ECEC487: Pattern Recognition
Effectivness of KMeans Clustering to Classify Players in the National Football League
Project Research Paper

Jared Balakrishnan

Drexel University
jared0@drexel.edu

December 10, 2019

Abstract

This paper investigates the effectiveness of clustering models based on the K-Means method in classifying NFL draft candidates according to their potential positions with respect to their physical characteristics given the fact that American Football has been historically a very physical sport. Additionally, performance was also studied through explanatory data analysis.

The model built in the course of the study utilizes the physical performance data of almost 2,000 players from the NFL Draft Combine, the annual event conducted by the NFL to recruit new players, most of whom tend to college athletes sourced from parsing the NFL Combine Results website.

Evidence gathered points to the fact that the clustering algorithms are a great way to classify the players in relation to the data used, although the efficiency was a little questionable. The model misclassified 262 players in total, which is not a very good model that could be used in the real world because of which further work will be required in order to optimize the solution.

1 Introduction

American Football is one of the most popular sports in the world, with as much as a total of 100.7 million viewers watching the Superbowl LIII feature between the New England Patriots and the Los Angeles Rams this year. Over the years, the National Football League (NFL) has grown to be the largest professional sports league in the world, bringing in a revenue of over USD 255 million in 2017, with each franchise being valued at a staggering USD 2.7 billion on average.

Earlier on, the sports industry was mostly about the entertainment aspect and loyalties between fans playing out in the numerous games and rivalries, but with the growth of technology, sports has emerged to be one of the pivotal areas which could reap the benefits of research in areas such as Machine Learning and Artificial Intelligence.

Fantasy Soccer and Fantasy Football have turned out to be an area where ordinary sports fans can assume the role of a team manager, recruiting players at the beginning and compete against their peers over the course of the league. Initially just modeled as a simulator game for fans, it has evolved into one that involves large sized financial rewards. The performance of the players that these fans can recruit fluctuate according to their performance in the real sporting league, thereby making it imperative to make proper recruitment decisions while putting together a roster.

When it comes to the NFL, amateur players are recruited from colleges in what could be termed a league-wide ”draft” conducted annually in the early days of Spring. Aspiring professional football players are subject to a rigorous selection process titled the *NFL Scouting Combine*, often called the NFL Combine for short prior to getting drafted by the different franchises. The NFL Combine recruits a total of 256 players per year over 7 rounds, by offering them the chance to showcase their physical as well as mental mettle to the coaching staff of the different NFL teams. Examples of the various tasks covered by the combine include the Bench Press, 40-yard dash, 3-cone drill, vertical jump, Wonderlic test (Intelligence test) and personal interviews.

Considering the fact that roster management is something that NFL franchises often spend fortunes on, a predictive model to analyze the position in which a player would be best fit can help teams make better decisions, as well as make proper trades with other teams if targeting a specific player. This benefit would also apply across to a scenario like *Fantasy Football* where people can be allowed to be create more robust rosters.

This paper therefore attempts to study the effectiveness of a K-Means based Clustering model in order to effectively classify the positions of NFL draftees based on their physical traits as measured at the NFL draft combine in the years of 2014 to 2018.

2 Related Work

The use of predictive analytics in sports is not new by any means, although it has not been around for a long time either.

The most notable success of sports analytics is notably the German software giant *SAP AG*'s collaboration with the DFB (Deutscher Fussball-Bund or the German Football Association) in setting up a big data room during the 2014 FIFA World Cup in Brazil. This so-called big data room was tasked with collecting data and insights that would help coaches and scouts to process vast amounts of data to find and assess key situations in each match so as to improve the player and team performances.

Surprisingly enough, the German National Team were crowned the world champions in that same tournament.

Similarly, the Philadelphia 76ers have a complete team dedicated to analytics and data science to chart strategies to improve the team's performance. The bulk of the work is contained in analyzing the play by play data to improve the in-game strategies for the team.

In his research on developing an optimal draft strategy for Fantasy Football teams, Papa Chakravarty found that in order for someone to win the league, it is imperative that the drafted team score at least 1.29 standard deviations than the league average[1].

