LUISS

Corso di laurea in
Data Science and Management

Cattedra Data Science in Action

Football Data Analytics: The Science of Talent.

Machine Learning Predictions for Player Scouting.

Prof. Alessio Martino

RELATORE

Prof. Paolo Spagnoletti

CORRELATORE

Domenico Albertini - 762411

CANDIDATO

Anno Accademico 2023/2024

"*Il successo non è un incidente. È lavoro duro, perseveranza, apprendimento, studio, sacrificio e, soprattutto, amore per ciò che stai facendo o imparando a fare"*

- Pelé

ABSTRACT

The increasing complexity of modern football has made player scouting a more sophisticated and data-centric process. Traditional methods rely on subjective observation, struggling with consistency and efficiency. To enhance talent identification, this study aims to develop a machine learning-based scouting model that objectively evaluates players using performance data from the 2023/24 season.

Performance indices were constructed and adjusted for league difficulty, ensuring standardized comparisons across competitions. A K-Nearest Neighbors (KNN) algorithm, based on these performance indices, identifies statistically similar players, highlighting those with comparable playing styles. Static and interactive visualizations further support analysts in comparing players effectively.

The model has been tested on multiple case studies, and results show it accurately captures stylistic similarities across different player positions and various technical and tactical profiles, demonstrating its potential in data-driven scouting. While this approach is robust, it can be further improved, particularly by incorporating additional more complex data. Nonetheless, this study successfully explores the increasingly growing relationship between the world of football and data analytics, demonstrating the potential of a structured, data-driven scouting methodology.

INDEX

CHAPTER 1:
INTRODUCTION

1.1 Modern Football Scouting Scenario: The Need for a Data-Driven Approach

Scouting and recruitment have always been fundamental pillars of success in football, with clubs continuously striving to identify, evaluate, and acquire players who best align with their tactical and strategic objectives. Traditionally, this process has relied heavily on subjective human observation, with scouts traveling worldwide to assess talent based on in-game performances, technical skills, and perceived potential. While this method has led to the discovery of legendary players, it is inherently limited by cognitive biases, inconsistencies in evaluation criteria, and constraints in data collection.

The study by Lawlor et al. (2021) explores the evolution of scouting, tracing its progression from traditional observational methods—which relied heavily on scouts' intuition and subjective judgment—to more data-driven and analytical approaches. It highlights how football recruitment is increasingly shaped by performance analysis and statistical modeling, while still recognizing that human expertise remains a crucial element in the decision-making process.

Therefore, the rise of data analytics in football has fundamentally transformed the scouting and recruitment landscape. Clubs now have access to vast databases of player statistics, allowing them to track performance across multiple leagues and competitions. Metrics such as Expected Goals (xG), pressing intensity, passing efficiency, and defensive contributions offer a more objective and quantifiable basis for player evaluation. However, despite these advancements, traditional scouting methods still struggle to process and interpret large-scale data efficiently.

It is precisely within this domain that the present study seeks to contribute, proposing the development of a scouting algorithm that leverages the potential of machine learning. The objective is to use performance data to analyze players, providing scouts with a solid and data-driven starting point for their evaluations. This approach aims to enhance both objectivity and efficiency in player identification, not to replace human judgment but to

complement it. In football, the eye remains irreplaceable because, as the saying goes, “eyes don’t lie.” Visual assessment captures nuances and contextual elements that raw data alone might misinterpret. However, in today’s football environment—characterized by an overwhelming amount of information, increasing tactical complexity, and meticulous preparation—an analytical support tool like the proposed algorithm becomes essential. The evolution of tactical frameworks and the ever-growing influence of game schemes have transformed player selection into a multidimensional process. The objective is no longer just to identify high-performing athletes, but to find players who fit into the specific tactical and strategic framework of a team. For instance, a club like Paris Saint-Germain, in search of a replacement for Kylian Mbappé, cannot simply sign the forward with the best goal or assist record. Instead, they must identify a profile with physical and technical attributes that align with Mbappé’s style of play—explosive speed, dribbling ability in tight spaces, and efficiency in finishing during fast transitions. Similarly, Real Madrid, when replacing a midfielder of Toni Kroos’s caliber, would not merely target an internationally renowned player but rather one who possesses elite playmaking skills, exceptional passing accuracy, and the ability to dictate the game’s tempo.

While this approach is intuitive for top clubs, which have virtually unlimited financial and scouting resources, the challenge is significantly greater for smaller clubs. However, for smaller football clubs, the problem becomes more complex. How can a team like Bologna replace young talents such as Ricardo Calafiori or Joshua Zirkzee without a global scouting network? Or how can Genoa, which relies on key players like Retegui and Gudmundsson, find adequate replacements after their departures? Is it enough to rely solely on the subjective judgment of scouts? And how many resources would be necessary to map the global football landscape?

This research is grounded in player data from the most recent complete season (2023/24), processing performance statistics to develop performance indices capable of quantitatively describing players’ abilities. The goal is to develop a user-friendly model that, given an input player, can identify similar players based on their performance indices.

The entire analysis was conducted using Python, a programming language particularly well-suited for this type of research due to its versatility, extensive data science libraries,

and strong support for machine learning applications. Python allows for efficient data processing, model development, and advanced visualization, ensuring that the scouting system remains both scalable and easily interpretable.

The study focuses specifically on outfield players (defenders, midfielders and attackers), while goalkeepers are included in the dataset, their performance evaluation requires a fundamentally different approach. The available data does not offer a sufficient level of detail to conduct an equivalent analysis for goalkeepers. Additionally, replacing an outfield player poses a greater tactical challenge than replacing a goalkeeper, as outfield players rely on dynamism, tactical adaptability, and positional versatility, making direct comparisons significantly more complex.

Finally, the goal of the algorithm developed in this study is not only to provide an objective starting point for evaluations but also to offer scouts a way to optimize their analyses, enabling clubs to reduce efforts and reduce resources required for scouting. In this way, the support of this tool can assist clubs in determining which player to sign during a transfer window, a decision that often not only defines the success of a single season but also contributes to shape unforgettable chapters in the history of football.

1.2 Thesis Structure

The thesis begins with a literature review, aimed at understanding the current state of the art and recent advancements in the application of machine learning to sports analytics. This section provides essential context by examining previous studies, methodologies, and the evolving role of data-driven approaches in football scouting.

Following the literature review, the data collection process is outlined, detailing the methods used to compile the initial dataset of football players. This includes a discussion on data sources, extraction techniques, and the preprocessing steps undertaken to ensure data reliability and completeness. The next phase focuses on data cleaning and manipulation, which are crucial steps in refining the dataset before conducting the core analysis.

A key component of the study is the development of performance indices, which serve as quantitative measures of player abilities. This section explains the rationale behind creating these indices, the methodology used to compute them, and the specific attributes

each index aims to quantify. Additionally, is explored in detail the analytical approach behind how these indices capture different aspects of player performance.

Once the performance indices are established, the next step involves validating their reliability and effectiveness. This phase is essential to ensure that the indices are accurate, representative, and analytically comprehensive, making them suitable for practical application in player scouting.

Following validation, the thesis moves on to the practical implementation of the scouting model, detailing the algorithm's structure, functionality, and deployment. Furthermore, static and interactive visualization tools are introduced to enhance interpretability, allowing scouts and analysts to interact with the results intuitively. These visual elements provide a clear and accessible way to analyze player comparisons and identify potential recruits.

Finally, the research includes several case studies that illustrate the model's performance across different player roles and characteristics. By applying the methodology to real-world examples, the effectiveness of the system is assessed, highlighting its practical value in various scouting scenarios.

CHAPTER 2:
LITERATURE REVIEW

2.1 The Role of Machine Learning and Data Analytics in Sports

In recent years, machine learning and data analytics have completely reshaped the sports industry, changing how teams, analysts, and decision-makers interpret performances, refine strategies, and make key decisions. The ability to process massive amounts of data in real-time has given teams a crucial competitive advantage, improving not only player evaluation but also tactical planning, injury prevention, and overall team management. From football and basketball to tennis and Formula 1, machine learning models have become an essential tool for sports analytics.

Among the most impactful applications of AI and machine learning in sports, several key areas stand out: Performance Analytics: leveraging advanced metrics to assess player contributions beyond conventional statistics; Tactical Analysis: Utilizing spatial data to refine formations, passing networks, and defensive structures for strategic optimization; Injury Prediction and Prevention: Detecting patterns in physical exertion and workload to help reduce injury risks; Referee Decision-Making: Supporting officials with technologies like VAR (Video Assistant Referee), Semi-Automated Offside Technology, and Goal-Line Technology, all of which rely on computer vision to improve accuracy and ensure fair decision-making.

2.2 The Growing Role of Data Science in Football

Football, the world's most popular sport, has been profoundly shaped by data-driven approaches. In the past, scouting and tactical planning relied largely on the subjective assessments of coaches and analysts. However, with the rise of machine learning and big data, clubs now have access to objective, quantifiable insights, allowing for more informed decision-making across multiple areas. In the following are investigated some of the most impactful areas where AI and machine learning have transformed football analytics.

2.2.1 Computer Vision for Tracking Data

One of the most groundbreaking advancements in modern football analytics is the integration of computer vision and optical tracking systems to gather both event and tracking data. Advanced deep learning techniques, particularly YOLO-based object detection models, have greatly enhanced the ability to extract real-time positional and event-based data from match footage. These models enable analysts to track player movements, team formations, and tactical adjustments with high precision, generating heatmaps and movement trajectories to evaluate positional effectiveness.

A key study by Tanapatpiboon et al. (2024) explores how different YOLO implementations (YOLOv5m6, YOLOv5l-tph, YOLOv5l-tph-plus) perform in identifying and tracking player positions from broadcast footage. The study finds that YOLOv5l-tph achieves the highest precision (0.9868), enabling the creation of highly accurate positional heatmaps that can be used by coaches and analysts to understand team and player dynamics. In general, player tracking systems utilize AI models to analyze player positioning, sprint intensity, and movement patterns, offering detailed insights into both individual and team performance. Similarly, ball tracking technologies monitor ball movement in real time, enabling precise evaluation of pass accuracy, shot quality, and ball progression metrics. These advancements are now widely commercialized, with companies like StatsPerform, Second Spectrum, and Tracab utilizing AI-powered vision systems to automatically collect detailed event data. This allows clubs to gain granular insights that have become indispensable for elite teams seeking to optimize performance and refine tactical strategies.

Top-tier football clubs equipped with advanced technological infrastructures nowadays have access even to real-time analytics tools that process match data instantaneously. These systems utilize AI-driven video analysis and tracking technologies to monitor player movements, ball trajectories, and tactical structures in real-time. By integrating this data into their decision-making workflows, coaching staff can make immediate tactical adjustments, refine player positioning strategies, and optimize in-game performance. This provides a significant competitive advantage, allowing coaches and analysts to gain real-time quantitative insights into the opponent's tactical setup during the match.

2.2.2 AI-Driven Player Performance Analysis and Tactical Insights

Once accurate player positioning data and movement data are extracted from match footage, the range of possible analyses becomes virtually limitless. Machine learning plays a crucial role not only in the computer vision processes that generate these coordinates—tracking player positioning, ball movement, and event occurrences—but also in the subsequent analytical models that leverage this data to extract meaningful insights. A particularly impactful examples of AI-driven football analytics are Expected Goals (xG) models. Expected Goals (xG) Models predict the probability of scoring based on multiple contextual factors, such as shot location, shooting angle, defensive pressure, and the type of assist leading to the shot. This metric is widely used to differentiate high-quality finishers from players who rely on a high volume of attempts. The study by Mead et al. (2023) provides a detailed analysis of Expected Goals (xG) models and their role in modern football analytics. The research highlights the impact of various contextual factors—such as shot location, angle, defensive pressure, and game situation—on the probability of a shot resulting in a goal. By integrating machine learning techniques, the study enhances traditional xG models, improving predictive performance and demonstrating the value of this metric in evaluating player and team efficiency.

Another impactful examples of AI-driven football analytics are Passing Networks and Ball Progression Models. These models use AI-generated visualizations to highlight key playmakers, passing structures, and ball distribution trends within a team. These models enable analysts to quantify a player's impact on ball circulation by examining their involvement in progressive passing sequences. By overlaying positional heatmaps with passing networks, teams can identify optimal build-up play strategies, pinpoint areas of congestion, and refine tactical approaches to improve ball movement efficiency. A study by Pei et al. (2023) explores how transformer-based deep learning models can predict passing heatmaps using player tracking data. This approach enhances traditional passing network analyses by incorporating predictive modeling, allowing teams to simulate and anticipate passing patterns in different tactical scenarios. The study demonstrates how AI can optimize passing strategies by forecasting ball circulation trends and identifying key zones of influence on the pitch.

Another impactful application of AI-driven football analytics is the generation of pre-match tactical reports, which provide teams with detailed, AI-generated insights into

opposition strategies. These reports analyze key aspects such as defensive setups, counter-attacking tendencies, and player movement patterns, offering coaching staff a data-driven foundation for match preparation. By leveraging these tactical reports, teams can anticipate an opponent's pressing intensity, passing tendencies, and defensive weaknesses. This information allows managers to develop tailored game plans, adjust formations, and exploit specific tactical vulnerabilities to gain a competitive edge.

2.2.3 AI-Driven Injury Prevention and Physical Performance Monitoring

One of the most innovative advancements in modern football analytics is the integration of machine learning into injury prevention and physical performance monitoring. Given the high intensity demands of professional football, injuries can significantly affect team performance and player careers. Traditionally, injury risk assessment depended on medical evaluations, past injury history, and subjective fatigue measurements. However, with the emergence of AI and big data analytics, teams can now shift from a reactive approach to a proactive one, allowing for early intervention and injury risk mitigation before issues escalate.

The study "Injury Prediction for Soccer Players Using Machine Learning" by Satvedi & Pyne (2022) provides an early example of how machine learning can be applied to predict injury risk in football. The researchers utilized a linear regression model to analyze factors such as minutes played, number of matches played, distance covered, and player position (both game-wise and season-wise), aiming to assess the impact of workload on injury likelihood. Using data from the English Premier League (EPL), the study found a significant correlation between excessive match load and increased injury risk, reinforcing the importance of load management strategies in professional football.

While the model successfully identified a correlation between high match load and increased injury risk, it also presented certain limitations. One major drawback was its reliance solely on match data, excluding crucial training load metrics that contribute significantly to player fatigue and injury susceptibility. Additionally, biomechanical factors such as sprint intensity, acceleration/deceleration forces, and physiological markers (e.g., heart rate variability, muscle fatigue) were not incorporated, limiting the model's ability to provide a holistic view of injury risk.

Building on this foundation, more advanced AI-driven approaches have emerged, integrating wearable technology and biometric tracking to improve injury risk prediction. GPS devices and biometric sensors continuously monitor player movement, heart rate variability, and muscle fatigue, providing real-time physiological data that were previously unavailable. This data allows clubs to identify potential injuries before they occur, enabling preemptive intervention through training modifications and personalized recovery protocols, ultimately reducing injury risks and optimizing player longevity. For instance, a recent study by Freitas et al. (2025) developed an automated injury identification and prediction system using Support Vector Machines (SVMs), Feedforward Neural Networks (FNNs), and Adaptive Boosting (AdaBoost). Conducted on a Portuguese first-division football team, the study leveraged GPS data and player-specific metrics to predict injuries with high accuracy—SVMs achieved a 74.22% accuracy rate. The study identified key predictive factors, such as player position, session type, velocity, acceleration, and player load, demonstrating the multifactorial nature of injuries. Crucially, the findings emphasized that effective injury prevention requires analyzing multiple data points simultaneously rather than relying on isolated metrics. By integrating machine learning-based injury prediction systems with load management strategies, clubs can optimize player availability, reduce recovery times, and minimize the risk of long-term injuries.

A practical example of AI-driven injury prevention in professional football is Liverpool FC, one of the world's most prestigious clubs. Since 2021, Liverpool has collaborated with Zone7, an AI-powered platform that analyzes complex datasets to predict and mitigate injury risks. As reported by iNews, Liverpool's adoption of Zone7 has been described as a "secret weapon" behind Jürgen Klopp's squad selection, playing a crucial role in reducing injuries and optimizing player availability[1]. The implementation of this technology has led to a significant reduction in injuries, helping to keep key players available for crucial matches. According to data from Premier Injuries, Liverpool nearly halved the number of serious injuries compared to the previous season, recording 6.9 injuries per 1,000 minutes played. This result is particularly notable given the club's high-intensity schedule, as Liverpool competed on four different fronts that season. Despite

[1] Read more: https://inews.co.uk/sport/football/liverpool-cut-injuries-secret-weapon-klopp-selection-1761181?srsltid=

the demanding fixture list, Liverpool suffered fewer injuries than 13 of their Premier League rivals, many of whom played significantly fewer matches[2].

2.2.4 Advanced Analytics and AI in Player Scouting and Recruiting

Scouting and recruitment have always been fundamental pillars of success in football, with clubs continuously striving to identify, evaluate, and acquire players who best align with their tactical and strategic objectives. Traditionally, this process has relied heavily on subjective human observation, with scouts traveling worldwide to assess talent based on in-game performances, technical skills, and perceived potential. While this method has led to the discovery of legendary players, it is inherently limited by cognitive biases, inconsistencies in evaluation criteria, and constraints in data collection. The study by Lawlor et al. (2021) explores the evolution of scouting, tracing its progression from traditional observational methods—which relied heavily on scouts' intuition and subjective judgment—to more data-driven and analytical approaches. They highlight how football recruitment is increasingly shaped by performance analysis and statistical modeling, while still recognizing that human expertise remains a crucial element in the decision-making process.

Therefore, the rise of data analytics in football has fundamentally transformed the scouting and recruitment landscape. Clubs now have access to vast databases of player statistics, allowing them to track performance across multiple leagues and competitions. Metrics such as Expected Goals (xG), pressing intensity, passing efficiency, and defensive contributions offer a more objective and quantifiable basis for player evaluation. However, despite these advancements, traditional scouting methods still struggle to process and interpret large-scale data efficiently. Machine learning algorithms addresses these challenges by offering objective, scalable, and data-driven insights into player recruitment. AI-driven scouting models analyze performance indicators without human bias, providing clubs with a wider pool of potential signings based on skill set rather than reputation.

[2] For further insights, you can watch the exclusive interview with Tal Brown, co-founder and CEO of Zone7, where he discusses the impact of their AI technologies on Liverpool's 2021/22 season: https://www.youtube.com/watch?v=RJEYLlx72i0&utm

It is precisely within this domain that the present study seeks to contribute. By leveraging AI methodologies in the scouting and recruitment process, this research aims to enhance the objectivity and efficiency of player identification. The following sections will review the state of the art and existing literature on AI-powered scouting, exploring key methodologies, applications, strengths, and limitations in this rapidly evolving field.

2.3 Machine Learning in Scouting and Recruitment: a Literature Review

Early studies in AI-driven talent identification primarily focused on physical performance metrics. A notable example is the study by Jauhiainen et al. (2019), which applied a one-class Support Vector Machine (SVM) to identify future elite players among 14-year-old junior footballers.

The dataset included 951 Finnish youth players, with only 14 classified as future academy-level talents. Since elite players represent a minority class in talent identification, the authors approached the problem as an anomaly detection task, training the model on general player data and testing its ability to distinguish rare elite profiles. The model achieved an AUC-ROC score of 0.763, with the best results obtained when incorporating physical test scores such as speed, agility, and technical skills.

While innovative in its application of unsupervised learning, the study's reliance on static physical attributes—such as sprint speed (10m, 20m, 30m times), agility tests, and countermovement jump height—without considering tactical or contextual factors, limited its real-world applicability to scouting. These metrics, while valuable for assessing raw athletic ability, fail to capture positional intelligence, decision-making, or adaptability within a tactical system. For example, two midfielders with identical sprint speeds might perform entirely different roles in a game—one excelling in pressing and ball recovery, while the other operates as a deep-lying playmaker controlling the tempo. The absence of game impact metrics—such as passing, shooting, tackling in efficiency and accuracy—restricts the model's ability to evaluate a player's true contribution within a team's tactical framework.

A step forward was made by Hashir Sayeed (2023), who proposed a two-stage machine learning framework designed to automate player evaluation by predicting both the optimal position of a player and their value. The study utilized FIFA player statistics, comprising 74 variables related to technical, physical, and economic attributes of

professional footballers. This included passing accuracy, dribbling ability, defensive contributions, speed, acceleration, market value, and wages. Player roles were reduced to nine major categories: Striker (ST), Midfielder (MF), Center Attacking Midfielder (CAM), Center Defensive Midfielder (CDM), Center Midfielder (CM), Winger (WN), Center Back (CB), Defender (DF), and Goalkeeper (GK).

The first stage of the framework applied a classification model (Artificial Neural Network) to predict the optimal position of a player based on their performance metrics. The classification performance varied significantly across positions, achieving an overall accuracy of 72%. The second stage introduced a regression model aimed at predicting a player's market value based on their attributes. This model also used an ANN, with the Mean Squared Error (MSE) and Mean Absolute Error (MAE) as evaluation metrics. The results showed: MSE = 0.63, indicating reasonable predictive capability.

While this framework introduced a valid automated pipeline for scouting and valuation, it presented some limitations. The classification model assigned players to pre-defined roles rather than identifying similar profiles based on data-driven similarities. This static role-based approach contrasts with modern scouting needs, where clubs seek players who fit specific tactical requirements rather than rigid positional labels. Additionally, the study did not account for the competitive difficulty of different leagues, failing to normalize performance metrics across varying levels of play. Lastly, interpretability and user experience were overlooked, as the framework lacked visual tools to help scouts compare players intuitively.

Another contribution in data-driven football scouting was introduced by Abhinav et al. (2024), who developed a scouting system based on Expected Goals (xG) and Expected Assists (xA). This approach utilized regression models and geometric distance calculations to evaluate attacking efficiency, aiming to identify undervalued offensive players whose underlying performance metrics exceeded their actual goal contributions. The study employed historical match data from multiple leagues to quantify the quality of scoring opportunities and passing effectiveness, rather than relying solely on raw goal and assist counts.

The xG metric estimates the probability of a shot resulting in a goal, taking into account factors such as shot location, angle, type of assist, and defensive pressure. Similarly, xA measures the likelihood of a pass leading to a goal-scoring opportunity, providing a more

nuanced assessment of creative players beyond traditional assist statistics. By integrating these metrics into a predictive ranking system, the model highlighted forwards and attacking midfielders who consistently generated high-quality scoring chances, even if their actual goal tally was lower due to poor finishing or tactical constraints.

While this methodology represented a significant advancement over traditional scouting approaches based on raw goal and assist counts, it also presents limitations and thus opportunities for further refinement. One of the key challenges is its limited applicability to defensive and midfield roles, as xG and xA are primarily designed to evaluate attacking contributions. However, football performance encompasses a broader set of attributes, including defensive positioning, pressing intensity, tactical adaptability, and ball progression, which are not explicitly captured by xG-based models. Expanding the framework to integrate additional performance metrics could enhance its ability to evaluate players across all positions. Moreover, the study does not currently adjust for league difficulty, meaning that xG values from different competitive contexts are treated equivalently. This could lead to variations in player evaluation, as the same xG value might hold different tactical and technical significance depending on the level of competition. Incorporating league difficulty adjustments could further refine the system's accuracy and cross-league comparability.

A notable recent advancement in data-driven football scouting is the Spatial Similarity Index (SSI) proposed by Gómez-Rubio et al. (2024), which leverages spatial and contextual data to compare players based on their on-field movement patterns This approach differs from traditional scouting metrics by focusing not on isolated performance statistics but on how players occupy space throughout a match. The study utilized tracking data from La Liga's 2019-2020 season, containing positional heatmaps that recorded player locations over time. The spatial information was transformed into a grid-based representation of the pitch, dividing the field into small cells where the time spent by each player was quantified. Using clustering techniques, the authors created a SSI, enabling teams to identify players with comparable positioning tendencies. This method allowed scouts to search for players who exhibit similar movement patterns, facilitating role-based recruitment by finding individuals who exhibit comparable spatial behaviors, (independent of club affiliation or tactical system) rather than relying solely on conventional performance statistics such as goals, assists, or successful tackles. The

study utilizes tracking data from La Liga's 2019-2020 season, incorporating positional heatmaps that capture player movements across different zones of the pitch. The methodology involves segmenting the field into a fine-grained grid system, where each cell represents the amount of time a player spends in a specific area.

While the SSI model represents a significant advancement in scouting methodologies, offering a novel way to analyze positional intelligence, there remain opportunities for further refinement. The model's primary strength lies in its ability to identify players with comparable movement patterns, yet it focuses exclusively on spatial positioning, without incorporating technical execution, decision-making under pressure, or tactical adaptability. As a result, two players with similar heatmaps may still exhibit distinct levels of effectiveness in their respective roles, depending on their technical proficiency and in-game intelligence. Additionally, even in this case, the study does not yet account for the relative difficulty of different leagues, treating all movement patterns as equivalent, regardless of variations in tactical complexity and competitive intensity. Integrating these additional layers of analysis could further enhance the model's applicability across diverse scouting contexts.

2.4 Bridging the Gaps in Football Scouting: Towards a More Effective Player Identification

In contrast to previous studies, the present work aims to introduce a comprehensive, customizable, and context-aware approach to football scouting. A central innovation of this study is the development and validation of performance indices, designed to quantitatively assess a player's abilities based on match-recorded performance statistics throughout the season. These indices are weighted according to league difficulty, ensuring cross-league comparability. Traditional scouting models often fail to account for the impact of competitive level on player performance, treating statistics from different leagues as equivalent, despite notable differences in tactical intensity and quality of opposition. By incorporating league difficulty adjustments, this research wishes to provide a more accurate and standardized assessment of player capabilities, enabling clubs to identify talent capable of transitioning effectively across different competitive environments.

A key methodological objective of this study is the development of a Machine Learning-based similarity model to facilitate dynamic player comparisons. Rather than relying on traditional classification models, which rigidly assign players to predefined positional categories, this approach analyzes multi-dimensional performance profiles, identifying statistical similarities in key attributes regardless of a player's hyper-specific position label. This adaptability is particularly valuable in modern tactical systems, where positional flexibility and role-specific contributions often outweigh conventional role classifications. By leveraging this model, the study seeks to enhance scouting efficiency, allowing clubs to identify players with comparable playing styles and performance outputs, ultimately providing a more practical and adaptable tool for real-world recruitment decisions.

Beyond these methodological advancements, this study also wants to address a critical challenge in machine learning-based scouting: interpretability. Advanced statistical models often lack transparency, making their insights difficult to translate into actionable decisions for scouts, analysts, and coaches. To overcome this limitation, this research integrates multiple visualizations that facilitate intuitive player comparisons, presenting comprehensive performance profiles in an accessible format. These visual tools enable users to explore player similarities, compare key performance indices, and assess tactical fit. By prioritizing user-friendliness and customization, this study ensures that data-driven scouting methodologies are not only statistically robust but also practically applicable to club decision-making processes.

CHAPTER 3:
DATA COLLECTION METHODOLOGY AND DATASET STRUCTURE

3.1 Data Collection Tool: The API

3.1.1 Introduction to the API

Data collection for this study was conducted using an API (Application Programming Interface). According to IBM,

"APIs simplify and accelerate application and software development by allowing developers to integrate data, services, and capabilities from other applications, instead of developing them from scratch".

The use of APIs has proven crucial in ensuring efficient and fast access to large volumes of data from external sources, making them an indispensable tool for data-driven research and analysis projects.

APIs operate through standard requests based on HTTP (HyperText Transfer Protocol), where users can specify certain parameters to obtain structured responses, typically provided in JSON (JavaScript Object Notation) format. This format is particularly suited for automated data processing using programming languages such as Python, enabling seamless integration between the API and the analytical tools employed.

3.1.2 API-Football

The API used for this project was *API-Football*[3], accessible through the *RapidAPI* platform. This tool was selected for its ease of use, the availability of detailed documentation, and the offer a free plan that allows testing its functionalities. Additionally, API-Football stands out for the breadth and quality of the information provided, covering details related to teams (both club and national), leagues, stadiums, coaches, transfers, trophies, betting odds, and most importantly, data on individual

[3] https://www.api-football.com.
The official API documentation, accessible at the following address: https://www.api-football.com/documentation-v3, provides detailed information on the available functionalities, including authentication methods, request types, and implementation examples.

players. For the specific purpose of this study, data related to players were extracted, including their personal details and, most importantly, their performance statistics.

