Clustering UFC Fighters

How effective are clustering algorithms on UFC Fighters?

Yuvaraj Selvam
Computer Science and Engineering Department
Arizona State University
Tempe Arizona USA
yselvam@asu.edu

ABSTRACT

The availability of large amounts of data in sports has made Sports Analytics a rapidly evolving topic. Data mining techniques can be applied in sports for a variety of different reasons like educating younger generation of players and technical staff, optimizing player training, evaluation of player performances, gaining a competitive advantage over opponent teams/individuals, etc. MMA, as a sport, is still in its nascent stages and is growing continuously. In such a setting, it becomes all the more important to use data mining techniques to analyze fight statistics and understand the sport better.

Contrary to popular belief, brute force alone will not always be sufficient to win a fight. One must have a clear tactical plan on how to approach his/her opponent. To devise a good fighting strategy for an opponent, one must understand their opponent's fighting style. Previous work has predominantly focused only on predicting winners of future fights. This work deals with clustering fighters based on their fighting style to better understand their strategies using K-Means++, Gaussian Mixture Model, and Agglomerative clustering algorithms based on their fight statistics and comparing their results. My findings indicate that Gaussian Mixture Model performs better than K-means++ and Agglomerative Clustering in this context.

KEYWORDS

Mixed Martial Arts, Clustering, K-Means++, Gaussian Mixture Model, Agglomerative clustering

1 Introduction

Mixed Martial Arts (MMA), as the name suggests, is a combat sport that incorporates techniques from multiple other combat sports. Broadly speaking, fighters employ boxing, Brazilian Jiu-Jitsu (BJJ), kickboxing, wrestling, Muay Thai, Taekwondo, and Karate in MMA fights. While MMA fighters are required to be skilled in various disciplines, most fighters predominantly stick to a single fighting style. A fighter's primary fighting style often becomes evident when we analyze a combination of few metrics from their fights. But, as a fighter incorporates more and more techniques from other disciplines in his/her fights, the distinction becomes indiscernible to the naked eye.

This project is an experiment on how various clustering algorithms cluster fighters that compete in Ultimate Fighting Championship (an American mixed martial arts promotion company, abbreviated as UFC) events based on the round-by-round statistics of their fights.

2 Related work

Most of the work on UFC datasets try to predict winners of future fights. Very little work has gone into clustering fighters. An article titled Clustering UFC Fighters by Fighting Style [1] talks about applying PCA to reduce dimensions and then using k-means to cluster the fighters. But it is implemented on a dataset that does not take round-by-round statistics into account and it does not compare results from multiple algorithms.

A paper titled Clustering of Basketball Players Using Self-Organizing Map Neural Networks [2], which was published in the Journal of Applied Research on Industrial Engineering, proposes the use of Artificial Neural Networks to cluster Basketball players based on their abilities. This paper suggests the use of playing position as the basis of difference for clustering. Finally, it compares the results from Self Organizing Maps (SOM) and K-means algorithms. It concludes that SOM is significantly superior over the K-means algorithm in this context.

Another article [3] discusses the idea of viewing the overall strategic approach and tactical approaches a fighter takes per round separately. This article also explains a fighter's style is embedded in the choices they make during their time in the cage. Further, it breaks the fighters down using K-means algorithm based on just 3 aspects–Standing time, control time and controlled time.

3 Methods

3.1 Dataset Collection

All the data that is needed for this project is available on UFC's Statistics website [4]. I scraped every single piece of fight-related information listed on the website, so that I will have more flexibility in the next phase of the project.

First, I tested to see if the website blocks IPs (Internet Protocol) that make continuous requests for long periods of time. As the website did not employ any rate limiting, I decided against mimicking real users. I directly retrieved the webpage's content programmatically using requests module in python. Then, I used Beautiful Soup (a web scraping python library) for tokenizing the content into HTML tags. Since the fight statistics were not directly available in a single page, I collected the links to all the completed UFC from parsed HTML tokens. Then, for each event, I collected the links to all the fights that happened in the event and extracted the overview of fight statistics listed in the page. Next, for each fight, I retrieved both fighters' statistics for every round. I stored all the extracted information locally in CSV format. Finally, once I had the round-wise statistics for all UFC fights, I collected details of every fighter that are not available in the fights page separately.

