12-2019

Function Space Tensor Decomposition and its Application in Sports Analytics

Justin Reising
East Tennessee State University

Recommended Citation

Reising, Justin, "Function Space Tensor Decomposition and its Application in Sports Analytics" (2019). *Electronic Theses and Dissertations.* Paper 3676. https://dc.etsu.edu/etd/3676

Function Space Tensor Decomposition and its Application in Sports Analytics

---

A thesis

presented to

the faculty of the Department of Mathematics

East Tennessee State University

In partial fulfillment

of the requirements for the degree

Master of Science in Mathematical Sciences

---

by

Justin Reising

December 2019

Jeff Knisley, Ph.D.

Nicole Lewis, Ph.D.

Michele Joyner, Ph.D.

Keywords: Sports Analytics, PCA, Tensor Decomposition, Functional Analysis.

ABSTRACT

Function Space Tensor Decomposition and its Application in Sports Analytics

by

Justin Reising

Recent advancements in sports information and technology systems have ushered in a new age of applications of both supervised and unsupervised analytical techniques in the sports domain. These automated systems capture large volumes of data points about competitors during live competition. As a result, multi-relational analyses are gaining popularity in the field of Sports Analytics. We review two case studies of dimensionality reduction with Principal Component Analysis and latent factor analysis with Non-Negative Matrix Factorization applied in sports. Also, we provide a review of a framework for extending these techniques for higher order data structures. The primary scope of this thesis is to further extend the concept of tensor decomposition through the use of function spaces. In doing so, we address the limitations of PCA to vector and matrix representations and the CP-Decomposition to tensor representations. Lastly, we provide an application in the context of professional stock car racing.

ACKNOWLEDGMENTS

To my “Nearest Neighbors”, for making this mathematical pursuit an invaluable experience and providing me with the courage and support to advance my knowledge and well-being. A special thanks to my friends and family for constant support with even the smallest of gestures that kept me going. To Dr. Knisley for his persistence, patience, and guidance in providing me with the tools and confidence for becoming an applied mathematical practitioner. To my colleagues at Joe Gibbs Racing for answering my seemingly never ending questions and the incredible opportunity to develop this work in the context of professional stock car racing. To my co-author in life, Gabrielle, for which none of this would be possible without your unselfish love and support which has shaped me into the man I am today.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Evolution of Big Data in Professional Sports

The age of newspaper box scores being the primary source of numerical information for professional athletes has long become an after-thought over the course of the last decade. Now, stakeholders throughout all levels of professional sports organizations are engaging with an overwhelming amount of data being collected from a multitude of sources [2]. From the front office to the field, court, or racetrack, the competitive advantage for clubs in top echelon team sports now rely heavily on the ability to mine, warehouse, and transform their data into actionable information.

Professional sports in the United States have taken a massive step in the procurement of data and have made major investments in human and computational resources to handle the data collected in the last decade [1]. This paradigm shift became more prevalent in the industry after the release of the book and movie *Moneyball*, which depicted the use of statistics to drive player acquisition decisions instead of traditional scouting practices within the Oakland Athletics Major League Baseball (MLB) organization during the 2002 season [1]. During that season, General Manager Billy Beane embraced statistics that went beyond the box score. This is now known as “Sabermetrics” and is used to inform scouting decisions to put players on the field with very limited capital. Sabermetrics were originally developed by Bill James in the 1970’s as “analytical musings” that have now evolved into what has become the field of “Sports Analytics” [3].

Baseball was not the only sport that began thinking of how to use technology and advanced mathematical techniques to increase the tally count in the win column. Of the major sports in the Unites States, the National Basketball Association (NBA), National Football League (NFL), Major League Soccer (MLS), National Hockey League (NHL) have all embraced the coming of age with analytics [1]. Some teams in each sport have been more apprehensive than others, but when dynasty teams seemingly form out of nowhere within a 5 year period, every team asks the same question: “What are they doing that we are not?” The answer is not just a couple of Physics and Mathematics PhD’s on payroll cranking out “insights”. It takes years to develop a data-driven culture within the organization and build the pipeline from raw data to actionable information.

Every professional sports team has a series of scouting methods for evaluating amateur and other professional players when it comes to physiological traits like speed, arm strength, etcetera. In baseball, the ability to do these things were typically observational in most cases and subjective measurements varied depending on which individual was doing the evaluation. Conventional baseball player evaluation is (or was) dependent on five “tools”: Speed, Arm Strength, Hit for Average, Hit for Power, and Fielding [3]. While baseball has always been the most prolific numbers game when it comes to statistics, there was still a subjective nature to them with all of the nuances of the game. However, since 2015, MLB mandated that all stadiums be equipped with sonar tracking systems that track object movements including players, bats, and the ball [19]. This ushered in a new age of big data that has never been seen before and pushed the envelope in the sports analytics domain from the “personal computer statistics” domain to the “cloud computing applied mathematics” domain. Over the course of the past 5 years, stadiums in the NBA, NFL, MLS, and NHL are now all equipped with this type of technology generating petabytes of data collectively each year.

One sport that has less public attention in the sports analytics industry is the National Association for Stock Car Auto Racing (NASCAR). NASCAR also has a very different structure than other sports. Teams are not located geographically with a “Home Field” or track in this case. Most teams operate from a central location in the Charlotte, North Carolina metropolitan area and travel to venues across the country on a weekly basis. This is not the only concept that is quite different than other professional team sports. As organizations have multiple teams at different levels, such as major league and minor league levels in baseball, NASCAR organizations can have multiple drivers competing at the same level. For example, Joe Gibbs Racing (JGR) is an organizational team consisting of four drivers that compete in the Monster Energy Cup Series (MECS) sponsored by the manufacturing team Toyota Racing Development (TRD). Each car-level team is comprised of a driver, crew members, with some unique corporate sponsorships. All future mentions of “teams” is in reference to organizational teams.

1.2 Motivation

The primary motivation for this paper is the increasing use of dimensionality reduction, low-rank approximation, and latent factor techniques applied in sports analytics with the goal of extracting hidden components in the underlying structure of data. However, the granularity of data can vary widely with different types of contextual indices such as individual events or competitors, situational characteristics, time-based intervals, etc. Current applications of decomposition techniques typically ignore such categorical parameters to suit the constraints for the numerical data types being analyzed. In sports, events that take place during competition are highly dependent on contextual factors. For example, in basketball, time left on the clock can influence shot selection or the inning, count, and runners on base for a pitch selection in baseball.

Principal Component Analysis (PCA) is one of the oldest methods applied for dimensionality reduction [13]. The primary goal of PCA is to reduce the dimensionality of a data set by extracting low dimensional sub-spaces while preserving as much variability as possible. PCA is a common technique that addresses the problem of dimensionality and sub-space leaning in data science, but its effectiveness is limited to numerical vector and matrix data structures. PCA is also computationally expensive for very large data sets. This causes PCA to lose power for large-scale, multi-relational data sets that are now more common in practice. When applied to multi-model data, the traditional PCA methods applied for matrices has been shown to be inadequate at capturing variance across different modes and burdened by increasing storage and computational costs [14].

There is an increasing demand for PCA type methods that can learn from tensor data while accounting for the multi-relational structure for multi-linear dimensionality reduction and subspace estimation [14]. In sports analytics, matrix factorization techniques have been used in many sports applications, such as in the NBA to identify latent factors of players that go beyond the standard five positions [5]. The extension of tensor decomposition to this type of problem is reviewed in more detail through the development of *tHoops*, which profiles shot selection tendencies relative to time on the clock [8]. This is a primary motivation exemplifying how subspace learning techniques can extend traditional shot chart analyses in the context of basketball. Collectively, this thesis provides a framework for addressing the limitations of PCA for tensors subspace learning.

Unlike in other sports, there is a lack of analytical frameworks publicized in NASCAR and motorsports in general. With large numbers of observations for each car, lap, track, and season from a multitude of data types and sources, utilizing multiple indexes in higher order data structures can lead to finer analysis of data. We further extend the application of multi-relational analysis for sports analytics and suggest approaches in the context of NASCAR.

1.3 Objectives

The objectives for this thesis are to describe in detail applications of dimensionality reduction and subspace learning applications in sports related contexts and to provide a framework for extensions of these methods utilizing concepts from functional analysis. A conventional approach for multi-relational data is to reshape the data (unfold) into a matrix structure and then apply classical PCA techniques. However this eliminates relational information from the folded index. The first primary goal of this thesis is to demonstrate the classical approach to sports related data in two different case studies and establish the context for extending analyses into higher-order data structures. The second goal for this thesis is to provide a mathematical basis for the traditional approaches through the lens of linear algebra concepts as well as the extension to tensor approaches through the lens of functional analysis concepts.

1.4 Outline of Thesis

The remainder of the thesis is organized as follows. Chapter 2 covers the mathematical background for the concepts introduced for matrix decompositions, an introduction for tensors, and introduction for function spaces. Chapter 3 provides two case studies with applications of PCA in baseball analytics and non-negative matrix (NMF) factorization in basketball and a review of *tHoops*; a tensor decomposition framework for basketball shot selection. Chapter 4 provides a comprehensive analysis including PCA and NMF approaches and introduces tensor decomposition via function spaces in the context of NASCAR analytics. Chapter 5 includes closing comments and suggestions for further work for the methodology.

2 MATHEMATICAL BACKGROUND

2.1 Special Matrices

In this chapter we introduce the mathematical background for developing our proposed framework for tensor analysis in sports analytics. The motivation for decomposing matrices or “data frames with numerical entries” is drawn from fundamental aspects of linear algebra with “special matrices”. For this paper, we assume the reader has exposure to basic undergraduate level linear algebra concepts in the following definitions for later references.

Definition 2.1 [10] *An $n \times n$ orthogonal matrix $\mathbf{Q}$ has orthonormal columns which means that $\mathbf{q}_i^T\mathbf{q}_j = 0$ and $\mathbf{q}_i^T\mathbf{q}_i = 1 \; \forall \; i, j \in 1, ..., n$. Equivalently,*

i) $\mathbf{Q}^T\mathbf{Q} = \mathbf{Q}\mathbf{Q}^T = \mathbf{I}_n$

ii) $||\mathbf{Q}\mathbf{x}|| = ||\mathbf{x}||$ *for all* $\mathbf{x} \in \mathbb{R}^n$

iii) $\mathbf{Q}^T = \mathbf{Q}^{-1}$

iv) The columns of $\mathbf{Q}$ are an orthonormal basis for $\mathbb{R}^n$

If $\mathbf{S}\mathbf{x} = \lambda\mathbf{x}$ such that $\mathbf{x} \neq 0$, then λ is an eigenvalue of $\mathbf{S}$ with eigenvector $\mathbf{x}$. This leads to the next definition.

Definition 2.2 [10] *If $\mathbf{S}$ is an $n \times n$ matrix with eigenvalues λ_i and eigenvectors $\mathbf{x}_i$, then*

i) Trace($\mathbf{S}$*)* $= \displaystyle\sum_{i,j=1}^{n} s_{i,j} = \sum_{i=1}^{n} \lambda_i$

ii) Determinant($\mathbf{S}$*)* $= \displaystyle\prod_{i=1}^{n} \lambda_i$

Theorem 2.3 [10] *If* $\mathbf{S}$ *is an* $n \times n$ *symmetric matrix, then,*

i) $\mathbf{S} = \mathbf{S}^T$ *implies that the eigenvalues are real*

ii) If $\lambda_i \neq \lambda_j \quad for \quad i, j \in 1, ..., n, \quad then \quad \mathbf{x}_i \cdot \mathbf{x}_j = 0.$ *(orthogonal eigenvectors)*

In this way, symmetric matrices, $\mathbf{S}$, are like real numbers in that every $\lambda \in \mathbb{R}$ and orthogonal matrices, $\mathbf{Q}$ are like complex numbers in that every $|\lambda| = 1$ [10]. While orthogonal and symmetric matrices are indeed “special” with their own unique properties, they are only part of the main attraction for many applications in data science.

2.2 Matrix Decomposition

Our first theorem is the Spectral Theorem, which is foundational for much of matrix decomposition.

Theorem 2.4 [10] ***The Spectral Theorem***. *Every symmetric matrix has the form,*

$$\mathbf{S} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T \tag{1}$$

Note that $\mathbf{Q}$ is the orthogonal eigenvector matrix of $\mathbf{S}$ and $\mathbf{\Lambda}$ is the diagonal matrix of corresponding eigenvalues. The Singular Value Decomposition is the extension of The Spectral Theorem for non-symmetric, non-square matrices.

Theorem 2.5 [10] *For an* $m \times n$ *matrix* $\mathbf{A}$ *with rank* $= r$*, the* ***Singular Value Decomposition (SVD)*** *of* $\mathbf{A}$ *is,*

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \quad s.t \quad \sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r \tag{2}$$

Just as the columns of $\mathbf{Q}$ are orthogonal in Theorem 2.4, the columns of $\mathbf{U}$ and $\mathbf{V}$ and also orthogonal. However, since $\mathbf{A}$ is not square, then the columns of $\mathbf{U}$ are orthogonal in $\mathbb{R}^m$ and the columns of $\mathbf{V}$ are orthogonal in $\mathbb{R}^n$. The vectors of $\mathbf{U}$ and $\mathbf{V}$ are referred to as the "left singular vectors" and "right singular vectors" respectively. Also, a special property of SVD is that it decomposes the matrix into a series of unique rank one pieces in (2) in order of importance [10]. Therefore, $\mathbf{A_k} = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$ is the best rank k approximation of $\mathbf{A}$.

Definition 2.6 *The* ***Frobenius Norm*** *is defined as* $||\mathbf{A}||_F = \sqrt{\sigma_1^2 + \sigma_2^2 + ... + \sigma_r^2}$.

Theorem 2.7 [10] ***The Eckart-Young Theorem****. For* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$*, If* $\mathbf{B}$ *has rank* k *and* $\mathbf{A_k} = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$*, then*

$$||\mathbf{A} - \mathbf{A_k}||_F \leq ||\mathbf{A} - \mathbf{B}||_F. \tag{3}$$

An immediate application of of Theorem 2.7 is Non-Negative Matrix Factorization (NMF). The goal of NMF is to approximate a non-negative matrix $\mathbf{A} \geq 0$ by a lower rank product of two matrices.

Definition 2.8 *A matrix* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *is a* ***Non-Negative Matrix*** *if* $a_{ij} \geq 0$ *for all* $i = 1, ..., m$ *and* $j = 1, ..., n$.

Theorem 2.7 applied to non-negative matrices leads to the following corollary.

Corollary 2.9 [10] ***Non-Negative Matrix Factorization*** *(NMF) For non-negative matrix* $\mathbf{A} \in \mathbb{R}^{m \times n}$*, there exists non-negative* $\mathbf{B} \in \mathbb{R}^{m \times r}$ *and non-negative* $\mathbf{C} \in \mathbb{R}^{r \times n}$*, such that* $\mathbf{A} \approx \mathbf{BC}$ *in that*

$$\min_{B,C} ||\mathbf{A} - \mathbf{BC}||_F^2 \quad \textit{exists.} \tag{4}$$

This approximation is a Linear Dimensionality Reduction (LDR) technique that requires the selection of a measure to assess the quality of the approximation [4]. The measure frequently chosen is the Frobenius norm of the error in the approximation (i.e $||\mathbf{A} - \mathbf{BC}||_F^2$). The choice of this error measure is primarily driven by the implicit assumption of the noise present in $\mathbf{A}$ being Gaussian and the low rank approximation given by Theorem 2.7, which is also known as the "Truncated SVD".

2.3 PCA and SVD

It is important to note the results of applying the SVD factorization to $\mathbf{A^T A}$ and $\mathbf{AA^T}$, which are square, symmetric, positive definite matrices:

$$\mathbf{A^T A} = (V\Sigma U^T)(U\Sigma V^T) = \mathbf{V\Sigma^T\Sigma V^T} = \mathbf{V\Sigma^2 V^T} \tag{5}$$

$$\mathbf{AA^T} = (U\Sigma V^T)(V\Sigma U^T) = \mathbf{U\Sigma^T\Sigma U^T} = \mathbf{U\Sigma^2 U^T} \tag{6}$$

The right-hand side of equations (5) and (6) are the SVD forms of Theorem 2.4, i.e $\mathbf{Q\Lambda Q^T}$. For the $m \times n$ matrix $\mathbf{A}$, the shape of $\mathbf{A^T A}$ is $m \times m$ and the shape of $\mathbf{AA^T}$ is $n \times n$. Lastly, it important to note that $\mathbf{V}$ contains the orthonormal eigenvectors of $\mathbf{A^T A}$, $\mathbf{U}$ contains the orthonormal eigenvectors of $\mathbf{AA^T}$, and the diagonal of $\mathbf{\Sigma^2}$ contain σ_1^2, σ_2^2, ... , σ_r^2, which are the non-zero eigenvalues of both $\mathbf{A^T A}$ and $\mathbf{AA^T}$. These special matrices, combined with Theorem 2.7 are key concepts of building an analytic framework for analyzing data.

Principal Component Analysis is a tool used in numerous settings with a wide variety of data types as a means of visualizing high-dimensional data structures [13]. Geometrically, PCA and SVD are closely related with one key step at the beginning for PCA, which is to center the data. However, if every feature of the data is of the same scale and metric, this step may be unnecessary.

Definition 2.10 [10] *The* ***Sample Covariance Matrix*** *of* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *is defined by*

$$\mathbf{S} = \frac{\mathbf{A}\mathbf{A}^{\mathbf{T}}}{m-1}. \tag{7}$$

After applying SVD to $\mathbf{S}$, we obtain $\mathbf{S} = \mathbf{U}\boldsymbol{\Sigma}^{\mathbf{2}}\mathbf{U}^{\mathbf{T}}$ and since $\mathbf{S}$ is a symmetric matrix, then the columns of $\mathbf{U}$ are eigenvectors of $\mathbf{S}$ by Theorem 2.4, which are also the left singular vectors of $\mathbf{A}$. Then by applying SVD to $\mathbf{A}$, the columns of $\mathbf{U}$ are the *principal components* of $\mathbf{A}$. As a consequence, then the eigenvalues of $\mathbf{S}$ equal to the squared singular values of $\mathbf{A}$, and the total variance of $\mathbf{A} = \sum_{i=1}^{r} \sigma_i^2/(n-1)$. The key observation of Theorem 2.7 in combination with SVD is that the first k singular vectors together account for the most variation in the data than any other set of k singular vectors. This is the key motivation for dimensionality reduction.