3.1.3 Description of the Data Collection Process

In the specific context of this study, interaction with the API-Football takes place by GET requests, designed to extract pre-existing data from the provider's database. These requests allow access to structured information dynamically, depending on the parameters specified by the user in the query. Below are the main parameters used in this interaction:

- The *API URL* to fetch data from: https://api-football-v1.p.rapidapi.com/v3/players [4]
- *Authentication Keys*: The host key and the personal key. To access the API, an authorization key (API key) is required, provided by the service. This key, unique to each user, ensures controlled access to the data, allowing the provider to monitor usage and enforce any request limits.
- *League ID*: Specifies the league for which player data is to be retrieved.
- *Season*: The year of the season of interest.
- *Page*: The data for each league is divided into multiple pages. To access all available information, an iterative while-loop was implemented, which cycles through the pages until all available data has been retrieved.

3.1.4 API Call Output

As anticipated, the response provided by the server through the API is returned in JSON format, a structured, readable, and widely used format that facilitates data processing using programming languages such as Python (adopted in this work). This format represents data as a nested set of key-value pairs, making it easy to extract, manipulate, and analyze information. Below is an example of a JSON response related to the player Jude Bellingham:

[4] The */player* endpoint specifies the need to access data related to individual players. Similarly, the */leagues* endpoint allows for retrieving information about competitions, while */venues* provides details on specific stadiums. This URL structure reflects the modularity of the API, making it easier to extract targeted data across different available categories.

{"get":"players","parameters":{"league":"140","page":"38","season":"2023"},"error
s":[],"results":20,"paging":{"current":38,"total":45},"response":[{"player":{"id"
:129718,"name":"J.Bellingham","firstname":"Jude Victor William",
"lastname":"Bellingham","age":21,"birth":{"date":"2003-06-29",
"place":"Stourbridge","country":"England"},"nationality":"England","height":"186
cm","weight":"75 kg","injured":false, "photo":"https:\/\/media.api-
sports.io\/football\/players\/129718.png"},
"statistics":[{"team":{"id":541,"name":"Real Madrid",
"logo":"https:\/\/media.api-
sports.io\/football\/teams\/541.png"},"league":{"id":140,"name":"La
Liga","country":"Spain",
"logo":"https:\/\/media.api-
sports.io\/football\/leagues\/140.png","flag":"https:\/\/media.api-
sports.io\/flags\/es.svg","season":2023},"games":{"appearences":28,"lineups":27,"
minutes":2324,"number":null,"position":"Midfielder","rating":"8.035714","captain"
:false},"substitutes":{"in":1,"out":9,"bench":5},"shots":{"total":49,"on":35},"go
als":{"total":19,"conceded":0,"assists":6,"saves":null},"passes":{"total":1500,"k
ey":48,"accuracy":48},"tackles":{"total":43,"blocks":5,"interceptions":21},"duels
":{"total":345,"won":189},"dribbles":{"attempts":85,"success":50,"past":null},"fo
uls":{"drawn":72,"committed":32},"cards":{"yellow":5,"yellowred":0,"red":1},"pena
lty":{"won":null,"commited":null,"scored":1,"missed":0,"saved":null}}]}

3.2 Dataset Creation

3.2.1 Data Scope Definition: Analysis Based on the Top 10 UEFA Leagues

For this study, the analysis focused on the top 10 European leagues based on the UEFA ranking[5]. This ranking is calculated based on the UEFA coefficient[6], a score assigned to leagues according to the performance of their clubs in European competitions (UEFA Champions League, UEFA Europa League, and UEFA Europa Conference League) over a five-season period. The primary objective of the UEFA ranking is to assess the overall quality and competitiveness of national leagues, determining the number of teams each league federation can qualify for UEFA competitions. According to such rankings, the top 10 leagues analyzed in this study are:

1. Premier League – England
2. Serie A – Italy
3. La Liga – Spain
4. Bundesliga – Germany

[5] https://www.uefa.com/nationalassociations/uefarankings/country/?year=2025

[6] For more information on the UEFA coefficient and its calculation methodology, please refer to: https://it.uefa.com/nationalassociations/uefarankings/country/about/#

5. Ligue 1 – France
6. Eredivisie – Netherlands
7. Primeira Liga – Portugal
8. Belgian Pro League – Belgium
9. Czech First League – Czech Republic
10. Super Lig – Turkey

3.2.2 Dataset Creation Procedure

During the API request phase, it was necessary to download player data for each league separately, following the pagination system provided by the API. For each league, the extracted data was saved in separate JSON files, each corresponding to a page of the output. Subsequently, to facilitate data usage and processing, these files were merged into a single JSON file per league, resulting in 10 complete JSON datasets (one for each analyzed league).

To further simplify data processing and analysis in Python, the JSON files generated for each league were converted into CSV (Comma-Separated Values) format. This transformation was essential because CSV offers significant advantages in terms of efficiency, readability, and ease of use for data analysis applications. The CSV file format is a common textual representation for tabular data. Each line in the file corresponds to a row in the table, and columns are separated by commas, allowing for simple and portable data exchange (Shafranovich, 2005).

This format is widely used for transferring and storing large datasets due to its structural simplicity and broad compatibility with programming languages such as Python, as well as the libraries needed for subsequent data processing and manipulation.

3.2.3 Dataset Definition and Variable Descriptions

To facilitate global analysis and reduce complexity in data management, the 10 CSV files generated for each league were concatenated into a single dataset. This approach allowed for the creation of a consolidated tabular structure, where each row represents a player, and the columns describe their personal, physical, and performance-related attributes during the season.

Below is the structure of the dataset along with the description of its variables:

a) Columns related to personal and demographic information:
   - `player_id`: Unique identifier of the player.
   - `name`: Full name of the player.
   - `firstname`: Player's first name.
   - `lastname`: Player's last name.
   - `age`: Player's age.
   - `birth_date`: Player's date of birth.
   - `birth_place`: Place of birth.
   - `birth_country`: Country of birth.
   - `player_nationality`: Player's nationality.

b) Columns related to physical attributes:
   - `height`: Player's height (in cm).
   - `weight`: Player's weight (in kg).
   - `player_photo`: Link to the player's profile picture.

c) Columns related to the team:
   - `team_id`: Unique identifier of the team.
   - `team_name`: Name of the team.
   - `team_logo`: Link to the official team logo.

d) Columns related to the league:
   - `league_id`: Unique identifier of the league.
   - `league_name`: Name of the league.
   - `league_country`: Country of the league.
   - `league_logo`: Link to the official league logo.
   - `league_flag`: Link to the flag of the league's country.
   - `league_season`: Football season of reference.

e) Columns related to performance statistics:

Appearances:

- `games_appearences`: Total number of matches played.
- `games_lineups`: Number of matches started as a starter.
- `games_minutes`: Total minutes played.
- `games_position`: Position on the field (goalkeeper, defender, midfielder, forward).
- `games_rating`: Average performance rating (scale from 0 to 10).

Substitutions:

- `substitutes_in`: Appearances as a substitute.
- `substitutes_out`: Times substituted off.
- `substitutes_bench`: Times included in the squad without playing.

Shots:

- `shots_total`: Total number of shots.
- `shots_on`: Shots on target.

Goals:

- `goals_total`: Total goals scored.
- `goals_conceded`: Goals conceded (for goalkeepers).
- `goals_assist`: Assists provided.
- `goals_saves`: Saves made (for goalkeepers).

Passes:

- `passes_total`: Total number of passes.
- `passes_key`: Key passes[7] made.
- `passes_accuracy`: Pass accuracy (percentage).

Tackles:

[7] A passing action that surpasses an opposing pressure line, creating a significant advantage for the team in the offensive phase. This definition emphasizes the concept of a "pressure line," which refers to the structured positioning of opposing players aimed at hindering the ball's progression.

- `tackles_total`: Total tackles made.
- `tackles_blocks`: Tackles that successfully blocked the opponent's play.
- `tackles_interceptions`: Ball interceptions made.

Duels:

- `duels_total`: Total duels contested.
- `duels_won`: Duels won.

Dribbling:

- `dribbles_attempts`: Dribbling attempts.
- `dribbles_success`: Successful dribbles.

Fouls:

- `fouls_drawn`: Fouls suffered.
- `fouls_committed`: Fouls committed.

Cards:

- `cards_yellow`: Yellow cards received.
- `cards_yellowred`: Red cards due to a second yellow.
- `cards_red`: Direct red cards.

Penalties:

- `penalty_scored`: Penalties scored.
- `penalty_missed`: Penalties missed.
- `penalty_saved`: Penalties saved (for goalkeepers).

CHAPTER 4:
DATA PREPARATION: CLEANING, SORTING AND GENERATION OF DERIVED VARIABLES

4.1 Data Cleaning

4.1.1 Data Manipulation

After generating the final dataset file, it was loaded and manipulated for subsequent analyses. For this process, the Pandas library was used, a widely recognized tool for data manipulation and analysis due to its flexibility, extensive set of built-in functions ability to process large datasets. Pandas allows operations on data structures called DataFrames, which are defined as two-dimensional labeled data tables with rows and columns, designed to simplify tasks such as filtering, aggregation, and data transformation[8].

At this stage, certain columns were excluded from the DataFrame as they did not provide relevant information for the analysis. Specifically, the variables `games_number`, `games_captain`, `penalty_won`, `penalty_committed`, `player_injured`, and `dribbles_past` were removed, as they contained only NaNs[9] or other unusable values. Then, variables were converted to numeric formats (whenever possible) to ensure consistency and make them suitable for use in the model.

4.1.2 Removing Duplicate Records

An essential step in dataset preparation was the removal of duplicate observations. To identify duplicates, the `player_id` variable was used, as it serves as the unique identifier assigned to each player.

The issue of duplicates primarily arose in cases where players were transferred during the summer transfer window before the start of the season. In these instances, the same player appeared twice in the dataset: once associated with their previous team, with performance

[8] For more information, see *https://docs.databricks.com/en/getting-started/dataframes.html* and *https://www.databricks.com/glossary/what-are-dataframes*

[9] NaN is an acronym for "Not a Number," indicating the absence of a defined numerical value for such variable(s).

statistics set to zero (as they had not played any matches), and once with their current team, containing actual recorded performance data. To address this issue, the dataset was first sorted by `player_id`, allowing duplicate entries to be displayed next to each other. Subsequently, a second sorting criterion was applied in descending order based on the variable `games_appearances`, which represents the number of matches played by each player. This second step ensured that, for each player, the record containing the most relevant data was identified. Below in *Figure 1* is an example with a subset of players.

player_id	name	birth_date	player_nationality	team_name	league_name	games_position	games_appearences	games_minutes
907	R. Lukaku	1993-05-13	Belgium	AS Roma	Serie A	Attacker	32.0	2648.0
907	R. Lukaku	1993-05-13	Belgium	Chelsea	Premier League	Attacker	0.0	0.0
1454	M. Guendouzi	1999-04-14	France	Lazio	Serie A	Midfielder	33.0	2371.0
1454	M. Guendouzi	1999-04-14	France	Marseille	Ligue 1	Midfielder	2.0	17.0
2802	Y. Sommer	1988-12-17	Switzerland	Inter	Serie A	Goalkeeper	34.0	3060.0
2802	Y. Sommer	1988-12-17	Switzerland	Bayern München	Bundesliga	Goalkeeper	0.0	0.0
2897	Kim Min-Jae	1996-11-15	Korea Republic	Bayern München	Bundesliga	Defender	25.0	1969.0
2897	Kim Min-Jae	1996-11-15	Korea Republic	Napoli	Serie A	Defender	0.0	0.0
36902	T. Reijnders	1998-07-29	Netherlands	AC Milan	Serie A	Midfielder	36.0	2829.0
36902	T. Reijnders	1998-07-29	Netherlands	AZ Alkmaar	Eredivisie	Midfielder	0.0	0.0

Figure 1 - Example subset of duplicate records

During the duplicate removal process, only the first record, corresponding to the one with the highest number of appearances, was retained.

The presence of duplicates could have introduced significant errors in subsequent analyses. In particular, during the calculation of player similarities, the inclusion of duplicate records could have led to unreliable results, negatively affecting the model's effectiveness. Therefore, removing duplicates was a crucial step in ensuring the consistency and accuracy of the dataset, providing a reliable foundation for the next phases of the study.

4.2 Generation of Additional Variables

By leveraging the variables already present in the dataset, it was possible to create new features capable of providing useful information that was not previously available. These new variables were designed to enrich the analysis and improve the predictive capabilities of the model that will be developed.

Below is a detailed description of the newly created variables, including their calculation and meaning:

- goals_per_game: This variable represents the number of goals scored by a player per match played. It was calculated by dividing the total goals scored (goals_total) by the total matches played (games_appearances). This metric helps assess a player's scoring efficiency in relation to the number of matches played.
- assists_per_game: similar to goals_per_game, this metric measures the number of assists provided by a player per match played. It was calculated by dividing the total assists (goals_assists) by the total matches played (games_appearances). This variable indicates a player's effectiveness in creating goal-scoring opportunities for teammates in relation to the number of matches played.
- shots_accuracy: This variable represents a player's shooting precision, calculated as the ratio of shots on target (shots_on) to total shots attempted (shots_total). This metric serves as an indicator of a player's ability to direct their shots toward the goal.
- dribbles_accuracy: This variable represents the percentage of successful dribbles, calculated as the ratio of successful dribbles (dribbles_success) to the total dribble attempts (dribbles_attempts). This metric is crucial for assessing a player's ability to successfully bypass opponents through dribbling.
- duels_accuracy: This variable measures a player's efficiency in duels, calculated as the ratio of duels won (duels_won) to total duels contested (duels_total). This metric helps assess a player's effectiveness in one-on-one situations.
- penalty_accuracy: This variable measures a player's precision in penalty kicks, calculated as the ratio of penalties scored (penalty_scored) to the total penalties taken (sum of penalty_scored and penalty_missed). This metric evaluates a player's accuracy from the penalty spot and, consequently, their reliability in converting penalties.

CHAPTER 5:
FEATURE ENGINEERING:
CONSTRUCTION OF PERFORMANCE INDICES

5.1 Development of Performance Indices

5.1.1 Purpose of Creating Performance Indices

The objective of this phase of the study is to construct performance indices capable of synthesizing multiple statistical variables into clearer and more representative indicators. These indices are designed to assess a player's abilities in specific aspects of the game while providing a comprehensive overview of their technical and tactical characteristics.

The need to create such indices arises from the necessity to aggregate related and complementary variables, weighting them appropriately to generate new features that encapsulate a player's overall ability within a particular context or playing style. For instance, variables such as `passes_total`, `passes_key`, and `passes_accuracy` can be combined to develop a synthetic index that measures a player's overall effectiveness in ball possession and playmaking.

Beyond simplifying data representation, these indices offer a crucial practical advantage: they enable direct player comparisons by condensing their skills into easily interpretable metrics. With these indices, it becomes easier to assess similarities between players, identify profiles that match specific tactical needs, and compare performances within the same position.

The aggregation of related variables into a single index has the following objectives:

- Reduces dataset dimensionality: an excessive number of variables can increase computational complexity and the risk of overfitting in machine learning models. Constructing synthetic indices allows for a more compact and relevant representation of information.
- Captures latent relationships between variables: certain player abilities cannot be fully described by a single variable. By combining multiple characteristics, indices can provide a more comprehensive measure of a player's skill set.

- Enhances predictive model performance: engineered features that capture meaningful relationships between variables enable machine learning models to better learn underlying patterns in the data, improving generalization capabilities.

5.1.2 Index Calculation Method

To compute the performance indices, a weighted approach was adopted, assigning different weights to relevant variables based on their ability to represent the specific skill being measured. To each variable was assigned a specific weight, determined according to its relative importance in describing a particular aspect of the game. For instance, in a playmaking index, `passes_accuracy` might be assigned a higher weight than `passes_total`, as pass precision is often more indicative of a player's ability to create and control the flow of play.

Once the indices were computed, they were normalized on a scale from 0 to 100 to ensure that the results were easily analyzable by the model. The value 100 was assigned to the player with the best performance within their position for the specific skill being measured, while 0 was attributed to the lowest performance. This approach allows for effective player comparisons, clearly highlighting those who excel and those who rank in the lower end of the scale.

However, it was crucial to calculate the indices separately for each position (goalkeepers, defenders, midfielders, and forwards). The rationale behind this choice lies in the significant differences in the characteristics and responsibilities required for players of different positions. For instance, variables such as goals, assists, and shots are typically much higher for forwards than for defenders. If these variables had been normalized together without distinguishing positions, the values for defenders would have been artificially compressed. In other words, the normalization scale would have been dominated by the extreme values of forwards, making it difficult to distinguish individual performances among defenders. Conversely, some forwards are particularly effective in defensive contributions, excelling in high pressing and ball recovery, with strong performances in statistics like duels won or tackles made. However, these contributions could be overshadowed if such variables were normalized alongside those of defenders, who naturally record higher numbers in these categories.

Therefore, each position has its own indices, which are calculated and normalized separately. This means that one player can achieve a score of 100 in playmaking as a defender, another can reach 100 as a midfielder, and another can obtain 100 as a forward. This approach aligns with the objective of this study, which is to compare players within the same overall position, ensuring consistent and meaningful comparisons. As a matter of fact, comparing a defender to a forward would be irrelevant, as the two positions require entirely different skills and responsibilities.

5.1.3 Adjusting Performance Indices for League Difficulty Level

In international football, scoring 20 goals in an elite league such as the Premier League or Serie A is a far greater challenge than achieving the same feat in the Czech Liga. This disparity stems from the quality level of each league, which is largely influenced by the financial resources available to clubs. Higher investments allow teams in top-tier leagues to acquire world-class players, thereby increasing both the competitive intensity and the tactical complexity of the game. As a result, it is crucial to account for league degree of difficulty when analyzing individual performances. Comparing two players without considering their competitive context can lead to misleading conclusions. For instance, a striker competing against the world's best defenders in the Premier League does not face the same level of difficulty as one playing in a less competitive league, even if both performances are impressive in their own.

As anticipated in Chapter 3, a useful method for assessing the relative difficulty of leagues is the use of the UEFA ranking and its coefficients.

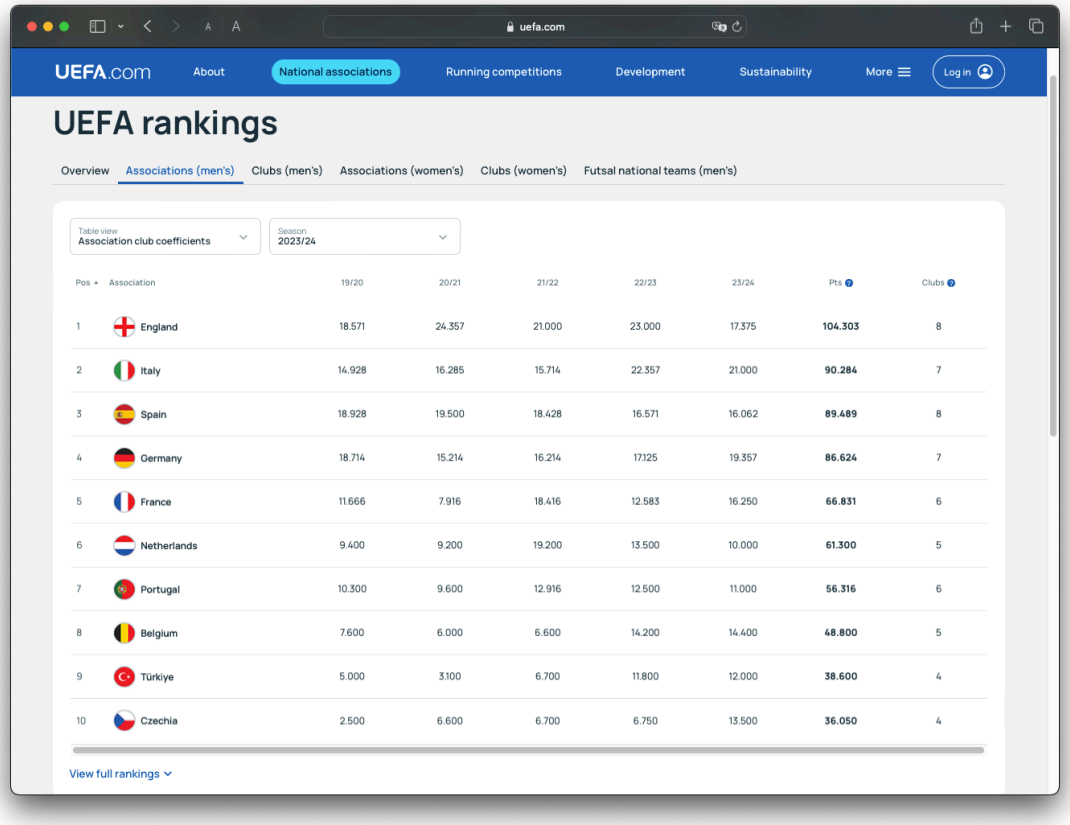


Figure 2 - UEFA rankings (up to season 2023/24)

In this study, various approaches were explored to incorporate UEFA coefficients into the calculation of performance indices. Initially, the computed indices were weighted directly by the UEFA coefficient of the respective league. However, this method overly favored players from top-tier leagues (particularly the Premier League, which ranks first), while unfairly disadvantaging exceptional talents competing in less prestigious leagues. In practice, this approach would have created an imbalance in the opposite direction.

A more balanced solution was to integrate the UEFA coefficient as an additional variable in the calculation of performance indices. For example, in computing the playmaking index, alongside weighted variables such as key passes, total passes, and other relevant metrics, the UEFA coefficient of the respective league was also included by adding it to the other variables.

Finally, to prevent excessive influence of this variable, the UEFA coefficient was divided by 10. For example, as shown in *Figure 2*, the Premier League's overall coefficient of 104.303 would be adjusted to a value of 10.4303. This adjustment ensures that the UEFA

coefficient contributes proportionally to the calculation, without overshadowing other crucial aspects of individual performance. This allows for the recognition of achievements in highly competitive leagues while still preserving the value of talented players in less competitive leagues.

The next section will provide a detailed analysis of all the constructed indices, offering an in-depth explanation of their structure and the variables that compose them.

5.2 In-Depth Performance Indices

5.2.1 Overall Offensive Strength Index

The Overall Offensive Strength Index was developed to quantify a player's overall effectiveness in the offensive phase. A set of variables describing different aspects of attacking play was combined. As previously mentioned, the formula involves the weighted sum of these variables, adjusting the contribution of each parameter to balance both the quantity and quality of performances.

In particular, goals scored (excluding penalties) form a fundamental part of the index, as they reflect a player's ability to convert chances into goals. Penalties are subtracted from the total goal count to avoid double counting, but they are still included with a specific weight to reward successful conversions and penalize missed attempts. Shooting accuracy, measured through shots on target and accuracy percentage, is given significant weight to highlight players who can consistently turn opportunities into dangerous scoring attempts. Duels won are also included in the index, although with a lower weight, to reflect the importance of physical contribution without overly penalizing players who excel primarily in technical skills. As previously mentioned, the UEFA coefficient is also incorporated to account for the difficulty level of the league in which the player competes. Below is the mathematical breakdown of the calculation and the weighting of each component:

$$
\begin{aligned}
&\text{Overall Offensive Strength Index} \\
&\quad = \big((\text{goals\_total} - \text{penalty\_scored}) \cdot 1\big) + (\text{shots\_total} \cdot 0.25) \\
&\quad + (\text{shots\_on} \cdot 0.5) + (\text{shots\_accuracy} \cdot 1.5) \\
&\quad + (\text{penalty\_scored} \cdot 1) - (\text{penalty\_missed} \cdot 0.75) \\
&\quad + (\text{duels\_won} \cdot 0.1) + (\text{goals\_per\_game} \cdot 0.75) \\
&\quad + \text{uefa\_coefficient\_value}
\end{aligned}
$$

Since this index is specifically designed to measure offensive performance, it is particularly relevant to analyze how the calculated scores are distributed among midfielders and attackers. To visualize this distribution, two separate bar plots were created (*Figure 3* and *Figure 4)*, representing the index values for the top 20 attackers and the top 20 midfielders.

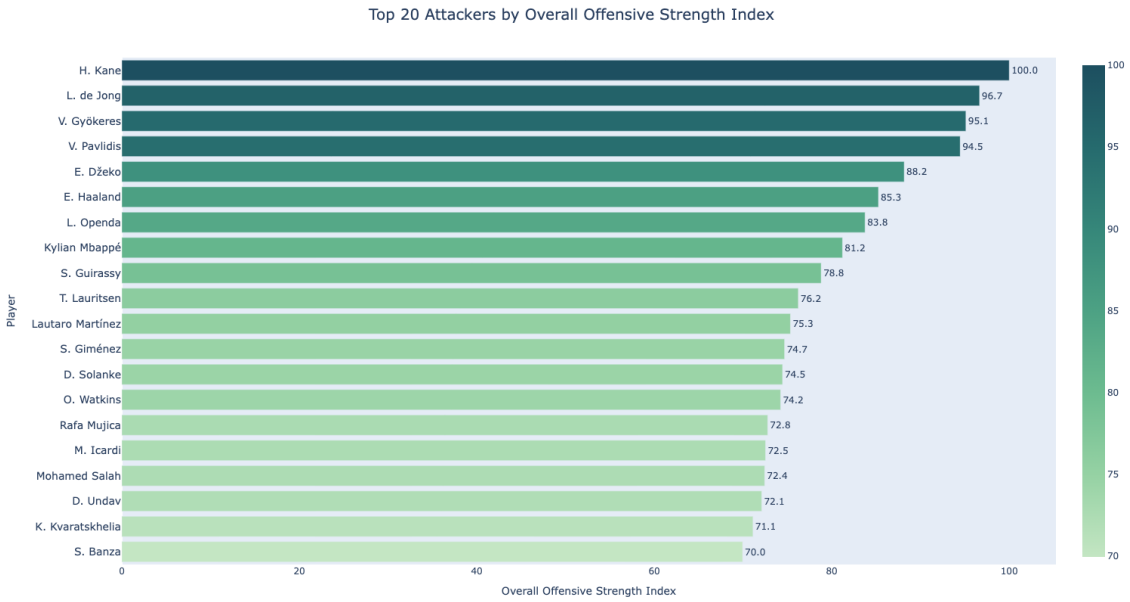


Figure 3 – Top 20 Attackers by Overall Offensive Strength Index

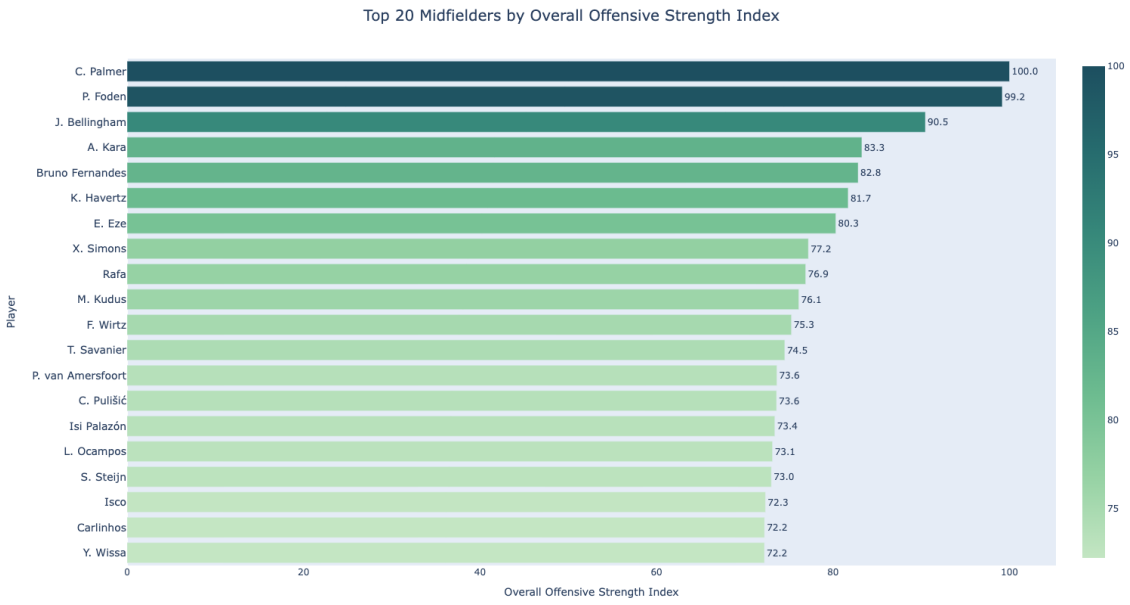


Figure 4 - Top 20 Midfielders by Overall Offensive Strength Index

The visualizations presented, as well as those that will be shown in this chapter, were implemented using Plotly, which enables interactive graph exploration. By hovering over the chart and the bar of interest, users can view detailed information about the player, including their team, league, and individual statistics related to the index. Below is an example (*Figure 5* and *Figure 6*).