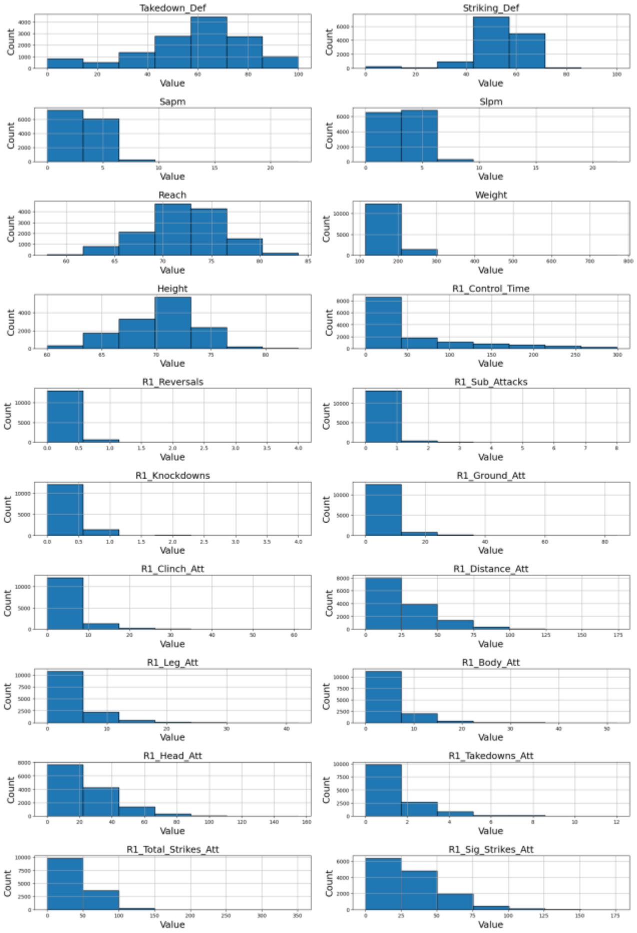

Figure 1: Distributions of a few numerical features in the dataset

3.2 Dataset Overview

The dataset has round-wise statistics for all the fights that happened between March 11, 1994 and November 12, 2022. In addition, the dataset has information for each fighter that are not captured in the round-wise statistics. Some of the features in this dataset are Total number of strikes landed, Significant strikes landed, Number of Takedown attempts, Takedown accuracy, Knockdown counts, Ground control time and Submission attacks. Figure 1 visualizes a few numerical columns in the dataset as histograms to better understand the dataset. Overall, this dataset has 49 attributes including statistics from 6888 fights that happened over 16000 rounds involving close to 3000 fighters.

3.3 Data Preparation

While collecting a lot of data points provides flexibility, it also introduces the need for considerable amount of preprocessing before the data can be fed to the clustering algorithms.

3.3.1 Data Restructuring. Statistics for each round in a fight were in a separate row. First and foremost, I pivoted the dataset to represent a whole fight in a single row. Then, each fight includes statistics for two fighters. As I am trying to cluster fighters, I split each row into two rows, one per fighter. I then applied data transformations to ensure compatibility with the clustering algorithms.

3.3.2 Data Cleansing. This dataset contains attributes that are ordinal, ratio and nominal in nature. Most of the columns in the dataset were of string type that are numbers in reality. These features required some manipulation before they can be type casted into numeric types. For example, weight of a fighter was represented as '185 lbs.' instead of just '185'. It also had strings like "—" to indicate blank values. I used regular expressions for pattern matching and replaced such values with appropriate data. Also, I split columns which were of the format '84 of 100' into two columns.

3.3.3 Unit Conversions. I converted time-based features, which were originally in terms of minutes and seconds, to just seconds so that they can be easily compared. Similarly, I converted height column to inches from feet and inches.

