The Hidden Gem

Titus Lee, Gary Park, and Logan Richardson

ECE 2410 – Introduction to Machine Learning

University of Virginia

Submitted: May 5, 2025

I. Experimental Set-up

The motivation for this experiment was to aid soccer team managers in scouting for younger and cheaper players who are similar to their star players. This provides a scouting recommendation for the identifying undervalued talent in the professional scene. The goal for this experiment is to identify the best form of clustering to identify a similar play style to that of their star player.

To implement this logic, nearest neighbors was used first, then comparing K-means clustering and hierarchical density-based spatial clustering of applications with noise (HDBSCAN) to determine which clustering algorithm best predicts the star player based on physical attributes and skill ratings. The group hypothesized that the HDBSCAN algorithm would be a better indicator of the lower rated player that best matches the play style of the target because this algorithm uses density based clustering and automatically selects the most stable number of clusters based on the hierarchy construction, as well as distinguishing noise and outliers within the data. K-means clustering requires the user to identify the number of clusters initially, which could add bias and give a worse predictor ultimately.

II. Methods

For this experiment, a dataset of approximately 17000 professional Fifa soccer players was used containing 50 different features ranging from physical attributes such as height, stamina, strength and so forth, along with actual play style ratings such as overall rating, dribbling, freekick. This soccer player data is from 2024 and provides recent data to be used by managers for the next soccer season.

This project utilizes K Nearest Neighbors (KNN), K-means clustering, and HDBSCAN to effectively find the top 1000 players that match a target and then create clusters based on their physical and skill attributes. To implement these algorithms the SKLEARN libraries for preprocessing using standard scaling and standard metrics, nearest neighbors, Kmeans, and a silhouette score library were imported to be utilized.

The procedure used for this code was to begin by preprocessing the data through filtering the features to that our group felt best for narrowing down skill ratings. To match the purpose of this project, the original table was filtered to only include players of less value and a lower age. Other features such as value, name, position, age and potential were also saved to be used as metadata after the clustering. After filtering those features to be used, The selected features in the dataset were standardized using StandardScaler, ensuring that both the filtered dataset and the target player features are normalized using the same scaling parameters for consistent comparison.

The next implementation was K nearest neighbor where the nearest neighbor SKLEARN function was used to find the 1000 nearest players to a given target player, then retrieving the corresponding unscaled player rows and adding distance as a feature in the column.

Next, K-means clustering was applied on the normalized data for the nearest 1000 players. The data is first grouped using k-Nearest Neighbors and then clustered into k groups based on playing style. A silhouette score between -1 and 1 is assigned to evaluate how well-defined the clusters are, and the top players from each cluster are displayed for analysis. The k-means was performed from 3 to 10 clusters and the silhouette score was assigned.

Finally, HDBSCAN was implemented on a subset of players selected using k-Nearest Neighbors based on normalized performance features. The algorithm was applied without specifying the number of clusters, but creating a minimum size of 3,5, and 7 clusters, allowing it to discover natural groupings in the data. Parameters like minimum cluster size and minimum samples were tuned to control the clustering sensitivity and noise detection, with outliers automatically labeled as noise (cluster = -1).

For the results section, the distance from the target player was plotted for the nearest players, along with the results of clustering through K-means and HDBSCAN to display. The silhouette scores were analyzed to compare scores and clustering results for HDBSCAN and K-means to determine which method was best for data with outliers and data closer in value.

III. Results

Principal Component Analysis (PCA) plots were generated to visualize the clusters formed by both k-means clustering and HDBSCAN, as shown in Figures 1 and 2. For average-tier players, both methods produced reasonable and interpretable groupings. However, for elite players such as Cristiano Ronaldo, HDBSCAN frequently failed to assign a cluster, labeling them as noise due to their statistical uniqueness. In contrast, k-means consistently assigned each player to a cluster, even if the player was an outlier.

Three representative players were selected to evaluate the performance of k-means and HDBSCAN for average players as shown in 3: Trippier, Carrillo, and Escudero. For each player, clustering parameters were tuned to maximize silhouette scores—specifically, the number of clusters k for k-means, and the `min_cluster_size` and `min_samples` parameters for HDBSCAN.

- **Trippier**: The optimal configuration was `k = 3` for k-means (silhouette score: 0.170), and `min_cluster_size = 3, min_samples = 1` for HDBSCAN (silhouette score: 0.199). The top five most similar players identified by k-means were B. Davies, S. Aurier, F. Coquelin, T. Alexander-Arnold, and M. Lemina. HDBSCAN returned a similar list, substituting L. Shaw for M. Lemina.
- **Carrillo**: The best performance was observed with `k = 5` (silhouette score: 0.167) and `min_cluster_size =`

7, `min_samples = 3` (silhouette score: 0.212). Both clustering methods produced the same top five comparable players: J. Izquierdo, Jese, T. Bongonda, R. Centurión, and D. Pelkas.

- **Escudero**: Optimal values were `k = 3` (silhouette score: 0.161) and `min_cluster_size = 3`, `min_samples = 1` (silhouette score: 0.197). The k-means method identified Montoya, Cedric, M. Doherty, M. Plattenhardt, and A. Conti as the closest players. HDBSCAN listed Montoya, Carles Planas, Cedric, M. Doherty, and M. Plattenhardt.