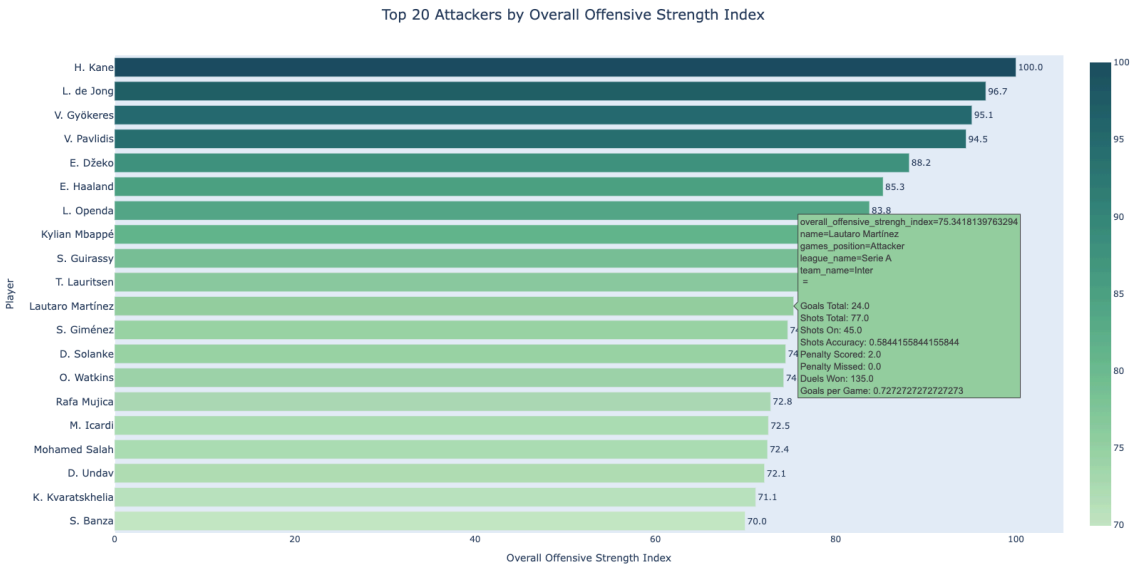


Figure 5 - Top 20 Attackers by Overall Offensive Strength Index - Hovering on Results

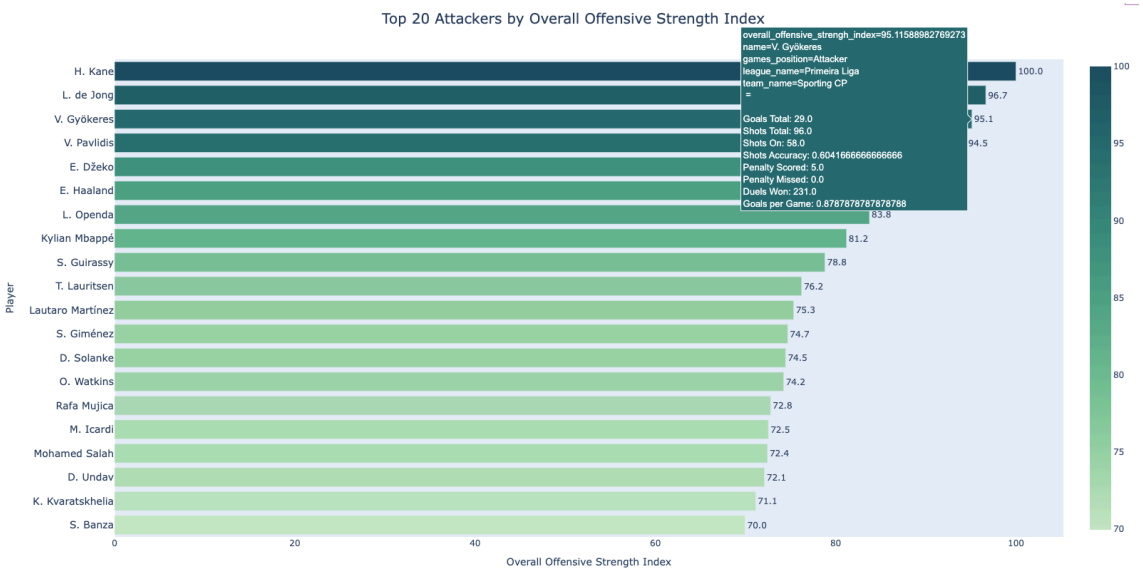


Figure 6 - Top 20 Attackers by Overall Offensive Strength Index - Hovering on Results

5.2.2 Overall Defensive Strength Index

The Overall Defensive Strength Index was developed to measure a player's overall effectiveness in defensive play, combining various variables that capture the key components of defensive performance.

The formula assigns greater weight to variables directly related to defensive play, such as total tackles, blocks, and interceptions. These variables directly reflect a player's ability to disrupt opposing plays. Duels won also play a key role in the index, carrying significant weight to emphasize the importance of physical contribution and competitiveness in one-on-one challenges. The quality of ball possession is also considered, incorporating parameters such as pass accuracy, total passes, and key passes. These elements reflect a player's ability to manage the ball and contribute to build-up play, even from the defensive phase. In modern football, defenders are no longer expected to solely defend, but also to initiate play from the back. At the same time, negative variables such as fouls committed, and cards received (yellow, double yellow, and direct red) negatively impact the index. The penalty applied is proportional to the severity of the card: yellow cards have a minor effect, while double yellow and direct red cards carry increasing weight. Finally, as in the previous indices, the UEFA coefficient has also been included.

The mathematical formula used to calculate the Overall Defensive Strength Index is as follows:

$$
\begin{aligned}
&\text{Overall Defensive Strength Index} \\
&\quad = (\text{tackles\_total} \cdot\ 1) + (\text{duels\_won} \cdot 1) \\
&\quad + (\text{passes\_accuracy} \cdot 0.5) + (\text{passes\_total} \cdot 0.1) \\
&\quad + (\text{passes\_key} \cdot 0.3) + (\text{tackles\_blocks} \cdot 1.75) \\
&\quad + (\text{tackles\_interceptions} \cdot 1.75) - (\text{fouls\_committed} \cdot 0.2) \\
&\quad - (\text{cards\_yellow} \cdot 0.25) - (\text{cards\_yellowred} \cdot 0.5) \\
&\quad - (\text{cards\_red} \cdot 0.75) + (\text{uefa\_coefficient\_value})
\end{aligned}
$$

Since this index is primarily defensive, it is more relevant to analyze how the calculated scores are distributed among defenders and midfielders. As in the previous case, two separate bar plots were generated, representing the index values for the top 20 defenders and top 20 midfielders (*Figure 7* and *Figure 8*).

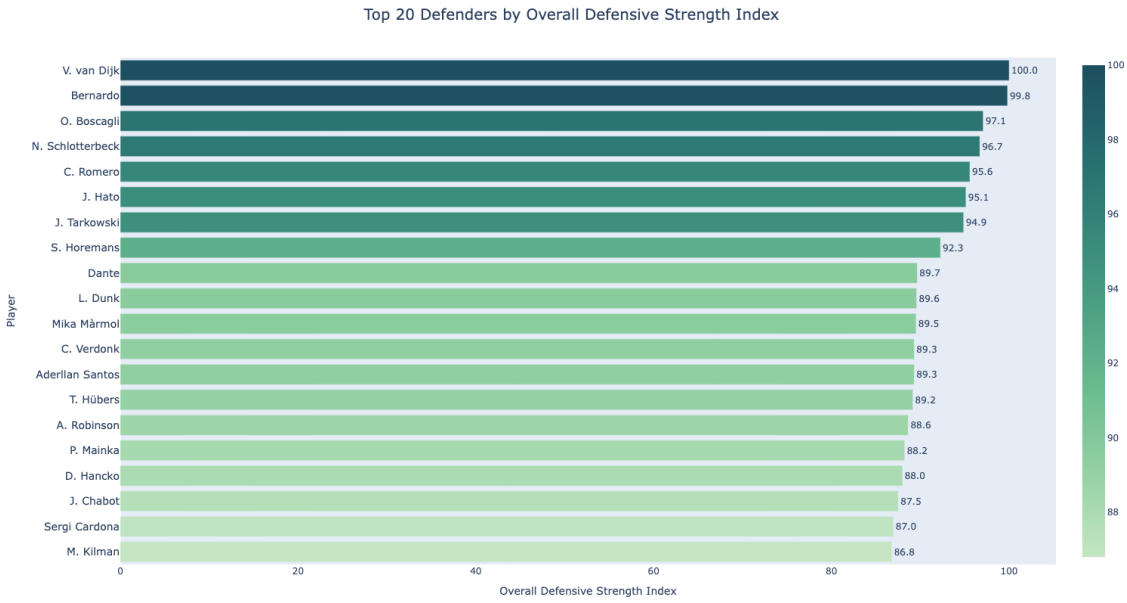


Figure 7 - Top 20 Defenders by Overall Defensive Strength Index

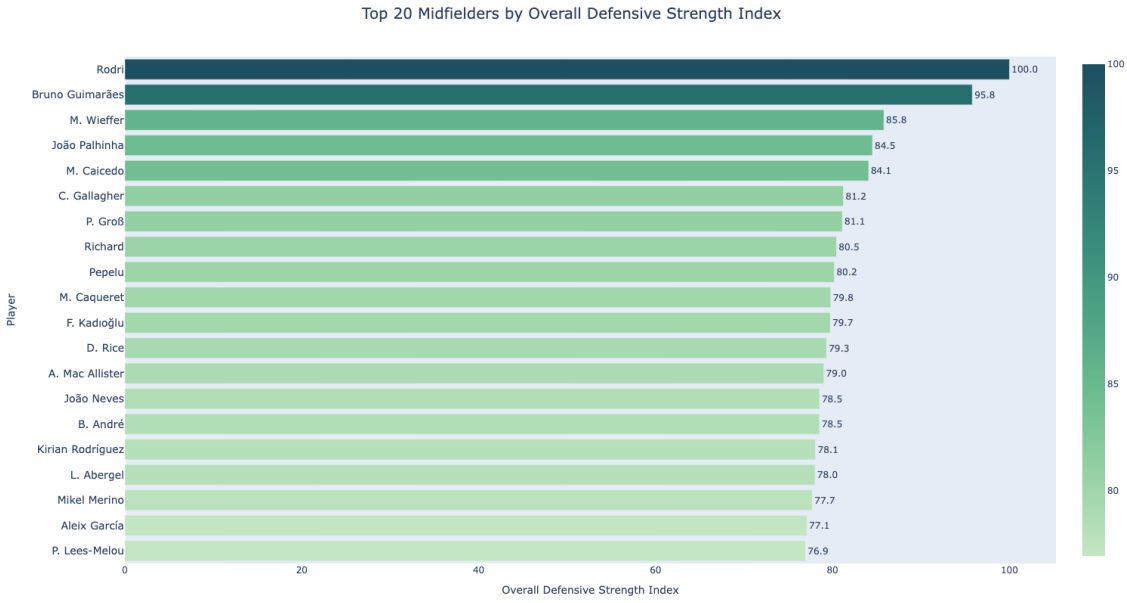


Figure 8 - Top 20 Midfielders by Overall Defensive Strength Index

5.2.3 Player Efficiency Index

The Player Efficiency Index was developed to measure a player's overall efficiency, rewarding productive actions and penalizing less effective behaviors. This index aims to summarize a player's overall contribution, emphasizing actions that have a positive impact on the game, such as scoring ability, assists, and accuracy in shooting and passing, while penalizing behaviors that negatively affect performance, such as fouls committed and cards received. Actions that contribute to achieving the team's objectives, such as total goals, assists, shots on target, and shooting accuracy, are assigned higher weights to reflect the value of efficient performances. Pass accuracy and key passes also play a significant role, emphasizing the importance of technical quality and tactical intelligence in the game.

At the same time, the Player Efficiency Index penalizes less efficient behaviors, such as fouls committed, which disrupt the flow of play, and cards received, applying increasing penalties based on the severity of the card. Red cards, for instance, have a greater impact than yellow cards, reflecting the significant consequences that a sending-off can have on the team. As in the other indices, the UEFA coefficient has been included.

The formula used to calculate the Player Efficiency Index is as follows:

$$\begin{aligned}\text{Player Efficiency Index} &= (goals\_total \cdot 2) + (goals\_assists \cdot 1.5) + (shots\_on \cdot 1.5) \\ &+ (shots\_accuracy \cdot 1.5) + (passes\_key \cdot 0.8) \\ &+ (passes\_accuracy \cdot 1.2) - (fouls\_committed \cdot 0.5) \\ &- (cards\_yellow \cdot 1.2) - (cards\_red \cdot 2) \\ &+ uefa\_coefficient\_value\end{aligned}$$

Since this index assesses a player's overall efficiency, even though it was calculated and normalized separately by position (like all other indices), the bar plot below presents the top 20 players regardless of their position (*Figure 9*).

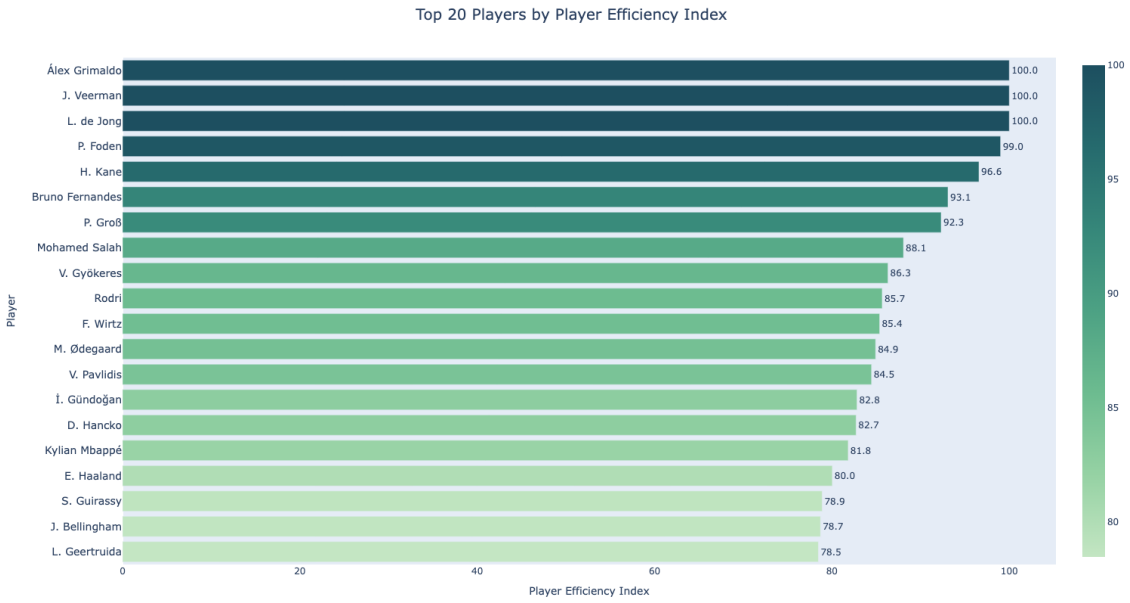


Figure 9 - Top 20 Players by Player Efficiency Index

5.2.4 Playmaking Index

The Playmaking Index was developed to measure a player's effectiveness in build-up play and ball possession management. This index focuses on a player's ability to maintain control of the ball, orchestrate play, and facilitate the team's transition to the attacking phase.

The index combines several variables that capture the key aspects of playmaking. Key passes, which directly contribute to creating goal-scoring opportunities, carry the highest weight, reflecting the importance of opening spaces and setting up teammates for shots. Pass accuracy, crucial for ensuring fluid ball circulation, is also heavily emphasized, while total passes help represent a player's overall involvement in possession. Other parameters, such as assists per game, successful dribbles, and duels won, have been included with lower weights to emphasize a player's ability to bypass opponents and maintain ball control. Ultimately, the UEFA coefficient has also been incorporated.

Playmaking Index

$$= (passes\_key \cdot 3) + (assists\_per\_game \cdot 0.5) + (passes\_accuracy \cdot 3) + (goals\_assists \cdot 0.5) + (passes\_total \cdot 1.5) + (dribbles\_accuracy \cdot 0.4) + (duels\_won \cdot 0.3) + uefa\_coefficient\_value$$

Since this index is closely tied to possession management and playmaking ability, the following graph (*Figure 10*) focuses on midfielders, presenting a bar plot showcasing the top 20 players in this role.

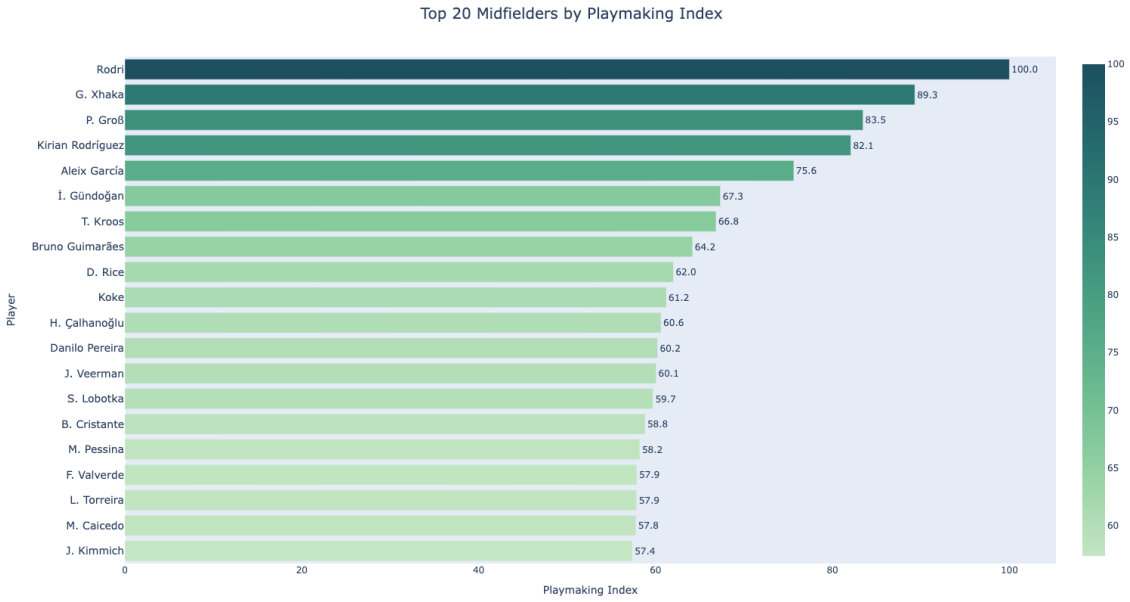


Figure 10 - Top 20 Midfielders by Playmaking Index

5.2.5 Shooting Efficiency Index

The Shooting Efficiency Index was designed to measure a player's shooting ability, combining variables that assess both accuracy and overall effectiveness in shooting. This index focuses on technical skills and the quality of shot execution, analyzing how well a player translates their attempts into high-quality or dangerous shots. A fundamental component of the index is the relationship between goals scored (excluding penalties) and total shots taken, weighted to give greater value to players who maintain high efficiency even with a high shooting volume. Shooting accuracy, measured

by the percentage of shots on target, is also emphasized, as it reflects a player's ability to direct their attempts toward the goal. Infine, il numero totale di tiri è incluso per considerare anche il contributo del giocatore in termini di volume offensivo. Come per gli altri indici, il coefficiente UEFA è stato considerato.

Finally, total shots are included to account for a player's offensive volume contribution. As with the other indices, the UEFA coefficient has been incorporated:

$$\text{Shooting Efficiency Index} = \left(\left(\frac{goals\_total - penalty\_scored}{shots\_total + 1}\right) \cdot 100 \cdot 2\right) + (shots\_accuracy \cdot 2) + (shots\_total \cdot 2) + uefa\_coefficient\_value$$

Since this index specifically measures shooting ability, it is particularly useful to analyze this skill among offensive roles. Therefore, the graphs below will display the top 20 midfielders and top 20 forwards, highlighting the players most proficient in generating high-quality shots within these two categories (*Figure 11* and *Figure 12*).

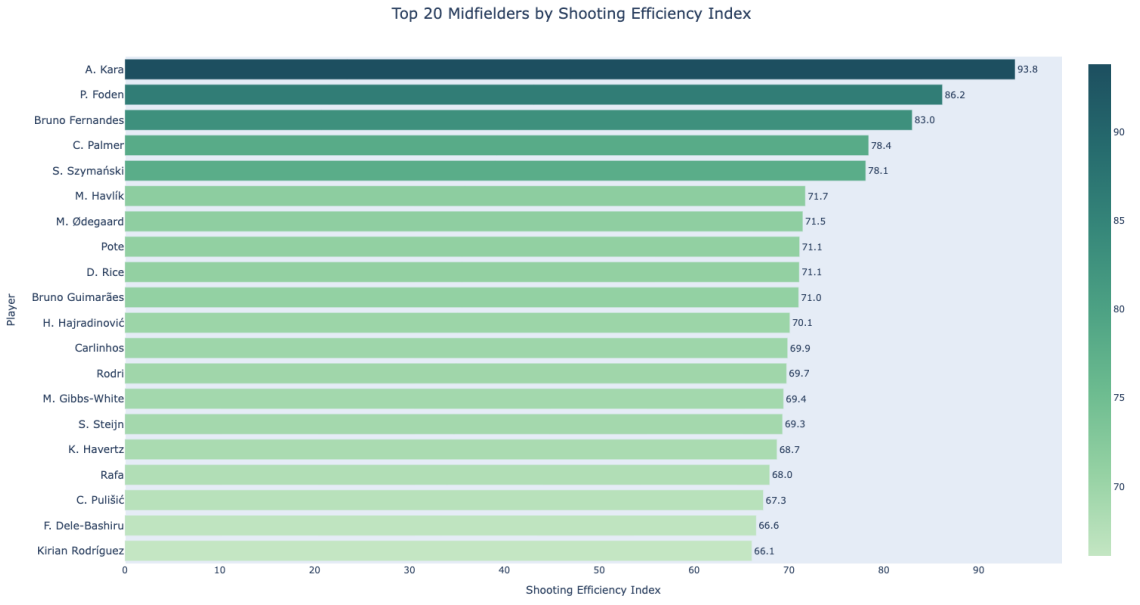


Figure 11 - Top 20 Midfielders by Shooting Efficiency Index

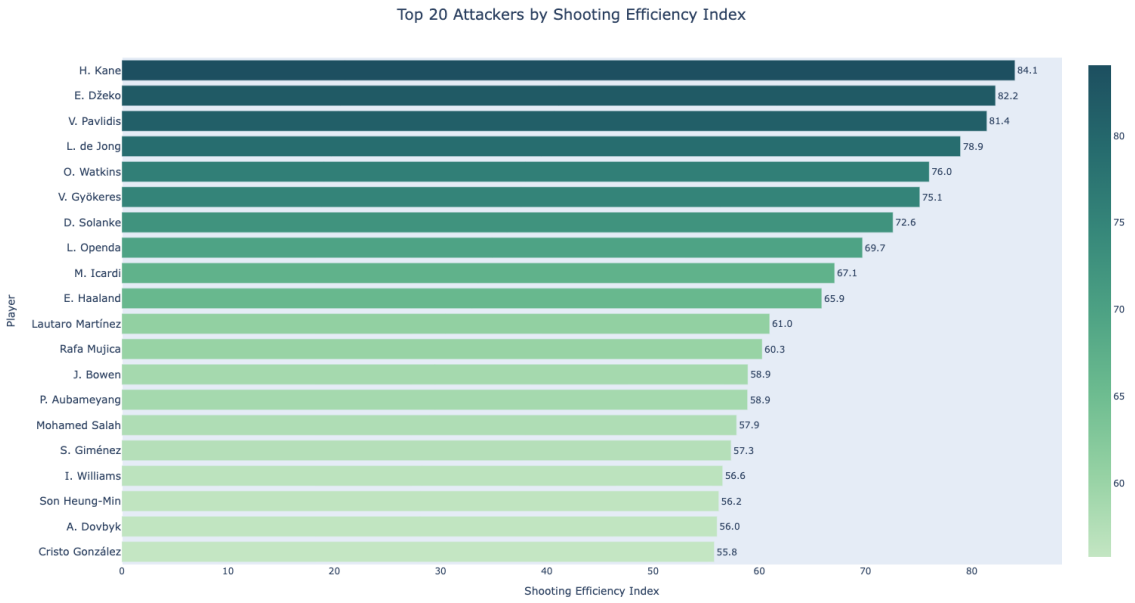


Figure 12 - Top 20 Attackers by Shooting Efficiency Index

5.2.6 Passing Efficiency Index

The Passing Efficiency Index was developed to measure a player's overall passing efficiency, focusing solely on the accuracy and quality of this specific skill. This index is based on variables that directly represent passing precision and effectiveness, assessing the player's contribution in this particular area. Although it may seem similar to the Playmaking Index, the key difference lies in their scope. The Playmaking Index considers a broader set of skills related to overall possession management, including dribbling, ball retention, and play construction to break defensive lines. The Passing Efficiency Index, on the other hand, focuses exclusively on passing itself.

While these two indices are correlated, the Passing Efficiency Index is more narrowly defined and specialized, enabling a more precise identification of players who excel in passing quality and efficiency, regardless of other technical or tactical skills.

The formula used to calculate the Passing Efficiency Index is as follows:

$$\text{Passing Efficiency Index} = (passes\_accuracy \cdot 0.7) + (passes\_key \cdot 0.5) + uefa\_coefficient\_value$$

Since this index is closely related to passing quality, the graph below will analyze the top 20 midfielders, as they are the players most responsible for distributing passes (*Figure 13*).

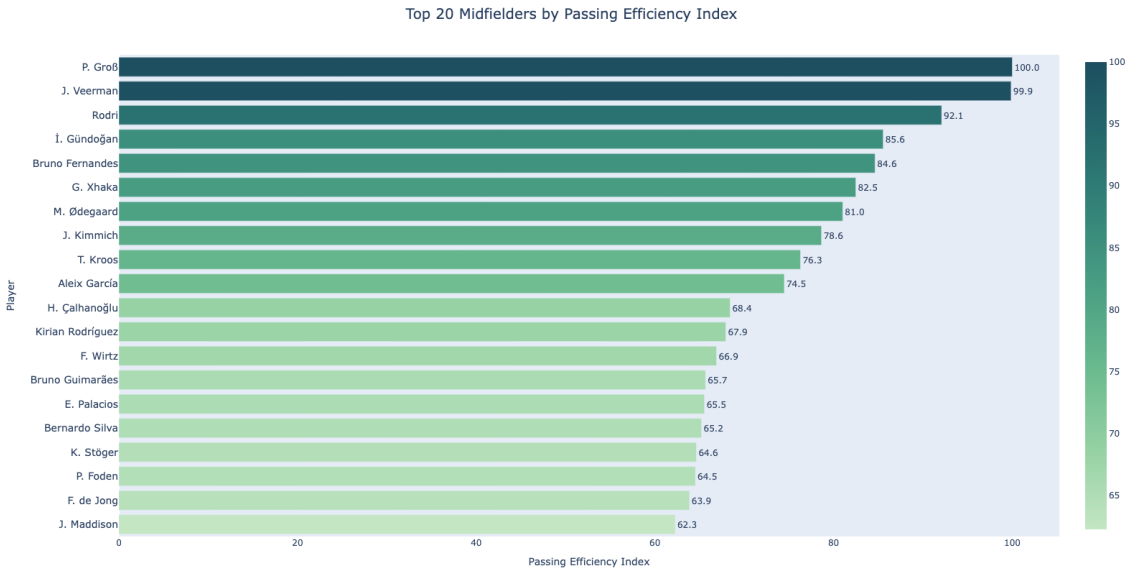


Figure 13 - Top 20 Midfielders by Passing Efficiency Index

5.2.7 Tackling Efficiency Index

The Tackling Efficiency Index was developed to measure a player's ability to execute tackles, focusing on both ball recovery effectiveness and the ability to do so cleanly, avoiding fouls and disciplinary actions. This index is based on variables that capture the different aspects of defensive play related to tackling, rewarding precise interventions while penalizing those that result in fouls or infractions.