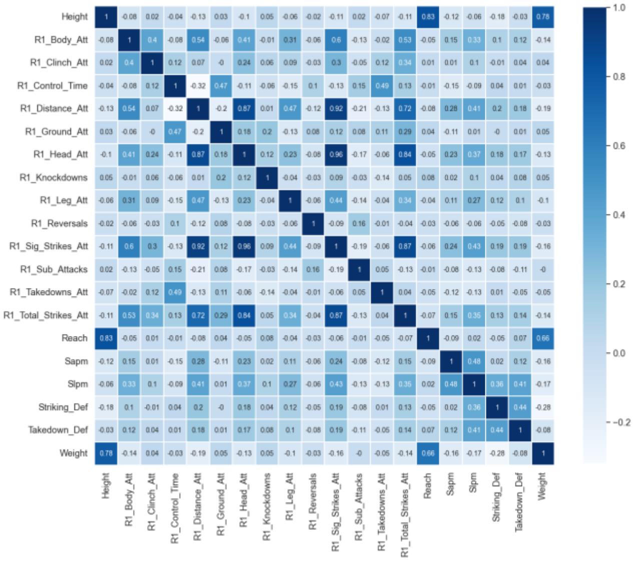

Figure 2: Feature Correlation Matrix

3.3.4 Feature Selection. The raw data includes a lot of data points that may not contribute to distinguishing fighting styles. Thus, it is crucial to select only the features that are appropriate for the task at hand. It is also important that the selected features capture all facets of the sport: striking, grappling and ground control—in terms of both offence and defense. As I am only interested in the choices a fighter makes in a fight, I dropped features that measure the success rates of the fighter's attempts. For example, features like number of takedowns attempted, number of strikes attempted are more suited to my needs than number of takedowns completed, or number of strikes landed. These insights helped me remove irrelevant and weak features in the dataset.

In order to identify redundant features, I plotted a correlation matrix for all the features in my consideration. Figure 2 shows some obvious correlations between height, weight and reach of a fighter. Apart from that, I understood that it would be redundant to have both number of strikes attempted from a distance and number of significant strikes attempted. Likewise, number of strikes attempted towards the opponent's head is highly correlated to the number of significant strikes. I used these findings to remove a few redundant features from the dataset.

After carefully selecting a subset of features using the above methods, I ended up with 29 dimensions to feed into the clustering algorithms.

3.3.5 Feature Scaling. I applied standard scaling on numeric features to better capture the insights encoded in the data. Feature scaling is a very crucial step in this context because, my goal is to test out K-Means, an algorithm that is all about calculating distances between data points.

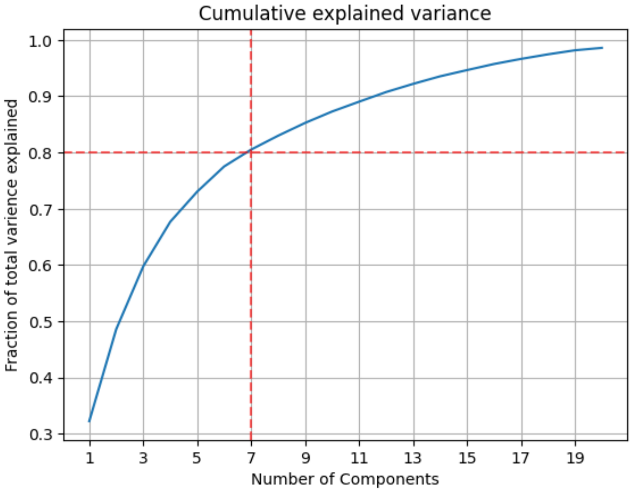

Figure 3: Total variance explained by 20 principal components

3.3.6 Dimensionality Reduction. I conducted Principal Component Analysis (PCA) on my feature set to explore the possibility of further bringing down the number of dimensions used for clustering. In order to find a good tradeoff between retaining as much variance as possible and having as less dimensions as possible, I plotted the total variance explained over number of principal components. As Figure 3 shows, the first 7 principal components explain 80% of the variance, which is a good tradeoff.

3.4 Clustering

After scaling and reducing the number of features in the data, my next step was to determine the total number of clusters, *k*. For that, I applied K-means++ with varying number of clusters ranging from 1 to 9 and computed their corresponding sum-of-squared errors (SSE). Figure 4 shows Sum of Squared Errors corresponding to number of clusters. Then, based on the location of the "elbow" in the graph, I chose *k* to be 3. In other words, I chose to cluster the fighters in dataset into 3 categories.