In terms of more advanced machine learning models, advanced neural networks (ANNs) have been employed to identify the methods used, as well as means to evaluate models and the intricate challenges involved in the prediction of sports results[2].

This study investigates the effectiveness of using a clustering model to classify players with physical data alone in a bid to help in the areas of sports betting and fantasy games.

3 Description of Data Used and Methodology

In order to build the model that would be used to classify players, dataset containing information from the NFL Draft Combine from years 2014 to 2018 was utilized. These datasets were scraped from the NFL Combine Results website[3] for the draft using a combination of Python and BeautifulSoup. The dataset contains the following features:

- Name: This is the name of the player trying out for the NFL Draft.
- Position: This is the position that the player selected to try out for.
- Height: This is the height of the potential player measured in inches.
- Weight: This is the weight of the potential player measured in pounds.
- Hand Size: This is the measurement of the length of a player's hand between the thumb and the end of the little finger, measured in inches.
- Arm Length: The arm length is the measured length of a player in inches.
- Dash: This is the amount of time it takes for a player to complete sprinting 40 yards in total, measured in seconds.
- Bench: This is the number of repetitions of the bench press exercise that a player is capable of performing.
- Vertical Jump: This is the number of inches that a player is capable of projecting themselves in the upward direction starting from a stationary position.
- Broad Jump: This is the number of inches measured when an athlete combines their speed, strength and agility in an attempt to leap as far as possible from a given take off point.
- Shuttle: The 20-yard shuttle, measured in seconds, is a test performed by the football players to evaluate their ability to change directions and quickness.

- Cones: This is the amount of time, in seconds, taken by an athlete to change directions when moving at a very high speed, by forcing them to change directions according to the placement of three cones on the field.
- Draft Position: This is the round of the NFL Draft that a player is selected by a franchise. The NFL draft is comprised of 7 rounds, implying that the value of this variable could range anywhere from 1 to 7, if they were selected. If the player went undrafted, the value of this variable for them would be equal to zero.
- Wonderlic Test: The Wonderlic test is used as a measure of the intelligence of the athletes.

The dataset utilized contains records for these various features for a total of 1961 players. Before carrying out any form of exploratory analysis or building a model, it is imperative that the integrity of the dataset is maintained. This is because of the fact that unclean data usually tends to exhibit the presence of large amounts of missing data or inconsistent formats across the dataset. The dataset utilized was observed to have a lot of missing values across an assortment of features because of which they had to be filled in for. Removing the datapoints containing missing values for the features was considered, but it was dropped when doing so reduced the size of the dataset by a large extent. These missing values were so filled in with the help of imputation. The imputation was carried out by grouping the players according to the positions they played in and then filling the missing values using the median measures for the features[4].

Once the dataset was cleaned, detailed analyses were carried out to measure the performance of players compared to the other players in their positions in order to see if they would offer us any form of evidence regarding the usefulness of a player at a certain position. Radar plots were constructed to see how players performed in their positions, one of which is shown below:

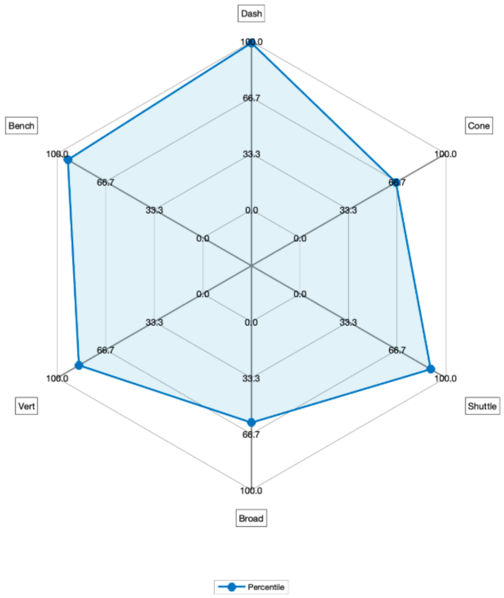

Figure 1: The Radar Plot for judging the performance of Dri Archer, a Wide Receiver. In spite of a stellar performance in the NFL Combine, Archer would go on to switching teams on a yearly basis and eventually get released by every franchise he played for.