More specifically, the index assigns significant weight to total tackles, blocking tackles, and interceptions, as these variables highlight a player's ability to disrupt opposing plays and regain possession for their team. Conversely, actions leading to fouls committed or cards received are penalized, with increasing weights based on severity (yellow, double yellow, or red cards). Unlike the Overall Defensive Strength Index, where cards have less impact due to the broader focus on overall defensive skills, disciplinary sanctions in this index are penalized more strictly, as the emphasis is specifically on clean and effective tackling. This choice is justified by the fact that the Tackling Efficiency Index aims to

assess the clean execution and overall success of the tackling technique itself. The ability to perform effective challenges without committing fouls or receiving cards is a key attribute that this index seeks to highlight. As with the other indices, the UEFA coefficient has been included.

The formula used to calculate the Tackling Efficiency Index is as follows:

$$\begin{aligned}\text{Tackling Efficiency Index} &= (tackles\_total \cdot 1) + (tackles\_blocks \cdot 2) \\ &+ (tackles\_interceptions \cdot 2.5) - (fouls\_committed \cdot 0.5) \\ &- (cards\_yellow \cdot 0.75) - (cards\_yellowred \cdot 1.25) \\ &- (cards\_red \cdot 1.75) + uefa\_coefficient\_value\end{aligned}$$

Since this index is closely related to tackling quality, the graph below will analyze the top 20 defenders, as they are the players most involved in this type of action (*Figure 14*).

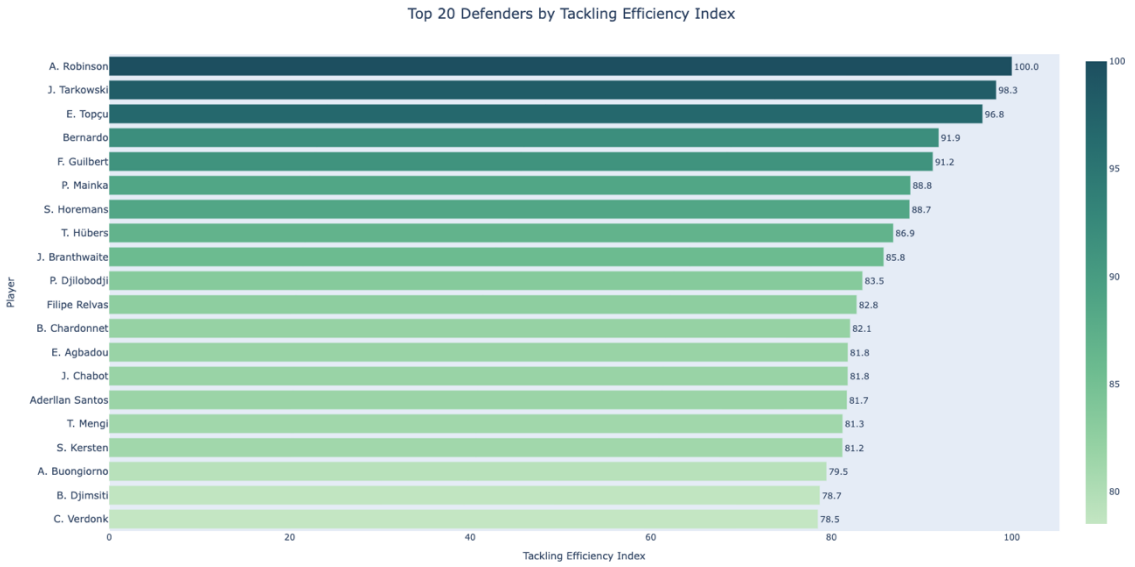


Figure 14 - Top 20 Defenders by Tackling Efficiency Index

5.2.8 Discipline Index

The Discipline Index was developed to assess a player's disciplinary behavior on the field, penalizing actions that lead to sanctions or irregular game interruptions. This index focuses exclusively on fouls committed and cards received, applying penalties

based on the severity of the infraction. Evaluating how frequently a player is sanctioned is crucial for this metric, as it also helps estimate their impact on future availability for the team. Yellow cards, double yellows, and direct reds can result in suspensions, reducing the number of matches in which the player can participate, ultimately negatively affecting the team's long-term performance. One of the key components of the calculation is the ratio between the number of cards received (yellow, double yellow, and red) and fouls committed[10]:

$$\frac{\text{cards\_yellow} + \text{cards\_yellowred} + \text{cards\_red}}{\text{fouls\_committed}}$$

This formula represents the number of cards received per foul committed. The idea is to assess how likely a player is to commit fouls that lead to disciplinary sanctions. For example, a player who commits many fouls but receives few cards will be considered relatively disciplined compared to one who earns cards with fewer fouls.

In addition to this key variable, cards received have also been included, with penalties proportional to their severity. Furthermore, fouls committed are considered, although with a lower weight, as they serve as an indicator of indiscipline. Yellow cards are often the result of an accumulation of fouls —in fact, a player who commits more fouls is more likely to receive a card.

To prevent distortions in the calculation, appearances were also included in the index weighting. Without this adjustment, players with zero or very few appearances would be ranked as the most disciplined, receiving a score of 100 simply because they did not spend enough time on the field to commit fouls or receive sanctions. At the same time, even highly disciplined players with many appearances and only a few cards would have been unfairly penalized with low scores. This would have introduced a bias in the results, making the assessment of a player's actual discipline less reliable.

Finally, unlike the other indices, the UEFA coefficient was not included in this calculation. The difficulty of the league is not necessarily linked to the number of fouls committed or cards received. A league considered less competitive may or may not have a more physical or foul-prone style of play compared to a higher-ranked league. For this

[10] If the number of fouls committed is zero, the denominator is set to 1 to prevent calculation errors caused by division by zero

reason, all players were evaluated equally, regardless of the league in which they compete.

The final mathematical formula used to calculate the Discipline Index is as follows:

$$\text{Discipline Index} = -\left(\left(\frac{cards\_yellow + cards\_yellowred + cards\_red}{fouls\_committed}\right) + (cards\_yellow \cdot 1.5) + (cards\_yellowred \cdot 2) + (cards\_red \cdot 2.5) + (fouls\_committed \cdot 0.5) - (games\_appearences \cdot 0.2)\right)$$

To ensure that the calculation represents a discipline index rather than an indiscipline index, a negative sign was added to the formula. This means that players with higher scores are the most disciplined, while those with lower scores, closer to zero, are the least disciplined. Without this adjustment, the interpretation would have been reversed.

In this case, it is useful to visualize both the most disciplined and the least disciplined players, particularly defenders, as they are the most likely to commit fouls or receive cards. For this reason, the graphs below display the top 20 most disciplined defenders and the bottom 20 in this index (*Figure 15* and *Figure 16*)

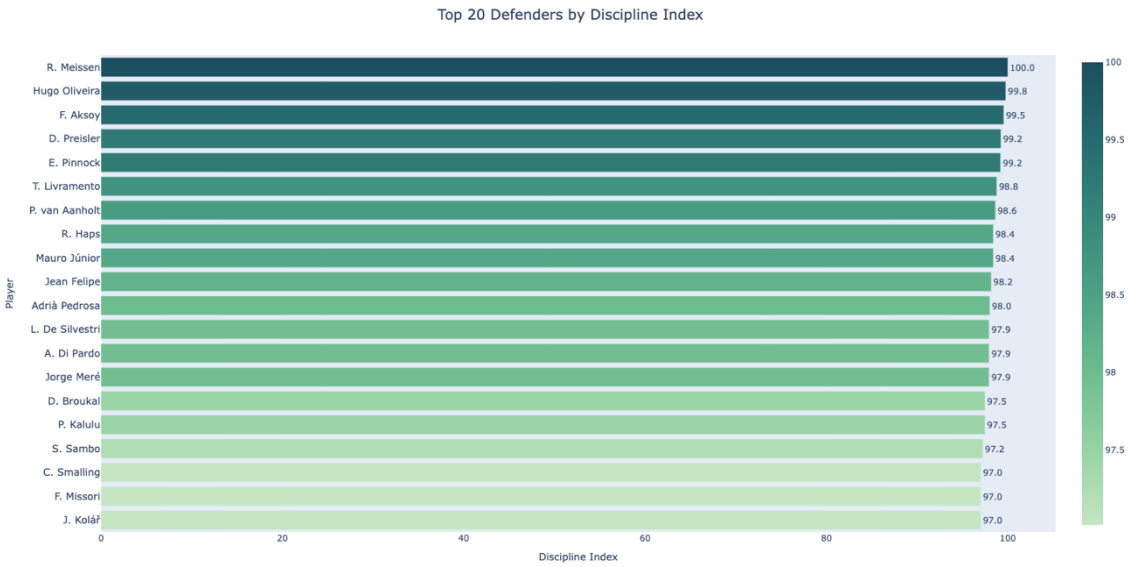


Figure 15 - Top 20 Defenders by Discipline Index

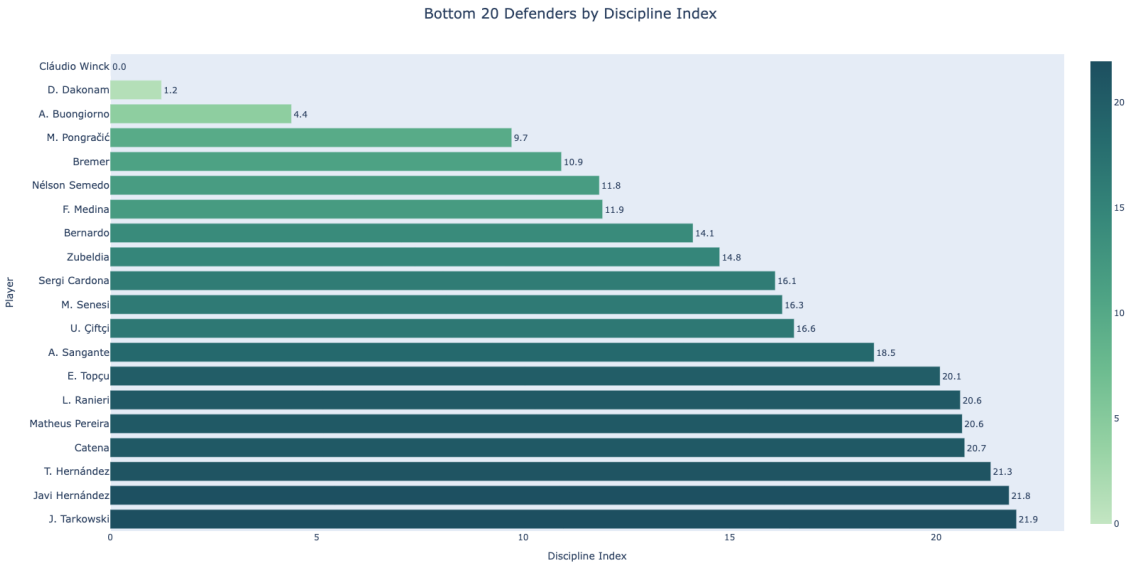


Figure 16 - Bottom 20 Defenders by Discipline Index

5.2.9 Physicality Index

The Physicality Index differs from the other indices as it focuses less on technical or tactical performance and more on a player's physical attributes. This index evaluates an athlete's height and weight, two variables that are increasingly crucial in modern football, where high physical intensity and frequent physical duels are essential. Beyond physical structure, the index also includes duels won, as being physically imposing alone is not enough, what truly matters is the ability to leverage one's body effectively on the field. Duels won, weighted with a lower value, reflect also how well a player utilizes their physicality during matches. In this case, the UEFA coefficient was not included in the calculation, as a player's physical attributes are obviously not influenced by the difficulty of the league in which they compete.

The formula used to calculate the Physicality Index is as follows:

$$\text{Physicality Index} = height\_cm + weight\_kg + (duels\_won \cdot 0.1)$$

Since this index is closely related to physical attributes, it is particularly relevant to analyze this metric across all outfield positions, as physical duels occur in each of them.

Therefore, the three bar plots below display the top 20 defenders, top 20 midfielders, and top 20 forwards according to this index (*Figure 17, Figure 18* and *Figure 19*).

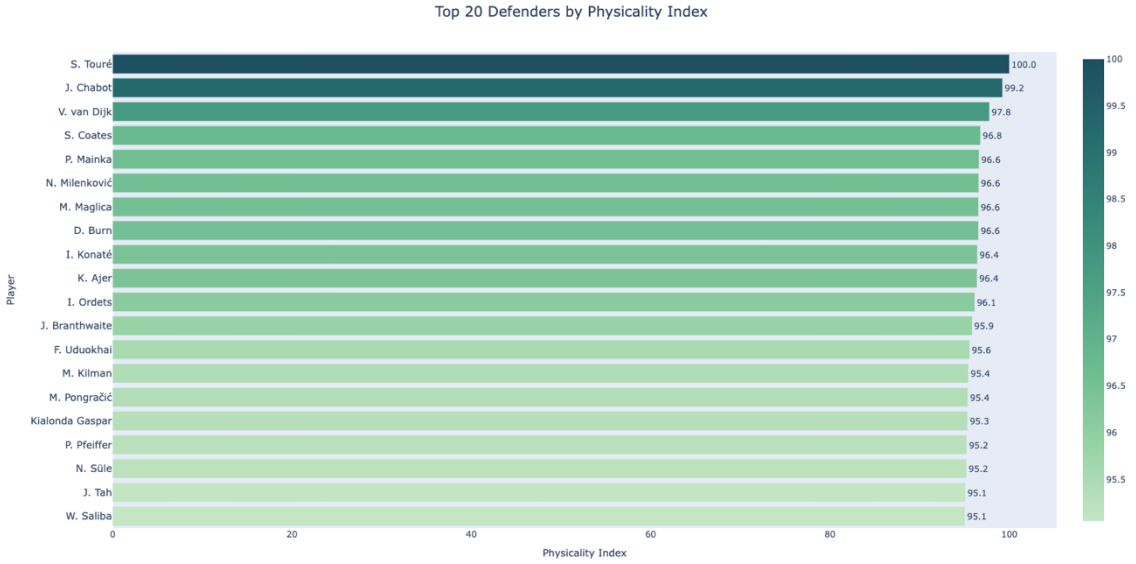


Figure 17 - Top 20 Defenders by Physicality Index

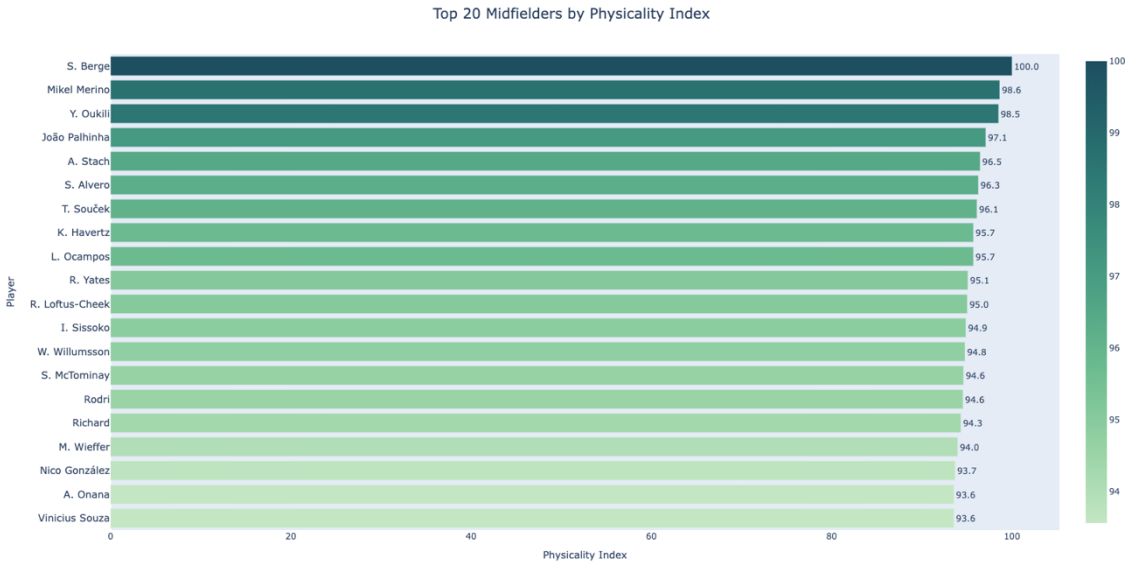


Figure 18 - Top 20 Midfielders by Physicality Index

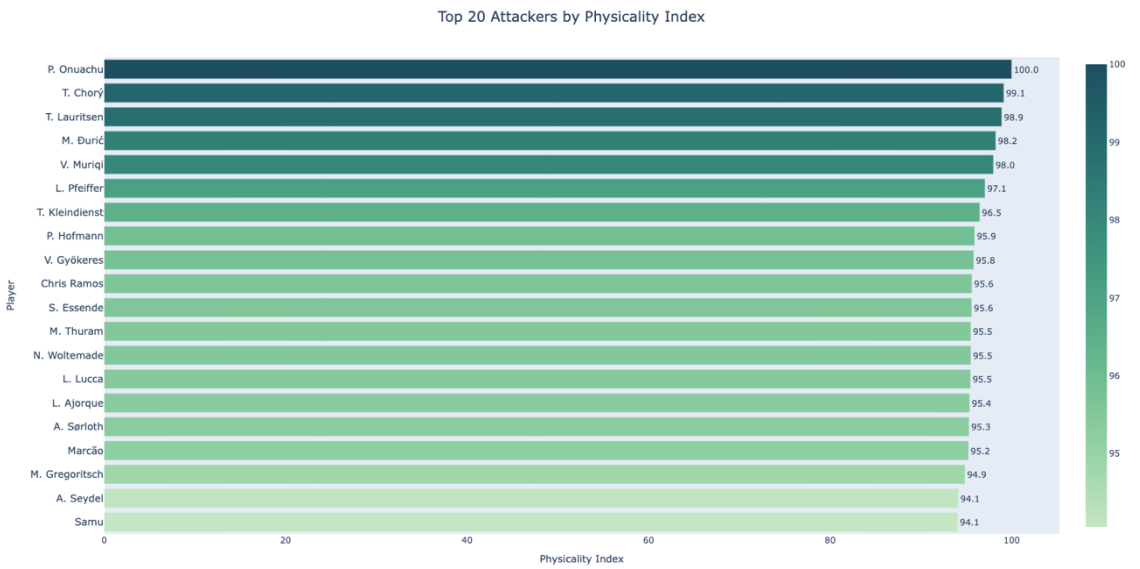


Figure 19 - Top 20 Attackers by Physicality Index

5.2.10 Offensive Contribution Index

The Offensive Contribution Index was developed to measure a player's overall contribution to their team's goals, considering both goals scored and assists. This index focuses exclusively on a player's direct involvement in offensive production, assigning greater weight to variables that reflect direct goal contributions.

Although it may seem similar to the Overall Offensive Strength Index, the key difference is that the Offensive Contribution Index is entirely focused on how much a player contributes to goal scoring, without considering other variables or excluding penalties. As a result, variables such as total goals and assists, as well as per-game averages, are given greater weight compared to those in the Overall Offensive Strength Index In this index, shots are evaluated in relation to goals scored through the goal-to-shot ratio, rewarding players who not only score but do so with high efficiency. This parameter helps identify players who contribute most to their team's offensive success, not just in terms of quantity but also in quality.

The UEFA coefficient has been included to account for the difficulty of the league in which the player contributes to goal scoring.

The formula used to calculate the Offensive Contribution Index is as follows:

$$\text{Offensive Contribution Index} = (goals\_total \cdot 1.5) + (goals\_per\_game \cdot 2) + (goals\_assists \cdot 1.5) + (assists\_per\_game \cdot 2) + \left(\frac{goals\_total}{shots\_total}\right) + uefa\_coefficient\_value$$

Since this index is closely tied to offensive production, the graphs below will analyze the top 20 attackers, top 20 midfielders, but also the top 20 defenders (*Figure 20, Figure 21* and *Figure 22*). The decision to include defenders is based on the fact that, in modern football, defenders who actively contribute to the attacking phase—such as attacking full-backs skilled in goal-scoring situations—are increasingly valued by teams.

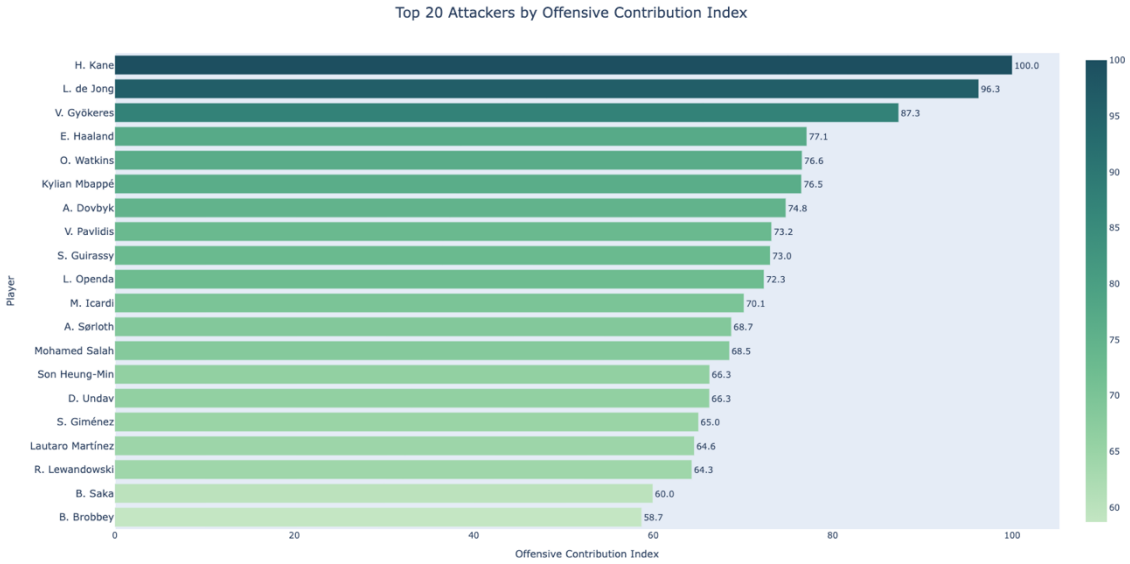


Figure 20 - Top 20 Attackers by Offensive Contribution Index

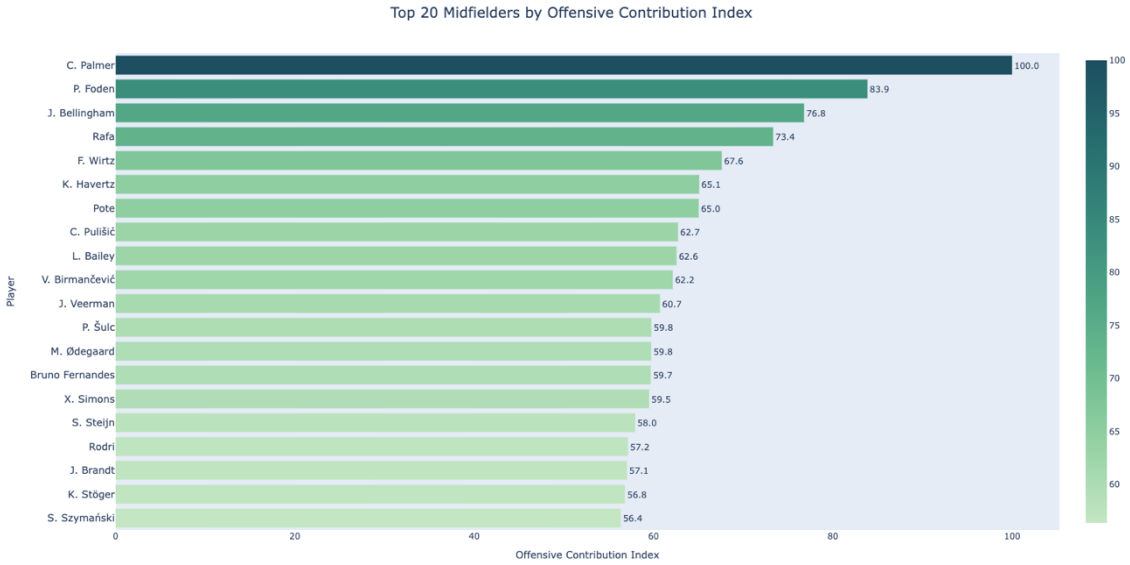


Figure 21 - Top 20 Midfielders by Offensive Contribution Index

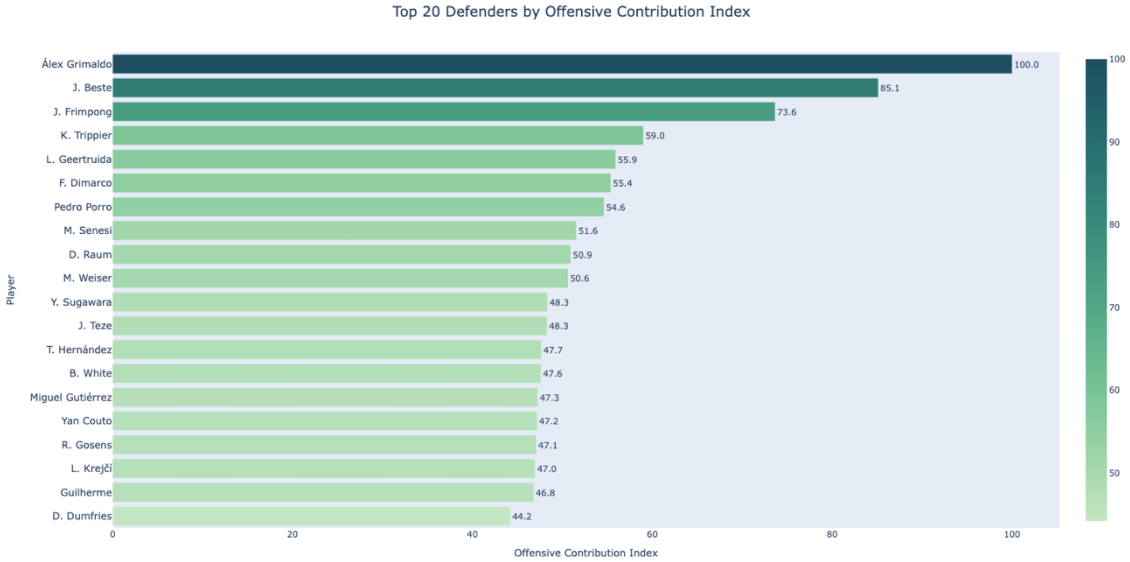


Figure 22 - Top 20 Defenders by Offensive Contribution Index

5.2.11 Consistency Index

The Consistency Index was developed to assess a player's stamina, using the available variables to measure their ability to maintain physical endurance throughout matches and across the season. This index aims to identify players who can stay on the

field longer, regularly playing a high number of minutes without being substituted. In doing so, it rewards physical resilience and a player's reliability as a key asset for his team.

The calculation of the index assigns a positive weight to total appearances and minutes played, while substitutions out are penalized. The goal is to reward players who can stay on the field for the full duration of the match without being replaced.

The formula used to calculate the Consistency Index is as follows:

$$\text{Consistency Index} = (games\_appearences \cdot 1) + (games\_minutes \cdot 1.5) - (substitutes\_out \cdot 0.3)$$

Since this index is closely tied to endurance and stamina, regardless of position (as a match lasts 90 minutes for all players), the bar plot below displays the top 20 players, without distinction by role (*Figure 23*).