2.4 Introduction to Tensors

Tensors are not a new mathematical object, but rather a generalization of matrices to higher number of indicies. Tensors and their decompositions were originally studied in the 1920's but remained in the abstract domain of mathematics until the explosion in computational capacity in the late 20^{th} century [6]. Over the course of the last decade, there has been a surge of applications in statistics, data science, and machine learning built on tensor representations of data. One of the most famous software developments in recent years is the machine learning platform *TensorFlow* developed and maintained by Google, Inc. [7]. TensorFlow provides back-end computational support for the "Keras" package, a popular machine learning package available in the Python and R programming languages. TensorFlow provides extensive training, support, and documentation to allow machine learning applications easier to develop, train, and deploy for practitioners. The name contains the fundamental structure of machine learning and data science which is the Tensor.

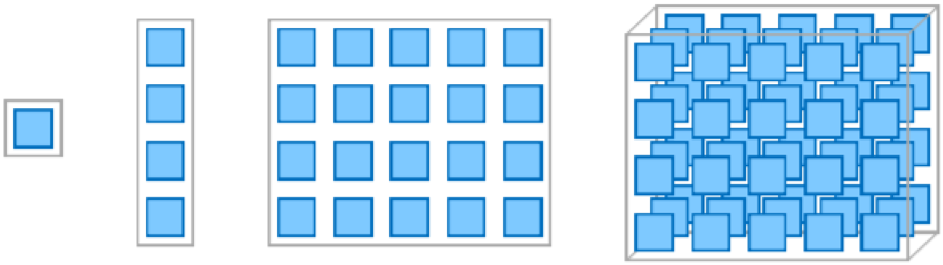

Figure 1: Tensor Order [6].

Tensors are multi-dimensional array structures in a field, such as ℝ. Figure 1 demonstrates the traditional progression of dimensions in the form of a scalar, vector, matrix, and tensor, denoted, $x \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^4, \mathbf{X} \in \mathbb{R}^{4\times 5}, \mathcal{X} \in \mathbb{R}^{4\times 5\times 3}$ respectively. More generally, scalars are referred to as a "0-Order Tensor", vectors are "1^{st}- Order Tensors", matrices are "2^{nd} - Order Tensors", and lastly, 3 dimensional structures are "3^{rd} - Order Tensors". The order of the tensor is the number of axes.

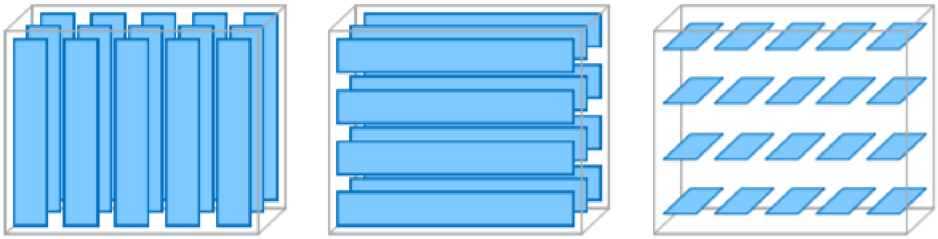

Figure 2: Tensor Fibers [6].

Indexing tensors allow for sub-components to be created by fixing one or more indices. For example, consider a third order tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$. *Fibers* are created by fixing all but two indices. Figure 2 demonstrates the vector fibers $\mathbf{x}_{:jk}$ (column), $\mathbf{x}_{i:k}$ (row), and $\mathbf{x}_{ij:}$ (tube). *Slices* are created by fixing all but one index. Figure 3 shows matrix slices $\mathbf{X}_{i::}$ (horizontal), $\mathbf{X}_{:j:}$ (lateral), and $\mathbf{X}_{::k}$ (frontal).

With fibers and slices of a tensor, it is easy to see that we can reshape tensors by rearranging these components with *vectorization* and *matricization* (unfolding). Given a matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$ vectorization is achieved by stacking the columns of $\mathbf{X}$ vertically, such as

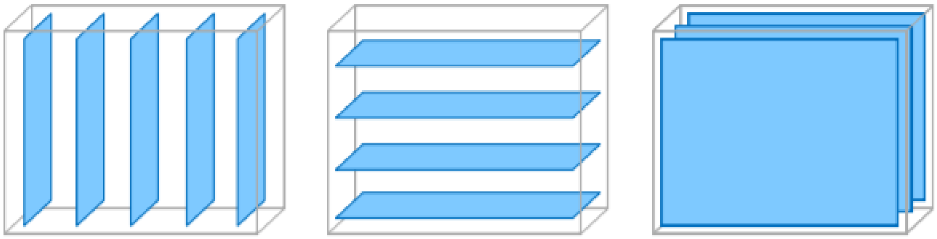

Figure 3: Tensor Slices [6].

$$\text{vec}(\mathbf{X}) = \begin{bmatrix} \mathbf{x_{:1}} \\ \mathbf{x_{:2}} \\ \vdots \\ \mathbf{x_{:n}} \end{bmatrix} \tag{8}$$

Tensors can also be vectorized and matricized in a similar fashion. One method we highlight is the mode-n matricization of a tensor. For $\mathcal{X} \in \mathbb{R}^{(I_1 \times I_2 \times \ldots \times I_N)}$, the mode-$n$ matricization of $\mathcal{X}$ is $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times (\prod_{m=1}^{N} I_m)}$. Let $x \in \mathcal{X}$ and $m \in \mathbf{M}$ where $\mathbf{M}$ is the unfolded tensor. Then the mapping of a mode-n matricization is,

$$x_{i_1, i_2, \ldots, i_N} \quad m_{i_n, j} \quad \text{where} \quad j = 1 + \sum_{\substack{k=1 \\ k \neq n}}^{N} \left((i_k - 1) \prod_{\substack{m=1 \\ m \neq n}}^{k-1} I_m \right) \tag{9}$$

For example, let $\mathcal{X} \in \mathbb{R}^{2 \times 2 \times 2}$ be composed of two frontal slices, $\mathbf{X}_{::1}, \mathbf{X}_{::2} \in \mathbb{R}^{2 \times 2}$.

$$\mathbf{X}_{::1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad \mathbf{X}_{::2} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}$$

Then the matricization index is denoted by the corresponding index of fibers that are used as columns in the associated matrix. For this example, the columns of $\mathbf{X}_{(1)}$ are the column fibers $\mathbf{x}_{:jk}$, $\mathbf{X}_{(2)}$ are the row fibers $\mathbf{x}_{i:k}$, and $\mathbf{X}_{(3)}$ are the tube fibers $\mathbf{x}_{ij:}$.

$$\mathbf{X}_{(1)} = \begin{bmatrix} a & b & e & f \\ c & d & g & h \end{bmatrix} \tag{10}$$

$$\mathbf{X}_{(2)} = \begin{bmatrix} a & c & e & g \\ b & d & f & h \end{bmatrix} \tag{11}$$

$$\mathbf{X}_{(3)} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \end{bmatrix} \tag{12}$$

In undergraduate linear algebra, the *Outer Product*, denoted $\circ$, is the product of two vector elements. For vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$, we obtain the following equation,

$$\mathbf{a} \circ \mathbf{b} = \mathbf{a}\mathbf{b}^T = \begin{bmatrix} a_1b_1 & a_1b_2 & ... & a_1b_n \\ a_2b_1 & a_2b_2 & ... & a_2b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_nb1 & a_nb_2 & ... & a_nb_n \end{bmatrix} \tag{13}$$

Equation (13) has a direct extension for tensor outer product. In the same way that an outer product of two vectors is a matrix, the general *tensor product* of N vectors (i.e first order tensors) produces an order N tensor.

Definition 2.11 [6] *The **Tensor Product** of N first order tensors produces a tensor $\mathcal{X}$ such that,*

$$\mathcal{X} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ ... \circ \mathbf{a}^{(N)} \quad \textit{where} \quad x_{i_1,i_2,...,i_N} = a^{(1)}_{i_1} a^{(2)}_{i_2} ... a^{(N)}_{i_N}. \tag{14}$$

Definition 2.12 [6] *A N-Order tensor is of **rank-1** if it can be strictly decomposed into the outer product of N first order tensors. More generally, the **Rank** of a tensor is the number of minimum first order tensors necessary to produce the tensor.*

Similar to Equation (13) and Equation (14), Figure 4 shows the rank one third order tensor $\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$. As higher order tensors can be generated by the outer

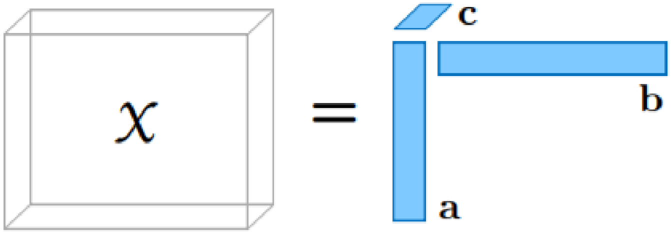

Figure 4: Rank 1 Third Order Tensor [6].

product of first order tensors, this can be extended to the product of different order tensors. There are multiple types of tensor products extending from Definition 13, but for this paper, we will introduce one primary product type.

Definition 2.13 [6] *The **Kronecker Product**, denoted* $\otimes$*, between two arbitrarily sized matrices* $\mathbf{A} \in \mathbb{R}^{R \times J}$ *and* $\mathbf{B} \in \mathbb{R}^{K \times L}$*, then* $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{IK \times JL}$*, is a generalization of the outer product defined in Equation 13.*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \dots & a_{1J}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \dots & a_{2J}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}\mathbf{B} & a_{I2}\mathbf{B} & \dots & a_{IJ}\mathbf{B} \end{bmatrix} = [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_1 \otimes \mathbf{b}_2 \quad \dots \quad \mathbf{a}_J \otimes \mathbf{b}_L] \tag{15}$$

2.5 Tensor Decomposition

Definition 2.14 [6] *A tensor decomposition is unique if there exists only one combination of rank-1 tensors that sum to $\mathcal{X}$ up to a common scaling and/or permutation indeterminacy.*

As previously discussed for the SVD in Definition 2.5, the primary restriction imposed for matrix decomposition with the SVD is the orthogonality of the left and right singular vectors, which makes the decomposition unique up to row and comlumn permutations. Similarly, a tensor decomposition is unique if it decomposes into one and only one arrangement of rank-1 tensors [6]. However, tensor decomposition can be unique under less restrictions than in the matrix case of the SVD. With the goal of low-rank approximation, consider the low rank tensor $\mathcal{X}$ in Figure 2.4, then each slice of the tensor,

$$\mathbf{X}_k = \sum_{r=1}^{R} (\mathbf{a} \circ \mathbf{b}) c_{kr} \tag{16}$$

is a low-rank matrix. Hence, a low-rank tensor is a collection of low-rank matrices with interrelations among the slices with different scaling, namely c_{kr} [6]. As a result, the relationship between slices make tensors much more rigid than matrices when it comes to conditions for uniqueness. This creates an opportunity for multiple decomposition approaches with different structural properties for generalizing SVD from matrices to higher order tensors.

Our primary objective is to generalize PCA and SVD from matrices to tensors and address the issues presented in regards to uniqueness. There are many approaches to tensor decomposition, but the scope of this paper discuss two common outer-product tensor decomposition methods with different properties: the *Canonical Polyadic Decomposition* (CP-Decomposition) and the *Tucker Decomposition.* The CP-Decomposition is typically used for latent factor analysis while Tucker is commonly applied for subspace estimation, compression, or dimensionality reduction [6].

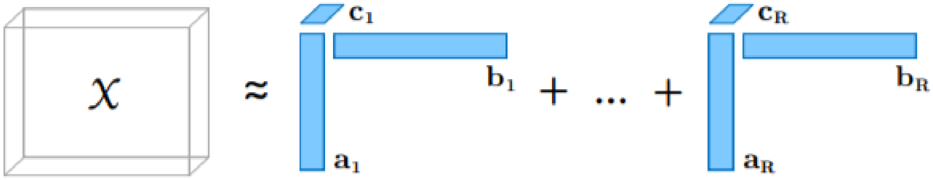

Figure 5: CP Decomposition [6].

First, we will highlight the CP-Decomposition, which is a rank decomposition. The key concept for this decomposition is the expression of tensor as the sum of rank-one tensors. For order-3 tensors, depicted in Figure 5, the CP-Decomposition is formalized as,

$$\min_{\hat{\mathcal{X}}} ||\mathcal{X} - \hat{\mathcal{X}}|| \quad \text{where} \quad \hat{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r = [[\mathbf{A}, \mathbf{B}, \mathbf{C}]]. \tag{17}$$

Note the similarity in Equation (17) and that of the NMF in Corollary 2.9. For the general case, the CP-Decomposition is formalized as,

$$\hat{\mathcal{X}} = \sum_{r=1}^{R} \lambda_r \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ ... \circ \mathbf{a}_r^{(n)} = [[\lambda; \mathbf{A}^{(1)}, \mathbf{A}^{(2)}, ..., \mathbf{A}^{(n)}]]. \tag{18}$$

The factors, $\mathbf{A}^{(n)}$, are normalized at unit length and the scalings are stored as λ_r. These λ_r are trial-specific scalings of the tensor and are the fundamental difference in the CP model and normal PCA. While there are many algorithms for computing the CP-Decompositon, the most common approach is the *Alternating Least Squares* (ALS) algorithm. This key idea is to optimize a factor matrix while holding all others constant and repeat for every factor matrix until a stopping criterion. But with the formalization of equation (18), the rank is necessary for approximation. There is no trivial algorithm in computing the rank of a tensor as it the problem in NP-hard [6]. In practice, most algorithms fit for multiple ranks and then choose the best approximation.

In direct contrast to CP-Decomposition is the Tucker Decomposition which decomposes the tensor into a "core" tensor for which there are different scalings along each mode. This is what makes Tucker akin to PCA and is sometime referred to as higher-order PCA.

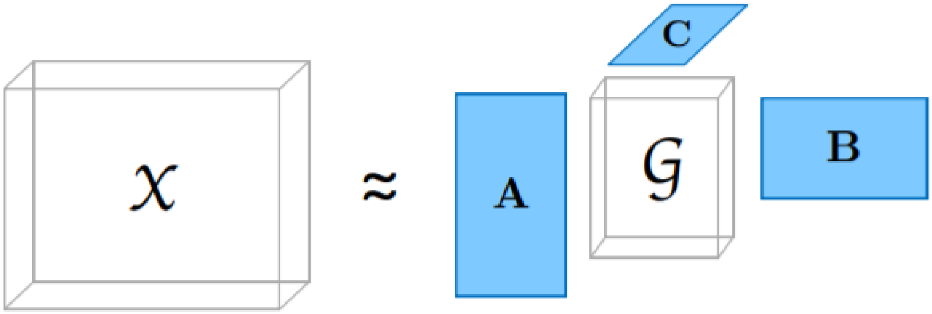

Figure 6: Tucker Decomposition [6].

Figure 6 depicts the model for the third order tensor decomposition. For this case, consider $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, then $\mathcal{G} \in \mathbb{R}^{P \times Q \times R}$, $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B}\ in \mathbb{R}^{J \times Q}$, and $\mathbf{C} \in \mathbb{R}^{K \times R}$.

The optimization problem then becomes,

$$\min_{\hat{\mathcal{X}}} ||\mathcal{X} - \hat{\mathcal{X}}|| \text{ where } \hat{\mathcal{X}} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} g_{pqr} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r = [[\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}]]. \tag{19}$$

The factors **A**, **B**, and **C** are often thought of the principal components for the respective axis. The core tensor, $\mathcal{G}$, is a compression of the original tensor and expresses the interaction between factors. In contrast to the CP-Decomposition, Tucker is generally not unique because of the arbitrary structure of the core constructed. However, if $g_{pqr} = 0$ for all $p \neq q \neq r$ in equation 19, then it would reduce to the CP-Decomposition. In the general case, the tucker decomposition is formalized as,

$$\min_{\hat{\mathcal{X}}} ||\mathcal{X} - \hat{\mathcal{X}}|| \text{ where } \hat{\mathcal{X}} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \cdots \sum_{r_N=1}^{R_N} g_{r_1, r_2, \ldots r_N} \mathbf{a}^{(1)}_{i_1 r_1} \circ \cdots \circ \mathbf{a}^{(N)}_{i_N r_N} = [[\mathcal{G}; \mathbf{A}^{(1)}, ..., \mathbf{A}^{(N)}]]. \tag{20}$$

The two key problems between these common techniques discussed lie with the loss of orthogonal components in CP Decomposition and the interpretability of the "core" from Tucker. Thus PCA tends to be poorly defined for tensors with order greater than two as the concept of "centering" becomes axis dependent. To bridge this gap we introduce a proof of concept to build a new approach to tensor PCA and utilize the inherent network structure of tensors. Next, we will introduce functional analysis concepts in tandem with matrix and tensor concepts to address these two pain points in extending multi-relational PCA.

2.6 Function Spaces

Before getting into function spaces in a general sense, it is necessary to expand on the linear algebra concept of a vector space. Vector spaces, such as $\mathbb{R}^n$, have useful properties that are the foundational elements of matrix decomposition techniques. We will build up vector spaces with additional properties to draw comparisons for function spaces, and ultimately, tensor decomposition with function spaces. We begin with the complement of the *outer product* from equation (13).

Definition 2.15 [11] *The **Inner Product** on a real vector space $\mathcal{V}$ is the mapping $<\cdot,\cdot>: \mathcal{V}\times\mathcal{V}\to\mathbb{R}$ such that for all $\mathbf{x},\mathbf{y},\mathbf{z}\in\mathcal{V}$ and $\alpha,\beta\in\mathbb{R}$,*

i) $<\mathbf{x},\alpha\mathbf{y}+\beta\mathbf{z}>=\alpha<\mathbf{x},\mathbf{y}>+\beta<\mathbf{x},\mathbf{z}>$ *(Linearity in Second Argument)*

ii) $<\mathbf{x},\mathbf{y}>=<\mathbf{y},\mathbf{x}>$ *(Symmetric)*

iii) $<\mathbf{x},\mathbf{x}>\geq 0$ *(Non-Negative)*

iv) $<\mathbf{x},\mathbf{x}>=0$ *if and only if* $\mathbf{x}=\mathbf{0}$ *(Positive Definite)*

Definition 2.16 [11] *A vector space with an inner product is called an **Inner Product Space**.*

Note that in $\mathbb{R}^n$, the inner product is also referred to as the *dot product* and defined as $<\mathbf{x},\mathbf{y}>=\sum_{i=1}^n x_iy_i$. Defining an inner product on a vector space induces a norm on the vector space.

