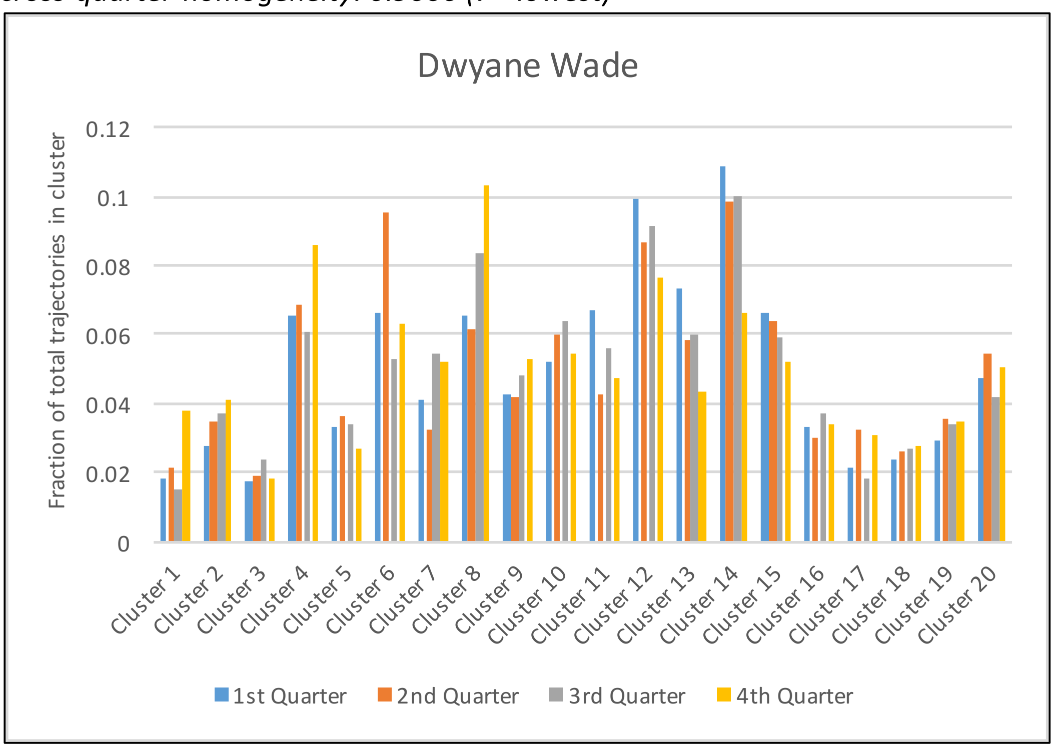
cross-quarter-homogeneity: 0.9471 (8th highest)



cross-quarter-homogeneity: 0.9018 (5th lowest)



cross-quarter-homogeneity: 0.9006 (7th lowest)



Bibliography

[1] Amick, S. (2014, March 05). NBA Caucus: Will analytics overtake the old school? Retrieved from https://www.usatoday.com/story/sports/nba/2014/03/05/mit-sloan-sports-analytics-conference-takeaways-phil-jackson/6087007/

[2] Brooks, J. (2018) Using Machine Learning to Derive Insights from Sports Location Data

[3] Amick, Sam. *NBA Caucus: Will Analytics Overtake the Old School?* 5 Mar. 2014, www.usatoday.com/story/sports/nba/2014/03/05/mit-sloan-sports-analytics-conference-takeaways-phil-jackson/6087007/.

[4] G.E. Hinton, R.R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. http://science.sciencemag.org/content/313/5786/504

[5] Turchenko, Volodymyr, and Artur Luczak. "Creation of a Deep Convolutional Auto-Encoder in Caffe." 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2017, doi:10.1109/idaacs.2017.8095172.

[6] https://swarbrickjones.files.wordpress.com/2015/04/conv_autoencoder.png