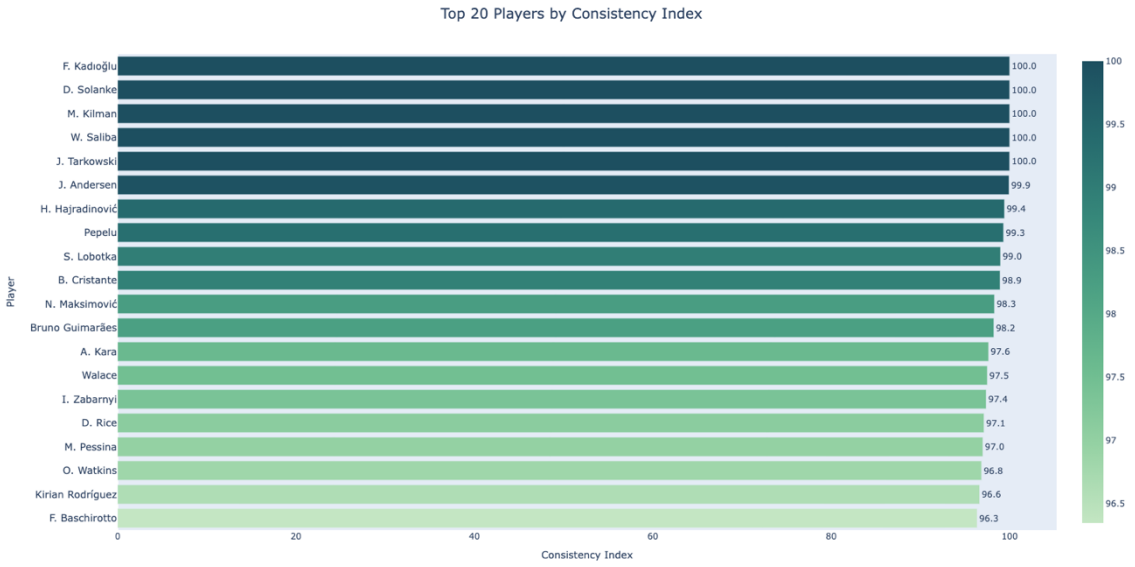


Figure 23 - Top 20 Players by Consistency Index

3.2.12 Clutch Performance Index

The Clutch Performance Index was developed to measure how decisive a player is on the field, focusing on actions that significantly impact the team's results. Unlike the

Offensive Contribution Index, which evaluates a player's overall involvement in goals, including their contribution per match (goals and assists per game) and does not differentiate penalties, the Clutch Performance Index focuses exclusively on a player's ability to make a difference, regardless of the consistency of their performances. This index highlights a player's impact in crucial moments, rewarding those who score decisive goals and provide key assists, as well as those who successfully convert penalties, which are often pivotal in matches. Penalties scored are assigned a lower weight than open-play goals but are still rewarded, as they demonstrate a player's ability to handle pressure in critical situations. Conversely, missed penalties are penalized, as they represent wasted opportunities that can often be decisive. The key distinction between this index and the Offensive Contribution Index lies in the timing and impact of a player's actions rather than their offensive consistency. While the Offensive Contribution Index measures a player's continuous involvement in the team's attacking output, the Clutch Performance Index aims to capture game-changing actions—those that influence the course of a match or prove decisive in the final result. Finally, as with the other indices, the UEFA coefficient has been included.

The formula used to calculate the Clutch Performance Index is as follows:

$$\begin{aligned}\text{Clutch Performance Index} &= \big((goals\_total - penalty\_scored) \cdot 1.5\big) \\ &+ (goals\_assists \cdot 1) + (penalty\_scored \cdot 0.75) \\ &- (penalty\_missed \cdot 0.5) + uefa\_coefficient\_value\end{aligned}$$

Since this index measures a player's ability to be decisive, the graph below will show the top 20 forwards and top 20 midfielders, as these positions play a key role in the crucial offensive phases of matches (*Figure 24*).

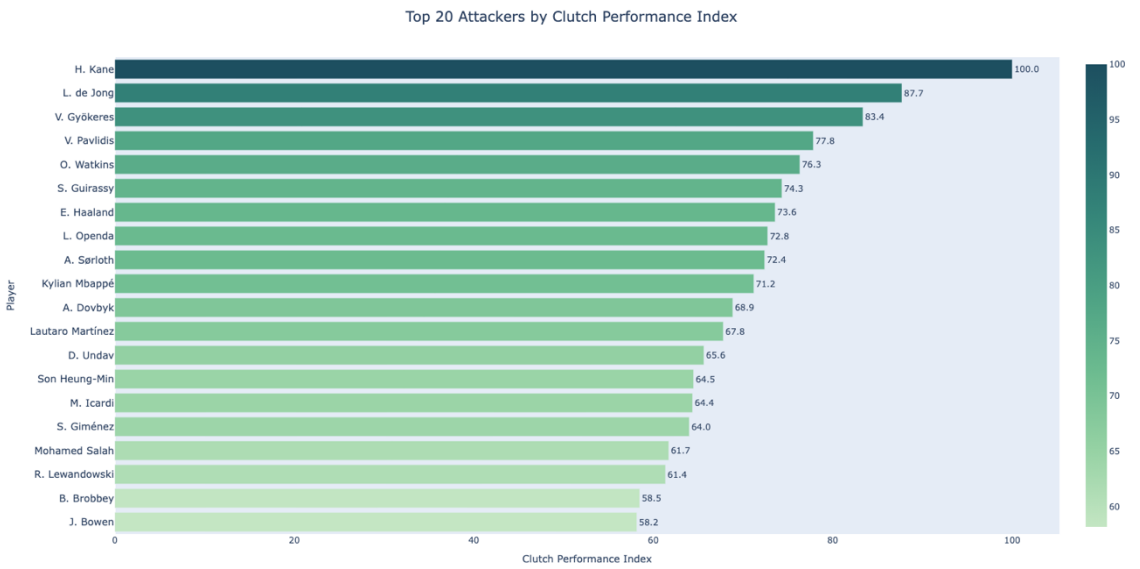


Figure 24 - Top 20 Attackers by Clutch Performance Index

5.2.13 Finishing Ability Index

The Finishing Ability Index was developed to measure a player's goal-scoring ability, focusing exclusively on variables related to finishing efficiency. Unlike the Offensive Contribution Index, which considers both goals and assists to evaluate a player's overall offensive contribution, and the Clutch Performance Index, which measures decisiveness, this index analyzes only goal-scoring ability, without accounting for other factors such as assists or shot volume.

The index rewards players who excel at "finding the back of the net", placing particular emphasis on variables such as goals scored from open play (total goals – penalty goals) and the goal-to-shot ratio. Penalties are also included but with a slightly lower weight than open-play goals, with penalty accuracy also factored in. The UEFA coefficient has been integrated into the calculation.

The formula used to calculate Finishing Ability Index is as follows:

$$\begin{aligned}\text{Finishing Ability Index} &= \big((goals\_total - penalty\_scored) \cdot 1\big) \\ &+ (penalty\_scored \cdot 0.75) + (penalty\_accuracy) \\ &+ \left(\frac{goals\_total}{shots\_total} \cdot 2\right) + uefa\_coefficient\_value\end{aligned}$$

Since this index is directly tied to finishing ability, the graph below will display the top 20 forwards, as they are the primary reference for goal-scoring proficiency (*Figure 25*).

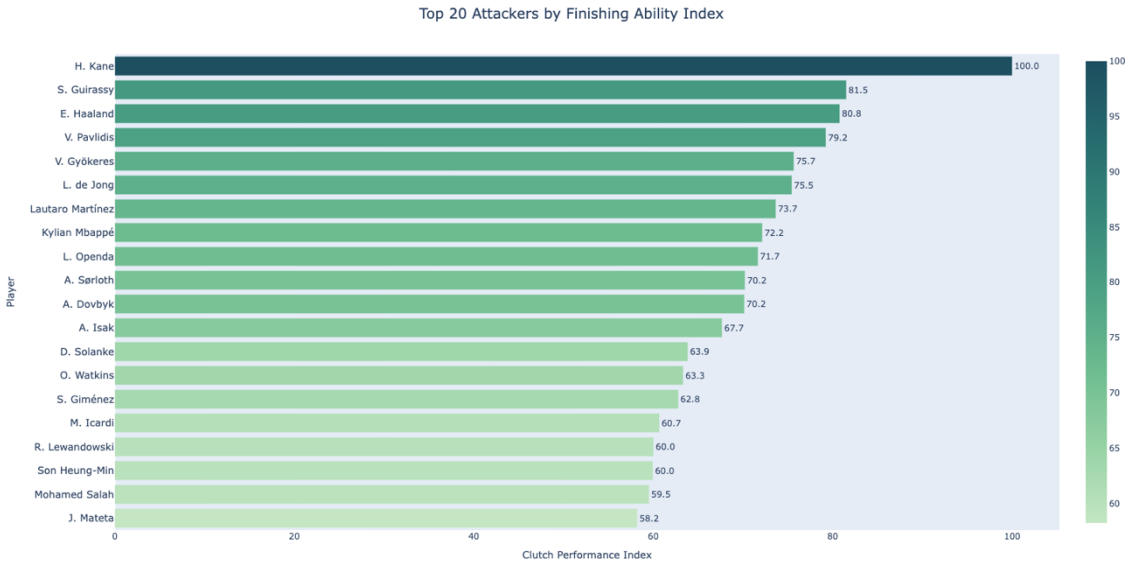


Figure 25 - Top 20 Attackers by Finishing Ability Index

5.2.14 Explosiveness Index

The Explosiveness Index was developed to measure a player's ability to break past opponents and create numerical superiority. This index combines various variables that reflect a player's capacity to challenge defenses through dribbling, shot attempts, duels won, and fouls suffered. This type of index is particularly relevant for attacking wingers, who are often responsible for creating numerical advantages on the wings, putting pressure on opposing defenses with sudden accelerations and dribbles, and then either taking a shot on goal or drawing valuable fouls in advanced areas of the field.

Successful dribbles form the core of the index and carry the greatest weight, as they are a defining trait of explosive players. Shots on target are also included, as they demonstrate a player's ability to create and capitalize on scoring opportunities after beating an opponent.

Duels won are included to highlight a player's ability to prevail both physically and technically in one-on-one situations. Another important element is fouls drawn, as highly

explosive players, due to their ability to beat opponents, often draw fouls, which can be highly valuable in securing set-piece opportunities in dangerous areas of the pitch. Once again, the UEFA coefficient has been included in the calculation.
The formula used to calculate the Explosiveness Index is as follows:

$$\text{Explosiveness Index} = (dribbles\_success \cdot 3) + (shots\_on \cdot 1) + (duels\_won \cdot 0.1) + (fouls\_drawn \cdot 0.3) + uefa\_coefficient\_value$$

Since this index reflects offensive impact, the bar plot below will analyze the top 20 forwards, as they are the players most associated with these attributes, particularly attacking wingers (*Figure 26*).

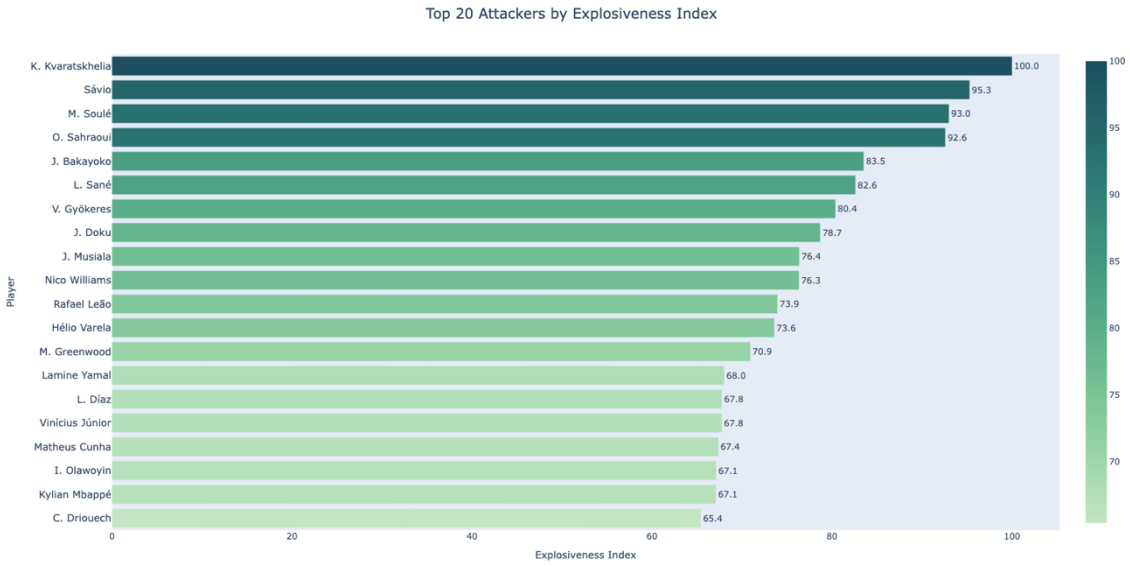


Figure 26 - Top 20 Attackers by Explosiveness Index

5.2.15 Strategic Play Index

The Strategic Play Index was developed to measure a player's ability to make intelligent and strategically valuable decisions during a match. This index combines variables from both defensive and offensive phases, providing a comprehensive overview of a player's tactical impact on the game's flow.

Actions that significantly contribute to the tactical dynamics of a match are rewarded. Key passes and pass accuracy reflect a player's role in playmaking and ball circulation, while interceptions and duels won highlight their ability to regain and retain possession. Fouls drawn are also valued, as they indicate moments where the player successfully gains an advantage for their team. Conversely, fouls committed are penalized, as they disrupt the flow of play or can put the team in difficult situations. Finally, as in previous indices, the UEFA coefficient has been included.

$$\begin{aligned}\text{Strategic Play Index} &= (passes\_key \cdot 1.5) + (passes\_accuracy \cdot 1) \\ &+ (tackles\_interceptions \cdot 1.2) + (duels\_won \cdot 0.75) \\ &+ (fouls\_drawn \cdot 1.3) - (fouls\_committed \cdot 0.8) \\ &+ uefa\_coefficient\_value\end{aligned}$$

Since this index combines offensive, defensive, and tactical abilities, it is particularly relevant for midfielders, as they are the most involved in coordinating both attacking and defensive actions and generally serve as the most strategic players on the field. Therefore, the bar plot below will show the top 20 midfielders according to this index (*Figure 27*).

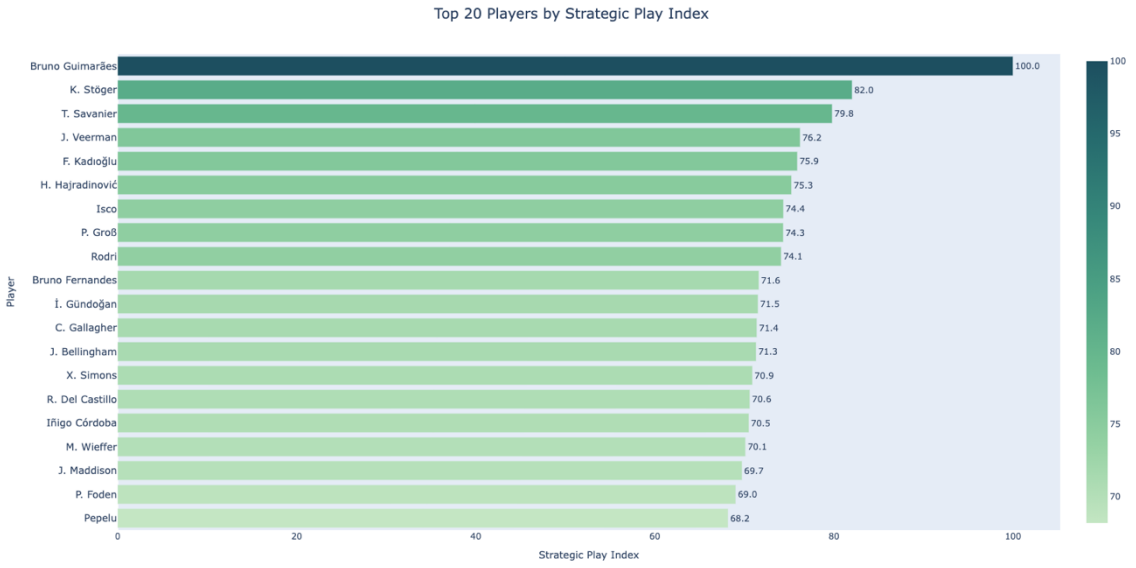


Figure 27 - Top 20 Players by Strategic Play Index

CHAPTER 6:
PERFORMANCE INDICES VALIDATION

6.1 Introduction to Validation

6.1.1 Why Validating?

Validating the indices is a crucial step to ensure they are truly representative and provide genuine insights into player performance. A well-designed index should accurately capture the characteristics it intends to measure, avoiding biases or arbitrary weight assignments. For this reason, it is essential to empirically test whether these indices can effectively differentiate between various types of players and highlight their distinct performance traits.

One of the most critical aspects in constructing these indices is the assignment of weights to the variables that compose them. Ideally, these weights should reflect the relative importance of each variable in determining the skill measured by the index. However, manually setting these weights introduces a degree of subjectivity that could impact the validity of the indices.

To minimize this arbitrariness, an initial attempt was made to develop an algorithm that would automatically determine the most appropriate weights for each variable. However, the results proved to be unsatisfactory: the generated indices did not accurately reflect the expected characteristics, and their use led to evaluations that were inconsistent with the footballing context. As a result, an alternative approach was necessary to verify the validity of the indices.

Specifically, a machine learning classification problem was formulated. The core idea was that if the developed indices contained meaningful information, then a classification model should be able to correctly predict a player's role using only the indices as input. This approach offers a practical way to assess whether the indices accurately capture the structural differences between players and effectively distinguish their roles on the field.

A particularly interesting aspect of this validation process is that the indices were normalized by role rather than across the entire dataset, as already described. Theoretically, this should have made it more challenging for the model to distinguish between a defender and a forward, as the index values were scaled within each role. As

explained earlier, for example, a forward who scored 50 goals, being the highest-scoring forward, could receive a value of 100 in the Offensive Contribution Index. Similarly, a defender who scored 6 goals, if he were the top-scoring defender, could also receive a value of 100 in the same index. Consequently, the model could not simply rely on index values alone to differentiate roles, it needed to understand how the indices behave across different positions. If the indices had been normalized across the entire dataset, the distinction between roles would have been more noticeable, making the model's task considerably easier.

However, despite this additional challenge, various approaches have been tested and refined to ensure that the foundations of player analysis are solid and reliable. These approaches will be presented in the following sections.

6.1.2 Classification Model for Validation

A classification model is a type of supervised learning algorithm that, given a set of input variables X (features), learns to predict a categorical output variable (target). In this case, the models are designed with the X variables representing all the developed indices, while the y variable corresponds to the player's position (`player_position`), as previously mentioned. A Supervised learning algorithm

> *"is a machine learning paradigm for acquiring the input-output relationship information of a system based on a given set of paired input-output training samples"*
>
> (Zhang, 2000)

This means that the algorithm learns the relationship between input and output using pre-labeled data, where each input example is associated with a known output label—in this case, the different player positions.

6.1.3 Train-Test Split

Before training any machine learning model, it is essential to divide the dataset into training and test sets—a process known as the train-test split. This step is crucial because it allows to evaluate how well the model generalizes to unseen data. Without this separation, there is a risk that the model might perform well simply because it memorizes the training data rather than learning meaningful patterns that can be applied to new cases (Hastie, Tibshirani, & Friedman, 2009): a phenomenon known as overfitting. So,

overfitting leads to excellent performance on training data but poor results when applied to new data. On the other hand, an insufficient training set may cause the specular problem of underfitting, where the model does not learn enough from the data, leading to poor predictive performance on new data.

To mitigate these risks, the dataset in this study was split into 70% training data and 30% test data. The training set is used to train the model, while the test set acts as an independent benchmark to evaluate its performance. By keeping the test data completely unseen during training, we obtain a more realistic estimate of how the model would perform, hence generalize towards new data (Hastie, Tibshirani, & Friedman, 2009). Importantly, stratified sampling was used to ensure that the distribution of player positions remained consistent across both sets. This is particularly important when dealing with imbalanced classes, as it prevents the model from being biased toward the most frequent categories. In this specific case, the dataset contains a higher number of midfielders and defenders compared to attackers and goalkeepers, leading to an imbalance in class distribution. This imbalance could seriously impact the model's performance, as it might end up learning patterns primarily from the more common roles (like defenders and midfielders) while struggling to understand the characteristics of underrepresented positions (such as attackers and goalkeepers). For example, if the training set randomly includes a high number of defenders but very few attackers, the model would have limited exposure to attacking players, making it harder to recognize their patterns. Then, when tested on a set where attackers are more prevalent, the model might fail to classify them correctly, simply because it did not get enough information about them during training.

6.1.4 Model Evaluation: Accuracy and Confusion Matrix

Once a machine learning model has been trained, it is essential to evaluate its performance using appropriate metrics. These metrics help determine how well the model generalizes to unseen data (the test set). In this study, two primary evaluation methods were used:

a) Accuracy: A direct metric that measures the percentage of correct predictions.

b) Confusion Matrix: A detailed view of the predictions, illustrating model's performance across each class.

The accuracy simply measures the proportion of correctly classified instances out of the total. It is calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

[11]

where:

- TP: True Positives
- TN: True Negatives
- FP: False Positives
- FN: False Negatives

In this study, accuracy served as a baseline indicator of how well the model could differentiate between different player positions. However, while accuracy is easy to interpret, it has its limitations, especially when dealing with imbalanced classes. Since the dataset contains more midfielders and defenders than other roles, a model could achieve high accuracy simply by favoring these majority classes while still struggling with less common ones.

To gain deeper understandings into the model's classification ability, a confusion matrix was used. Instead of providing just a single score like accuracy, the confusion matrix shows how the model performs for each class, highlighting which position are most frequently confused with others. By visualizing the confusion matrix as a heatmap, we can immediately identify which player positions are misclassified the most.

6.2 Classification Models: Implementation and Results

6.2.1 Logistic Regression

The Logistic Regression model was selected as the first approach to evaluate the validity of the indices. Logistic regression is a widely used statistical model for classification tasks, particularly in binary and multi-class classification problems. It estimates the probability that a given input belongs to a specific class, making it a

[11] This formula is widely recognized in the field and is detailed in resources such as the article "Classification: Accuracy, Precision, Recall" from Google's Machine Learning Crash Course.

fundamental tool for problems where the target variable is categorical (Hosmer, Lemeshow & Sturdivant, 2013). Logistic regression is particularly useful because it provides interpretable results, making it a strong baseline model before moving to more complex machine learning algorithms. Unlike linear regression, which predicts continuous values, logistic regression applies the sigmoid function to map predictions to a probability range between 0 and 1. This makes it ideal for classification tasks, as predictions can be thresholded to assign classes.

The model achieved an overall accuracy of 93%, indicating that the proposed indices contain meaningful information that allow the model to distinguish the player positions effectively. The corresponding confusion matrix, shown in *Figure 28*, provides further insights into the classification performance:

- Goalkeepers were correctly classified in nearly all cases (25/27), with only two misclassifications as defenders. This remains an insignificant error rate, reinforcing that the performance indices are distinct from those of outfield players, even though, as previously mentioned, this study does not primarily focus on goalkeepers.
- Defenders achieved a high classification performance, with 389 correctly classified instances, and 10 misclassified as midfielders and 20 as attackers. The occasional misclassification as midfielders is reasonable, as certain defenders exhibit attributes like midfielders, such as passing ability and involvement in build-up play. However, the misclassification of defenders as attackers is more problematic, as it suggests the model struggles to fully distinguish between defensive and offensive roles.
- Midfielders were predominantly classified correctly (396 cases), with minimal misclassifications. The occasional misclassification as attackers (8 cases) or defenders (10 cases) is expected, as certain midfielders take on more offensive or defensive responsibilities, naturally blurring positional distinctions.
- Attackers were well classified overall (289 correct), but 18 were misclassified as defenders, which represents the biggest logical issue in these results. Unlike midfielders being confused with adjacent roles, attackers and defenders have

fundamentally different responsibilities, making this misclassification harder to accept.

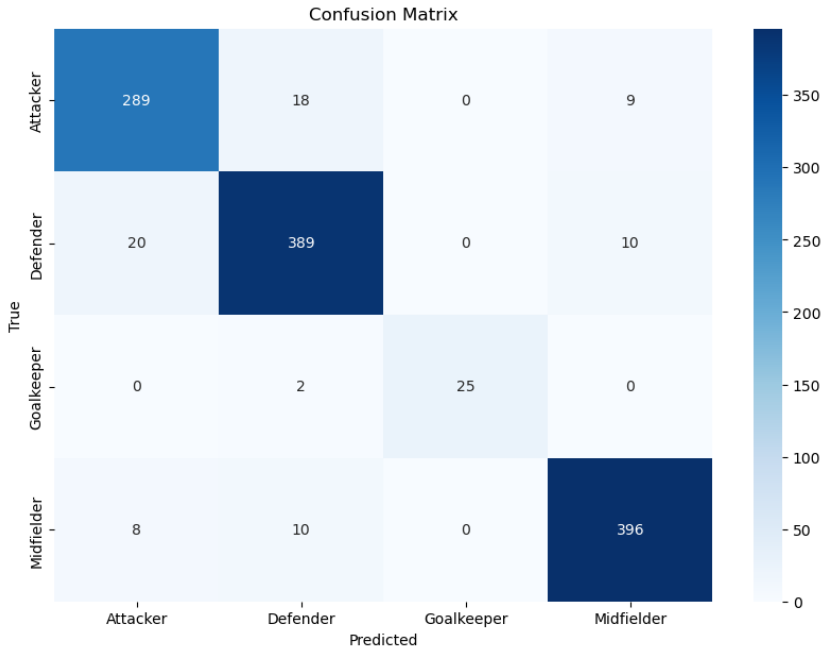


Figure 28 - Confusion Matrix of the Logistic Regression Model for Indices Validation

```
Logistic Regression: Accuracy = 0.93
```

The most critical limitation observed is the misclassification of attackers as defenders and vice versa, which does not align with typical football characteristics. A likely explanation for this issue is class imbalance—there are more defenders in the dataset than attackers, causing Logistic Regression to struggle with underrepresented classes. Since Logistic Regression assumes linear decision boundaries, it may not be the best approach for handling complex, non-linear relationships in the data. The model's difficulty in handling imbalanced datasets leads it to overclassify the majority class (defenders), incorrectly assigning some attackers to this category. However, to determine whether this issue is inherent to the Logistic Regression model or indicative of a flaw in the way the indices were structured, it is necessary to test an alternative approach.

6.2.2 Random Forest Classifier

The Random Forest Classifier was considered as an alternative model to address classification performance. Unlike Logistic Regression, Random Forest is an ensemble learning[12] method that builds multiple decision trees[13] and combines their outputs to make more robust and accurate predictions (Breiman, 2001).

Random Forest proves to be highly effective for classification tasks due to several key advantages. First, it can capture non-linearity, making it well-suited for modeling relationships between player performance indices and their respective position, unlike logistic regression. Moreover, by averaging predictions from multiple decision trees, Random Forest reduces overfitting, ensuring greater generalization to unseen data compared to single decision tree models.

The Random Forest model achieved an overall accuracy of 93%, the same as the Logistic Regression model. However, despite having identical accuracy, a deeper analysis of the confusion matrix displayed in *Figure 29* reveals that Random Forest provides a more precise classification and makes fewer unreasonable errors.

The most significant improvements are that Random Forest significantly reduces the misclassification between attackers and defenders, an issue that was more common in Logistic Regression. Instead, when errors occur, they are mostly between midfielders and defenders.

This misclassification pattern, as already mentioned earlier, is more intuitive in a football context. Many attacking midfielders play very advanced roles, making their statistical profiles resemble those of attackers. Similarly, defensive midfielders operate close to the defensive line, often sharing attributes with defenders. As a result, these misclassifications reflect real tactical overlaps, whereas the confusion between attackers and defenders seen in Logistic Regression was less justifiable.