In addition, correlation matrices for the players with respect to the round in which they were drafted were also examined to see if any insights could be gathered. The correlation matrix relating the physical attributes and the draft round positions for the quarterbacks is as shown below:

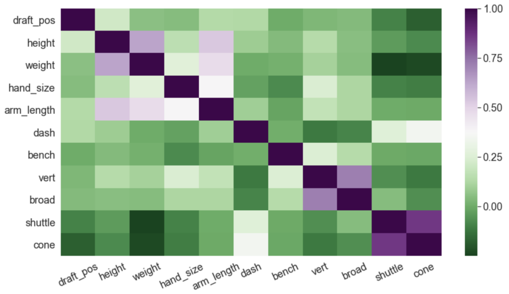

Figure 2: Correlation matrix for Quarterbacks in the NFL draft combine from 2014-2018.

It is imperative to note that in accordance with the purpose of helping sports fan build rosters that could garner financial rewards through avenues like Fantasy Football, offering them a clean list of players classified according to the positions they play in and are capable of playing in (given their verstaility) would help them a lot in making more data-driven decisions. So, a clustering model based on the Manhattan Distance metric was built in order to be able to better predict a lineup or position that a player can play in.

The Manhattan Distance is defined as the between two points measured along the axis at right angles. The distance between two points with x and y coordinates at (x_1, y_1) and (x_2, y_2) is given by[5]:

$$|x_1 - x_2| + |y_1 - y_2|$$

The aforementioned model was built largely using a Python technology stack. The major libraries used in the implementation of the model include *NumPy*, *Pandas*, *scikit-learn*, *matplotlib*.

The dataset, once cleaned up is set up for pre-processing before the model is built. A new dataframe was created with just those columns that would help in determining clusters. In order to enforce this, the Wonderlic Test and the Draft Position features were dropped altogether. The Wonderlic Test feature was dropped given the fact that it was missing for 88% of the players. This is due to the fact that the test is voluntary. The Draft Position variable was dropped since it doesn't add anything of substance. It does aid in the prediction of the draft round, but that is outside the scope of this paper.

Another instance of feature engineering involved the encoding of the categorical Position feature into an integer in the range of 0 to 10. The motivation for doing so was the fact that upon preliminary analysis, there were almost 23 unique positions with capabilities that intersected a lot. This if taken as it is would necessitate the formation of a very large number of clusters which is undesirable, and could potentially turn out to be a very overspecified model. So, in order to carry this encoding out, the following positions were merged into unified labels:

- OLB, ILB, LB were put together into the *Line Backer* category.
- CB, S, SS and FS were put together into the *Safeties* category.
- OL, OG, C and OT were condensed together as *Linemen.*
- DT, NT, DE and DL were condensed together as *Defensive Linemen.*
- K, LS and P were fused into *Kickers.*

This condensation of positions therefore resulted in the clusters being that of Wide Receivers, Safeties, Tight End, Running Back, Line Backer, Linemen, Quarterback, Defensive Linemen, Kicker, Defensive Back and Full Back.

Then the positions of the players were encoded to a number in the range of 0-10. After this encoding was finished, the features were then extracted and subject to Principal Component Analysis. This is because of the number of features involved meant that dimensionality reduction had to be applied.

In order to find the ideal value for the number of components to be chosen to be used in the PCA analysis, the following explained variance plots were plotted as shown below:

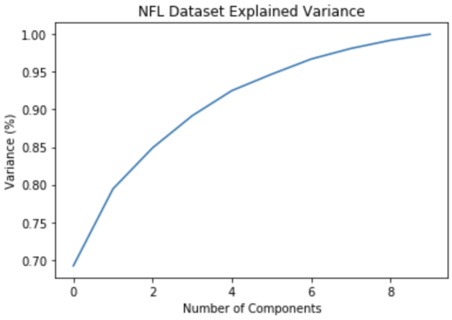

Figure 3: The Explained Variance Plot explaining the choice of 6 as the number of components to be used in the PCA. It could be seen from the curve that it's at 6 components that the explained variance is over 95%.