Corollary 2.17 [11] *Every inner product space, $\mathcal{V}$, is a normed vector space with the norm defined by,*

$$||\mathbf{x}||_2=(<\mathbf{x},\mathbf{x}>)^{1/2}=\left(\sum_{i=1}^n x_i^2\right)^{1/2}. \tag{21}$$

More generally, for $1\leq p<\infty$, the p- norm on $\mathbb{R}^n$ is defined by,

$$||\mathbf{x}||_p=(<\mathbf{x},\mathbf{x}>)^{1/p}=\left(\sum_{i=1}^n |x_i|^p\right)^{1/p}. \tag{22}$$

For $p = \infty$, the ∞-norm, or *maximum norm* is defined by,

$$||\mathbf{x}||_\infty = \max\{|x_i|\}_{i=1}^n. \tag{23}$$

Definition 2.18 [12] *A* ***Banach Space*** *is a normed vector space that is a complete metric space with respect to the metric derived from its norm.*

Altogether, we begin with the vector space $\mathbb{R}^n$, define the inner product via the dot product, which induces the norm defined in equations (22) and (23). This norm, combined with the fact that $\mathbb{R}^n$ is a complete metric space (i.e every Cauchy sequence converges), yields the important result of $\mathbb{R}^n$ is a finite-dimensional Banach Space. For the case that $p = 2$, we obtain a special type of Banach Space.

Definition 2.19 [12] *A* ***Hilbert Space*** *is a complete inner product space.*

It is important to note that every Hilbert space is a Banach space with respect to the norm defined in equation (21). As we have demonstrated, since $\mathbb{R}^n$ is an inner product space and completed by the norm in equation (21), then it is a finite-dimensional Hilbert space. With these key concepts on hand, we move on to the introduction of function spaces. Similar to how vectors operate in a vector space, functions operate in function spaces with defined mappings.

Definition 2.20 [21] *A* ***function space*** *is the set of all real-valued functions on a set X, denoted $\ell(X) = \{f : X \to \mathbb{R}\}$.*

Note that we can extend the concepts of function spaces as a mapping of a set to the complex numbers ($\mathbb{C}$), but for the scope of applications in this paper, we will only consider the real numbers ($\mathbb{R}$). The set X is commonly referred to as the *indexing set*. Also, more in line with our application, consider X to be finite, then we can we list the elements of the indexing set such that for some $n \in \mathbb{N}$ and some $f \in \ell(X)$,

$$X = [x_1, x_2, ..., x_n] \rightarrow [f(x_1), f(x_2), ..., f(x_n)]. \tag{24}$$

This is known as the *array representation* which allows us to view these lists of objects as vectors and establishes a bijective relationship between $X \in \mathbb{R}^n$ and $l(X)$. For example, consider the set $X = \{x_1, x_2, ..., x_n\}$, where $f, g \in \ell(G)$ and $\alpha, \beta \in \mathbb{R}$. Then, $\alpha f(x_i) + \beta g(x_i) \in \mathbb{R}$. In fact, we get all of the same properties as we do for the traditional sense of a vector space, which lead to three key concepts regarding function spaces.

Theorem 2.21 [21] *A function space $\ell(X)$ is a vector space.*

Corollary 2.22 [21] *If X is a finite set, then $\ell(X) \cong \mathbb{R}^n$ where $n = |X|$.*

Corollary 2.23 [21] *If X is a finite set, then for all functions $f, g \in \ell(X)$, $fg \in \ell(X)$.*

Extending function spaces defined as vector spaces, in Corollary 2.22, we obtain the isometric property for function spaces of finite sets. However, in contrast to traditional vector spaces, Corollary 2.23 introduces the additional property of the product of functions being closed in a function space. From this, we can define an inner product on a function space $\ell(X)$ for a finite set X,

$$< f, g >= \sum_{i=1}^{n} f(x_i)g(x_i) \quad \text{for all} \quad f, g \in \ell(X). \tag{25}$$

Consequently, this induces a norm defined by,

$$||f||_2 = \left(\sum_{i=1}^{n} |f(x_i)|^2 \right)^{1/2} . \tag{26}$$

Thus, $\ell(X)$ is a complete inner product space with respect equations (25) and (26), and by Definition 2.19, a Hilbert space. With the initial notion of a function space $\ell(X) = \{f : X \to \mathbb{R}\}$ we also obtain familiar inequalities from traditional vector spaces.

Theorem 2.24 [11] *$< f, g >$ is positive definite, and satisfies the **Cauchy-Schwarz Bunyakovsky Inequality**,*

$$| < f, g > | \leq ||f|| \ \ ||g|| \quad \textit{for all} \ \ f, g \in \ell(X). \tag{27}$$

It immediately follows that the triangle inequality still holds in $\ell(X)$ as well.

$$||f + g|| \leq ||f|| + ||g|| \quad \text{for all} \ \ f, g \in \ell(X) \tag{28}$$

Definition 2.25 *For $< f, g >$ defined on $\ell(X)$, a set of functions $\{u_n\}$ is an **orthonormal set** if*

$$< u_m, u_n >= \delta_{mn}. \tag{29}$$

Theorem 2.26 [12] *Every finite dimensional Hilbert space has an orthonormal basis.*

Corollary 2.27 [21] *For a set X where $|X| = n$, $\ell(X) = \{f : X \to \mathbb{R}\}$, there exists $\{u_n\}$ that is an orthonormal basis for $\ell(X)$.*

Definition 2.28 [12] *Let $\mathcal{V}$ and $\mathcal{W}$ be vector spaces.The mapping $\mathcal{T} : \mathcal{V} \to \mathcal{W}$ is a **linear transformation** if*

$$\mathcal{T}(\alpha\mathbf{v} + \beta\mathbf{w}) = \alpha\mathcal{T}(\mathbf{v}) + \beta\mathcal{T}(\mathbf{w}) \quad \textit{for all} \ \ \mathbf{v}, \mathbf{w} \in \mathcal{V} \ \ \textit{and} \ \ \alpha, \beta \in \mathbb{R}. \tag{30}$$

If $\mathcal{V}$ is a function space on a finite set X with $|X| = n$, $\ell(X) = \{f : X \to \mathbb{R}\}$, and $\mathcal{W} \cong \mathbb{R}^m$, then $\mathcal{T}$ maps from the set of functions on $x \in X$ ($f \in \ell(X)$), to an element $\mathbf{w} \in \mathcal{W}$. Let $\mathbf{f} = [f(x_1), f(x_2), ..., f(x_n)]$. Then

$$\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \mathcal{T}(\mathbf{f}) = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix}. \qquad (31)$$

Theorem 2.29 [12] *If $\mathcal{T}$ is a linear transformation from an n dimensional vector space $\mathcal{V}$ into a vector space $\mathcal{W}$, then given any orthonormal basis $\{ \mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_n \}$ of $\mathcal{V}$, the linear transformation $\mathcal{T}$ is given by,*

$$\mathcal{T}(\mathbf{v}) = \sum_{k=1}^{n} < \mathbf{v}, \mathbf{e}_k > \mathbf{a}_k \quad for \quad all \quad \mathbf{v} \in \mathcal{V}, \quad \mathcal{T}(\mathbf{e}_k) = \mathbf{a}_k. \qquad (32)$$

3 RESEARCH

The goal of this section is to describe the current applications of the mathematical methodologies described in the mathematical background in the domain of "Sports Analytics". Principal Component Analysis and Non-Negative Matrix Factorization techniques provide a framework for sports analysts to develop methods for characterizing and profiling players, teams, and sport specific actions like "pitches" in baseball or "shots" in basketball. Professional sports organizations like MLB and NBA have made data publicly available for where this type of analysis has gained attention in public articles and blogs. The NFL has incorporated an annual competition called "The Big Data Bowl" that mimics Kaggle competitions for free and open discussion and collaboration with football related data collected by the NFL.

The direct access to data for casual fans and amateur practitioners of data science and machine learning is a significant factor in the development of the techniques described in this paper in the field of sports analytics. In this section, we will first investigate two sport specific case studies for general applications of Principal Component Analysis and Non-Negative Matrix Factorization. Then we will review the applications of tensor decomposition frameworks with a generalised example and in relation to the tHoops framework [8]. In their work, Pelechrinis et al. propose a method of analyzing spatio-temporal sports data using tensor decomposition methods to create profiles with respect to competitor metric.

3.1 PCA in Sports Analytics

Terminology used in baseball pitch analytics like “spin rate”, “spin direction”, and “release velocity” have become common at all levels of engagement with the game since the introduction of StatCast in 2015. In this case study, the anatomy of pitch types are analyzed by their measured metrics collected from the sophisticated tracking systems that make up StatCast. The data set chosen for this analysis is a subset of the pitch data from the 2016 MLB season that consists of pitches thrown on Mondays (See Appendix 1.1 for data source and feature descriptions).

Table 1: 2016 Monday Baseball Pitch Data Header

	probCalledStrike	releaseVelocity	spinRate	spinDir	locationHoriz	locationVert	movementHoriz	movementVert
1	0.98	94.20	2, 044.22	205.48	-0.37	2.93	-6.93	8.28
2	0.74	97.10	1, 966.32	220.14	0.34	3.22	-7.48	7.35
3	0.97	96.50	2, 127.17	198.82	0.39	2.27	-5.22	9.79
4	1	95.60	1, 947.11	198.73	-0.004	2.38	-7.24	8.40
5	1	95.60	1, 903.08	205.50	0.27	2.42	-6.79	9.37
6	0.32	98.30	2, 038.06	206.73	-0.21	1.43	-8.30	7.96

Table 2: 2016 Monday Pitch Summary

Feature	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
probCalledStrike	73,569	0.48	0.43	0.00	0.01	0.95	1.00
releaseVelocity	73,569	88.51	5.93	60.00	84.70	92.90	105.00
spinRate	73,569	2,201.33	318.24	159.04	2,062.02	2,396.80	3,472.37
spinDir	73,569	183.78	61.35	0.01	148.75	222.64	359.99
locationHoriz	73,569	−0.04	0.86	−4.05	−0.63	0.56	3.97
locationVert	73,569	2.27	0.93	−2.54	1.68	2.87	6.75
movementHoriz	73,569	−0.79	6.40	−16.21	−6.21	4.55	20.42
movementVert	73,569	5.25	5.25	−16.21	2.83	9.04	17.85

After some data cleaning and feature selection, we obtain the descriptions of key pitch metrics for 73,569 pitches thrown in the 2016 season. The goal of this analysis is to uncover the underlying structure of pitches based on the physical attributes measured for each pitch type through a decomposition of the “Pitch Data Matrix” for all pitches.

Table 3: Pitch Correlation Matrix

	probCalledStrike	releaseVelocity	spinRate	spinDir	locationHoriz	locationVert	movementHoriz	movementVert
probCalledStrike	1	0.08	0.02	0.03	-0.03	0.17	-0.01	0.09
releaseVelocity	0.08	1	0.09	0.29	-0.03	0.25	-0.27	0.71
spinRate	0.02	0.09	1	-0.21	0.09	0.07	0.13	-0.07
spinDir	0.03	0.29	-0.21	1	-0.18	0.05	-0.73	0.29
locationHoriz	-0.03	-0.03	0.09	-0.18	1	−0.13	0.16	0.01
locationVert	0.17	0.25	0.07	0.05	-0.13	1	-0.03	0.25
movementHoriz	-0.01	-0.27	0.13	-0.73	0.16	-0.03	1	-0.20
movementVert	0.09	0.71	-0.07	0.29	0.01	0.25	-0.20	1

In general exploratory data analysis of the pitch data, it is not uncommon to see correlations between features. In this case, as seen in Table 3, horizontal movement and spin direction have a moderate negative correlation at approximately 73 percent. This is a perfectly normal association in how pitches are thrown and move based on the tilt of the axis for which the spin revolves. Also, release velocity and vertical movement have a moderate correlation of approximately 71 percent. Again, the association is justified by example of four seam fastballs having an approximate flat back spin while curveballs and other non-fastball type pitches have a combination of degrees of lateral and downward movement. Many people familiar with baseball know pitch types and spin directions intuitively, or have a unique understanding of the “Magnus Effect” and it’s applications in baseball.

While no two pitches from two different pitchers are exactly the same, there are classes of pitch types that have been generally accepted based on the movement or grip type of the pitch. With this heuristic being so prevalent before technology allowed for these metrics to be recorded during live games, it is reasonable to look for clusters of pitches within the data.

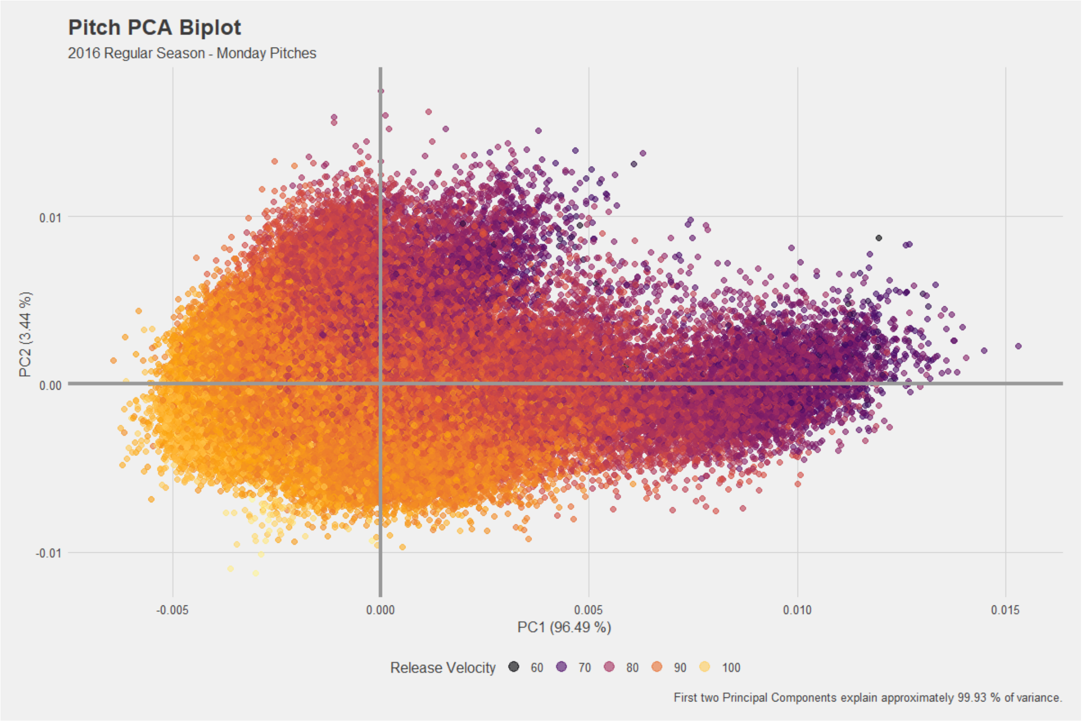

Figure 7: Pitch PCA Biplot - Velocity.

In Figure 7, it is clear that the first principal component is creating separation based on the release velocity of the pitch. However, there is still a lot of overlap in the pitches, which is expected with the variety of pitchers included in this data set. Altogether, there are 611 individual pitchers contributing to the global data set of pitches. Someone with extensive domain knowledge of baseball and pitching may see another structure beneath heat map in Figure 7. This person would know that pitch types vary between left-handed and right-handed pitchers. For example, a two-seam fastball from a left-handed pitcher would actually move in the opposite direction of a two-seam fastball from a right-handed pitcher.

Figure 8 really begins to tell the story of pitch structure between a certain class of

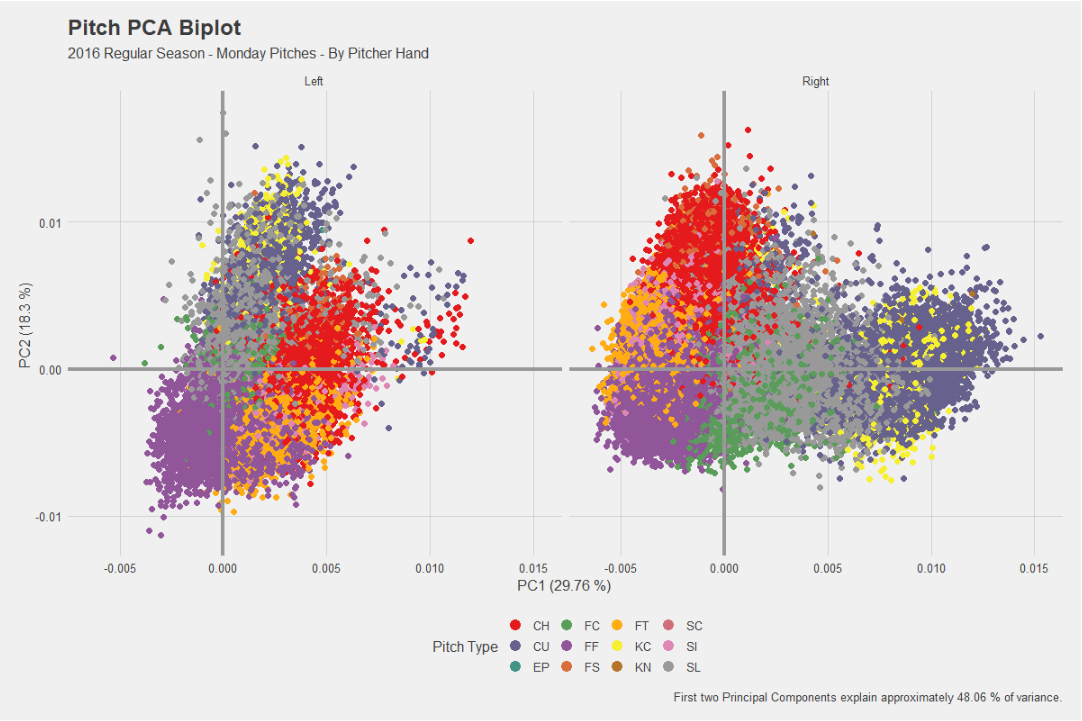

Figure 8: Pitch PCA Biplot - Pitch Type by Pitcher Hand.

pitcher, being “left-handed” v.s “right-handed”. It can be seen that the “FF” pitch type (Four-Seam Fastball) is in the bottom left of the each plot. This pitch is the most similar between the right-handed and left handed pitchers in terms of release velocity and horizontal movement. Note, the symmetry of every other pitch type between left and right handed pitchers reflecting along an invisible $y = x$ line on the PCA Biplot for each pitcher class. This type of analysis is leading to more structural insights behind the anatomy of pitch types based on the information provided.

The contrast plot of left and right handed pitchers in Figure 8 is an example of creating another axis in which to analyze the decomposition of pitch types. By peeling back another layer for these principal components in which to separate, more information becomes visible. However, principal component analysis does require some contextual knowledge of how the data is collected in order to make valid interpretations. Another distinction to possibly consider would be the contrast between individual pitchers.