[12] Ensemble Learning is a machine learning technique that combines multiple base models to improve predictive performance. Instead of relying on a single algorithm, ensemble methods aggregate predictions from several models, reducing variance, bias, and the likelihood of overfitting (Dietterich, 2000)

[13] Decision Trees are a machine learning technique used for classification and regression tasks. They function by recursively splitting the dataset into subsets based on feature values, creating a tree-like structure where each node represents a decision rule and each leaf represents a predicted outcome (Quinlan, 1986)

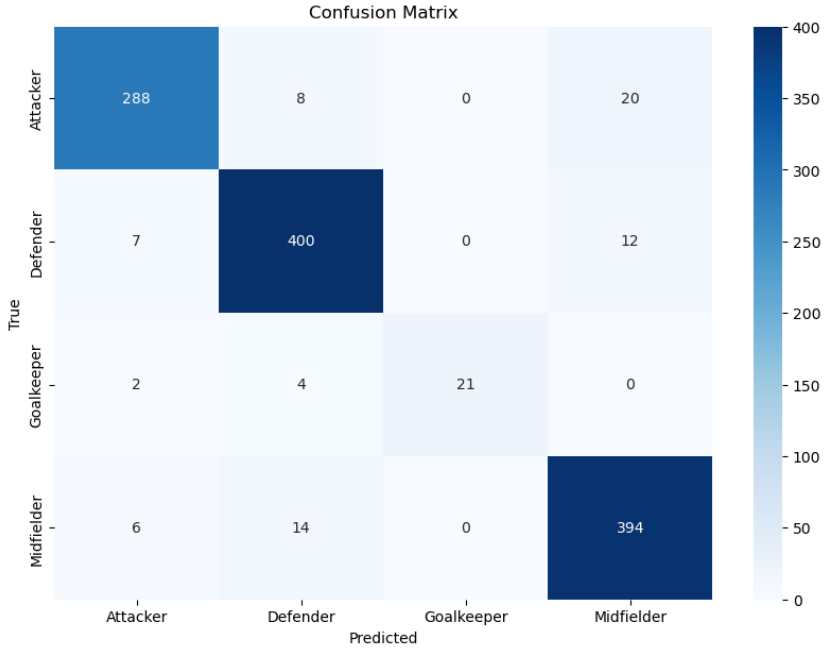


Figure 29 - Confusion Matrix of the Random Forest Model for Indices Validation

```
Random Forest: Accuracy = 0.93
```

More precisely, by analyzing the confusion matrix (*Figure 29*), we can observe significant differences between the Random Forest model and the Logistic Regression model, despite both achieving an overall accuracy of 93%.

One of the most notable improvements is the reduction in the misclassification of attackers as defenders:

- Logistic Regression misclassified 20 defenders were misclassified as attackers in Logistic Regression, whereas Random Forest slightly increased this to 7.
- Similarly, Logistic Regression misclassified 18 attackers as defenders, whereas Random Forest reduced this number to only 8.

This represents a major improvement, as attackers and defenders typically have very distinct statistical profiles. A model that confuses these roles suggests difficulties in differentiating offensive and defensive attributes, something that Random Forest has clearly improved upon.

Instead, the misclassifications that remain in Random Forest primarily involve midfielders being mistaken for attackers or defenders.

- Midfielders misclassified as attackers: Logistic Regression: 8 cases, Random Forest: 6 cases (slightly improved).
- Midfielders misclassified as defenders: Logistic Regression: 10 cases, Random Forest: 14 cases (slightly increased).

However, this is not a critical issue, as previously discussed, as midfielders often exhibit characteristics that overlap with both attackers and defenders. Specifically, the model slightly increases the confusion between midfielders and attackers, this is not an issue, as it aligns perfectly with real football dynamics. Many midfielders operate in the final third ("*trequartisti*", not by chance are they called "attacking midfielders"), contributing heavily to goals and assists, making it entirely logical that they could be categorized as attackers.

One particularly surprising and problematic result in the Random Forest model is the misclassification of goalkeepers as defenders:

- Logistic Regression misclassified 2 goalkeepers as defenders, while Random Forest misclassified 4 as defenders and 2 as attacker.

This is an illogical error, as goalkeepers are fundamentally distinct from outfield players. While this study does not focus primarily on goalkeepers, the fact that Random Forest incorrectly assigns more of them to the defender and attacker position suggests that the model is still not fully optimized. A well-functioning classification model should never confuse goalkeepers with defenders or attackers, as their statistical profiles are inherently different.

In summary, while both models achieve the same accuracy, Random Forest corrects unrealistic mistakes, particularly in distinguishing attackers from defenders, making it a more reliable model for positional classification. Instead of critical misclassifications, errors now occur where statistical overlap is natural, such as between midfielders and adjacent roles. This further confirms that the developed indices effectively capture player role characteristics, and that Random Forest is better suited than Logistic Regression in handling complex, non-linear relationships in football performance data.

However, the misclassification of goalkeepers suggests that Random Forest is not the final solution to validate the indices. This unexpected error highlights the need to explore another advanced ensemble model to validate indices. To further refine the validation and eliminate such mistakes, Gradient Boosting will be tested as the next step.

6.2.3 Gradient Boosting

The Gradient Boosting Classifier was implemented as the next step to further refine classification performance. Unlike Logistic Regression and Random Forest, Gradient Boosting is an ensemble learning method that builds multiple weak learners (decision trees) in a sequential manner, where each tree corrects the errors of the previous one (Friedman, 2001). This iterative approach allows Gradient Boosting to minimize prediction errors more effectively than traditional models. Instead of averaging predictions like Random Forest, Gradient Boosting focuses on reducing errors step by step, making it a powerful approach for tasks that require high precision (Hastie, Tibshirani & Friedman, 2009).

Compared to the previous models, Gradient Boosting offers a new classification approach, making it a more adaptive alternative to both Logistic Regression and Random Forest. While the earlier models demonstrated the validity of the developed indices, neither provided a fully optimized classification of player positions.

The Gradient Boosting model achieved an accuracy of 95%, improving upon the 93% accuracy of both Random Forest and Logistic Regression. However, the real improvement is not just in the numerical accuracy, but in the quality of the predictions and the logical consistency of errors.

One of the most significant improvements is the further reduction of unrealistic misclassifications. Gradient Boosting refines the classification of outfield players, producing errors that are more justifiable in a football context.

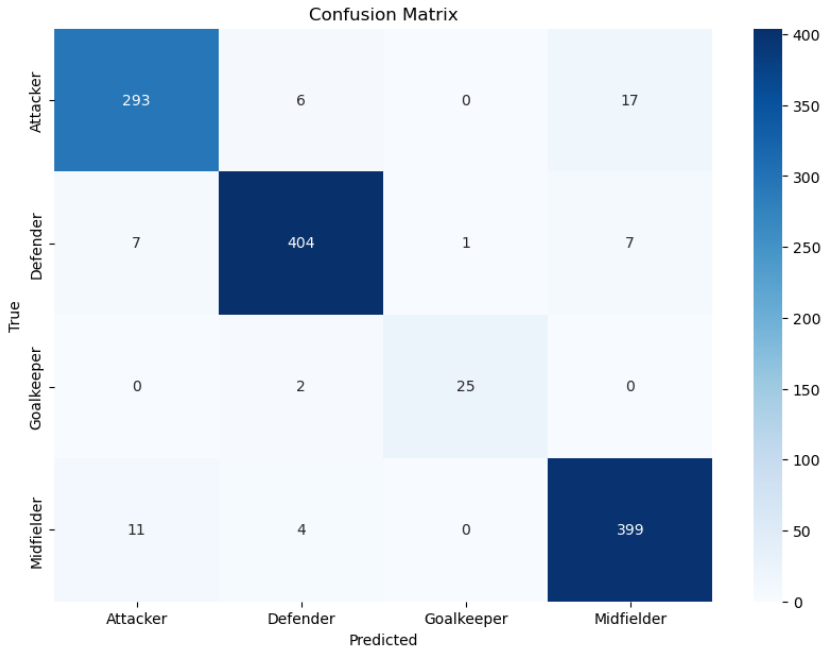


Figure 30 - Confusion Matrix of Gradient Boosting Model for Indices Validation

```
Gradient Boosting: Accuracy = 0.95
```

Analyzing the Confusion Matrix in *Figure 30,* it is possible to assess:

- Attackers are now classified with greater precision (293 correct): Only 6 were misclassified as defenders, compared to 8 in Random Forest and 18 in Logistic Regression. This confirms that the model better differentiates attacking players from defensive ones, reducing the biggest issue observed in the previous models.
- Defenders are still classified with high accuracy (404 correct), with only 7 being misclassified as midfielders, an error that is much more understandable for the reasons already explained. This result is the same of Random Forest and better than Logistic Regression (10). Regarding defenders being misclassified as attackers, the excellent result achieved with Random Forest (7) is maintained, which is already a significant improvement over Logistic Regression (20).
- Midfielders continue to be the most flexible category, with 4 misclassified as defenders. This represents an improvement even in the expected overlap between defensive midfielders and defenders, which was still present in Logistic Regression and Random Forest. While the model slightly increases again in the

confusion between midfielders and attacker (11), this is not an issue, as it aligns perfectly with real football dynamics, as already explicated.

- Moreover, the goalkeeper misclassification issue, which was a major flaw in Random Forest, has been almost entirely eliminated, with only 2 goalkeepers mistakenly classified, compared to the 6 total misclassified in Random Forest. Given that goalkeepers are not the primary focus of this study, this small margin of error is negligible.

In summary, unlike Logistic Regression, which struggled with linear assumptions and frequently misclassified attackers as defenders, Gradient Boosting captures more complex relationships between performance indices, making these misclassifications far less frequent. Compared to Random Forest, the sequential learning approach of Gradient Boosting corrects errors more effectively. Instead of treating all decision trees equally, it learns from its mistakes, refining its classification with each iteration. This is evident in how it reduces the most problematic errors, such as the confusion between attackers and defenders and the goalkeeper misclassification issue.

In conclusion, while the accuracy has even increased by 2%, the real gain is in the logical consistency of the model's decisions. The remaining misclassifications occur between adjacent roles which are naturally overlapping in real football scenarios. This suggests that Gradient Boosting is the most reliable model tested so far, as it successfully distinguishes between player positions without introducing unrealistic misclassifications.

6.2.4 Ensuring Model Robustness: Testing Gradient Boosting Across Multiple Random States

Despite the strong performance of all three models, we are not yet able to state with absolute certainty that the developed indices are truly representative. In statistical modeling and machine learning, it is considered best practice to test the same model multiple times using different random states[14], even when keeping the same train-test split. This is because a single accuracy score from a single experiment on a model does

[14] In statistics, the term *random state* refers to a parameter that controls the randomness involved in algorithms that utilize random processes. "`random_state`: controls the randomness of the estimator. When set to a specific integer, it ensures reproducibility by making the results deterministic across multiple runs. This is particularly useful for model validation and comparison." (Scikit-learn, n.d.)

not guarantee that the model's performance is consistently reliable. There is always a possibility that the observed accuracy is a result of a lucky random state, which happens to align particularly well with the dataset's structure. Machine learning models, especially ensemble methods like Gradient Boosting, can exhibit variance in their results depending on how the data is split, how randomly initialized parameters influence learning, and how decision boundaries are formed.

To eliminate the possibility of randomness influencing the results, the Gradient Boosting model, which has so far shown the best performance, was tested an additional nine times, each time with a different random state. The goal was to confirm that the indices developed in this study are truly representative and explanatory, hence that the satisfactory classification results are not due to a random chance. By evaluating the model's performance across these multiple trials, it becomes possible to assess its stability, verifying whether the classification accuracy remains consistently high and whether the logical consistency of errors is preserved.

6.2.5 Evaluating Model Stability: Results from Multiple Random States

We now proceed to examine the results of these nine additional Gradient Boosting runs, assessing whether the model's accuracy and misclassification patterns remain consistent across different random states, thereby confirming the robustness and reliability of the developed indices.

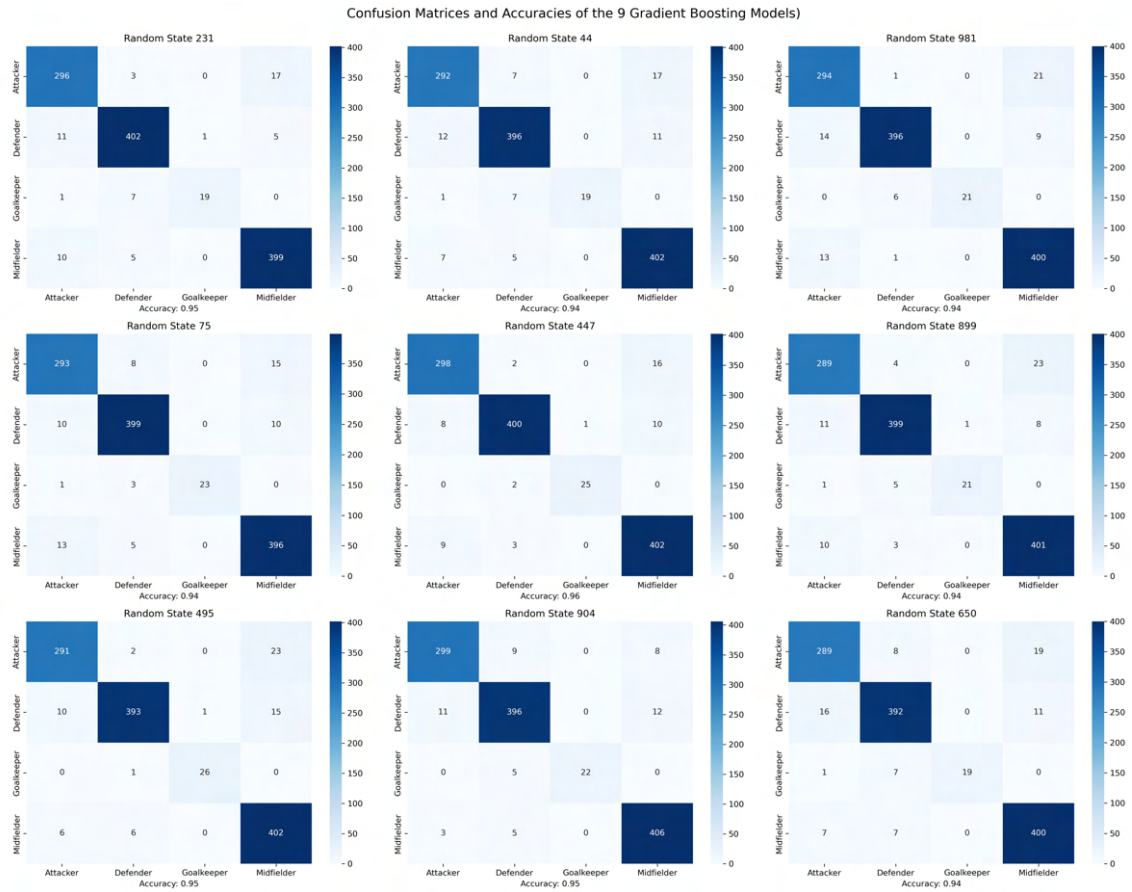


Figure 31 - Confusion Matrices and Accuracies of the 9 Gradient Boosting Models to Ensure Model Robustness

Summary of Accuracies:

```
Run 1 with Random State 631: Accuracy = 0.96
Run 2 with Random State 857: Accuracy = 0.94
Run 3 with Random State 773: Accuracy = 0.95
Run 4 with Random State 378: Accuracy = 0.95
Run 5 with Random State 576: Accuracy = 0.94
Run 6 with Random State 975: Accuracy = 0.95
Run 7 with Random State 127: Accuracy = 0.96
Run 8 with Random State 169: Accuracy = 0.93
Run 9 with Random State 527: Accuracy = 0.95

Mean Accuracy: 0.95
```

After running nine additional Gradient Boosting models with different random states, the results confirm that the model's performance remains consistently high, reinforcing the validity of the developed indices (*Figure 31*). The accuracy scores range between 0.94

and 0.96, with no extreme variations, indicating that the model's effectiveness is not dependent on a specific random state but is instead a robust and generalizable result.

The following are the main considerations that emerge from testing the Gradient Boosting model across multiple random states. Across the nine additional runs, the accuracy fluctuates slightly but remains within a narrow range (0.94 – 0.96). This confirms that the model's performance is not the result of a single "lucky" random state, but rather a stable outcome that consistently supports the classification of player roles. The confusion matrices displayed in *Figure 31* indicate that the types of misclassifications remain stable across different runs. The most common errors occur between midfielders and attackers or defenders and midfielders, which aligns with real-world positional overlaps. Importantly, unrealistic errors remain minimal and consistent across all random states.

To further consolidate the analysis, a mean confusion matrix (*Figure 32*) was computed by averaging the results of the nine Gradient Boosting models tested with different random states. This visualization provides a comprehensive overview of how the model performs on average, eliminating the variability introduced by individual runs and offering a clearer picture of classification consistency.

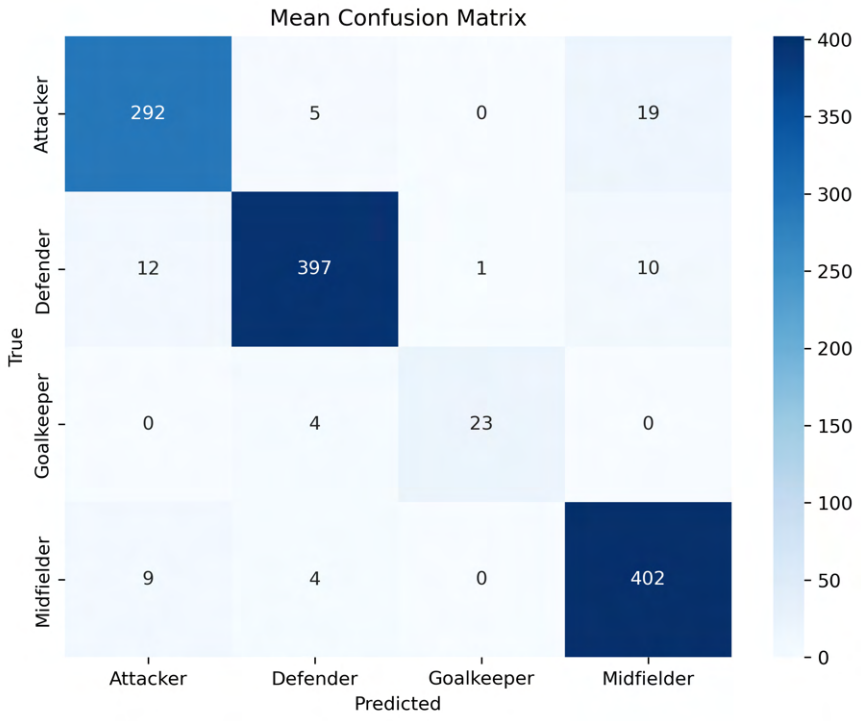


Figure 32 - Mean Confusion Matrix of the 9 Gradient Boosting Models

```
Multiple Gradient Boosting Models Mean Accuracy = 0.95
```

These are, therefore, the final considerations.

- The misclassification of attackers as defenders and vice versa has been significantly reduced, confirming that the model successfully captures the most critical distinctions between offensive and defensive roles.
- Defenders and midfielders maintain high classification precision, with occasional misclassifications that align with the natural tactical overlap.
- Goalkeeper misclassification has been reduced to an insignificant level (4 cases), demonstrating that even for a role not central to this study, the indices remain distinct enough to prevent major errors.
- The only persistent misclassifications involve midfielders and attackers (9 and 19 cases, respectively). However, this is a logical and expected misclassification.

6.3 Conclusions of the Validation Process

Across all models tested—Logistic Regression, Random Forest, and Gradient Boosting—the classification performance has been consistently high, demonstrating that the developed indices are effective in distinguishing player roles. The robustness of these indices has been further reinforced by testing Gradient Boosting across multiple random states, where accuracy remained stable between 92% and 95%, with no major fluctuations.

If the indices were weak or poorly constructed, the performance would have shown greater instability across different iterations. Instead, the results confirm that the indices are reliable, representative, and predictive, successfully capturing meaningful patterns in player performance. The errors observed have been logical, aligning with real football dynamics, rather than being random inconsistencies.

Another notable finding is that was face one of the most significant initial challenges described: despite the role-based normalization of indices, the model still achieved remarkably high accuracy. Initially, it was expected that normalizing indices within each position rather than across all players would make classification more challenging. This should have theoretically made it harder for the model to differentiate between roles. However, the results indicate the opposite—the model effectively captured role-specific patterns, demonstrating that the indices successfully encode distinct and meaningful characteristics that differentiate player types.

The indices have now been fully validated. They effectively describe a player's characteristics and position-specific attributes, proving to be a solid analytical foundation. With this validation step complete, the indices are now ready to be applied to the core objective of this study: the development of a scouting algorithm capable of identifying similar players based on these indices.

6.4 Benchmarking Against Raw Features: Testing Models Without Indices

6.4.1 Why Benchmarking?

To further assess the effectiveness and added value of the developed indices, the same three models—Logistic Regression, Random Forest, and Gradient Boosting—were tested again, but this time using the original performance variables, rather than the computed indices. The target variable (y) remains the player position, while the input features (X) now consist of the raw, normalized performance metrics, without any pre-aggregated indices.

The goal of this experiment is to determine whether the indices truly enhance the classification process or if other results could have been achieved using only the raw features. If the indices are meaningful, the performance on raw data should not be significantly different, neither substantially better nor worse. This would indicate that the indices successfully capture and structure the key information present in the raw features without introducing bias or losing valuable details. Rather than making the model inherently more powerful, the indices serve to organize the data in a way that reduces noise, eliminates redundancy, and simplifies interpretation and comparison, ultimately making the scouting process more effective. By contrast, using the raw features forces the model to extract patterns directly from a larger and potentially noisier dataset, without the benefit of pre-computed indicators that encapsulate key performance attributes.

Theoretically, it is possible to hypothesize the following scenarios:

- If the raw feature models perform significantly better, it would suggest that the indices fail to capture relevant information and may have oversimplified the data.
- If the performance remains the same or is only marginally better/worse, it will confirm that the indices effectively compress relevant player attributes into meaningful categories, without causing information loss.

- If the raw feature models perform worse, this would further validate the usefulness of the indices, showing that aggregating key statistics into structured indicators enhances classification accuracy.

6.4.2 Analysis of Results: Raw Features vs. Performance Indices

The results of applying Logistic Regression, Random Forest, and Gradient Boosting using only the raw, normalized performance features are summarized in the confusion matrices shown in *Figure 33*.

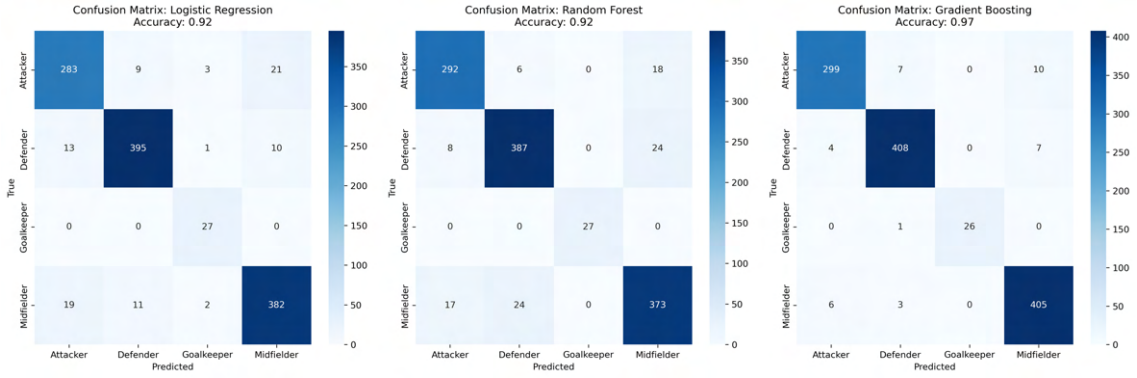


Figure 33 – Confusion Matrices of Logistic Regression, Random Forest, and Gradient Boosting Models with Raw Features

```
Logistic Regression with raw features: Accuracy = 0.92
Random Forest with raw features: Accuracy = 0.92
Gradient Boosting with raw features: Accuracy = 0.97
```

The following key observations emerge from comparing the performance of the models trained on raw features:

1) Logistic Regression (Accuracy: 92%)
   - The model struggles with the same issues observed in previous tests, particularly in misclassifying attackers as defenders (21 cases) and midfielders as attackers (19 cases).
   - Interestingly, goalkeepers were classified perfectly (27/27 correct), suggesting that their raw statistics remain distinct enough for the model to separate them from outfield players.

Overall, the results suggest that Logistic Regression performed better when using the indices rather than the raw features. This is likely because the indices, being aggregated from multiple performance metrics, helped smooth out variability and introduced more linear patterns in the data. In contrast, the raw features have a higher degree of non-linearity, making it even harder for Logistic Regression, being a linear model, to effectively separate player roles.

2) Random Forest (Accuracy: 92%):

   Overall performance remains nearly unchanged, with only a 1% difference in accuracy, indicating that the indices did not cause any loss of valuable information. The model's ability to classify player positions is similar whether using raw features or structured indices. However, the indices appear to provide a more structured and interpretable representation of player attributes, which helps refine positional distinctions, particularly in cases where raw statistical overlap makes classification more challenging.

   - Defender misclassification increased, with 24 defenders misclassified as midfielders, compared to 10 in the index-based model. This suggests that the indices helped refine the distinction between defenders and midfielders, which is now less clear when using raw features.
   - Attacker classification slightly worsened, with 18 attackers misclassified as defenders, compared to only 7 in the index-based model. This further supports the effectiveness of the indices in separating offensive and defensive roles.
   - Midfielders remain the most flexible category, but their classification accuracy is nearly identical in both models, suggesting that the indices primarily enhanced the distinction between extreme roles (attackers and defenders) rather than midfielders, who naturally overlap between different playing styles.

   These results confirm that while Random Forest is capable of learning directly from raw features, the aggregated indices provide a structured advantage by improving the separation of key player roles.

3) Gradient Boosting remains the best-performing model but showing a clear advantage in handling raw data compared to Logistic Regression and Random Forest. Notably, the model trained on raw features achieved a 2% increase in overall accuracy compared to the best Gradient Boosting model trained on indices (97% vs. 95%).

- Defenders achieved the highest precision (408 correct), with only 7 misclassified as midfielders, maintaining strong performance. This result is slightly better than the mean confusion matrix (397 correct, 10 misclassified as midfielders).
- Attackers saw their best classification performance (299 correct), with only 10 misclassified as midfielders. This marks a slight improvement over the mean confusion matrix (292 correct).
- Midfielders remained highly accurate (405 correct), with only 6 misclassified as attackers and 3 as defenders. This is comparable to the mean confusion matrix (402 correct), showing that both raw features and indices represent midfielders effectively.
- Goalkeepers saw an improvement in classification accuracy (26 correct, 1 misclassified as a defender), compared to the mean confusion matrix (23 correct, 4 misclassified as defenders). While this difference is not so significant, it suggests that indices may slightly relief in differentiating goalkeepers from outfield players.

6.4.3 The Importance of Constructing Performance Indices

Given that the Gradient Boosting model achieved slightly higher accuracy when trained on raw features, a legitimate question arises: why not use the original performance variables instead of the aggregated indices in the scouting model? If raw features provide a (marginally) better classification of player positions, would not they also be more effective for identifying similar players?

The answer lies in the fundamental purpose of this study. The goal is not just to build the most accurate model but to create a scouting algorithm that can effectively compare players and identify similar profiles. For this purpose, the interpretability and comparability provided by the indices are far more valuable than a slight increase in classification accuracy of variables validation.

Raw performance variables, while rich in information, can be highly fragmented, redundant, and difficult to interpret. Football performance is composed of multiple interrelated action: passing, shooting, tackling, dribbling, which cannot be fully understood in isolation. By aggregating related statistics into structured indices, the model gains a more holistic and interpretable representation of a player's abilities.

Moreover, indices reduce dimensionality, making player comparisons more practical. Instead of comparing hundreds of raw metrics, scouts and analysts can compare players based on concise, high-level attributes such as playmaking capability, finishing ability, or tackling efficiency. This structured approach allows for a more intuitive and actionable scouting process, where players can be matched based on well-defined performance profiles rather than a chaotic mix of isolated statistics.

Additionally, indices improve model stability and generalizability. Raw features may introduce more variability across different datasets, making the scouting model less consistent when applied to different leagues, competitions, or player samples. Indices provide a standardized framework, ensuring that the model remains reliable even in varying scouting environments.

In summary, while raw features may have resulted in a small increase in classification accuracy, the use of indices in the scouting model is crucial for enhancing interpretability, improving player comparability, reducing redundancy and ensuring a more practical application in talent identification.

6.4.4 Final Assessment: Validating the Role of the Indices

The results obtained from training models on raw features align with the expectations set forth at the beginning of this analysis. While the Gradient Boosting model achieved a slightly higher accuracy when using raw performance variables, the improvement was marginal (+2%), confirming that the indices effectively compress relevant player attributes into meaningful categories without causing information loss.

This outcome validates the structured approach used in creating the indices. Rather than oversimplifying the data or omitting key performance details, the indices provide a concise yet comprehensive representation of player attributes, making patterns easier to detect and enhancing interpretability. This ensures that, even with a slight trade-off in accuracy, the scouting model will benefit from a more practical and insightful framework for identifying similar players, supporting better decision-making in talent identification and recruitment.