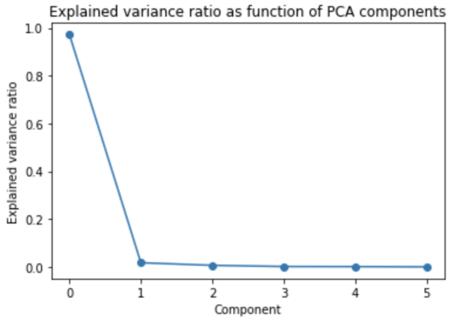

Figure 4: The Explained Variance Ratio Plot. This plot essentially returns the percentage of variance explained by each of the selected components.

The value of the number of components was so chosen that 95% of the variation was explained, eventually translating into 6. Once this is done, the visualization of these clusters could be accomplished. In order to view the clustering of these players in the two-dimensional space, 2 components were used. This is where the clustering process began. The initial clustering was carried out to see the spread of the data points as clusters using the default in-built K-Means clustering feature provided by *scikit-learn*, which is as shown below:

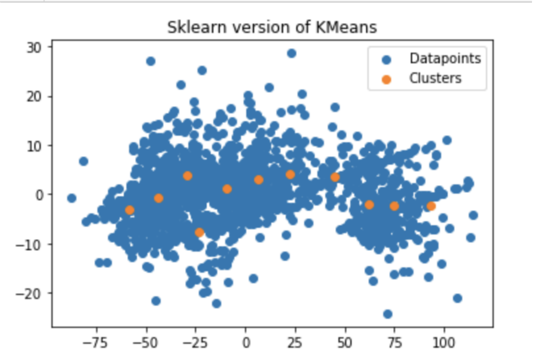

Figure 5: The distribution of the NFL dataset and the clusters.

Upon this, a KMeans clustering algorithm was implemented that took the number of clusters as an input along with a threshold value of 0.0001. When the difference between two iterations of the clustering algorithm falls below this threshold, the algorithm terminates. After the execution of the algorithm, the clustering algorithm takes the cluster data in the form of arrays and plots them, the result of which is discussed in the subsequent section.

The distortion function of the clustering was then analyzed in order to evaluate the quality of the clustering. When assessing the performance of a clustering model, it is imperative that the evaluation function be closely related to the clustering criteria since such a relationship could prevent adverse effects on the validation process. In the case of the K-Means function, this chosen criterion usually tends to be the minimization of the *distortion* of the clusters[6].

From a mathematical perspective, the distortion is defined as the sum of the squared distances between each observation vector and its dominating centroid. When the K-Means clustering model is run, during every iteration the algorithm refines the choices of the centroids so as to reduce the distortion. It is also to be noted that this measure of clustering quality is also the factor that determines when the clustering algorithm ceases to run[7]. When the change is less than a set threshold, the K-Means clustering model is not making any further progress and terminate.

4 Results

The results of the clustering model to deduce position players is as shown below:

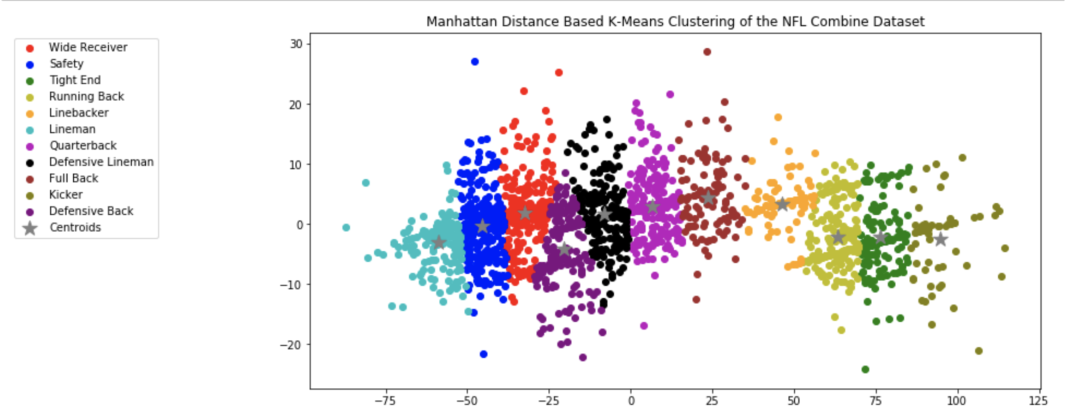

Figure 6: The two-dimensional plot of the players classified into the different positions as developed by the model.