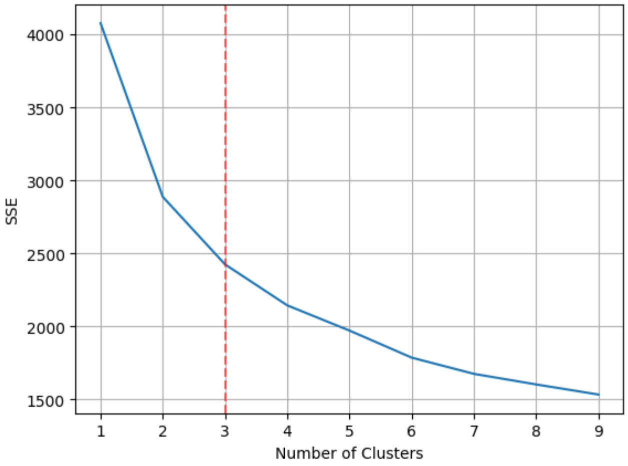

Figure 4: Sum of Squared Errors vs. Number of clusters

Once I selected the number of clusters, I then proceeded to cluster fighters using 3 different methods—K-means++, Gaussian Mixture Model and Agglomerative Clustering. While clustering, I considered the fact that fighters may fight differently in each of their fights depending on the fighting style of their opponent. This means that assigning a single cluster to each fighter directly could skew the results. So, I decided to assign a cluster to each fighter-fight pair separately and then take the most frequently occurring cluster for a fighter as his/her fighting style.

3.4.1 K-means Clustering. K-means is one of the most popular machine learning algorithms for its simplicity. It first starts by selecting *k* samples from the data as the centroids without replacement. Then, it calculates the distance between all the points in the data and the earlier chosen centroids. Then, it groups every point with its closest centroid and recalculates the centroid for that group by taking an average of points in the cluster. Since the crux of the algorithm is distance-based, it is important to make sure the data has been standardized.

I ran the K-means algorithm for 20 times with different centroid seeds each time. I chose to have a maximum of 500 iterations per run to ensure that the cluster centroids converge completely.

3.4.2 Gaussian Mixture Model. Gaussian Mixture is a model-based approach to clustering. It constructs a model for each cluster in the dataset and tries to fine-tune how well the model fits the data. Since the model it uses to represent the clusters is a mixture of Gaussian distributions, it is called the Gaussian Mixture Model (GMM). A major advantage of Gaussian Mixture over K-means is that it can handle non-linearity in the data. At the same time, I was able to notice that GMM was the most computationally expensive of all the methods I compared.

3.4.3 Agglomerative Clustering. Agglomerative Clustering is a type of hierarchical clustering, also known as Agglomerative Nesting (AGNES). The basic idea of this method is that, initially, every point in the dataset is a cluster on its own. Then, two clusters that are close to each other are merged together to form a larger cluster. This is repeated until there is only one large cluster containing all the data points. Now, there are many ways to decide the "closeness" of two clusters. For example, the average of the differences between the points in two clusters could determine how close they are. I used the 'complete' linkage to determine the closeness of two clusters, which uses the maximum value of all pairwise distances between the elements in the two clusters. This results in compact clusters.

3.4.4 Assigning Labels to Clusters. I analyzed the clusters from each algorithm and compared the similarities and dissimilarities of the fighters in those clusters. I was able to assign appropriate labels to these clusters, namely Grappler, Tactician and Power Puncher. I then hand-picked a few fighters for each label and compared them with labels assigned by the clustering algorithms.

4 Results

I visualized the first three principal components of the data as a 3D scatter plot and color-coded them based on the clustering results from each algorithm. This is shown in Figure 5.

Now, to quantify the clustering performances, I calculated the silhouette scores for each method and listed them in Table 1.

Table 1: Silhouette Scores

Clustering Algorithm	Silhouette Score
K-means	0.225286
Gaussian Mixture	0.062508
Agglomerative (complete)	0.265400

Furthermore, I tabulated the labels assigned by each algorithm alongside the actual label for the handpicked fighters in Table 2.