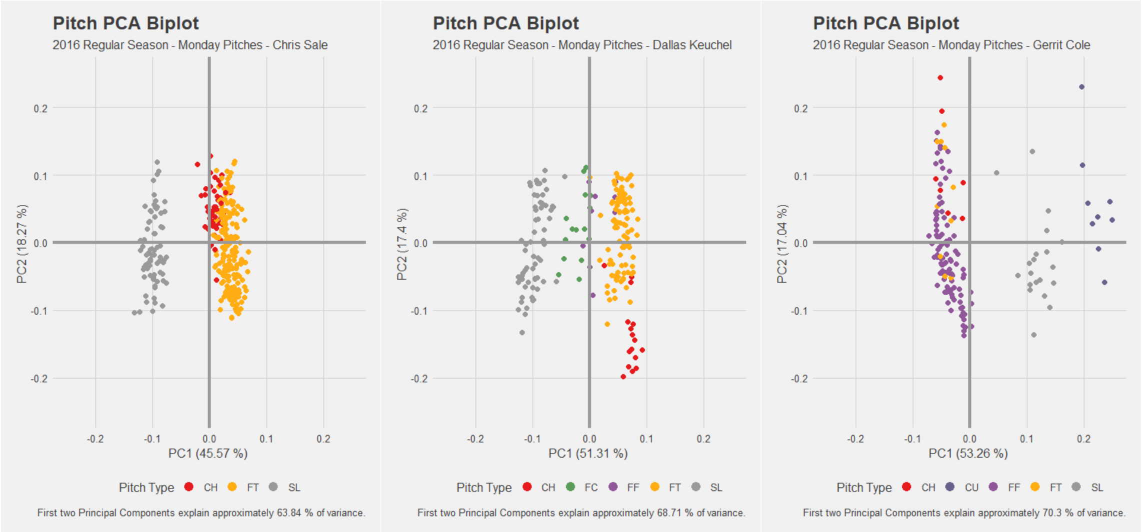

Figure 9: Pitch PCA Biplot - Pitcher Comparison.

Figure 9 shows the contrast between three prominent major league pitchers Chris Sale, Dallas Keuchal, and Gerrit Cole. The key focus in this comparison is in the separation of pitch classes between the pitchers and how they vary. For example, Chris Sale's Slider (SL) is quite distinct from his other two pitches. Also, his Change-Up (CH) and Two-Seam Fastball (FT) seem to be closely related but have variation between the two pitches. In particular, these two pitches have similar movements with different release velocities. For Dallas Keuchel, there are similarities in the spread of the pitch types to that of Chris Sale, except for the small cluster of Change-Ups in the fourth quadrant. Finally, Gerrit Cole throws a lot of Four-Seam Fastballs with Change-Ups and Two-Seam Fastballs mixed in the left cluster. His Slider and Curveball (CU) are the distinct pitches here.

Upon further investigation of these principal components between the pitchers in Figure 9, the commonality between them is the separation of pitches along principal component one. This means the most variation between each of the pitches is comprised of “Spin Direction”. For this reason, the pitch types for Chris Sale and Dallas Keuchel clusters are spread similarly along principal component one while Gerrit Cole has a reflected mapping. This is because Gerrit Cole is right-handed and Chris Sale and Dallas Keuchel are left-handed. To address the coordinate flip for Change-Ups between Chris Sale and Dallas Keuchel, PCA was applied to each subset of data for each pitcher individually. As a result, the principal axes are not orthogonal between pitchers because they have structurally different pitches.

This leads to a very important concept in how to handle the additional axes for consideration in Tensor PCA; how to maintain orthogonality of the principal axes. The main difference between the visualizations in Figure 8 and Figure 9 is in how the layers were constructed. For Figure 8, PCA was applied to the global data set, then indexed by left and right handed pitchers for comparison. For Figure 9, the data was first indexed for individual pitchers, then PCA was applied. In terms of tensors, this is an sample extension of the CP Decomposition in which there was pitcher-indexed re-scaling applied prior to PCA.

3.2 NMF in Sports Analytics

In 2013, the NBA mandated that each team's venue be outfitted with player tracking systems [8]. Since then, many techniques have been applied to the spatial data collected in the NBA. Shot charts are a notable graphic for sports media outlets to display during television broadcasts for fan experience. However, basic shot charts over the course of a game or season can become cluttered and lose their value for insight. For example, an initial exploration of the spatial component of shots taken by players in the 2014-15 NBA season may take the form of one of the following types of visualizations depending on the scope.

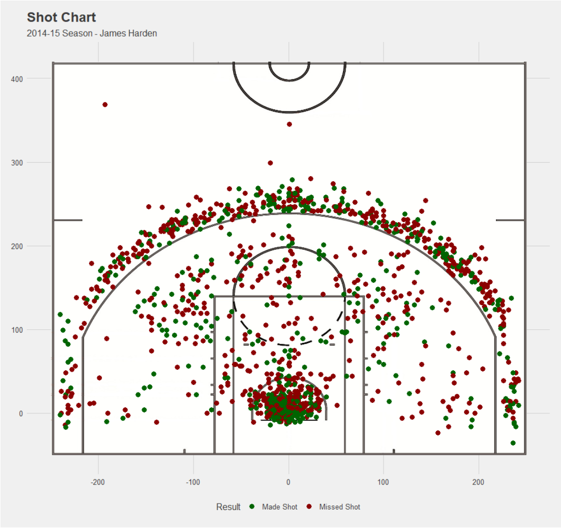

Figure 10: Shot Chart - James Harden

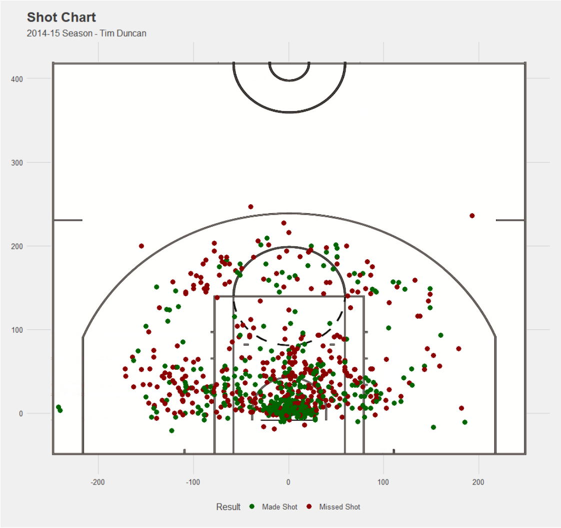

Figure 11: Shot Chart - Tim Duncan

In Figure 10, we can see a scatter of shots taken by James Harden in the 2014-15 season. For shot charts of this type, there tend to be clusters of dense areas on the court for around the basket for lay-ups and along the three-point line as these are common areas for which shots are taken. However, the frequency of mid-range shots between these two areas can vary between players such as with Tim Duncan's shot chart in Figure 11. The casual fan of the NBA, is familiar with player positions and how certain types of player fit a "mold". Consider a player like Tim Duncan, who stood at 6'-11", that was a typical post player that played the "Center" position for his entire 19 year career. Figure 11 confirms a common bias that post-players do not deviate far from the basket when taking a shot.

Without knowing the players' identity, after comparing the shot charts in Figure 10 and Figure 11, one could hypothesize that they were taken by two different types of players based on their shot selection areas. Prior to spatial data becoming available, this type of "player binning" was left to domain experts like coaches or scouts that evaluate players. In recent years, more applications like principal component analysis and non-negative matrix factorization have offered insights with more objective sentiment by uncovering mathematical structure beneath the shot charts.

Figure 12 is a generalized heat map of successful shots made in that season by the 347 individual players in the dataset. The zone, "Restricted Area 2" (RA2) in Figure 12 is directly under the basket and where the most shots are taken in the 2014-15 NBA season. This is an intuitive observation as well since lay-ups and dunks happen there exclusively. In the examples so far, we can only determine that players take shots of minimal distance from the basket with high frequency as do Harden and Duncan. But outside of that, they differ where they take secondary shots. It is appropriate to hypothesize if there are other players that have similar shot patterns to the two prominent players described above.

In a blog post entitled *Using Machine Learning to Find the 8 Types of Players*

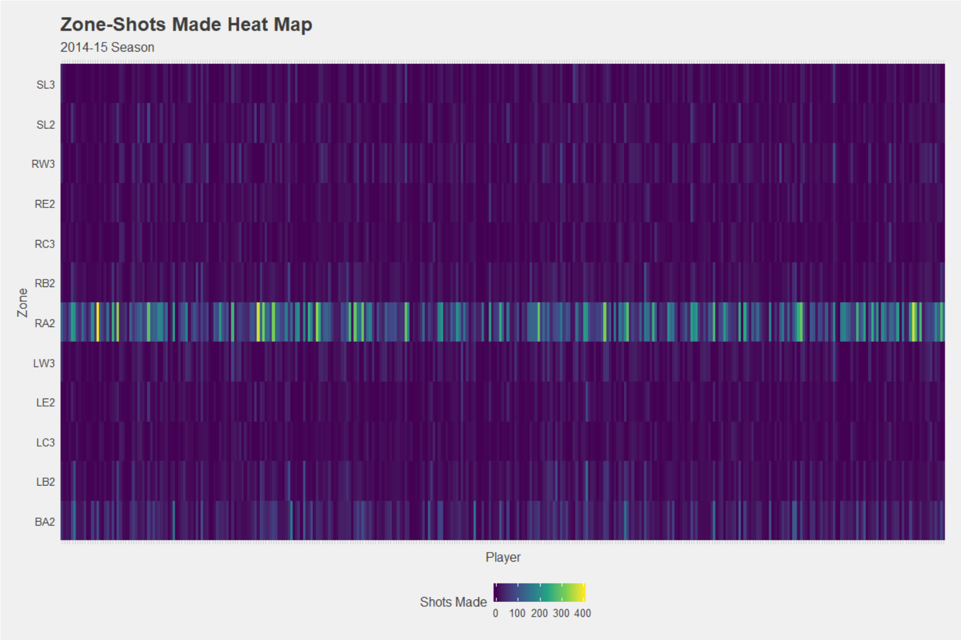

Figure 12: Court Zone Heat Map

in the NBA , the author describes multiple machine learning algorithms applied to a similar NBA basketball data set to expand the definition of player types from their simple position labels [16]. In the same spirit, we are going to apply non-negative matrix factorization to shooting data to try to uncover different types of shooters based on their spatial shot selections.

In this application, we begin with a matrix of values for counts of shots taken in each zone for each player depicted in Figure 12. This matrix, $\mathbf{A} \in \mathbb{R}^{347\times 12}$ consists of 347 players and 12 court zones. By Corollary 2.9, there exists $\mathbf{B} \in \mathbb{R}^{m\times r}$ and $\mathbf{C} \in \mathbb{R}^{r\times 12}$ such that $\mathbf{A} \approx \mathbf{BC}$. We use the following objective function to approximate r,

$$\min_{B,C} ||\mathbf{A} - \mathbf{BC}||_F^2 \quad \text{subject to} \quad b_{ij}, c_{ij} \geq \quad 0 \;\; \forall \;\; i, j. \tag{33}$$

When applying non-negative matrix factorization, observe that the product of $\mathbf{B}$

$$\begin{array}{cc} & \mathbf{A} \\ & \text{Zones} \\ \text{Players} & \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \end{array} \begin{array}{c} \approx \\ \\ \approx \end{array} \begin{array}{cc} & \mathbf{B} \\ & \text{Factors} \\ \text{Players} & \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mr} \end{bmatrix} \end{array} \begin{array}{cc} & \mathbf{C} \\ & \text{Zones} \\ \text{Factors} & \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{r1} & c_{r2} & \cdots & c_{rn} \end{bmatrix} \end{array}$$

Figure 13: NBA NMF Latent Factors

and **C** matrices is a low rank ($r < n$) approximation of the original matrix. In this case, the decomposition of **A** associates weights to latent factors between the players and court zones.

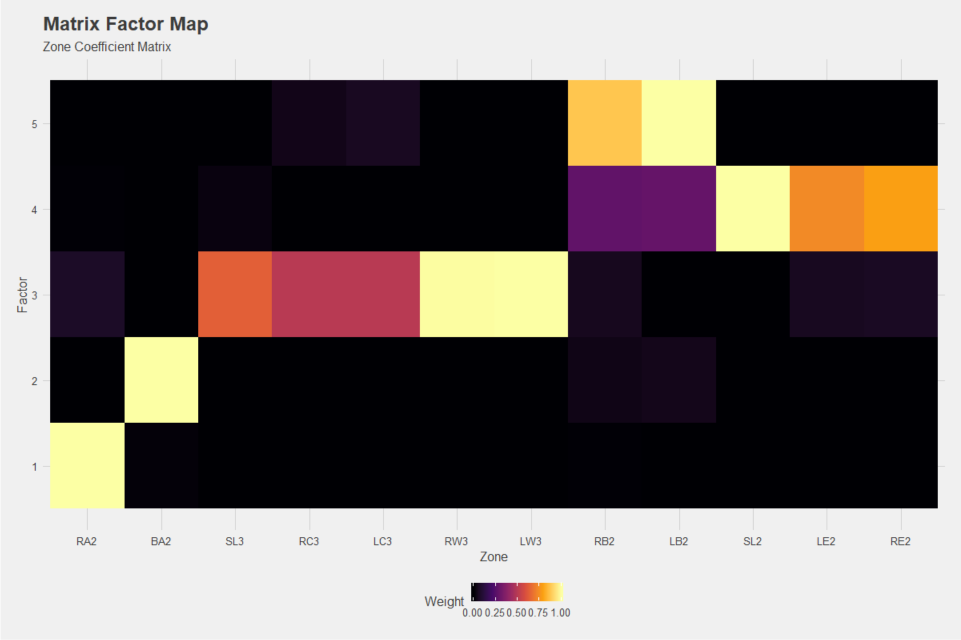

Figure 14: Factor Coefficient Map

The factors weights in Figure 14 partition into notable groupings in terms of shot locations. Factor 1 is heavily weighted for just zone “RA2” which is the restricted area directly beneath the basket. Subsequently, factor 2 is dominated by "BA2", which is a mild radial extension of RA2. Factor 3 is composed of all 3 point areas on the court. Factor 4 is made up of mid-range areas around the top of the key and lastly, factor 5 is composed of the mid-range baseline shots. It is important to note here that these official court zones were used by Pelechrinis et al. as they include natural boarders on the court [8]. These factors are capturing a structural relationships that exists between the zones underlying the heat map in Figure 12.

Player	Value	Position
Andre Drummond	1.00	C
Tyreke Evans	0.930	SG
DeAndre Jordan	0.919	C
Enes Kanter	0.827	C
Anthony Davis	0.825	PF
DeMarcus Cousins	0.788	C
LeBron James	0.785	SF
Derrick Favors	0.781	PF
Russell Westbrook	0.749	PG
Greg Monroe	0.748	PF

Table 4: Factor 1 Weights

Player	Value	Pos
Stephen Curry	1.00	PG
Klay Thompson	0.831	SG
Kyle Korver	0.769	SG
James Harden	0.724	SG
JJ Redick	0.695	SG
Damian Lillard	0.684	PG
Trevor Ariza	0.675	SF
Danny Green	0.663	SG
Wesley Matthews	0.598	SG
Robert Covington	0.574	SF

Table 5: Factor 3 Weights

In Table 4, we can see a dynamic group of players that are grouped together that share similar qualities. This factor is heavily weighted by shots taken in the Restricted Area. More notable players like that of Andre Drummond, Anthony Davis, DeMarcus Cousins, LeBron James, and Russell Westbrook, are known for attacking the rim with aggressive dunks. By their listed positions, four of them are labeled traditional Centers (C) while others are Power Forwards (PF) and Shooting Guards (SG). This factor describes players that favor shooting the ball directly underneath the goal in the Restricted Area.

Table 5 shows the top ten weights of players for Factor 3, which profiles 3 point shooters. For the casual fan, it should be no surprise who the top two players in this category are. Since the 2014-15 season, teammates Stephen Curry and Klay Thompson have acquired the collective nickname of the “Splash Brothers” as result of having higher than average 3 point shooting percentages. In addition to being a constant long range threat duo, Curry, Thompson, and the Golden State Warriors also began their 5 year NBA Finals appearance run in 2015.

3.3 Tensor PCA Techniques

The case studies demonstrating PCA and NMF in analyzing baseball pitches and basketball shots allow analysts to gain insights into the structure of the data collected. For these unsupervised models, analysis does not yield “absolutes” but rather patterns to inform further research. This can open a dialogue between researchers and domain experts in creating new questions about the data. As dimensionality, complexity, and volume of data in sports analytics increase, these techniques help researchers ask better questions that can isolate the signal from the noise. In this section, we introduce tensor decomposition techniques with the same objective of identifying hidden latent factors beneath the surface of the data.

As previously mentioned Pelechrinis et al. [8] introduced a multi-aspect analytical framework for analyzing spatio-temporal basketball data called “tHoops”. Their work is a direct extension of the basketball case study described in this paper to address a gap created in the NMF approach by taking into account the temporal aspect that can affect a player's shot selection. In this particular review, we will focus on the general application to the shot selection process by players.

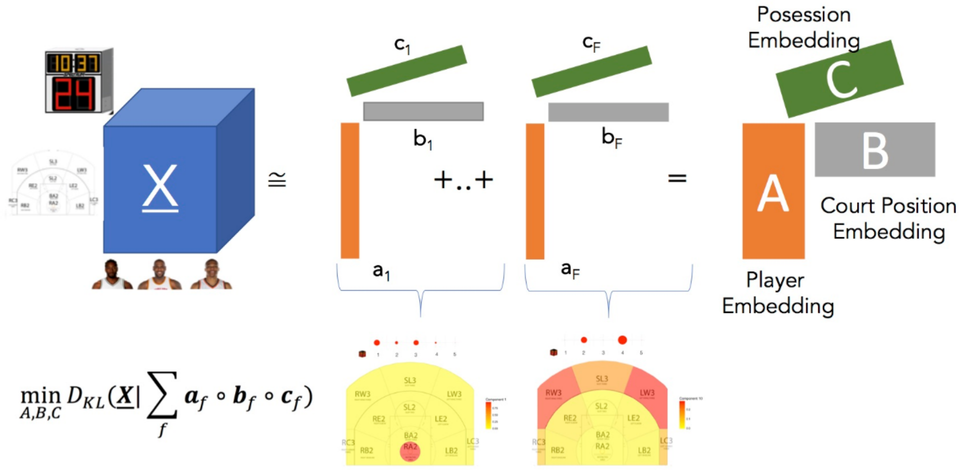

Figure 15: tHoops Framework [8]

Figure 15 is a complete description of the tHoops framework where $\underline{\mathbf{X}}_{ijk}$ is the number of shots that player i took from court zone j during time k. Pelechrinis et al. assert that "tHoops identifies prototype patterns in the data, expressed as triplets of vectors corresponding to the three dimensions of $\underline{\mathbf{X}}$ respectively." [8] The coefficients of each vector in the "Player Embedding" matrix correspond to a "soft membership" of each player to the respective component (pattern) [8]. Each component is a temporal-spatial pattern decomposed from the tensor $\underline{\mathbf{X}}$.