As could be seen from the above figure, the 1,961 players are classified into the eleven different clusters. The error rate for the misclassification was calculated; that is, the number of players who were incorrectly pooled together. The Manhattan Distance metric is shown to have an error rate of 13.4%. This translates into 262 misclassified players. It is understood that the use of the Euclidean distance metric would have provided a classification with a lower margin of error.

As mentioned prior, the evaluation function of the clustering algorithm, the distortion function for the K-Means clustering model built to classify players is as shown below. As could be seen from the plot, at the presence of 11 clusters, the value of the distortion function is very low, which is desirable.

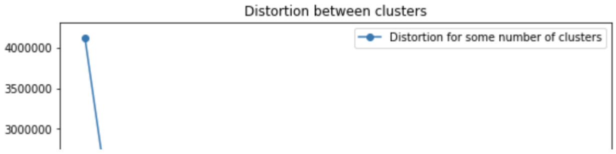

Figure 7: Distortion Curve of the Classification Model. The point where there is the most rapid decrease in the distortion function is often called the elbow point, and usually tends to be a good visual method to evaluate a clustering model.

5 Conclusion

The effectiveness of using K-Means based cluster algorithms for the NFL draftees was studied. Exploratory data analysis didn't prove to be the best of methods to classify players based on positions since a lot of players getting released by their franchises conflicted with stellar performances at the Combine.

The K-Means clustering is an interestingly questionable method to classify players. This is because of the fact that while it was able to classify players pretty well, it also failed to classify 262 players out of a 1,961 players which is something that needs to be addressed in order to be effective at all times.

This also implies that there are several extraneous variables such as psychological factors other than just the physical performance of the NFL Draft Combine that cannot be explained by this model, thereby necessitating the need to explore more methods to solve this problem.

6 References

[1] P. Chakravarthy, "Optimizing draft strategies in fantasy football", Harvardsportsanalysis.files.wordpress.com, 2012. [Online].
Available: https://harvardsportsanalysis.files.wordpress.com/2012/04/fantasyfootballdraftanalysis1.pdf. [Accessed: 22- Oct- 2019].

[2] Bunker, Rory P., and Fadi Thabtah. “A Machine Learning Framework for Sport Result Prediction.” Applied Computing and Informatics, vol. 15, no. 1, 2019, pp. 27–33., doi:10.1016/j.aci.2017.09.005.

[3] “32 Years of NFL Scouting Combine Data.” NFL Combine Results, 25 Mar. 2019, http://nflcombineresults.com/.

[4] G. Drakos, “Handling Missing Values in Machine Learning: Part 2,” Medium, 05-Oct-2018.
Online. Available:
https://towardsdatascience.com/handling-missing-values-in-machine-learning-part-2-222154b4b 58e.

[5] Pandit, Shraddha, and Suchita Gupta. “A Comparative Study on Distance Measuring Approaches for Clustering.” International Journal of Research in Computer Science, vol. 2, no. 1, 2011, pp. 29–31., doi:10.7815/ijorcs.21.2011.011.

[6] “K-Means Clustering and Vector Quantization (Scipy.cluster.vq).” K-Means Clustering and Vector Quantization (Scipy.cluster.vq) - SciPy v0.14.0 Reference Guide, https://docs.scipy.org/doc/scipy-0.14.0/reference/cluster.vq.html.

[7] Pham, D T, et al. “Selection of K in K-Means Clustering.” Journal of Mechanical Engineering Science, vol. 219, no. C, 27 Sept. 2004, doi:10.1243/095440605X8298.