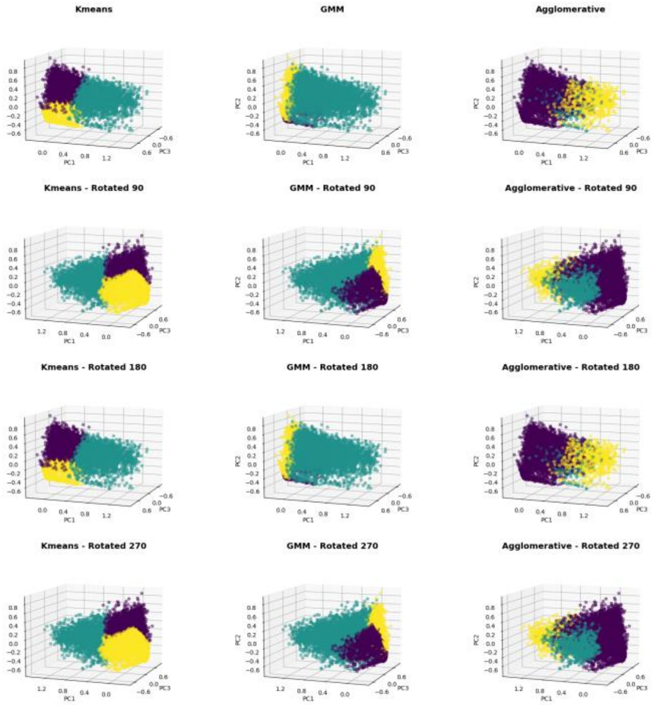

Figure 5: 3D Visualization of results from 3 clustering algorithms

Table 2: Cluster Labels for Hand-Picked Fighters

Fighter	Labels			
	K-Means	GMM	AGNES	Actual
Demetrious Johnson	Tactician	Grappler	Tactician	Grappler
Georges St-Pierre	Grappler	Grappler	Tactician	Grappler
Khabib Nurmagomedov	Grappler	Grappler	Grappler	Grappler
Anderson Silva	Power Puncher	Tactician	Tactician	Tactician
Conor McGregor	Power Puncher	Power Puncher	Tactician	Tactician
Israel Adesanya	Tactician	Tactician	Tactician	Tactician
Francis Ngannou	Power Puncher	Power Puncher	Tactician	Power Puncher
Derrick Lewis	Power Puncher	Power Puncher	Tactician	Power Puncher
Justin Gaethje	Tactician	Power Puncher	Tactician	Power Puncher

5 Discussion

Based on just the silhouette scores, Agglomerative clustering is best suited for this dataset. This may not necessarily be the case though. This may have been due to the suboptimal choice for the linkage parameter. As the 'complete' algorithm tries to keep the clusters compact and because of the how the data is, most of the fighters might have ended up in the same cluster. This may have resulted in a good silhouette score.

Table 2 tells us that Gaussian Mixture Model performed the best out of the three as its labels match the actual label most. K-means labels also match with actual labels to some extent. It also tells us that Agglomerative clustering method was indeed not the most accurate despite having the best silhouette score among the three algorithms.

5 Future Work

I plan on further enriching the dataset by collecting data from other sources which may affect the clustering. For example, metrics like total round time, total time spent on the feet were not included in this analysis which can be very helpful in normalizing the existing features. Also, it would be interesting to test out clustering algorithms like Self Organizing Maps (SOM), Mean Shift Clustering, Spectral Clustering on this dataset and comparing them with these results. It is worth noting that, some of these fighters can belong to multiple clusters. So, applying non-exclusive clustering algorithms could also produce compelling results.

6 Conclusion

Given that MMA fighters use skills from multiple fighting disciplines, both K-means and Gaussian Mixture Model performed well to cluster the fighters meaningfully. Both of these algorithms were able to find nuances in the data that can be useful in analyzing the sport. This also means that data mining techniques can be effectively used in MMA analysis.

REFERENCES

[1] David Wismer. 2022. Clustering UFC Fighters by Fighting Style. Retrieved December 3, 2022 from https://davidrwismer.medium.com/clustering-ufc-fighters-by-fighting-style-1f65102b4821.

[2] Babaee Khobdeh Soroush and Yamaghani Mohamad Reza, 2021. Clustering of Basketball Players Using Self-Organizing Map Neural Networks. Journal of Applied Research on Industrial Engineering, E-ISSN: 2676-6167 | P-ISSN: 2538-5100.

[3]. Jason Leib. 2021. Fast data model for classifying MMA fighting styles. Retrieved December 3, 2022 from https://www.mmai-analytics.com/articles/breaking-down-the-data-science-of-the-fast-models/

[4] UFC Statistics. Stats | UFC. Retrieved December 3, 2022 from http://www.ufcstats.com/.