Similar to how a tensor is a generalization of a matrix, the generalization of SVD in n-modes is the Canonical Polyadic (CP) or PARAFAC Decomposition [8]. Figure

15 depicts how $\underline{\mathbf{X}}$ is expressed as the sum of rank-1 tensors.

$$\underline{\mathbf{X}} \approx \sum_{f=1}^{F} \mathbf{a}_f \circ \mathbf{b}_f \circ \mathbf{c}_f. \tag{34}$$

In summary, **a**, **b**, and **c** correspond to players, court zones, and period respectively. Each of the F components can be thought of as *clusters* and the coefficients of the indices in the component correspond to a degree of association within the cluster. The larger the coefficient, the closer the association [8]. The component represents a "soft representation" of players that tend to take shots from similar court zones at similar times. Consequently, the **b** and **c** vectors correspond to latent factors of spatio-temporal patterns derived from $\underline{\mathbf{X}}$.

The primary advantage the tensor decomposition approach has over the matrix factorization techniques is the ability to consider multiple aspects simultaneously, which allows for richer context for latent patterns. One could argue that we could compare shot charts directly for a subset of shots taken during a particular game time, such as the fourth period. But as discussed in the Pitch PCA analysis, doing PCA (or NMF) locally vs globally are two different projections that approximate different data structures. Therefore, in taking such an approach, we would have inadvertently thrown out possibly valuable information related to other time periods that can interact with shot selections in the fourth period alone. Consequently, as more contextual factors are taken into consideration, i.e higher dimensions, we lose more information, making it more difficult to identify quality patterns.

It is important to emphasize that just as with applying PCA and NMF, tHoops (or tensor decomposition in general) is an unsupervised learning method that makes it difficult to assess the model. These methods require additional contextual knowledge related to the sports, players, and data acquisition methods to interpret the results in a meaningful way. In other words, using a latent pattern to inform a defensive strategy against a particular player/team at a particular time does nothing for an individual that cannot develop defensive strategies. Value is a consumer driven metric, and in data science, it is assessed by the decision-maker.

4 APPLICATION TO NASCAR

Background research for applications of data science and machine learning techniques in NASCAR are scarce in the public domain. There certainly is not a shortage of data collected by the teams to implement common algorithms and analyses, but unlike the data collected with MLB, NBA, and NFL, the NASCAR data is not made available through public API's or other databases. As a consequence, any novel techniques applied and/or developed by individual teams are kept in house exclusively. NASCAR is an incredibly competitive sport where teams are notorious for pushing the envelope to get as much speed out of the car as possible. Every aspect of the car's body is now measured with laser precision as well as human inspection for mechanical components by NASCAR officials before and after each race. There have also been major changes in the style of racing since 2017 with the introduction of "Stage Racing" and subsequently, a new points system and playoff structure. Lastly, NASCAR has recently announced the introduction of the "Next Gen" car that will become standard for all teams to adopt in 2021. With these recent and future changes on the horizon, teams are eager to maximize the utility of the data available by NASCAR across multiple levels within the organization.

4.1 Data Source

In early 2018, prior to the inaugural Monster Energy Cup Series (MENCS) race at Daytona International Speedway, NASCAR announced that timing, scoring, location, and telemetry data collected from cars during practice, qualifying , and race events would be shared among teams [15]. Prior to this announcement, this type of data was collected by a company called Sports Media Technology (SMT) that provides data-driven applications accessible to television partners that allow fans to engage with a data-rich viewing experience in real time [18]. However, teams in the garage only had access to their own data that was collected from the Electrical Control Unit (ECU) on the car and could only access it by physically plugging into the car after the race. Other data streams were collected via data scraping techniques by some teams which likely prompted the announcement as a response from NASCAR to even the playing field [15].

The raw data is made accessible to teams by NASCAR from private servers in real time during races. From there, teams can use the data for custom or third party applications and data warehousing at individual teams' expenses. The data used in this thesis was made available by an industry liaison, in particular, we focus on the "Timing and Scoring" data for every regular season race in the Monster Energy Cup Series for the 2018 season. In total, there were 36 races considered in this data set. Races that did not count towards the NASCAR points scoring structure, specifically the "Clash", "Dual" and "All-Star" races, were not considered.

The raw data contains 362,075 observation and 30 features. Timing and scoring data is primarily index by lap as each competitor crosses the Start-Finish line. Table 6 shows the first 6 observations of the Timing and Scoring data set which includes 14 categorical features, 15 numerical features and one date-time feature. Table 7 contains lap counts for each stage and track lengths for each race. (Note: $*$ indicates playoff races.)

Table 6: 2018 NASCAR Timing and Scoring Data Header

	SeriesID	SeriesNameShort	EventName	TrackNameShort	TrackName	TimingSessionID	SessionNameShort	SessionName	RunType	StartDateTime
1	1	MENCS	18HOM	HOM	Homestead Miami Speedway	2983	Race	Ford EcoBoost 400	Race	1542555000
2	1	MENCS	18HOM	HOM	Homestead Miami Speedway	2983	Race	Ford EcoBoost 400	Race	1542555000
3	1	MENCS	18HOM	HOM	Homestead Miami Speedway	2983	Race	Ford EcoBoost 400	Race	1542555000
4	1	MENCS	18HOM	HOM	Homestead Miami Speedway	2983	Race	Ford EcoBoost 400	Race	1542555000
5	1	MENCS	18HOM	HOM	Homestead Miami Speedway	2983	Race	Ford EcoBoost 400	Race	1542555000
6	1	MENCS	18HOM	HOM	Homestead Miami Speedway	2983	Race	Ford EcoBoost 400	Race	1542555000

	CarNumber	CompetitorName	Make	LapNumber	LapFlag	LapTime	LapPassTime	LapPassLocation	LapStanding	LapStandingFinal
1	00	Landon Cassill(i)	Chv	1	Green	35.311	38.32	Track	33	33
2	1	Jamie McMurray	Chv	1	Green	34.176	36.355	Track	21	21
3	10	Aric Almirola	Frd	1	Green	33.894	34.864	Track	10	10
4	11	Denny Hamlin	Tyt	1	Green	33.503	33.503	Track	1	1
5	12	Ryan Blaney	Frd	1	Green	33.844	35.437	Track	14	14
6	13	Ty Dillon	Chv	1	Green	34.376	37.411	Track	30	30

	LapFastTime	LapFastLap	LapTimeToLeader	LapsToLeader	LapTimeToAhead	LapsToAhead	LapPitStopCount	LastLapPitLap	LapStartPosition	LapsLed
1	35.311	1	4.817	0	0.213	0	0	0	32	0
2	34.176	1	2.852	0	0.067	0	0	0	21	0
3	33.894	1	1.361	0	0.007	0	0	0	10	0
4	33.503	1	0	0	0	0	0	0	1	1
5	33.844	1	1.934	0	0.006	0	0	0	15	0
6	34.376	1	3.908	0	0.008	0	0	0	31	0

Table 7: Race Track Characteristics

Track	Event	Stage1	Stage2	Stage3	FinalStage	TrackLength(mi)
Atlanta	18ATL	85	170	NA	325	1.50
Auto Club	18CAL	60	120	NA	200	2.00
Bristol	18BRI1	125	250	NA	500	0.53
Bristol-2	18BRI2	125	250	NA	500	0.53
Charlotte	18CHA	100	200	300	400	1.50
Charlotte-2*(road)	18CHR	25	50	NA	109	2.28
Chicagoland	18CHI	80	160	NA	267	1.50
Darlington	18DAR	100	200	NA	367	1.36
Daytona	18DAY1	60	120	NA	200	2.50
Daytona-2	18DAY2	40	80	NA	160	2.50
Dover	18DOV1	120	240	NA	400	1.00
Dover-2*	18DOV2	120	240	NA	400	1.00
Indianapolis	18IND	50	100	NA	160	2.50
ISM (Phoenix)	18PHO1	75	150	NA	312	1.00
ISM (Phoenix)-2*	18PHO2	75	150	NA	312	1.00
Kansas	18KAN1	80	160	NA	267	1.50
Kansas-2*	18KAN2	80	160	NA	267	1.50
Kentucky	18KEN	80	160	NA	267	1.50
Las Vegas	18LAS1	80	160	NA	267	1.50
Las Vegas-2*	18LAS2	80	160	NA	267	1.50
Martinsville	18MAR1	130	260	NA	500	0.52
Martinsville-2*	18MAR2	130	260	NA	500	0.52
Miami*	18HOM	80	160	NA	267	1.50
Michigan	18MIC1	60	120	NA	200	2.00
Michigan-2	18MIC2	60	120	NA	200	2.00
New Hampshire	18LOU	75	150	NA	301	1.06
Pocono	18POC1	50	100	NA	160	2.50
Pocono-2	18POC2	50	100	NA	160	2.50
Richmond	18RIC1	100	200	NA	400	0.75
Richmond-2*	18RIC2	100	200	NA	400	0.75
Sonoma	18SON	25	50	NA	110	1.99
Talladega	18TAL1	55	110	NA	188	2.66
Talladega-2*	18TAL2	55	110	NA	188	2.66
Texas	18TEX1	85	170	NA	334	1.50
Texas-2*	18TEX2	85	170	NA	334	1.50
Watkins Glen	18WAT	20	40	NA	90	2.45

Initial data exploration of the complete 2018 season of timing and scoring data can be quite cumbersome with the variety of categorical variables available. For example, lap times can vary widely between tracks and even with this mind, visualizing the distributions of lap times between them requires some extra leg work.

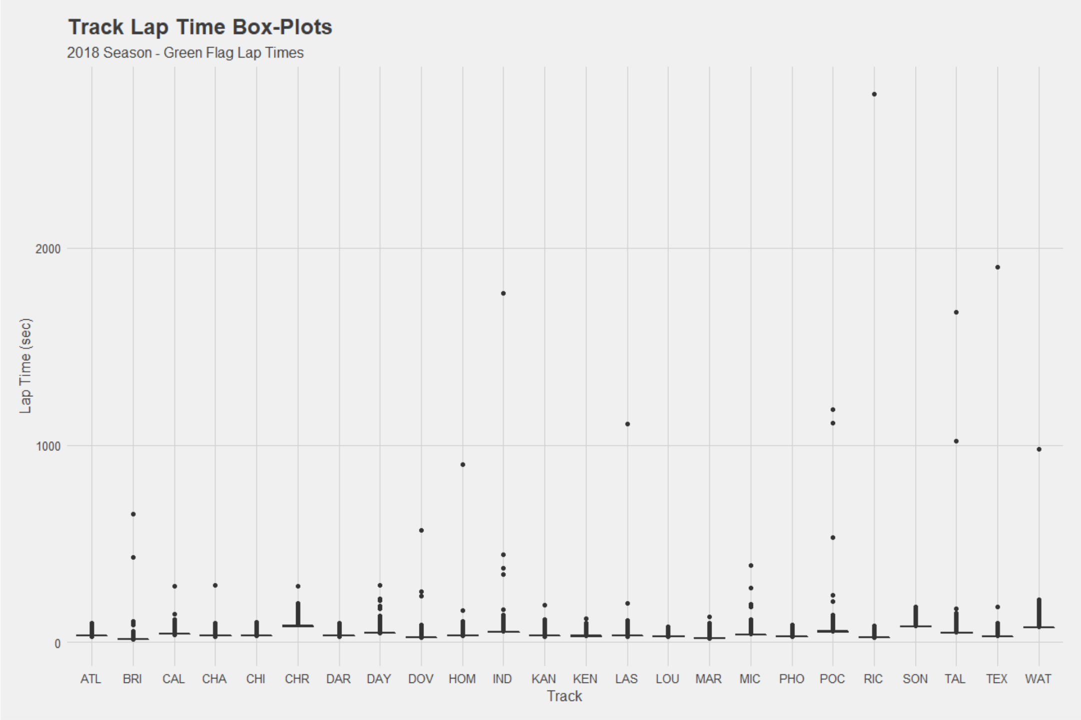

Figure 16: 2018 Season Lap Time Box Plots

Figure 16 shows the box plots for green-flag lap times between tracks. There are significant outliers for lap times that cause the distributions to flatten. This is caused by the measurement parameters for this data set, namely, crossing the Start-Finish line. For cars that begin a lap and crash to the point where the car can no longer be driven, then the car may not come back across the Start-Finish line. Depending on the type of analysis done on lap times, these incidents would require unique handling from track to track based on other observational knowledge that is not recorded in the data, such as cars being involved in wrecks.

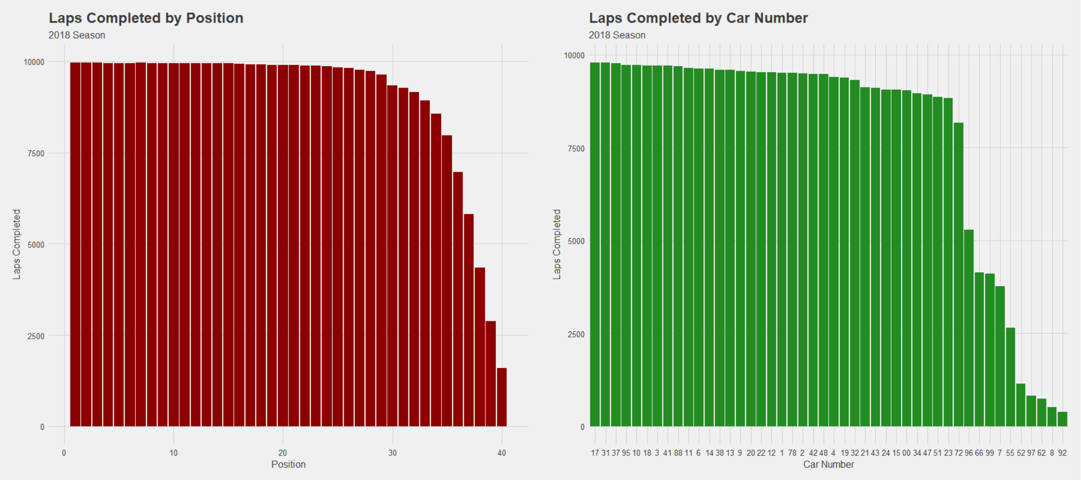

Figure 17: 2018 Season Track Lap Time Box Plots

Total laps completed by position and by car number can be seen in Figure 17. Two notable observations is the fall off of laps completed in the later positions and for certain car numbers. For position based lap counts, this is attributed to cars finishing between positions 30 and 40 are lapped by lead lap cars or do not finish the race due to wrecks or mechanical failures. For car number based lap counts, slight decreases in lap counts are the result of similar reasons. However, the sharp drop seen for some cars is due to part-time participation in races. For this thesis, we focus more on the frequency of lap counts relative to driver, position, and track.

4.2 Processed Data

Pre-processing is the most crucial and often time-consuming step in a data science. Inconsistencies in the data arise for a variety of factors, such as missing values, nonuniform indexes for grouping variables, etcetera. Some of the challenges with NASCAR data in can be dealing with drivers who do not finish races because of mechanical failures or wrecks or identifying discrepancies in lap times between green flag laps and caution laps. In addition to multi-aspect nuances that affect the "cleanliness" of the data due to the nature of the sport, some teams have part-time drivers due to sponsorship deals and other contractual negotiations. For the 2018 season, each race consists of 40 competitors that can enter the race, however, only 35 drivers competed in each of the 36 races in the 2018 season. Similar to tHoops, which considered shot counts in various court sections by players, we investigated counts of laps completed by drivers in each position through the 2018 season.

In Figure 18, the heat map is ordered by positions drivers completed laps in most frequently in for the 2019 regular season. These drivers maintained the top-10 positions with much higher frequency over the course of the season. The driver rating system in NASCAR, similar to a quarter-back rating in the NFL, is heavily weighted by "Average Running Position" [17]. With the presentation of this heat map, we would expect drivers ratings to follow a similar trend aside from points acquired from stage wins and wins.

It is not uncommon for drivers that typically contend in higher positions during races log laps in the back of the field due to varying pit strategies, failing pre-race inspections, or poor qualifying. The heat map in Figure 18 shows how positions with

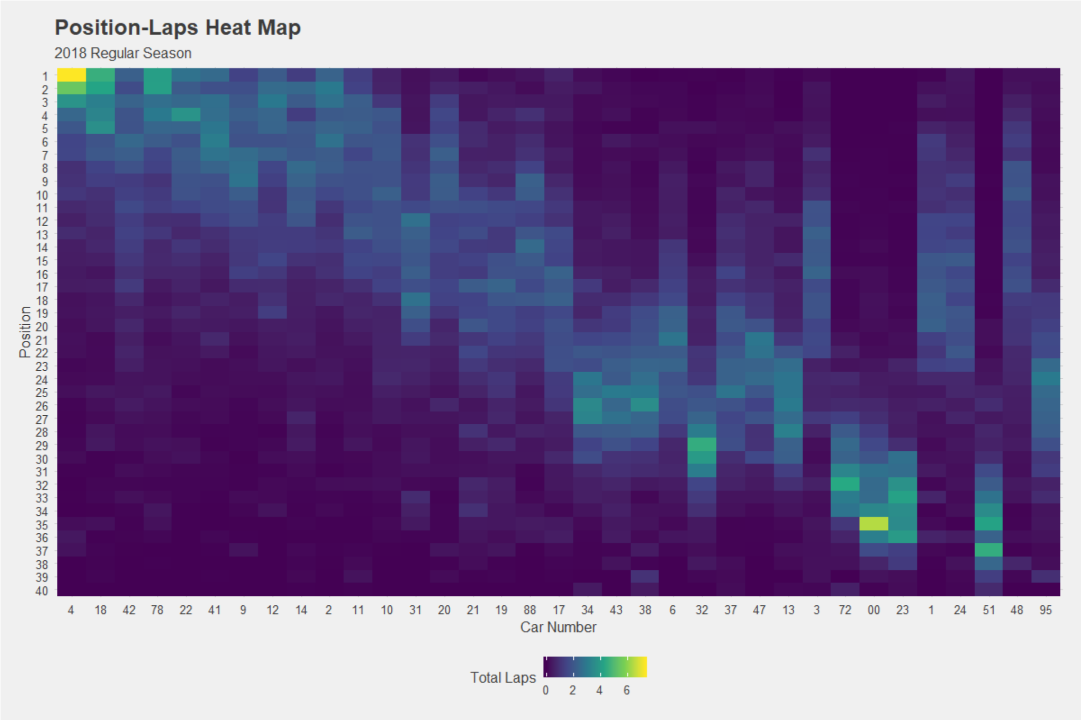

Figure 18: Position Heat Map

higher densities are near the top and bottom positions. In particular, note that the “00” car had 2,116 laps recorded in the 35th position and recorded zero or single digit lap counts in all position greater than 20th. Meanwhile, the “4” car led 1992 laps over the course of the season and recorded single or double digit laps below the 25th position. It is also important to keep in mind that laps recorded in a position is not only dependent on the driver’s ability and car performance, but also the ability of the drivers and cars around them. The frequency of successfully blocking passes or failing to pass cars can cause higher densities for lap counts in positions and conversely for lower densities.