To assess clustering performance for elite players, an additional three were selected as shown in 4: Neymar Jr, L. Messi, and Cristiano Ronaldo. As before, the parameters were tuned to maximize silhouette scores.

- **Neymar Jr**: The optimal configuration was `k = 3` for k-means (silhouette score: 0.265), and `min_cluster_size = 3`, `min_samples = 1` for HDBSCAN (silhouette score: 0.115). K-means identified K. Mbappé, Coutinho, M. Salah, P. Dybala, and H. Son as the most similar players. HDBSCAN labeled Neymar Jr. as noise and did not assign him to any cluster.
- **L. Messi**: The optimal configuration was `k = 3` for k-means (silhouette score: 0.231), and `min_cluster_size = 5`, `min_samples = 2` for HDBSCAN (silhouette score: 0.127). K-means returned E. Hazard, S. Agüero, P. Dybala, Neymar Jr, and L. Insigne as the closest matches. HDBSCAN again labeled Messi as noise and did not assign him to any cluster.
- **Cristiano Ronaldo**: The best configuration was `k = 4` for k-means (silhouette score: 0.225), and `min_cluster_size = 7`, `min_samples = 3` for HDBSCAN (silhouette score: 0.081). K-means identified S. Agüero, R. Mahrez, L. Insigne, H. Son, and P. Aubameyang as the most similar players. HDBSCAN again failed to assign a cluster, labeling Ronaldo as noise.

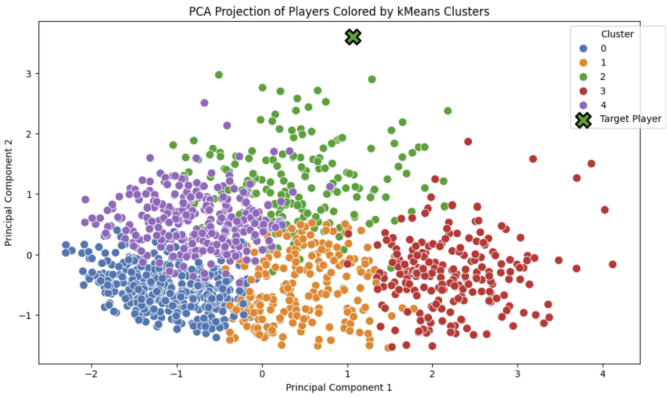

Fig. 1. PCA Projection of Players Coloroed by Kmeans Clustering

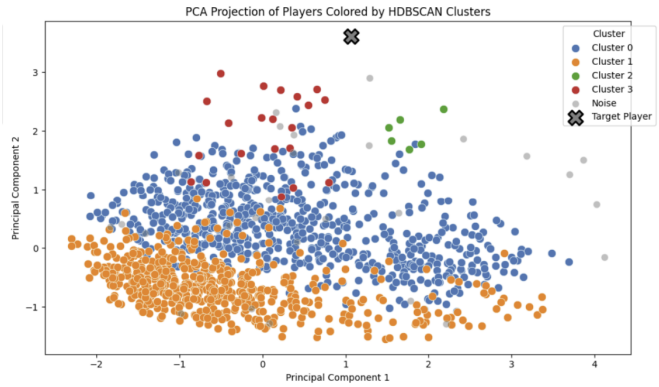

Fig. 2. PCA Projection of Players Colored by HDBSCAN

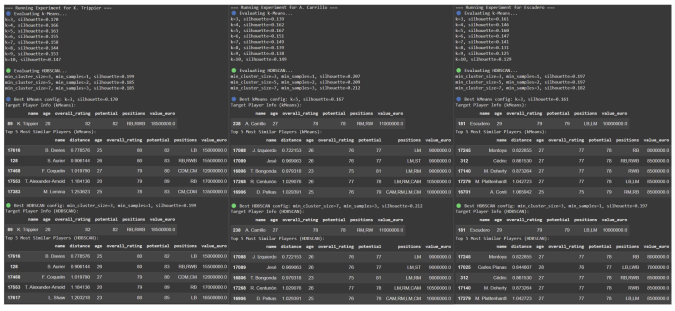

Fig. 3. Comparison of Average Player Results through Clustering Methods

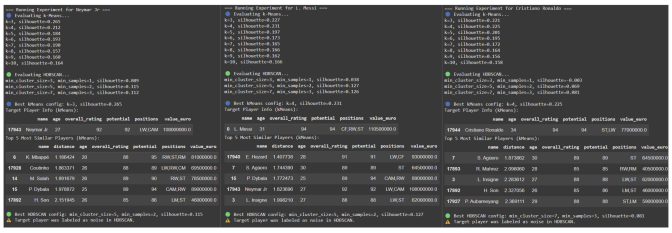

Fig. 4. Comparison of Elite Player Results through Clustering Methods

IV. Learning Objectives

The key takeaways of this project are that algorithm performance depends on data, where different algorithms suit different data structures and distributions. Metrics and domain knowledge are essential and meaningful clusters rely on selecting the proper features depending on the hypothesis in question. As for the specific clustering algorithms, K-means clustering works well for clearly separated, high-performing outliers; HDBSCAN clustering adapts better to noise and overlapping groups, making this density-based clustering more ideal for average players and data with few outliers.