CHAPTER 7:
DESIGNING A PRACTICAL AND EFFICIENT SCOUTING MODEL

7.1 Evaluating Model Options for Player Scouting

7.1.1 Overview

After successfully constructing and validating the performance indices in the previous chapter, it is now possible to develop the core scouting model of this study. The goal, as already enlightened, is to create a machine learning algorithm capable of identifying similar players based on these indices. By leveraging the validated and role-normalized indices, the model will analyze the relationships between players and return the most statistically similar profiles.

The scouting model is designed to take a player's unique ID as input and return a list of the most similar players, based either on all indices or on a user-selected subset of indices. This flexibility is crucial because, in a real-world scouting scenario, player comparisons can serve different objectives:

- *Full-Profile Similarity Search*:
  The model considers all indices to find players who are similar in an overall sense, meaning their performance characteristics closely look like the selected player across all aspects of the game. This can be useful for general scouting, where teams want to identify well-rounded replacements or alternative transfer targets.

- *Role-Specific Similarity Search*:
  In many cases, a club is not looking for a perfect clone of a player but rather someone who excels in specific attributes. For example, if a team wants to replace a highly explosive winger who has contributed a significant number of goals and assists, it could not be relevant to compare defensive or playmaking indices.

This dual approach ensures that the model provides actionable insights tailored to different scouting needs. As we will see, the model performs well even when considering

all indices, delivering highly realistic results. The recommended scouting methodology proposed in this study is to first analyze a player's full-profile similarity and then refine the search by selecting the most relevant indices, allowing for a more targeted and role-specific analysis.

7.1.2 Finding the Right Model for Player Comparison

Before selecting the best approach for our scouting model, it is essential to evaluate different types of machine learning models that could be used to measure player similarity. The challenge is to choose an algorithm that can effectively compare players based on their performance indices, ensuring that the retrieved matches are accurate, interpretable, and actionable for scouting purposes.

What kind of model is needed for player similarity? The scouting model does not require a classification algorithm (e.g., Logistic Regression, Random Forest, Gradient Boosting), as its purpose is not to predict a pre-label variable (like "player positions" before) but to retrieve the most similar players based on their performance profiles. Instead, we need a distance-based retrieval method that can effectively identify players with comparable attributes.

A suitable model should:

- Measure numerical similarity based on structured indices.
- Handle different subsets of indices dynamically (e.g., comparing players based on specific traits like finishing or explosiveness).
- Be interpretable, so scouts can understand the results.

With these criteria in mind, several approaches were theoretically evaluated.

I. Clustering Models (e.g., K-Means, DBSCAN)

- Why Consider It? Clustering algorithms can group players into predefined categories, revealing hidden structures in the data.
- Why not consider it? Despite their ability to identify patterns in data, clustering algorithms are not well-suited for individualized player comparison. Since clustering assigns each player to a fixed group, it does not allow for dynamic

similarity searches based on a specific reference player. Additionally, if new players are added to the dataset, the clusters must be recomputed, making the approach less practical for real-time scouting applications.

II. Dimensionality Reduction Models (e.g., PCA)

- Why Consider It? Dimensionality reduction techniques are useful for compressing high-dimensional data into a smaller space while preserving as much information as possible. This can make similarity comparisons efficient and provide valuable insights into player relationships through visualization techniques.
- Why not consider it? The primary drawback of dimensionality reduction methods is the loss of interpretability. When raw performance metrics are transformed into a lower-dimensional space, their original meaning is obscured, making it difficult to understand the reasoning behind player similarities. Moreover, these methods are not designed for direct similarity retrieval but rather for data exploration, meaning that additional steps would be required to implement them in a scouting system.

III. Deep Learning Models (e.g., Neural Networks, Autoencoders)

- Why consider it? Deep learning models have the capacity to learn highly complex, non-linear patterns. This could potentially allow for more advanced similarity search, capturing intricate player attributes that traditional methods might overlook.
- Why not consider it? Despite their potential, deep learning models introduce unnecessary complexity to this problem. The dataset size does not justify the use of deep networks, as simpler models can already achieve high accuracy. Additionally, deep learning models are often considered "*black boxes*", meaning that their decision-making process is difficult to interpret. This lack of transparency would make it challenging for scouts to understand why and how a certain player was suggested. Furthermore, deep learning models require

significant computational resources and fine-tuning, making them inefficient for real-time scouting applications.

IV. K-Nearest Neighbors (KNN) – The Optimal Choice

- Why consider it? Unlike some previously discussed methods, K-Nearest Neighbors (KNN) in this application would have been used as an instance-based learning algorithm that does not require a training phase. Instead, it dynamically searches for the most similar players every time a query is made. This makes it particularly well-suited for scouting, where the goal is to find players who resemble a given reference player in terms of performance attributes. Additionally, KNN offers a high degree of flexibility, allowing to search for overall similarity or focus on specific attributes.
- Why not consider it? One drawback is that KNN does not scale well with very large datasets, as it requires computing distances for every query in real-time. Another potential limitation is that KNN treats all features as equally important unless explicitly weighted. Lastly, KNN is sensitive to feature scaling, meaning that if features are on different scales, the distance calculations may be skewed.

KNN emerged as the best choice because it meets all the key requirements for this study. First, it provides high interpretability, as the retrieved players are simply those with the smallest mathematic distance from the reference player. This allows scouts to easily understand why certain players were suggested. Additionally, all the main limitations of KNN are effectively mitigated in this study. The dataset size remains manageable (around 4 thousand records), preventing computational inefficiencies, and feature scaling is not an issue since all indices have already been normalized. Furthermore, the model allows users to select specific indices for comparison, addressing the potential concern that KNN treats all features equally. This is not a limitation but an advantage in our study. The goal is precisely to treat all indices equally by default, ensuring that similarity is based on a holistic assessment of a player's attributes. If the user is particularly interested in specific characteristics, the comparison can be customized by selecting only the most relevant indices, ensuring that the model remains highly adaptable to different scouting needs.

In the appendix, K-Nearest Neighbors is defined in detail. The theoretical foundations of the algorithm, including its core principles and general functioning, are briefly explored, while a more in-depth analysis is provided on its theoretical application to this specific dataset and the problem at hand. These aspects are detailed in the appendix, allowing the discussion here to remain focused on its practical implementation within this scouting model.

7.2 From Input to Output: From User Query to Actionable Model Results

7.2.1 A User-friendly approach: Simplifying Player Search

The next step is to discuss how the model is practically implemented. The primary objective is to create a user-friendly system that allows easily for player searches and comparisons. To retrieve similar players, the model requires the `player_id` as input, since it is the only unique identifier that distinguishes players unambiguously. Relying solely on a player's last name would lead to potential ambiguities due to cases of homonyms[15]. However, a fundamental issue arises: users typically do not know the unique player ID stored in the dataset, so they need to search using the player's last name, which is the most common way footballers are recognized. Therefore, the primary objective is to create a user-friendly system that allows for player's ID research through the last name.

However, even surname-based searches introduce several challenges:

- *Spelling variations*: Many footballers, especially those from diverse linguistic backgrounds, have surnames that are difficult to spell correctly (e.g., Khvicha Kvaratskhelia).
- *Special characters*: Some names contain accents, apostrophes, or unique symbols that may not be readily available on a user's keyboard (e.g., Hakan Çalhanoğlu).

[15] For instance, common surnames like Rossi, Fernandes or Rodríguez could correspond to several different players across various leagues and seasons.

- *Dataset formatting*: Some players' names are stored with their full legal surname, which may not match the commonly known version (e.g., Kylian Mbappé appears in the dataset under `lastname` as "Mbappe Lottin").

To make player searches more accessible, a fuzzy string matching stage (Navarro, 2001) was implemented to allow users to search for players without needing to enter their exact names. At its core, a fuzzy string matching routine handles imprecise searches by finding possible matches based on the similarity scores between a query (i.e., the user query) and a set of target strings (i.e., the player surnames contained in the dataset).
Below are the three aforementioned example cases (*Figure 34*, *Figure 35* and *Figure 36*): the outputs shown are the actual results generated by the model. For each query, the system provides:

- *User Input*: The string entered by the user.
- *Closest Matched Last Name*: The best match found in the dataset using fuzzy logic.
- *Summary DataFrame*: A table displaying relevant details about all matching players, including:
    - Player ID (the unique identifier required for KNN retrieval).
    - Last Name (as stored in the dataset).
    - Full Name (for additional clarity).
    - Nationality (to differentiate between players with similar names).
    - Club, League, and League Country (to ensure the correct player is selected).

```
Last name searched by the user: 'mbappe'

Most similar last names found:
- 'mbappe lottin' (similarity name: 90.0%)
- 'mbappe lottin' (similarity name: 90.0%)
- 'ba' (similarity name: 90.0%)
```

- 'pepe' (similarity name: 77.14285714285715%)
- 'mbaye' (similarity name: 72.72727272727273%)
- 'mbaye' (similarity name: 72.72727272727273%)
- 'rapp' (similarity name: 67.5%)
- 'kappel' (similarity name: 66.66666666666667%)
- 'bassey ughelumba' (similarity name: 60.000000000000001%)
- 'lomba neto' (similarity name: 60.000000000000001%)

player_id	lastname	name	player_nationality	league_name	league_country	team_name
386287	Mbappe Lottin	E. Mbappe	France	Ligue 1	France	Paris Saint Germain
278	Mbappe Lottin	Kylian Mbappe	France	Ligue 1	France	Paris Saint Germain
20968	Ba	O. Ba	Senegal	Süper Lig	Turkey	Istanbul Basaksehir
3246	Pepe	N. Pepe	Côte d'Ivoire	Süper Lig	Turkey	Trabzonspor
14444	Mbaye	M. Mbaye	Senegal	La Liga	Spain	Cadiz
403883	Mbaye	M. Mbaye	Senegal	Ligue 1	France	Metz
24843	Rapp	N. Rapp	Germany	Bundesliga	Germany	Werder Bremen
62092	Kappel	L. Kappel	Suriname	Süper Lig	Turkey	Pendikspor
152967	Bassey Ughelumba	C. Bassey	Nigeria	Premier League	England	Fulham
1864	Lomba Neto	Pedro Neto	Portugal	Premier League	England	Wolves

Figure 34 – Last name similarity search results for Kylian Mbappé

Last name searched by the user: '*calha*'

Most similar last names found:

- 'calhanoglu' (similarity name: 90.0%)
- 'calabria' (similarity name: 80.0%)
- 'calafiori' (similarity name: 80.0%)
- 'dos santos magalhaes' (similarity name: 72.0%)
- 'escalante' (similarity name: 72.0%)
- 'magalhaes ribeiro ferreira' (similarity name: 72.0%)
- 'can' (similarity name: 72.0%)
- 'pereira magalhaes dos santos' (similarity name: 72.0%)
- 'gomes ferreira magalhaes' (similarity name: 72.0%)
- 'lima magalhaes' (similarity name: 72.0%)

player_id	lastname	name	player_nationality	league_name	league_country	team_name
1640	Calhanoglu	H. Calhanoglu	Türkiye	Serie A	Italy	Inter
1627	Calabria	D. Calabria	Italy	Serie A	Italy	AC Milan
157052	Calafiori	R. Calafiori	Italy	Serie A	Italy	Bologna
22224	dos Santos Magalhaes	Gabriel Magalhaes	Brazil	Premier League	England	Arsenal
47388	Escalante	G. Escalante	Argentina	La Liga	Spain	Cadiz
41320	Magalhaes Ribeiro Ferreira	Andre Ferreira	Portugal	La Liga	Spain	Granada CF
864	Can	E. Can	Germany	Bundesliga	Germany	Borussia Dortmund
159511	Pereira Magalhaes dos Santos	Gabriel Pereira	Brazil	Primeira Liga	Portugal	GIL Vicente
310094	Gomes Ferreira Magalhaes	Miguel Maga	Portugal	Primeira Liga	Portugal	Guimaraes
41089	Lima Magalhaes	Matheus	Brazil	Primeira Liga	Portugal	SC Braga

Figure 35 - Last name similarity search results for Hakan Çalhanoğlu

---

Last name searched by the user: 'kvara'

Most similar last names found:

- 'kvaratskhelia' (similarity name: 90.0%)
- 'kara' (similarity name: 88.88888888888889%)
- 'kara' (similarity name: 88.88888888888889%)
- 'karafiat' (similarity name: 80.0%)
- 'kamara' (similarity name: 72.72727272727273%)
- 'varane' (similarity name: 72.72727272727273%)
- 'kamara' (similarity name: 72.72727272727273%)
- 'khadra' (similarity name: 72.72727272727273%)
- 'varela' (similarity name: 72.72727272727273%)
- 'karaca' (similarity name: 72.72727272727273%)

player_id	lastname	name	player_nationality	league_name	league_country	team_name
483	Kvaratskhelia	K. Kvaratskhelia	Georgia	Serie A	Italy	Napoli
50235	Kara	A. Kara	Türkiye	Süper Lig	Turkey	Kasimpasa
119139	Kara	E. Kara	Austria	Süper Lig	Turkey	Samsunspor
66037	Karafiat	O. Karafiat	Czechia	Czech Liga	Czech-Republic	Mlada Boleslav
1904	Kamara	B. Kamara	France	Premier League	England	Aston Villa
742	Varane	R. Varane	France	Premier League	England	Manchester United
22007	Kamara	H. Kamara	Côte d'Ivoire	Serie A	Italy	Udinese
145476	Khadra	R. Khadra	Germany	Ligue 1	France	Reims
278375	Varela	A. Varela	Argentina	Primeira Liga	Portugal	FC Porto
49941	Karaca	E. Karaca	Türkiye	Süper Lig	Turkey	Alanyaspor

Figure 36 - Last name similarity search results for Khvicha Kvaratskhelia

---

The first case highlights a homonym issue, specifically involving Kylian Mbappé and his brother Ethan Mbappé, both of whom played for Paris Saint-Germain (PSG) in the 2023/24 season. Through fuzzy matching, the model correctly retrieves first both Kylian and Ethan, ensuring that the user can choose the intended player. The player ID remains the decisive element for disambiguation. Moreover, since the dataset stores Kylian's last name as "Mbappé Lottin", a user searching for "mbappe" will not obtain an exact match without an approximate approach. The subsequent cases further demonstrate how the model resolves difficult searches involving non-Latin characters (e.g., "Çalhanoğlu") and complex transliterations (e.g., "Kvaratskhelia"), reinforcing its practical usefulness in a scouting context.

With this functionality in place, the system successfully ensures that users can precisely retrieve the correct player's last name. Once the player's approximate name is entered, the fuzzy matching algorithm provides the closest matches, displaying their corresponding player ID. The player ID can now be fed into the scouting model, enabling the retrieval of the most similar players based on the performance indices.

7.2.2 User Input

Once the user has obtained the unique player ID, they can immediately use it as input to retrieve the most similar players. As soon as the player ID is entered, the model processes the query and returns the results instantly.

As previously mentioned, the user has the flexibility to exclude specific indices from the comparison, particularly those that may not be relevant for the specific analysis. By default, all indices are included in the similarity search, ensuring a comprehensive comparison of players. However, if certain attributes are deemed unnecessary, these can be easily excluded by commenting out the respective variables in the Python code using the # symbol[16]. For instance, the default setup considers all performance indices:

```
index_columns = [
'overall_offensive_strengh_index',
'overall_defensive_strengh_index',
'playmaking_index',
'player_efficiency_index',
```

[16] In Python, any text following # on the same line is ignored during execution. This is commonly used for adding explanatory notes within the code or temporarily disabling specific lines without deleting them.

```
'shooting_efficiency_index',
'passing_efficiency_index',
'tackling_efficiency_index',
'discipline_index',
'physicality_index',
'offensive_contribution_index',
'consistency_index',
'clutch_performance_index',
'finishing_ability_index',
'explosiveness_index',
'strategic_play_index'
]
```

However, if the goal is to focus on attacking attributes while disregarding defensive and disciplinary aspects, the user can modify the selection as follows:

```
index_columns = [
    'overall_offensive_strengh_index',
    #'overall_defensive_strengh_index',
    'playmaking_index',
    'player_efficiency_index',
    'shooting_efficiency_index',
    'passing_efficiency_index',
    #'tackling_efficiency_index',
    #'discipline_index'
    'physicality_index',
    'offensive_contribution_index',
    'consistency_index',
    'clutch_performance_index',
    'finishing_ability_index',
    'explosiveness_index',
    'strategic_play_index'
]
```

In this case, with the # symbol, the variables `overall_defensive_strengh_index`, `tackling_efficiency_index`, and `discipline_index` have been excluded.

Another customizable parameter in this model is *k* (number of nearest neighbors), which determines the number of similar players retrieved. By default, in this implementation and in all subsequent examples, *k* is set to 20, meaning the 20 most similar players to the

target will be displayed. However, this is not a fixed constraint, and the user can easily modify this value based on their scouting needs. A scout conducting a more in-depth analysis may wish to expand the search beyond the top 20 to include a broader set of potential targets, while in other cases, narrowing it down to a smaller number may be preferable for focusing only on the closest matches. For visualization purposes, in this study and in the following examples, $k = 20$ has been chosen to ensure clarity in plots and comparisons. However, increasing or decreasing k does not alter the model's accuracy, but the higher the value of K, the less similar the last retrieved results will be to the reference player, as the search includes progressively more distant neighbors. Since KNN developed is not a classification or regression model but a similarity-based retrieval system, adjusting k simply determines how many nearest neighbors are retrieved during the search process. The underlying similarity calculations remain unchanged, ensuring that the model consistently identifies the most comparable players regardless of the chosen value of k.

7.2.3 Output Structure: How Results Are Displayed

The output is a structured DataFrame containing 20 rows, each representing a player ranked by similarity. This table provides essential details, including the player's unique ID, name, position, league, team, age, nationality, and the computed similarity metrics.

Below is an example of the model's output for the previously discussed case: Khvicha Kvaratskhelia. The following table (*Figure 37*) displays the 20 most similar players based on all the indices.

player_id	name	games_position	age	player_nationality	league_name	league_country	team_name	euclidean_distance	similarity_score	height_cm	weight_kg	overall_offensive_strengh_index	overall_defe
483	K. Kvaratskhelia	Attacker	23	Georgia	Serie A	Italy	Napoli	0.0	100.0	183	76	71.12232953808231	88.1923440
266657	Sávio	Attacker	20	Brazil	La Liga	Spain	Girona	37.01961655444794	86.06976385011613	179	66	48.57543216467349	84.0599920
290549	J. Bakayoko	Attacker	21	Belgium	Eredivisie	Netherlands	PSV Eindhoven	53.488579405259884	79.87260237174014	179	76	56.219128580251684	71.8350376
323936	M. Soulé	Attacker	21	Argentina	Serie A	Italy	Frosinone	57.639996729468535	78.31045156993781	182	0	55.448481103530334	94.1558490
22236	Rafael Leão	Attacker	25	Portugal	Serie A	Italy	AC Milan	58.518053829954646	77.98004450737778	188	81	49.50401093247668	65.1467568
138787	A. Gordon	Attacker	23	England	Premier League	England	Newcastle	59.72629570298258	77.52539110510112	183	72	58.15579439297933	75.0688322
2489	L. Díaz	Attacker	27	Colombia	Premier League	England	Liverpool	63.29696727447181	76.18177107782816	178	65	56.86017517462678	69.0423324
47294	I. Williams	Attacker	30	Ghana	La Liga	Spain	Athletic Club	65.20022012013399	75.46558965671544	186	81	64.42335234615861	65.1405623
291024	Hélio Varela	Attacker	22	Portugal	Primeira Liga	Portugal	Portimonense	68.68914449898972	74.1527305557827	176	69	47.81140309255768	77.0849379
1460	B. Saka	Attacker	23	England	Premier League	England	Arsenal	69.93389309099499	73.68433991731226	178	72	69.33853833572414	96.7915293
99576	O. Sahraoui	Attacker	23	Norway	Eredivisie	Netherlands	Heerenveen	70.0484041979172	73.64125014735569	170	65	46.84448903601828	69.9241962
1165	Matheus Cunha	Attacker	25	Brazil	Premier League	England	Wolves	70.60378214693276	73.43226510908968	183	76	60.8973301662443	65.5881191
2215	A. Laurienté	Attacker	26	France	Serie A	Italy	Sassuolo	71.29117394300094	73.17360413585568	171	59	50.34241824453986	63.7608718
644	L. Sané	Attacker	28	Germany	Bundesliga	Germany	Bayern München	73.73323092009375	72.25467429409625	183	80	46.92232043364705	58.1089809
10500	Pepê Aquino	Attacker	27	Brazil	Primeira Liga	Portugal	FC Porto	75.38058811934799	71.63478470731268	175	69	39.4966436264843	83.1324360
284324	A. Garnacho	Attacker	20	Argentina	Premier League	England	Manchester United	75.87327567637362	71.44938964771279	180	73	49.59268580043341	60.4750435
181812	J. Musiala	Attacker	21	Germany	Bundesliga	Germany	Bayern München	76.01555194286188	71.39585203238417	184	72	45.98183524560757	67.9151670
10009	Rodrygo	Attacker	23	Brazil	La Liga	Spain	Real Madrid	76.76270112643965	71.11470475061469	174	64	55.280219580354874	57.4761984
386828	Lamine Yamal	Attacker	17	Spain	La Liga	Spain	Barcelona	77.08627722741757	70.99294521017075	180	72	44.77093020312372	82.9890809
2799	A. Guðmundsson	Attacker	27	Iceland	Serie A	Italy	Genoa	80.15865083223991	69.83683140243053	177	80	46.46307557470289	60.0651785

Figure 37 – Dataframe of the Top 20 Similar Attackers to K. Karatskhelia

The output is presented as a DataFrame consisting of 20 rows, ranked from the most to least similar player based on the computed similarity (the Euclidean distance). Each row contains key details such as:

- Player ID
- Name
- Position
- League
- Team
- Age
- Nationality
- Performance indices used in the similarity computation

To provide a more interpretable comparison of player similarity, the output also includes two key numerical metrics:

- Euclidean Distance
- Similarity Score

The Euclidean Distance represents the direct numerical distance between the target player, and each retrieved similar player within the multi-dimensional feature space[17]. This distance measure, computed by the KNN model, is crucial because it does not just indicate the order of similarity but also quantifies how similar each suggested player is to the reference player. A smaller Euclidean distance signifies a higher degree of similarity, meaning that the suggested player closely matches the target player's profile. Conversely, a larger Euclidean distance indicates a weaker resemblance. While Euclidean distance is highly informative, it has a notable limitation in terms of interpretability. Since lower values indicate higher similarity, the measure can feel counterintuitive, especially when visualizing results. For instance, in a bar plot ranking similar players, those with the highest similarity would have the shortest bars, while those with lower similarity would have longer bars. This can make the results less intuitive to interpret immediately.

[17] For more details on the calculation of Euclidean distance and the metric itself, please refer to the appendix.

To overcome this, a Similarity Score has been introduced to provide a more user-friendly representation of player similarity. This score is computed as:

$$\text{similarity score} = \left(1 - \frac{\text{euclidean\_distance}}{\text{max\_distance}}\right) \times 100$$

This transformation normalizes the Euclidean distance into a percentage-based similarity metric, ensuring that:

- 100% represents a perfect match (the target player compared to themselves)
- Higher values indicate greater similarity.
- Lower values indicate weaker similarity.

This approach inverts the interpretation of distance, making it more intuitive: higher similarity scores mean players are more alike, aligning with the natural way users expect similarity rankings to be presented.

7.2.4 Overview Visualization: A Clearer Representation of the Output

As previously illustrated, the output is a DataFrame containing all the performance indices. However, while the DataFrame presents comprehensive data, it lacks immediate interpretability. Simply scanning through the table requires constant back-and-forth comparisons between names and indices, making it difficult for a scout to extract quick insights about player similarity.

To enhance readability and provide a clearer understanding of which players are most similar and how much similar they are, an interactive bar plot (implemented again with the library *Plotly*) is generated alongside the DataFrame. This type of visualization ranks the k most similar players based on the similarity score, offering an intuitive representation of similarity levels.

The bar plot displayed in *Figure 38* already highlights how much more effective the visualization is compared to the previous example of DataFrame output.

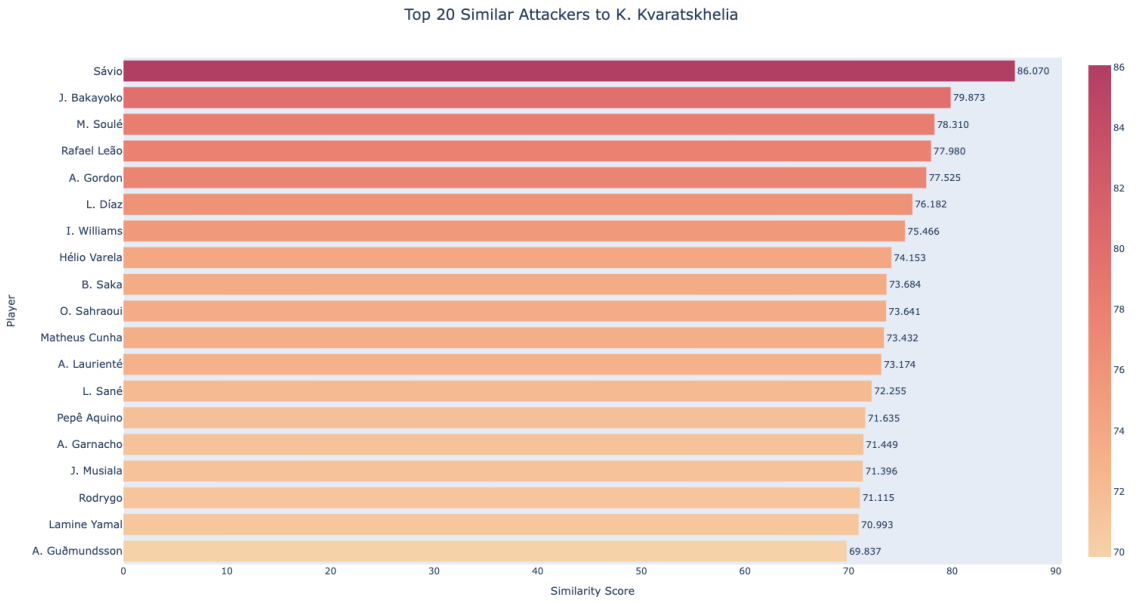


Figure 38 – Bar plot of the Top 20 Similar Attackers to K. Kvaratskhelia

These are the same results of the previous output but presented in a far more intuitive and visually accessible format. Instead of manually scanning rows and columns to compare numerical values, users can immediately grasp the similarity rankings, and thanks to Plotly's interactivity, they can simply hover over any of the bars to view the same detailed data from the previous DataFrame. This includes both general player information (such as age, nationality, team and league) and all performance indices used for the similarity calculation. In *Figure 39* is the same example, showing the tooltip when hovering over Luis Díaz's result.

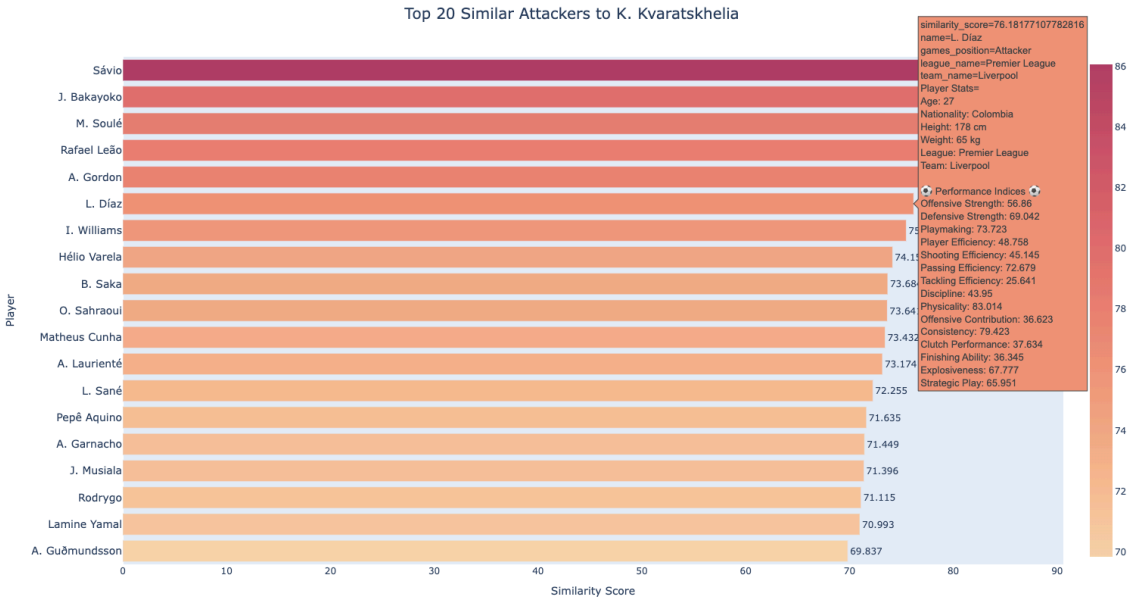


Figure 39 - Top 20 Similar Attackers to K. Kvaratskhelia - Hovering on Results

This visualization serves as an initial, high-level comparison tool, offering a quick and intuitive way to assess overall similarity. Compared to the raw DataFrame, the bar plot immediately highlights the most relevant matches, allowing scouts to focus on the most similar players briefly rather than manually comparing multiple numerical values.