4.3 PCA in NASCAR

A typical approach with PCA is as an investigative tool for variance between features. In this case, the primary interest is the variation in laps completed between positions among drivers in the Monster Energy Cup Series. Similar to MLB pitches and NBA shot selections, variations in laps completed in positions can vary widely depending on a number of factors. Wrecking a car early in a race or leading the majority of the race can lead to seriously asymmetric distributions of position lap counts and both can happen to a highly skilled drivers. The goal of PCA in this case is to relate drivers by reducing the dimensionality of the positions into principal components (PC's).

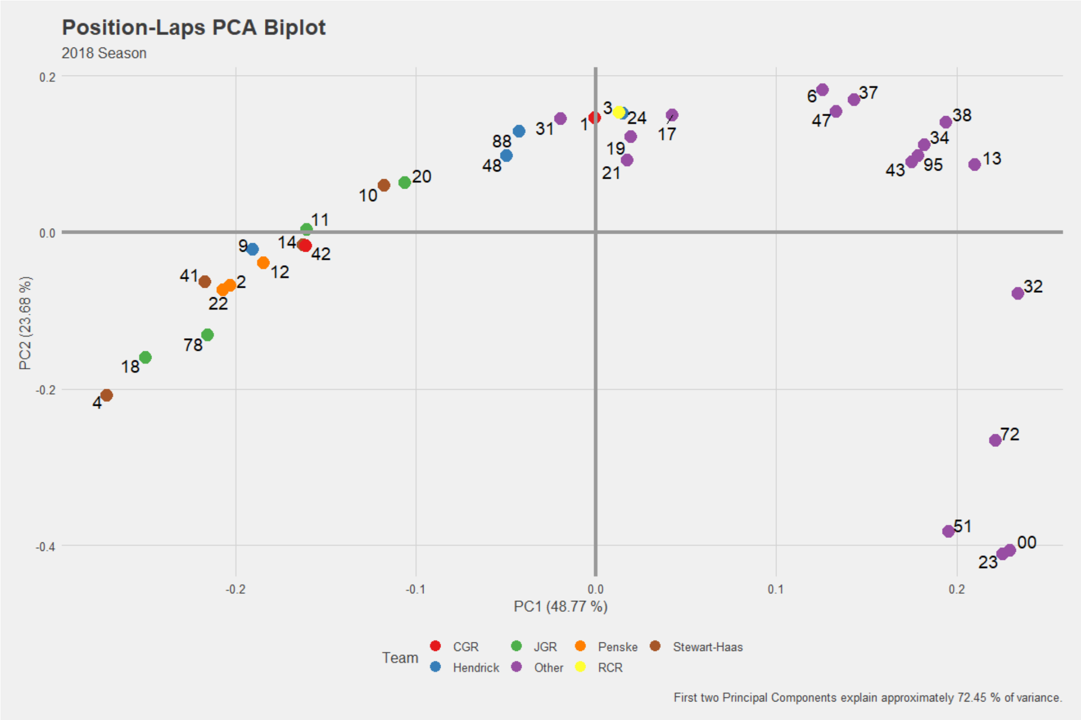

Figure 19: Position-Laps PCA Biplot.

The PCA Biplot in Figure 19 has an interesting spread among the drivers with the more dense cluster of drivers influenced by negative components of PC 1 and more dispersed along the positive components. As PCA is an exploratory unsupervised method, we can begin to ask questions about this cluster of drivers in quadrant 3 that can lead to other analyses. For a researcher with domain knowledge and watches races routinely, two things are immediately obvious about this cluster of drivers. Each of them, with the exception of the 41, 42, and 9 cars had multiple race wins. Also, this cluster corresponds to cars with higher densities in higher positions in the heat map. We can further investigate the principal components individually to identify patterns in our data.

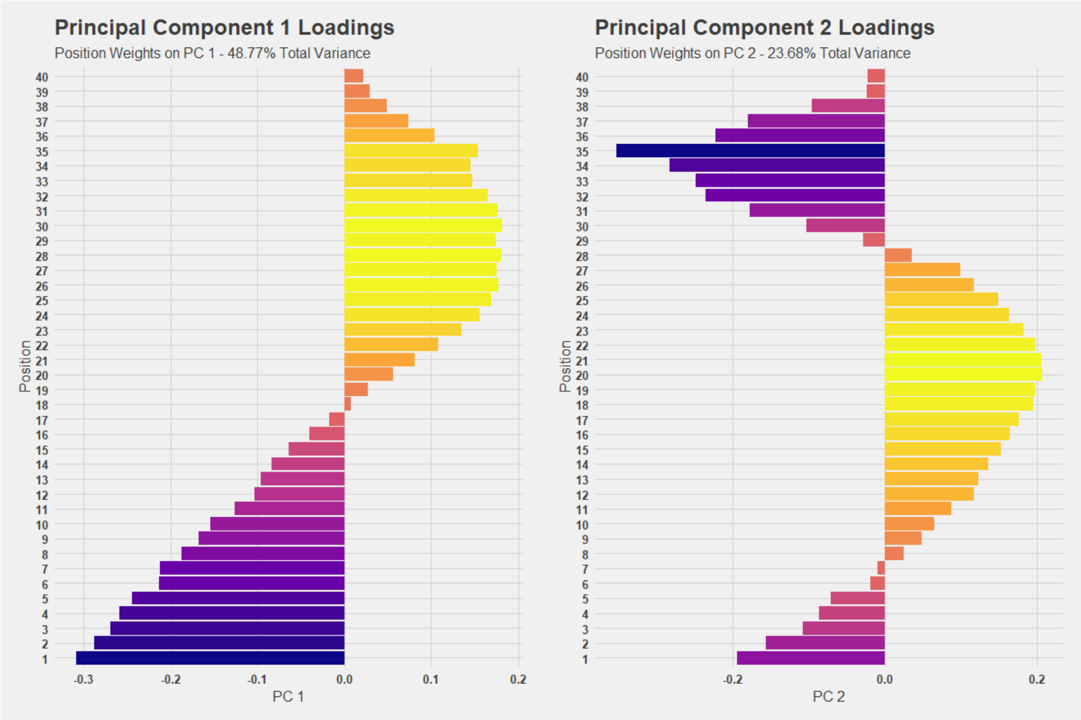

Figure 20: Principal Component Loadings.

Figure 20 is a visual representation of the the element values for the first two columns of **U** resulting from the SVD. These orthogonal vectors are linear combinations of the original features (Positions) projected into the column space. Principal Component 1 is relatively split into two major feature weights for positions greater than 18 and with negative values and positions less than 18 with positive values. In a general sense, we can conclude that approximately 48 percent of the variation in laps completed by position is explained by the frequency of recording laps above position 18 or not. Principal Component 2 has a slightly different shape with negative values for the top 7 and bottom 10 positions, but positive values for positions between 7 and 30. This corresponds to the position densities previously noted in the Figure 18 heat map observed where cars that run in positions above 10 and below 30 more frequently tend to have less variation. Meanwhile, positions in the the middle of the pack tend to change position more frequently.

4.4 NMF in NASCAR

While applying PCA gave us insights regarding the variation between position lap counts, we are further interested in how drivers are rated in regard to their lap counts and if there are subclasses of drivers. We have already seen an interesting grouping of drivers by generating the PCA Biplot in Figure 19. Since we are dealing with strictly non-negative counts of laps, NMF is appropriate method in extracting other latent factors from the data. We will take the same approach as in the basketball case study but with “Drivers” and “Positions” instead of “Players” and “Zones”.

In Figure 22, we can see three distinct groupings for each of the factors by position.

$$\begin{array}{ccccccc} & \mathbf{A} & \approx & & \mathbf{B} & & \mathbf{C} \\ & \text{Positions} & & & \text{Factors} & & \text{Positions} \\ \text{Drivers} & \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} & \approx & \text{Drivers} & \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mr} \end{bmatrix} & \text{Factors} & \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{r1} & c_{r2} & \cdots & c_{rn} \end{bmatrix} \end{array}$$

Figure 21: NASCAR NMF Latent Factors

By the weights associated to these positions, we can associate factor one to a tendency to complete laps in the top 10. Factor two shows a high tendency in positions 20 through 25, while factor three highlights positions 30 through 35. In contrast to PCA, where we observed a distinction between top 20 and bottom 20, NMF offers a slightly more granular separation with an additional grouping.

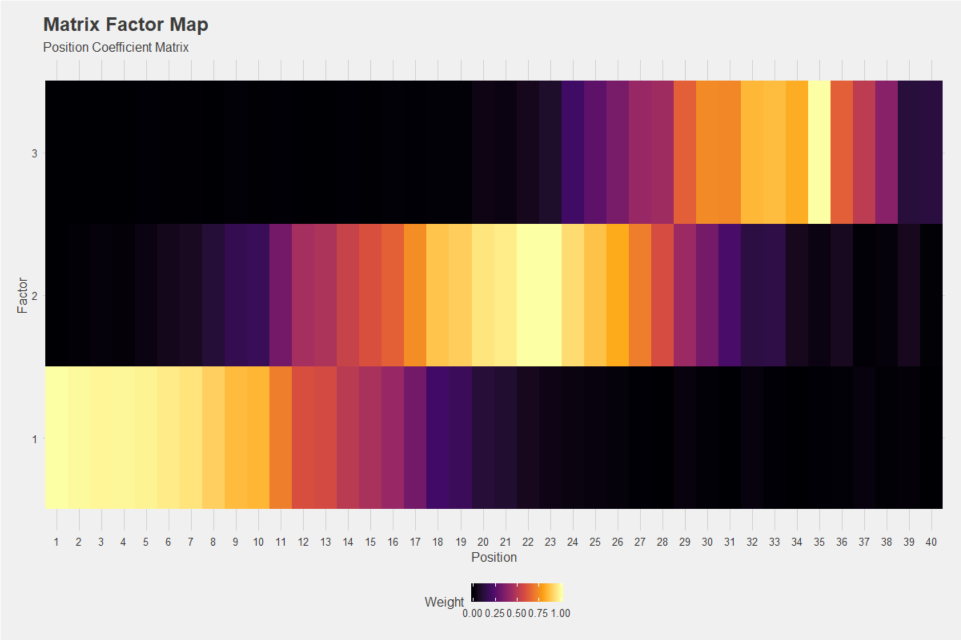

Figure 22: NMF Position Map

Figure 23 depicts the driver coefficients from the factorization where we also see clear groupings of drivers. The drivers with the highest weights for factor one correspond to the the factor groupings in Figure 22. Attention is immediately drawn to the highly weighted drivers, as we are now able to cluster drivers' position tendencies. The most notable feature of this mapping is the cluster of drivers for factor 1, in that each of the top 14 drivers in this cluster made the playoffs. The outliers that just made it into the playoffs were the 88 and 3 cars. The moderate weights for these cars indicates that these driver primarily run in the middle of the pack but also spends a considerable time in the top 10 as well.

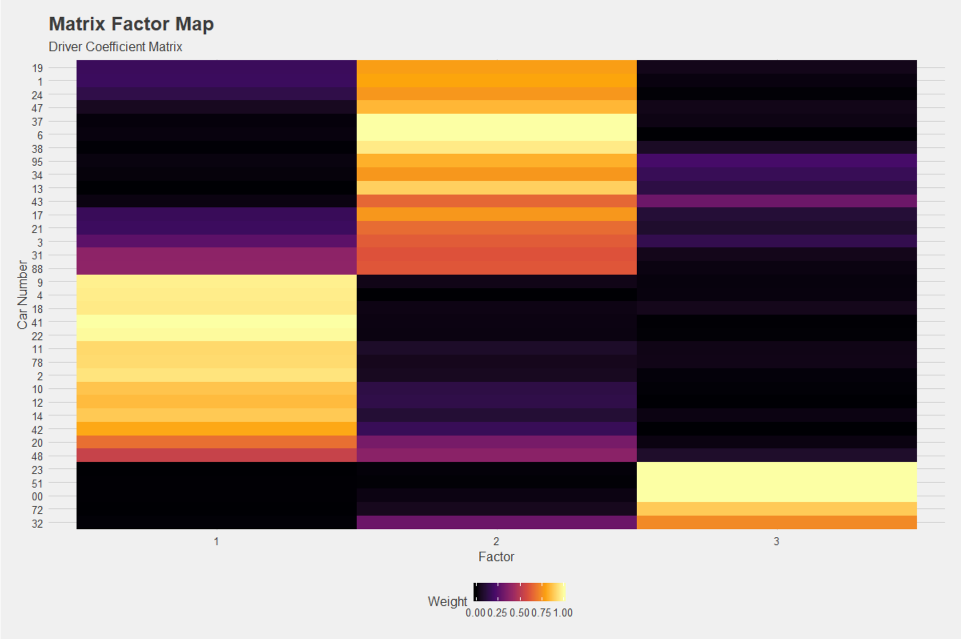

Figure 23: NMF Driver Map

In comparing Tables 8 and 9, we see familiar names and car numbers in the Factor 1 grouping as we did in the primary PCA cluster. However, with the NMF approach,

Car #	Value	Competitor Name
41	1.000	Kurt Busch
22	0.990	Joey Logano
9	0.967	Chase Elliott
4	0.960	Kevin Harvick
18	0.954	Kyle Busch
2	0.940	Brad Keselowski
78	0.919	Martin Truex Jr.
11	0.915	Denny Hamlin
14	0.880	Clint Bowyer
10	0.869	Aric Almirola
12	0.849	Ryan Blaney
42	0.807	Kyle Larson
20	0.666	Erik Jones
48	0.536	Jimmie Johnson
88	0.388	Alex Bowman

Table 8: Factor 1 Driver Weights

Car #	Value	Competitor Name
37	1.000	Chris Buescher
6	1.000	Matt Kenseth
38	0.954	David Ragan
13	0.895	Ty Dillon
47	0.839	AJ Allmendinger
95	0.825	Regan Smith
1	0.801	Jamie McMurray
19	0.782	Daniel Suarez
24	0.765	William Byron
17	0.764	Ricky Stenhouse Jr.
34	0.764	Michael McDowell
21	0.659	Paul Menard
43	0.648	Bubba Wallace
3	0.623	Austin Dillon
88	0.611	Alex Bowman

Table 9: Factor 2 Driver Weights

there is more distinction made between groups of drivers. There is evidence that the 4 and 18 cars were the dominant cars in 2018 according to the raw lap counts depicted in Figure 18, but get edged out by three other drivers according to factor one in Table 8. Each of the drivers in factor one were playoff contenders by the end of the season with the exception of the 88 car Alex Bowman. Surprisingly, the 3 car managed to sneak into the playoffs despite being ranked so low in factor two, the "Middle-of-the-Pack" group.

Factor two consists of drivers that could have made a late run in the season for a playoff bid. In fact, in the current 2019 season at the time of writing the 24 and 88 cars made the playoffs with relative ease. The 19 car driven by Daniel Suarez had a strong year and switched teams from Joe Gibbs Racing to Stewart-Hass Racing and just missed the playoffs. These is evidence suggesting there's more to being weighted heavily in a group than laps counts in position alone that describe the performance of drivers. For this reason, it would be advantageous to investigate the tensor structure type of this data with track as an additional axis.

4.5 Tensor Decomposition in NASCAR: Part 1

For this application, we extend the application of PCA to the heat map in Figure 18 to a higher order tensor structure. Where we previously investigated lap counts by position for each driver in the form of a heat map or incidence matrix, we will add the additional feature of “Track” as a third axis in the form of an order three tensor. Figure 24 shows an example structure for this tensor indicating the axes where the indices will consist of the number of laps recorded by driver, position, and track in the 2018 season. First, we will introduce a proof of concept and formally define the algorithm for Function Space Tensor Decomposition in the general case of an order three tensor. Note that this methodology can be extended for real or complex higher order tensors. Then we will demonstrate the application for this NASCAR data with the goal of identifying latent patterns with respect to driver, position, and track and discuss the patterns in context.

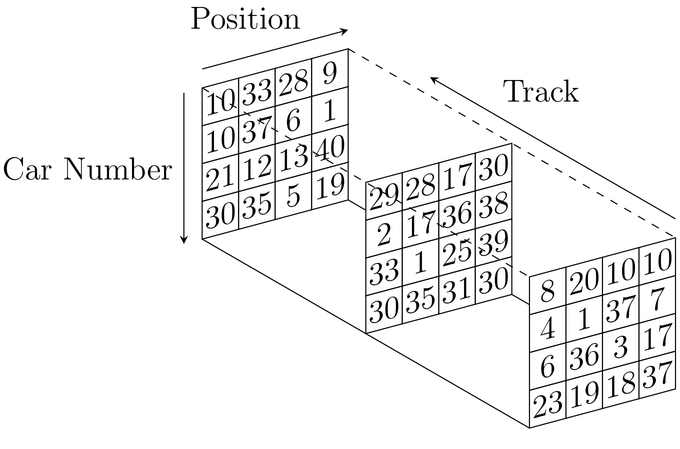

Figure 24: Position, Car, and Track Tensor Example

Consider an order-3 tensor $\mathcal{X} \in \mathbb{R}^{M \times N \times R}$. Let, $\mathbb{X}$ be the tensor such that,

$$\mathbb{X} = [\mathbf{X}_{m::}]_{m=1}^{M} = [x_{m,n,r}]_{m,n,r=1}^{M,N,R} \tag{35}$$

Given the tensor $\mathbb{X}$, we can define an action on the vector space $\mathbb{R}^M$ by,

$$\mathbb{X}\mathbf{v} = \sum_{k=1}^{M} \mathbf{X}_{k::} v_k \tag{36}$$

Then $\mathbb{X}$ is a linear transformation and a mapping $\mathbb{X} : \mathbb{R}^M \to \mathbb{R}^{N \times R}$. In addition, $\mathbb{X}$ has an inner product, and norm defined by,

$$\langle \mathbf{X}_{\mathbf{k}::}, \mathbf{X}_{\ell::} \rangle = \sum_{ij} \mathbf{X}_{kij} \mathbf{X}_{\ell ij} \quad \text{and} \quad ||\mathbf{X}_{k::}|| = (\langle \mathbf{X}_{\mathbf{k}::}, \mathbf{X}_{\ell::} \rangle)^{\frac{1}{2}} \tag{37}$$

Therefore, $\mathbb{X}$ is a normed inner product space, that generalizes the concept of a "column space" of a matrix. As a result, $\mathbb{X}$ has an adjoint given by mapping, $\mathbb{X}^T\mathbb{X} : \mathbb{R}^M \to \mathbb{R}^M$, which is a matrix of inner products of the original tensor slices:

$$\mathbb{X}^T\mathbb{X} = [\langle \mathbf{X}_{\mathbf{k}::}, \mathbf{X}_{\ell::} \rangle]_{k,\ell=1}^{M} \tag{38}$$

Consequently, $\mathbb{X}^T\mathbb{X}$ is a symmetric positive semi-definite matrix and SVD yields $\mathbb{X}^T\mathbb{X} = \mathbf{V\Sigma^2V^T}$. By Theorem 2.4, the eigen-decomposition of the symmetric matrix $\mathbb{X}^T\mathbb{X}$ indicates that the columns of $\mathbf{V}$ form an orthonormal basis for $\mathbb{X}^T\mathbb{X}$ where the singular values are ordered $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_M \geq 0$. Then,

$$\mathbb{X}^T\mathbb{X} = \mathbf{V\Sigma^2V^T} \implies \mathbf{U} = [\mathbf{u_1}|\mathbf{u_2}|...|\mathbf{u_M}] \quad \text{where} \quad \mathbf{u}_k = \frac{1}{\sigma_k}\mathbb{X}\mathbf{v}_k \tag{39}$$

Lastly, since the columns of $\mathbf{V}$ form and orthonormal basis for $\mathbb{X}^T\mathbb{X}$, the principal components are obtained by,

$$\mathbf{P} = [\mathbf{p_1}|\mathbf{p_2}|...|\mathbf{p_M}] \quad \text{where} \quad \mathbf{p}_k = \mathbb{X}\mathbf{v}_j \tag{40}$$

Consequently, the “principal components” are tensors that are one order lower than $\mathcal{X}$. In this case, since our tensor $\mathcal{X}$ is a 3^{rd}-Order tensor, the princiapl components are 2^{nd}-Order tensors, i.e matrices. Note that by defining the action of the tensor in equation (36) as a linear transformation on the vector space of the first index, namely $\mathbb{R}^M$ in the formal definition, we are free to use each of the indices to define this action. As a result, we obtain orthogonal projections of the corresponding adjoint matrix of inner products relative to the respective action defined. This allows us to view the principal components in each subspace. This can be thought of as pivot for constructing the principal components by initializing the feature space index of the tensor in equation (36).

In Figure 24, our tensor $\mathcal{X}$ is defined by the axes “Car Number”, “Position”, and “Track”. There are 35 distinct drivers, 24 individual tracks, and 40 positions corresponding to $\mathcal{X} \in \mathbb{R}^{35\times 24\times 40}$. So each element of the tensor $x_{m,n,r}$ corresponds to the number of laps that Driver m completed at Track n in Position r. Then $\mathbf{X}_{m::}$ is relational heat map of laps counts by position and track for driver m, $\mathbf{X}_{:n:}$ is the relational heat map of lap counts by driver and track, and $\mathbf{X}_{::r}$ is the relational heat map of lap counts by driver and track for each position. We begin the analysis by investigating the first feature space in the tensor structure for the NASCAR application as our pivot, “Car Number”.

We define the action on the “Car Number” feature space, $\mathbb{R}^{35}$ by,

$$\mathbb{X}\mathbf{v} = \sum_{k=1}^{35} \mathbf{X}_{k::} v_k \tag{41}$$

Applying the computations in the preceding algorithm yields principal components in the subspace for two variables, in particular, the adjacent indices, “Position” and

"Track". Therefore, rather than principal components being vectors in the matrix PCA case, our principal components are 2-dimensional arrays. We retain orthogonality in the function space defined by the mapping $\mathbb{X}^T\mathbb{X}$ from SVD. The resulting singular values from SVD applied to $\mathbb{X}^T\mathbb{X}$, indicate the amount of variance explained for the lap counts in the tensor relative to our pivot feature space, "Car Number"(Or rather "as projected onto the principal axes in the car number feature space"). Each singular value in decreasing order corresponds to a principal component.

$$\begin{gathered} s_i = \frac{\sigma_i^2}{||\Sigma^2||_2} \text{ where} \sum_{i=1}^{35} s_i = 1 \\ s_1 = 0.5586, s_2 = 0.2001, s_3 = 0.0795, ..., s_{35} = 4.72 \times 10^{-7} \end{gathered} \tag{42}$$

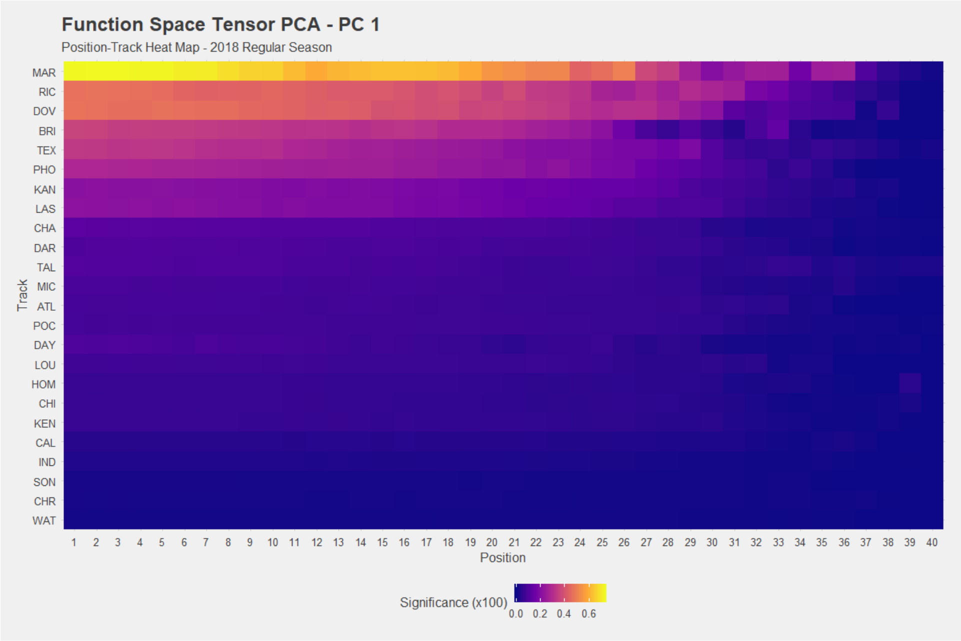

Figure 25: Function Space Tensor PC 1

The first principal component for the decomposition of the tensor can be seen in Figure 25. This component explains approximately 55.88 percent of the variation in lap counts in the feature space. These principal components can be thought of a profiles ranked by their singular values. Note that in this heat map, there are variations in significance between tracks and positions. The highest weighted tracks in order are Martinsville (MAR), Richmond (RIC), Dover (DOV) and Bristol (BRI). These tracks have a particular commonality between them, with the exception of Dover, in that MAR, RIC, and BRI are all short tracks that are approximately 0.5 mile in length that run 400 or 500 lap races. Another interesting observation is that Sonoma (SON), Charlotte Roval (CHR), and Watkins Glenn (WAT) are the least weighted tracks overall and are all road courses. Road courses tend to be longer in track length and run between 90 and 110 laps in total at these tracks.

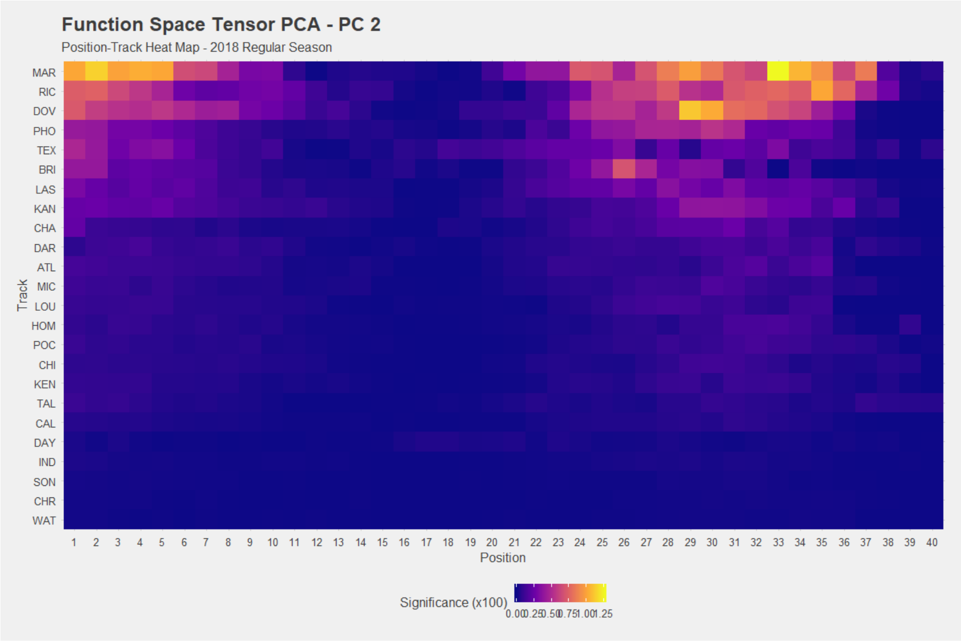

Figure 26: Function Space Tensor PC 2

In Figure 26, the second principal component explains approximately 20 percent of the variation in lap counts in the feature space and shows separation largely between the top ten positions and the positions between 25 and 37. We also saw in the NMF analysis a similar separation of position tendencies in Figure 22. As previously mentioned, tracks have many characteristics that teams account for that affect decisions that influence position at any given time during a race. Specifically, at MAR, RIC, or BRI, lap times are can range between 15 and 25 seconds. If a car is forced to make a green flag pit stop, then that driver will inevitably be multiple laps down by the end of the pit stop. Conversely, at road courses, lap times can range between 75 and 85 seconds, for which a green flag pit stop can be completed without losing a lap behind the leader.

In summary, the ordering of the weights by track really only correspond to raw total laps counts, but the fall off across positions is directly explained by the fact that cars in the back of the pack tend to get lapped and subsequently, tend to record less laps than those who stay on the lead lap. However, in each of the subsequent principal components, the same six tracks are dominating the significance because of the variation in lap counts between tracks. Since raw lap counts are dominating the variation, the low lap counts at the road courses WAT, CHR, and SON get practically washed out as background noise.

The reason this is important is because in contrast to the *tHoops* framework, which used spatial data points relative to players and time, races do not have consistent time intervals across every track. This would be analogous to how *tHoops* binned time into quarters, but games having varying quarter lengths which would significantly affect the number of shots taken. This is what makes NASCAR such a unique sport to do analysis.

Although these results are skewed by the high lap totals for short tracks, they confirm that this approach is indeed capturing the variation. Laps themselves are not uniform across each track. Tracks have different shapes, lengths, surface types, and banking degrees in the turns. For example, leading 50 laps at Bristol Motor Speedway is approximately 25 miles and would reflect 10 percent of the entire race. Meanwhile, 50 laps at Daytona International Speedway is approximately 125 miles and 25 percent of the race. Each track offers different characteristics that affect car performance and situational decision making by drivers and crew chiefs that can wildly influence position at any given time. In order to get deeper insights, we need to address the scale of lap units for each track.

4.6 Tensor Decomposition in NASCAR: Part 2

Feature scaling can be rather troublesome in higher order data structures. In the case for our tensor, we would need to address which index to perform the scaling respectively. Note that we have already addressed variation in races competed by drivers by filtering out drivers that competed in every race. Positions have a natural maximum of 40 available participants due to qualifying restrictions. As we have seen, it isn't the number of tracks that is the issue but the number of laps between them. Note that depending on how axes are defined for general tensors will dictate a scaling methodology which will be covered in the further work discussion. In this case, we simply re-scaled the number of laps as a percentage of the total laps run at each track in the season. In doing so, we get more informative principal components for the underlying tensor network structure as "profiling components".

The sequence of the approach is still the same, and we will first consider the pivot mapping from the "Car Number" feature space and define the action on $\mathbb{R}^{35}$ by,

$$\mathbb{X}\mathbf{v} = \sum_{k=1}^{35} \mathbf{X}_{k::} v_k \tag{43}$$

After applying SVD to the adjoint $\mathbb{X}^T\mathbb{X}$, we obtain the following variance measures from the singular values for each of the principal components:

$$s_1 = 0.5912, s_2 = 0.1638, s_3 = 0.0707, ..., s_{35} = 0.0009$$

Note that the first 3 principal components account for approximately 82 percent of the variation with decreasing amounts accounted by each of the subsequent singular values.

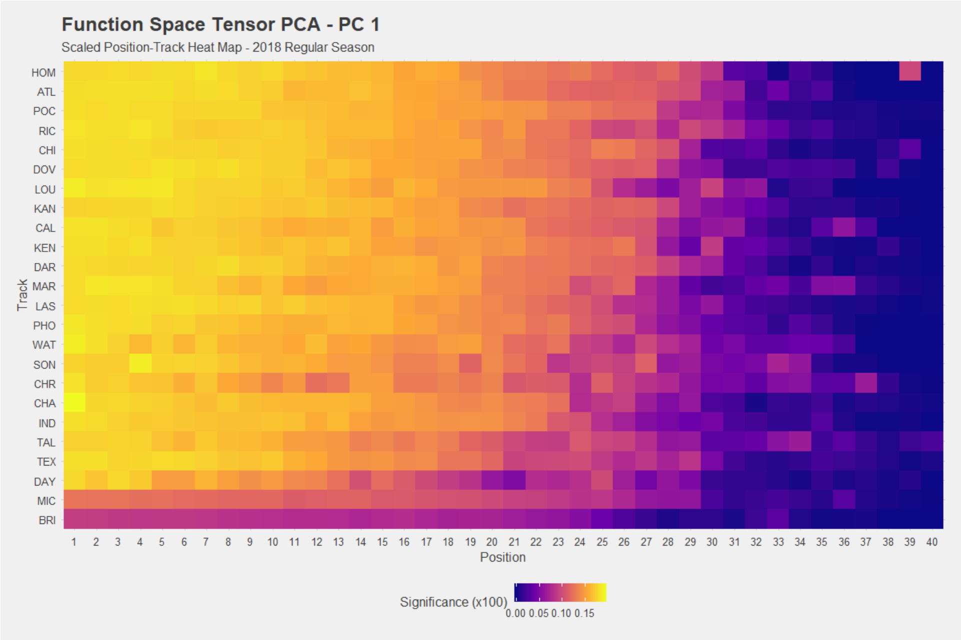

Figure 27: Track-Position Profile 1

The first principal component is depicted in Figure 27 where we can see almost a contrast heat map than before, but with the same general qualities. Again, the tracks are ordered by cumulative significance and we see the same fall off trend for positions for each track. In general, we tend to see higher percentages of laps recorded at higher positions, but in this case BRI has much lower overall significance. Bristol is a very unique track in relation to every other track in that given that it is a half-mile track, it is the only track that has pit stalls on both the front stretch and the back stretch. Depending on the location of a driver's pit stall relative to the Start-Finish line, this could significantly impact they enter and exit pit lane. Because these position records are taken as a car crosses the Start-Finish line, the magnitude of positions lost or gained can be heavily impacted for any given pit lane event.

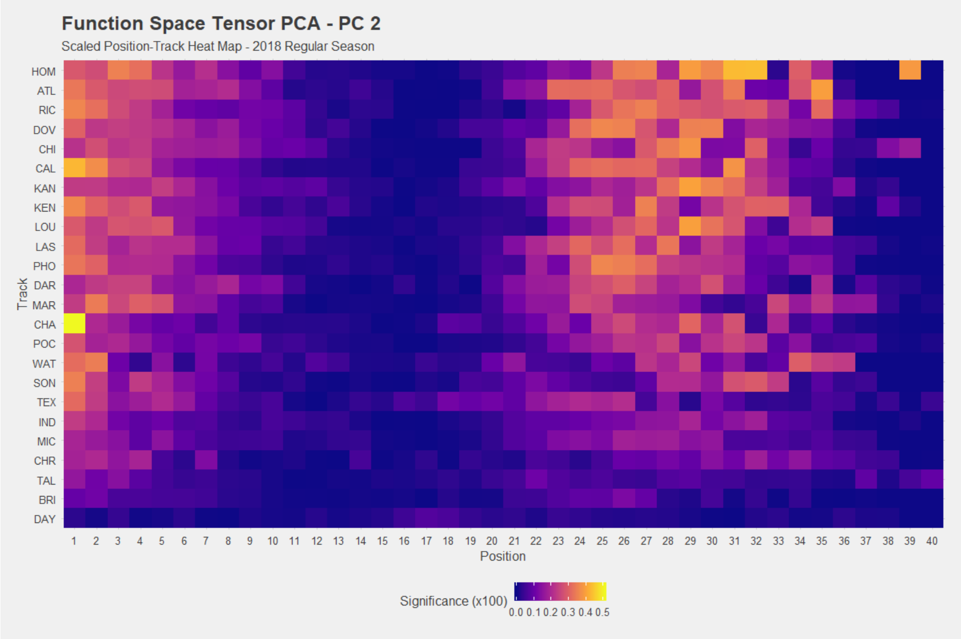

Figure 28: Track-Position Profile 2

Principal component two accounts for approximately 16 percent of the variation and we see the same trend in the separation between the front runners and back of the pack. It is important to note the consistent insignificance we see in last few positions that can be easily explained by drivers that wreck early in a race and are unable to finish will consequently have lap completion rates close to zero. There are two observations that stick out in this plot regarding Charlotte (CHA) and Daytona (DAY). The highly significant cell in CHA position 1 is of particular interest in that Kyle Busch (18) led 377 of the 600 laps. Also, Daytona is well-known for "The Big One" as it pertains to crashes that occur at almost every race there. At the 2.5 mile super-speedway, speeds exceed 200mph regularly as drivers form single file lines around the track. At these speeds, and the proximity of the cars, a miscalculation maneuvering through traffic can, and does, cause significant pile ups where higher numbers of cars do not finish the race. Bristol also shares this characteristic due to the limited lateral track space for cars to fight for position, but at much lower speeds. In summary, this component is capturing the positions where highest percentage of laps are recorded for each track.

Next, we consider a different pivoting strategy to obtain principal components in a different feature space. This time we focus on the "Track" pivot to develop the notion of "Driver-Position" profiles. This time, we define the action on $\mathbb{R}^{24}$ by,

$$\mathbb{X}\mathbf{v} = \sum_{k=1}^{24} \mathbf{X}_{:k:} v_k \tag{44}$$

After applying SVD to the adjoint $\mathbb{X}^T\mathbb{X}$, we obtain the following variance measures from the singular values for each of the principal components:

$$s_1 = 0.9149, s_2 = 0.0015, s_3 = 0.0011, ..., s_{24} = 2.77 \times 10^{-4}$$

Here we obtain approximately 93 percent of the variation by the first two principal components, and more notably, approximately 91 percent in principal component one alone.

Through the analysis of the the NASCAR data so far with the results from PCA and NMF revealing the separation of drivers relative to positions, we have developed a sense of how drivers typically perform. The high amount of variation explained by a 2-dimensional array principal component in Figure 29 shows a similar heat map to the original heat map in Figure 18. However, when we account for the additional feature "Tracks" and utilizing the tensor structure, the first principal component "Car-Position Profile" has a much clearer picture of where drivers tend to run, at

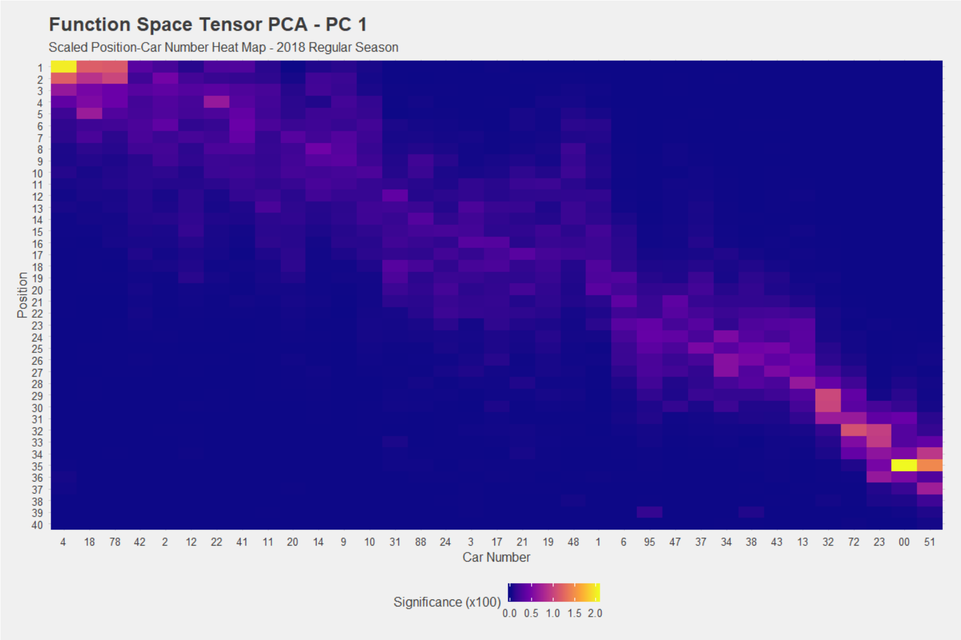

Figure 29: Driver-Position Profile 1

least in 2018. In the context of the 2018 playoffs, the 4, 18, and 78 car were each competing for the championship in Homestead as part of the final four drivers in the NASCAR playoff structure. The fourth driver edged their way into the championship round ahead of three drivers that ranked higher in Figure 29 by winning a crucial race in the final lap prior to the championship. The 22 car made the final playoff round and ended up winning the race at Homestead for the 2018 championship. As is a common theme in sports analytics, capitalizing on analytical trends can get a team to the playoffs, but once there, a single "black swan" event can change the likeliest of outcomes.

Lastly, we will consider the pivot strategy to obtain principal components in the "Position" feature space. This will allow us to develop the notion of "Driver-Track

Profiles". This time, we define the action on $\mathbb{R}^{40}$ by,

$$\mathbb{X}\mathbf{v} = \sum_{k=1}^{40} \mathbf{X}_{::k} v_k \tag{45}$$

After applying SVD to the adjoint $\mathbb{X}^T\mathbb{X}$, we obtain the following variance measures from the singular values for each of the principal components:

$$s_1 = 0.5193, s_2 = 0.2081, s_3 = 0.0957, ..., s_{40} = 4.64 \times 10^{-5}$$

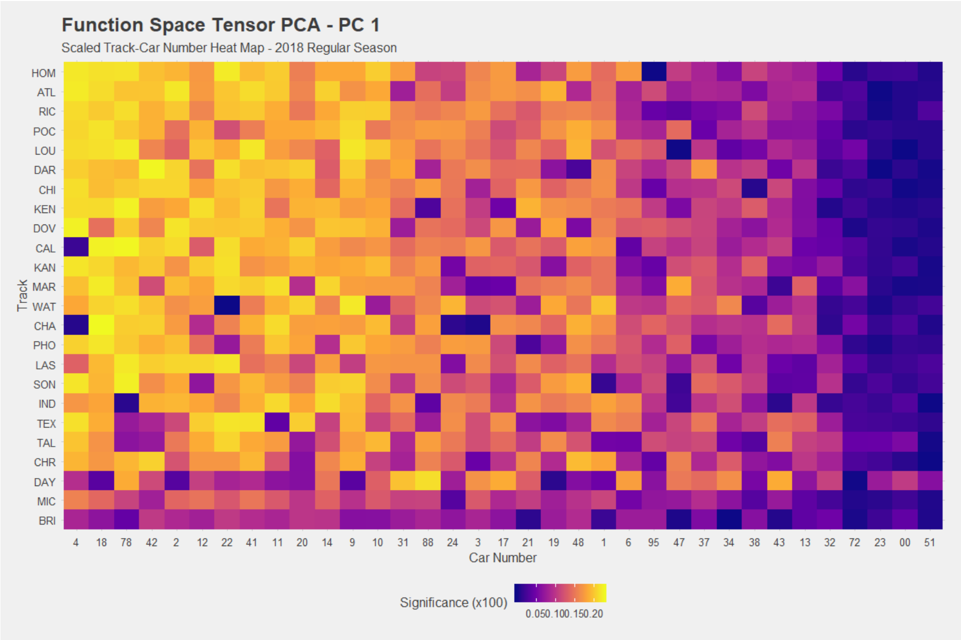

Figure 30: Driver-Track Profile 1

The first principal component in this feature space in Figure 30 accounts for approximately 52 percent of the variation and depicts trends in how well drivers run at each track in 2018. We are able to pick out some hot spots, such as the 22 car at Homestead (HOM), the 9 car at Watkins Glenn (WAT), and to 18 car at

Charlotte (CHA) to name a few. However, this component requires some substantial contextual knowledge about the drivers and tracks to interpret. Keep in mind that the value for each tensor element is the percent of total laps held by each driver, in each position at each track. It is noteworthy to pay attention to the low significant cells for drivers, especially the drivers that we are used to seeing at the top positions. Some key events took place to cause these cold zones such as the 4 car finishing 24^{th} at California (CAL) and 40^{th} at Charlotte (CHA) due to a crash early in the race. The 22 car crashed on lap 1 at Watkins Glenn (WAT) and finished last. The 78 car struggled at Indianapolis finishing 40^{th} due to a mechanical failure on lap 41 and crashes caused short outings for the 18, 9, and 2 cars at Daytona (DAY).

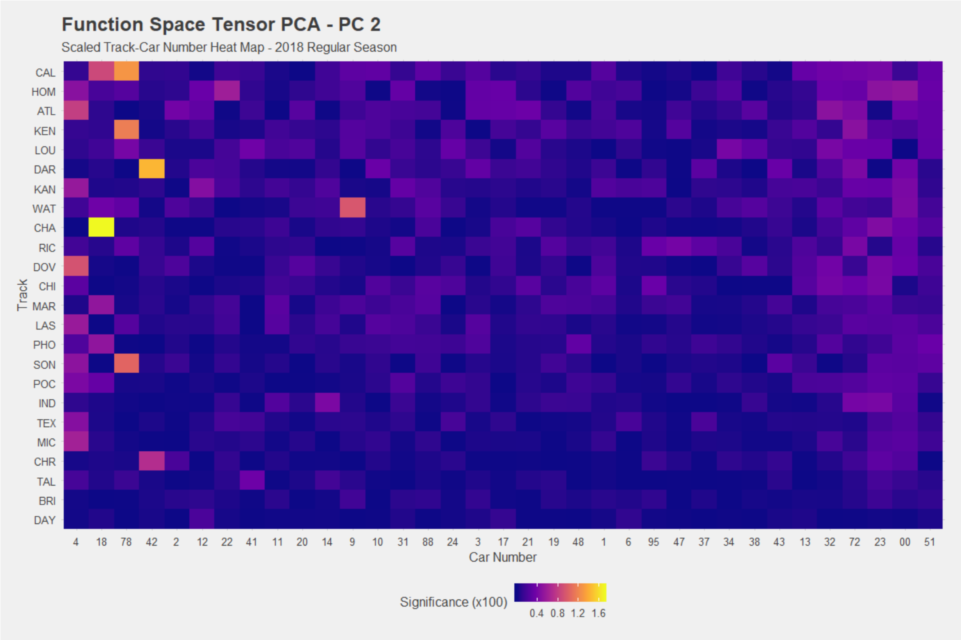

Figure 31: Driver-Track Profile 2

Principal component two accounts for approximately 21 percent of the variation and has similar hot spots for drivers and tracks that were noted in principal component one with much less noise. In fact, a closer look at these drivers for races at these tracks reveals exactly where drivers led the most laps. In some cases, such as the 18 car at Charlotte (CHA), the 9 car at Watkins Glenn (WAT), and the 78 car at Pocono (POC) and Kentucky (KEN), these hot spots correlate to wins. Leading the most laps is relatively correlated with winning the race, however, in California, the 18 and 78 cars both led for the majority of the race until the 78 car ran away with the lead in the last 32 laps winning by a margin of approximately 11 seconds. The 42 car led 242 laps at Darlington (DAR) and gave up the lead to the 2 car with 22 laps to go in the race. In summary, this second component profiles were drivers dominated leading races.

There is no doubt that just as with PCA or NMF, interpreting principal components and latent factors generated by these methods requires necessary contextual knowledge to pick up on patterns with utility. For example, utilizing the "Driver-Track Profiles" can guide research into other aspects of the race to target certain drivers and investigate their trends in other relational data sets that can lead to other competitive insights. The indices chosen for this tensor decomposition method can be extended into other feature analyses such as investigating lap times over the course of a race as non-uniform time series. The underlying network structure for tensors allows for more flexibility than the CP and Tucker Decomposition approaches while maintaining rigid decompositions without sacrificing interpretability.

5 CONCLUSION

Further work in this area is inevitable stemming from a number of fields such as multi-linear algebra, network dynamics, and functional analysis. Tensor PCA and decomposition techniques are in high demand with the increased frequency of multi-relational data structures utilized in practice. As machine learning and computational methods continue to be developed, tensor structures will remain as the fundamental structure for applications in the field various analytical context including sports analytics. A couple of things to consider moving forward would be to develop the mathematical framework more formally and include general finite order tensors of real and complex valued vector spaces.

The next area to address is the computational aspect of the application. In this thesis, we use general python packages such as *pandas, numpy, matplotlib, scipy*, and *Tensorly.* We also incorporate R for other tasks with data manipulation and plotting packages included in the *tidyverse* as well as the *rTensor* package for converting data frames into tensor objects. There are many other packages, in other languages as well, designed to carry out tensor products, decompositons, and manipulations such as vectorization and matricization covered in Chapter 2. Additional packages may be available to better utilize the network structure to further develop the function space methodology more efficiently to complement existing packages or built into new ones.

BIBLIOGRAPHY

[1] Hanchett, D. (2012). *Playing Hardball With Big Data: How Analytics Is Changing The World of Sports.* EMC, pp. 2. Retrieved from https://www.behance.net/gallery/25846183/Playing-Hardball-with-Big-Data.

[2] Mondello, M., Kamke, C. (2014). *The Introduction and Application of Sports Analytics in Professional Sport Organizations.* Journal of Applied Sport Management, 6(2) Retrieved from https://search.proquest.com/docview/1730027881?accountid=10771

[3] Rees, L., Rakes, T., Deane, J. (2014) *Using Analytics To Challenge Conventional Baseball Wisdom.* Journal of Service Science, 8(1) Retrieved from https://clutejournals.com/index.php/JSS/article/view/9493

[4] Gillis, N. (2014). *The why and how of nonnegative matrix factorization.* Regularization, Optimization, Kernels, and Support Vector Machines, 12(257), 257-291.

[5] Miller, A., Bornn, L., Adams, R., Goldsberry, K. (2014) *Factorized Point Process Intensities: A Spatial Analysis of Professional Basketball.* ArXiv, abs/1401.0942.

[6] Rabanser, S., Shchur, O., Gnnemann, S. (2017). *Introduction to Tensor Decompositions and their Applications in Machine Learning.* ArXiv, abs/1711.10781.

[7] (2019) *TensorFlow* Retrieved from https://www.tensorflow.org/

[8] Papalexakis, E.E., Pelechrinis, K. (2017). *tHoops: A Multi-Aspect Analytical Framework Spatio-Temporal Basketball Data Using Tensor Decomposition.* ArXiv, abs/1712.01199.

[9] Strang, G. (1993). *The Fundamental Theorem of Linear Algebra.* The American Mathematical Monthly, 100(9), 848-855. doi:10.2307/2324660

[10] Strang, G. (2019). *Linear Algebra and Learning From Data* Wellesley, MA: Wellesley-Cambridge Press

[11] M. Stone P. Goldbart, (2004) *Mathematics for Physics* (Cambridge University Press, Cambridge 2004), ISBN: 9780521854030

[12] Hunter, J.K. Nachtergaele, B. (2001) *Applied Analysis.* (Singapore: World Scientific)

[13] Gray, V. (2017). *Principal component analysis: methods, applications and technology* . New York: Nova Science Publishers, Inc.

[14] Zare, Ali, et al. (2018) "Extension of PCA to higher order data structures: An introduction to tensors, tensor decompositions, and tensor PCA." ArXiv, abs/1803.00704.

[15] Pelecky, D.L (2018). *Data Sharing: Fact and Fiction.* Retrieved from https://medium.com/building-speed/data-sharing-fact-and-fiction-f71902be636c

[16] Cheng, A. (2017). *Using Machine Learning to Find the 8 Types of Players in the NBA.* Retrieved from https://fastbreakdata.com/classifying-the-modern-nba-player-with-machine-learning-539da03bb824

[17] Beaver, D. (2019) *Exploring NASCARs Driver Rating.* Retrieved from https://sports.yahoo.com/exploring-nascar-driver-rating-180210052.html

[18] (2019) *Sports Media Technology, INC.* Retrieved from https://www.smt.com

[19] (2019) *MLB Statcast* Retrieved from http://m.mlb.com/glossary/statcast

[20] Li, J., Bien J., Wells, M. (2018) *rTensor: An R Package for Multidimensional Array (Tensor) Unfolding, Multiplication, and Decomposition* Journal of Statistical Software, Articles, 87(10). 1-31. doi:10.18637/jss.v087.i10

[21] A. N. Kolmogorov (1958) On linear dimensionality of topological vector spaces, Dokl. Akad. Nauk SSSR, 120:2, 239241

APPENDICES

1 Appendix A - Data Sources

1.1 Case Study: MLB Pitch Analysis

Data retrieved from https://www2.stat.duke.edu/courses/Summer17/sta101.001-2/uploads/project/project.html

1.2 Case Study: NBA Shot Analysis

Data retrieved from https://github.com/kpelechrinis/NBA_Shot_Data [8].

2 Appendix B - Code Implementation

2.1 Python Code

```
# Import necessary packages
%matplotlib inline
from matplotlib import pyplot as plt
from scipy import linalg
import numpy as np
import pandas as pd

# Import data and select relevant features
Timing18df = pd.read_csv('TimingLaps2018Normalized.csv')
tdf = Timing18df[['TrackNameShort','CarNumber','LapStanding','LapPct']]

# Create lists of indices
CarNumbers = tdf.CarNumber.unique()
CarNumbers.sort()
Positions   = tdf.LapStanding.unique()
Positions.sort()
Tracks = tdf.TrackNameShort.unique()

# Create list of group keys
XDf = tdf.groupby(['CarNumber','TrackNameShort','LapStanding'])
Tgroups = list(XDf.groups.keys())

# Assign lengths of indices
m = len( tdf.CarNumber.unique() )
n = len( tdf.TrackNameShort.unique() )
r = len( tdf.LapStanding.unique() )

# Create Multi-Dimensional Array
X = np.zeros( (m,n,r), dtype = float )
for i in range(m):
    print('.', end = '')
    for j in range(n):
        for k in range(r):
            group = (CarNumbers[i], Tracks[j], Positions[k] )
            if( group in Tgroups):
                X[i,j,k] = XDf.get_group(group).iloc[0]['LapPct']
t = X.astype(float)

# Note: Adjust m and k for pivot index
```

```
m = t.shape[0]
THT = np.array([ [ (t[k,:,:]*t[l,:,:]).sum() for k in
   range(m)] for l in range(m)] )

# Apply SVD and obtain singular values
V,Sigma,  V    = linalg.svd( T  T )
np.cumsum(Sigma**2)/sum(Sigma**2)

# Obtain Principal Axis and Principal Component
# Note: Adjust V column index for corresponding Principal
   Axis
U_0 = sum([t[k,:,:]*V[k,0] for k in range(m) ])
P0 = U_0**2 / sum((U_0**2).flatten())

# Plotting Principal Component Heat Map
# Note: Adjust labels for relative pivots and axes
fig, ax = plt.subplots(figsize = (10,10))
plt.imshow(P0, cmap = 'plasma')
plt.colorbar( fraction=0.025, pad=0.04, label = '
   Significance')
plt.xlabel('Position')
plt.ylabel('TrackId');
plt.yticks(ticks = list(range(0,n)),labels = Tracks)
```

VITA

JUSTIN REISING

Education:

B.S. Mathematics, King University,
Bristol, Tennessee 2011

M.S. Mathematical Sciences, East Tennessee State University
Johnson City, Tennessee 2019

Professional Experience:

Graduate Assistant, ETSU,
Johnson City, Tennessee, 2018–2019

Race Tools and Technology Analyst, Joe Gibbs Racing
Huntersville, North Carolina, 2019–present

Publications:

J. Reising,“Function Space Tensor Decomposition
and its Application in Sports Analytics”
Dissertations and Theses,
(2019).