However, this alone is not yet sufficient for a truly effective comparison. While the bar plot offers a strong overview, it does not allow for an in-depth, side-by-side examination of multiple attributes. A scout attempting to compare players in detail would need to constantly shift focus between bars, repeatedly checking and memorizing numerical values—an approach that is neither practical nor efficient.

This is why an additional visualization step for an in-depth player comparison is necessary. The next section introduces a final and more powerful graphical comparison tool, specifically designed to enable a simultaneous, multi-attribute evaluation of selected players. This additional visualization step directly addresses the need for comprehensive and efficient player comparisons, ensuring that all relevant performance indices can be analyzed in parallel rather than sequentially.

7.3 Final In-Depth Player Comparison Visualization

7.3.1 Radar Plot

Radar plots, also known as spider charts or web charts, are commonly used for multidimensional data visualization, allowing for an immediate and intuitive comparison of multiple variables. This type of plot is particularly effective in scenarios where different attributes must be evaluated simultaneously, offering a structured way to highlight strengths and weaknesses within a given dataset.

A radar plot consists of a series of axes radiating from a common center, with each axis representing a different variable. The values along each axis are plotted relative to a uniform scale, and the points are then connected to form a polygon. This structure of the generated polygon allows for a quick comparison of strengths and weaknesses across multiple metrics.

Radar plots are among the most widely used visualization tools in sports analytics, particularly in football player performance assessment. They allow for the simultaneous representation of multiple performance indicators, offering a compact and intuitive way to compare players across different skill domains. They are particularly valuable for benchmarking athlete performance, providing an intuitive view of ability distributions across different categories (Stanojevic, 2022).

Radar plots have also been applied in medical diagnostics, where they facilitate comparative health assessments and patient condition monitoring (Smit, 2014). Additionally, they are widely used in business performance analysis, helping to visualize key performance indicators (KPIs) and assess competitive positioning.

In this study, radar plots are used to represent a player's performance across the multiple performance indices, allowing for a visual and analytical comparison between different footballers selected from the model's results. Each axis in the radar chart corresponds to a specific performance index, enabling a graphical representation of a player's skill set.

Compared to the bar plot visualization discussed of the model output, the radar plot offers a more immediate and structured way to compare multiple performance indices simultaneously. While the bar plot was interactive and provided access to all individual index values, it required the user to hover over each bar to retrieve the data and mentally keep track of multiple values when making comparisons. This made it difficult to assess overall patterns briefly and required continuous back-and-forth navigation between

players, as already shown. The radar plot solves this issue by displaying all key performance indices at once, making it significantly easier to identify differences and similarities between players without relying on memory or manual comparisons. By observing the shape of the radar plot, scouts can instantly identify which attributes stand out as key strengths and which indicate potential weaknesses in a player's profile.

Using the same example as before, the target player Kvaratskhelia, the radar plot in *Figure 40* visualizes his performance indices across all the attributes.

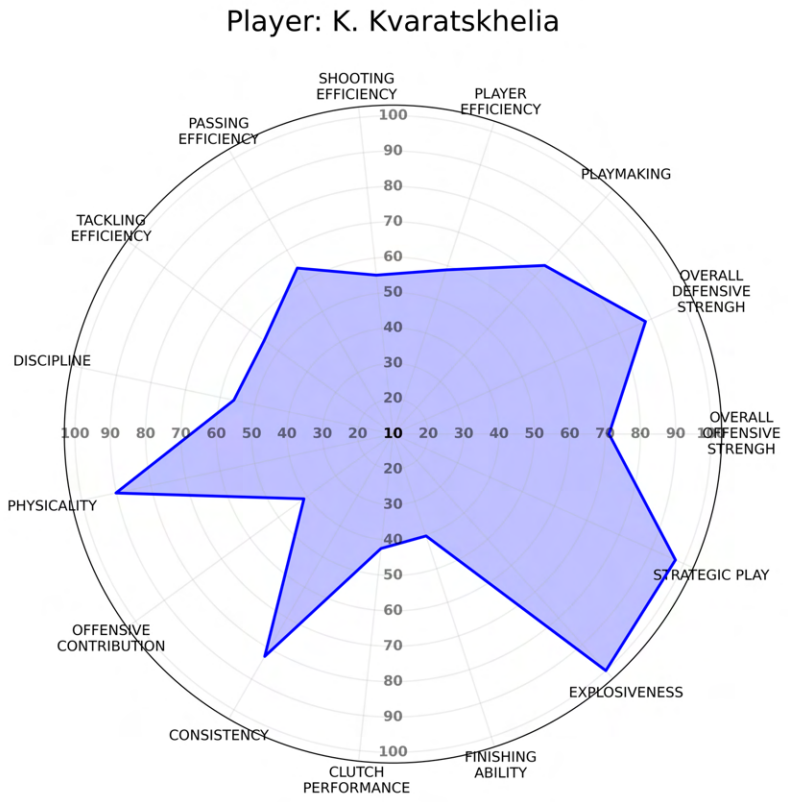


Figure 40 - Radar Plot of K. Kvaratskhelia

Visually, Kvaratskhelia's player profile emerges distinctly. For instance, it is evident that he is among the top attackers in explosiveness, consistently making strategically decisions and demonstrating good passing ability. However, the plot also reveals that he struggles more in tackling attributes and does not exhibit the same level of proficiency in finishing as in other offensive metrics.

While this representation effectively captures an individual player's profile, it does not yet facilitate direct comparison with other similar players retrieved by the model. To truly assess potential alternatives, a multi-player radar plot is required, where Kvaratskhelia's radar plot can be directly overlaid with that of the most similar players.
The next section will introduce the final comparison visualizations, designed to provide a structured and visually intuitive tool for scouting, allowing for immediate insights into key differences and similarities across players.

7.3.2 Comparative Analysis: Multi-Player Radar Plot

Returning to the model's output, the user now has access to two key tools:
- A DataFrame ranking the most similar players to the target, providing a structured overview of similar players.
- A bar plot visually presenting these rankings, where each player's similarity score is highlighted for an immediate and intuitive interpretation of their relative resemblance.

With the player IDs of the top-ranked similar players readily available in the DataFrame, the user can now select specific players for a detailed comparison through a multi-player radar plot.
The model allows for the input of up to two additional player IDs, alongside the target player, to generate a radar plot that overlays their performance profiles. Limiting the comparison to a maximum of three players ensures clarity and interpretability. To enhance comparative analysis, the radar plots are displayed to the user in two formats:

a. Up to three separate side-by-side radar plots, ensuring that individual player profiles remain clearly distinguishable while still being easily compared.
b. A single, overlaid radar plot, where all selected players are plotted on the same axes, allowing for an immediate visual comparison of their strengths and weaknesses across all attributes.

As mentioned above, the limit of three players is set to ensure clarity and interpretability in both visualization formats. In the side-by-side layout, exceeding three players would reintroduce the problem of constant scrolling and back-and-forth comparisons, ultimately diminishing the efficiency of the visualization. By keeping the comparison concise, the user can quickly assess differences in performance and make informed scouting decisions without unnecessary cognitive load. In the overlaid radar plot, including too many players

would lead to visual disorder, making it difficult to distinguish individual attributes and extract meaningful insights. When multiple player profiles are plotted on the same axes, excessive overlap would obscure differences, reducing the effectiveness of the comparison. Thus, limiting the comparison to three players strikes a balance between comprehensiveness and clarity, ensuring that key performance attributes remain distinguishable while still allowing for an effective evaluation of similarities and differences.

To demonstrate the practical application of the comparative radar plots, the following examples use the same target player, Khvicha Kvaratskhelia, and compare him to two players selected from the top 20 most similar players retrieved by the model:

- Sávio (red) – The most similar player to Kvaratskhelia, ranked first in the similarity list. Also known as Savinho, a recent acquisition by Manchester City at the time of analysis.
- Albert Guðmundsson (green) – The least similar player among the top 20 recommendations. That still shares important attributes with Kvaratskhelia, as the model identified.

This can be an interesting contrast: Sávio is nearly identical to the target and Guðmundsson is on the lower bound of similarity within the suggested list. This allows us to analyze both strong and weaker matches and assess the effectiveness of the model's recommendations through these plots.

Side-By-Side Radar Plot:

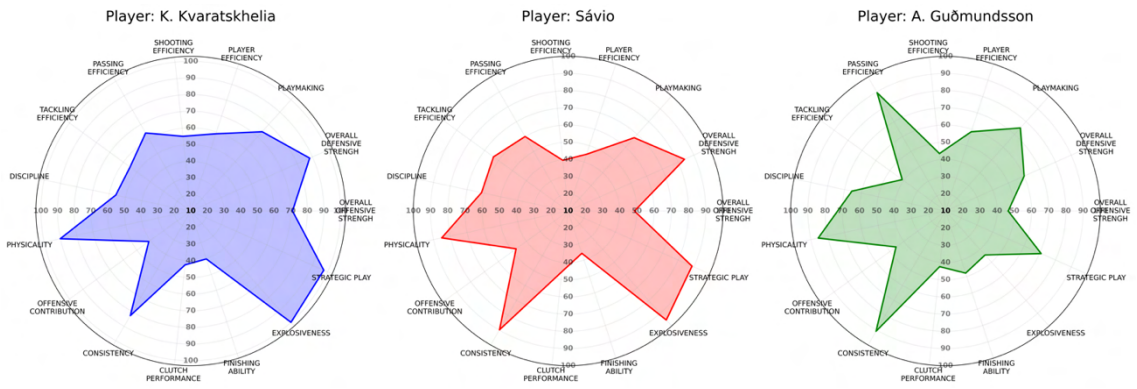


Figure 41 - Side-By-Side Radar Plot

The *Figure 41* illustrate the Side-By-Side Radar Plot comparison. This a structured, comparative view, where scouts can evaluate attributes without distraction from excessive visual intersections.

Advantages:

- Clear individual assessment of each player's shape and strengths. Each player's radar plot remains fully visible, ensuring that their skill distribution is not obscured by overlapping visual elements. This makes it easier to identify overall attributes distribution immediately without interference from other players.
- No overlap issues, ensuring that every metric remains distinctly visible. Since each player's data is displayed separately, all performance indices are clearly distinguishable, reducing the risk of misinterpretation due to excessive visual clutter. This is especially beneficial when comparing players with highly contrasting styles, as their unique characteristics remain intact.

Limitations

- Slower direct comparison: Unlike the overlaid version, where differences and similarities are immediately visible, the side-by-side layout requires the user to visually shift their focus between separate plots to compare attributes numerically. This back-and-forth movement can make it slightly less efficient for rapid decision-making.
- Limited scalability for multiple comparisons: While side-by-side visualization works well for two or three players, adding more plots would require scrolling or zooming out, making the comparison less fluid and harder to process. Beyond a certain number of players, this approach loses effectiveness, as the user has to manually track and remember attributes across multiple charts.

Overlaid Radar Plot:

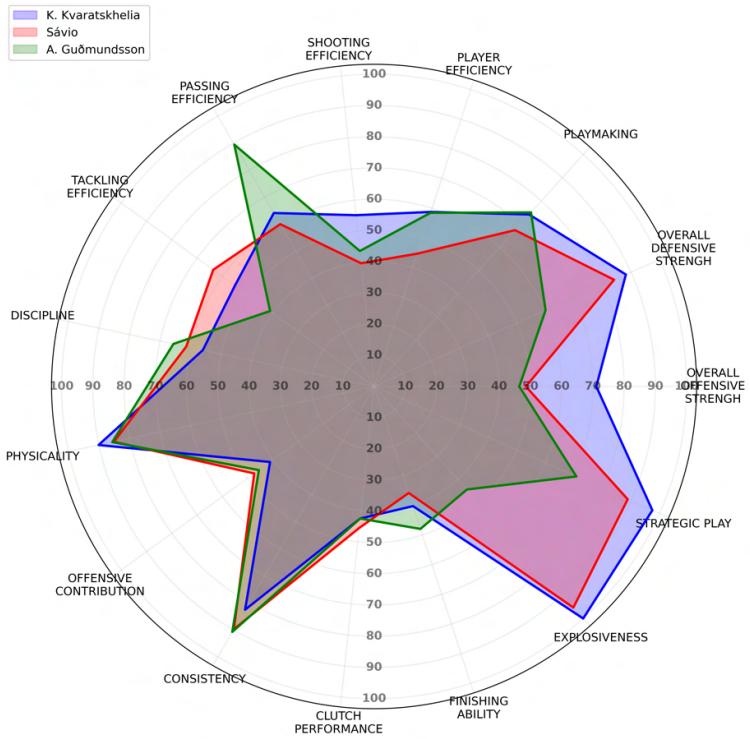


Figure 42 - Overlaid Radar Plot

The *Figure 42* shows the Overlaid Radar Plot comparison.

Advantages:

- Highly efficient for direct comparison: Since all players are plotted together, it is much easier to compare them than in separate visualizations.
- Super immediate interpretation: The user instantly sees which player excels in specific skills over the others and where the profiles overlap or diverge. Similar attributes are indicated by overlapping sections of the radar plot. Differences become clear in areas where one player's shape extends beyond the others.

Limitations:

- Loss of individual skill set distribution: Due to the multiple intersections of plots, it becomes harder to clearly visualize a single player's overall shape. The more overlapping occurs, the more difficult it is to isolate and assess the unique characteristics of an individual player.

- Best suited for players with similar roles (like this analysis): If players have very different profiles, the overlapping can make interpretation more complex rather than more intuitive.

In conclusion for cases where understanding the unique attributes of a single player is crucial, the side-by-side radar plot allows for a clearer focus on each player's individual strengths and weaknesses. For comparisons and immediate pattern recognition, the overlaid radar plot remains the more practical option. By providing both visualizations (side-by-side and overlaid), the scouting process remains flexible and adaptable, allowing analysts to choose the most effective approach based on the players being compared. A more detailed discussion on how to interpret these results, assess player profiles, and explore different types of comparisons will be addressed in the following chapter, which will focus on analyzing the outputs, testing new results with different target players and extracting meaningful insights from the various visualizations.

7.3.3 Enhancing Radar Plot Design: Aesthetics and Additional Information

Up to this point, the scouting model has been designed to be both efficient (delivering rapid results), effective (offering meaningful comparisons between similar players) and interpretable (providing clear input-output mechanisms and intuitive radar plots). However, an essential aspect of usability is also user experience, ensuring that the tool is not only functional but also engaging and visually enriching. To further enhance the two radar plots, aesthetically appealing, and useful player information has been incorporated into the visualizations.

As illustrated in Chapter 3, the dataset includes more than performance metrics and statistical indicators. It also contains player-specific images, including a profile photo of each player, a team logo and a league logo. The Logos inclusion provides immediate context regarding the club and league in which the player is currently competing.

The process remains entirely automated, ensuring that the user experience remains direct and rapid. Just like before, after retrieving the top-K most similar players from the model's output, the user will be asked to select which players to analyze and compare in detail. Once selected, the chosen players will be displayed in both side-by-side and overlaid radar plots, now enhanced with their respective images and additional key

information. In *Figure 43* and *Figure 44* are the previous comparison visualizations, now integrated with these additional visual and contextual elements.

Side-By-Side Radar Plot:

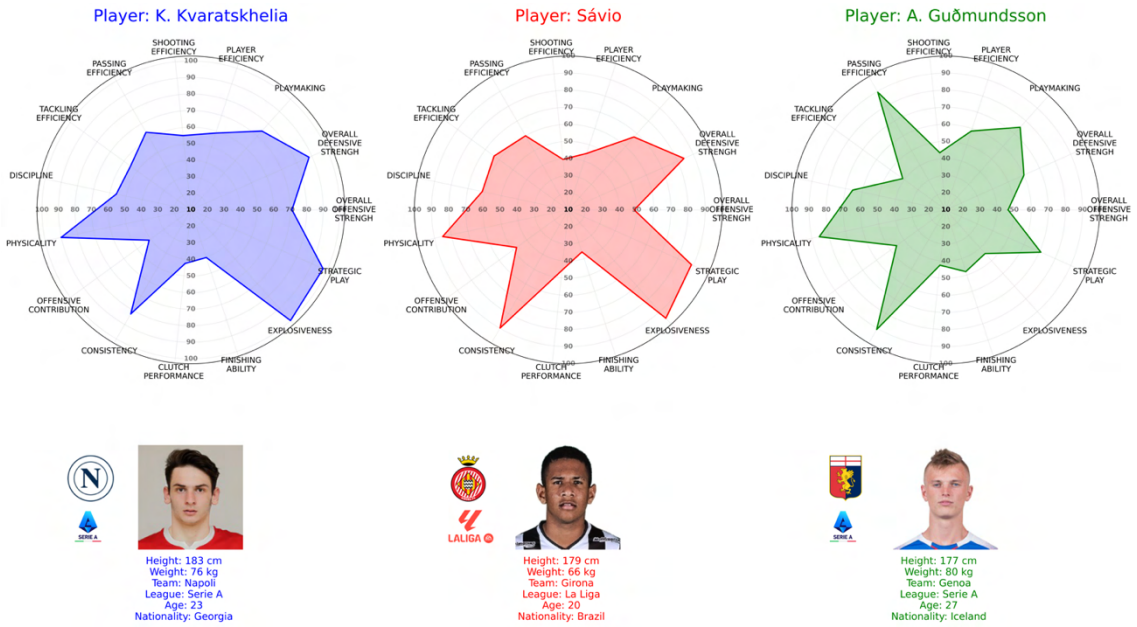
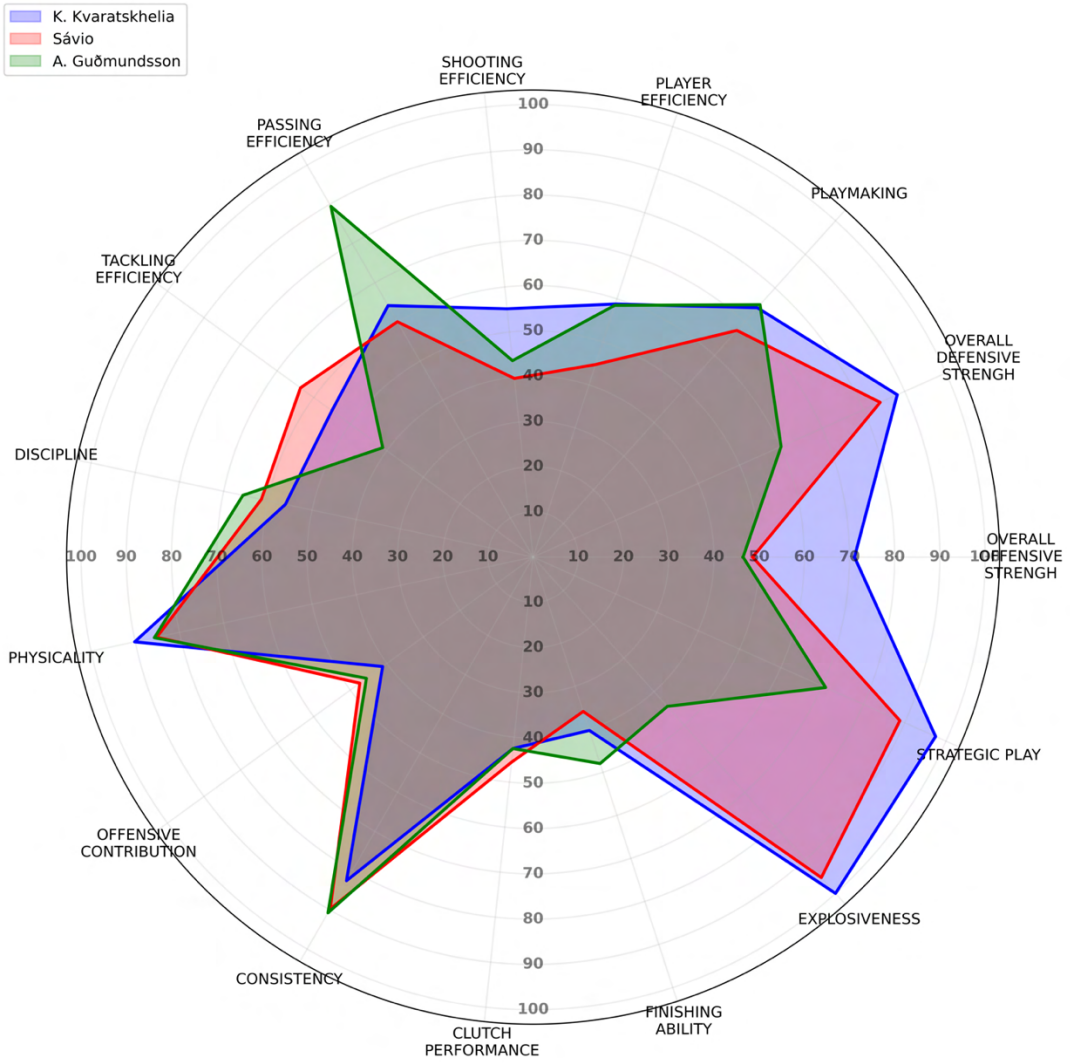
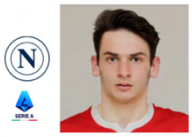


Figure 43 - Side-By-Side Radar Plot

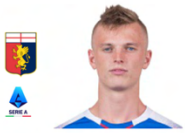
Overlaid Radar Plot:





Name: K. Kvaratskhelia
Height: 183 cm
Weight: 76 kg
Team: Napoli
League: Serie A
Age: 23
Nationality: Georgia

Name: Sávio
Height: 179 cm
Weight: 66 kg
Team: Girona
League: La Liga
Age: 20
Nationality: Brazil



Name: A. Guðmundsson
Height: 177 cm
Weight: 80 kg
Team: Genoa
League: Serie A
Age: 27
Nationality: Iceland

Figure 44 - Overlaid Radar Plot

Beyond visual identifiers, key biographical and physical attributes are also integrated, including:

- Age, that is particularly important in player comparisons, as younger players typically have more room for development, making them potentially more valuable in scouting and transfer decisions.
- Nationality, knowing players' nationality may arise further investigation into their performances in their national team. Moreover, many leagues and UEFA competitions impose quotas on homegrown, foreign, and non-EU players.
- Height and weight, as these physical characteristics can be crucial in scouting decisions. A player's physique influences their playing style, positional suitability and adaptability in different role-tasks. For instance, taller and stronger players may have an advantage in aerial duels, while lighter, more agile players might excel in dribbling and quick directional changes. Including these attributes in the comparison allows scouts to immediately assess whether a player fits the physical profile required for a specific tactical role or team needs. This is particularly relevant when scouting for a player who not only possesses the desired performance characteristics but also meets specific physical criteria, ensuring they align with the demands of a given playing style or league.

Conclusively, the final visualizations have been enhanced to create a more engaging and useful user experience, ensuring that both the model's analytical depth and its usability are maximized.

In the following chapter, the model will be further explored through the testing of various players, simulating a real scouting approach. By experimenting with different player profiles and positions, testing different approaches and comparisons, the practical applications of the model will be assessed, and its results will be analyzed and compared to further confirm its effectiveness.

CHAPTER 8:
REAL-WORLD USE CASES IN FOOTBALL SCOUTING: TESTING THE MODEL AND PLAYER ANALYSIS

All additional data cited and presented beyond the dataset of the analysis, such as a player's estimated market value in euros, salary, and specific role, have been sourced from *Transfermarkt* (https://www.transfermarkt.it). It is one of the most reputable football databases, widely used by analysts, scouts, and journalists for player valuations, transfer histories, and performance statistics. It has comprehensive data collection and regular updates.

8.1 Testing the Model on Attackers

8.1.1 Case Study: Replacing Khvicha Kvaratskhelia at Napoli

The previous chapter examples involved Khvicha Kvaratskhelia, and this choice was not from random. The Georgian winger was at the center of a major €70 million transfer from Napoli to PSG, leaving a significant gap in Napoli's attacking setup. Given Kvaratskhelia's crucial role in the team, the club now faces the challenge of identifying a suitable replacement—a task where a data-driven scouting tool like this model becomes particularly valuable.

To begin, let's re-examine the model's results (*Figure 38)*. By considering all performance metrics, we can immediately observe a striking pattern: nearly all the top 20 most similar players are pure wingers.

1 - Sávio - Winger
2 - J. Bakayoko - Winger
3 - M. Soulé - Winger
4 - Rafael Leão - Winger
5 - A. Gordon - Winger
6 - L. Díaz - Winger
7 - I. Williams - Winger
8 - Hélio Varela - Winger

9 - B. Saka - Winger
10 - O. Sahraoui - Winger
11 - Matheus Cunha - Striker/Winger
12 - A. Laurienté - Winger
13 - L. Sané - Winger
14 - Pepê Aquino - Winger
15 - A. Garnacho - Winger
16 - J. Musiala - Winger
17 - Rodrygo - Winger
18 - Lamine Yamal - Winger
19 - A. Guðmundsson - Second Striker/Winger

The high concentration of wingers among the top-ranked players confirms that the model correctly captures playing style and role similarity, despite not using explicit position labels. The dataset does not differentiate between wingers, second strikers, center-forwards, or attacking midfielders, yet the model correctly retrieves players with similar tactical roles.

Therefore, it is possible to assess:

- 18 out of 20 players are pure wingers, confirming that the model highlights playing style and role similarity even without relying on explicit position labels.
- Matheus Cunha and Albert Guðmundsson are the only two players who are not pure wingers, but both still fit the profile of attacking, creative forwards with high dribbling ability and offensive output. Additionally, both have played as wingers on multiple occasions, further aligning with Kvaratskhelia's role and style of play.
- The top-ranked result is Sávio, a highly promising young winger who had an exceptional season at Girona and was recently signed by Manchester City—suggesting that the model aligns well with real-world scouting decisions.

With these results in mind, it is now time to return to the specific comparison presented in the previous example: analyzing Kvaratskhelia alongside his most and least similar recommended players from the top 20: Sávio and Albert Guðmundsson. Using radar plots, it is possible to conduct a more detailed skill-based evaluation to draw meaningful conclusions.

The radar plot comparison (*Figure 43* and *Figure 44*) immediately highlights that Sávio's overall skill distribution closely mirrors that of Kvaratskhelia, with their plots almost overlapping in multiple key attributes. This suggests that they move similarly on the field, excelling in comparable areas and fulfilling a nearly identical tactical role. In contrast, Guðmundsson's distribution deviates more significantly, confirming that, while still somewhat similar, his playstyle differs in key aspects.

Key Observations:

- Dribbling and One-on-One Ability: Both Kvaratskhelia and Sávio exhibit high Explosiveness, a key attribute that define wingers who frequently engage in one-on-one duels. These high values indicate their ability to accelerate quickly, change direction, and make intelligent attacking movements. In contrast, Guðmundsson shows a lower score in Explosiveness and Strategic Play. Instead, his playstyle appears more oriented toward Passing Efficiency and Playmaking, indicating a preference for positional play and combination passing ability rather than beating defenders in isolation.
- Final Third Decision-Making: Sávio and Kvaratskhelia display similar levels of Offensive Contribution and Playmaking, reinforcing their roles as high-impact creators in the final third. Their similar Player Efficiency level further highlight their effectiveness in retaining possession and making decisive actions. However, Guðmundsson's significantly higher Finishing Ability suggests that he may function better as a secondary striker or inside forward, positioning himself more to convert chances.
- Defensive Work Rate: Surprisingly, both Kvaratskhelia and Sávio have relatively strong Overall Defensive Strength and Tackling Efficiency for attacking wingers, indicating their willingness to track back and contribute defensively. Guðmundsson, however, shows lower Defensive Strength and Tackling Efficiency, suggesting a reduced contribution in defensive phases. This contrast could be crucial for scouting decisions, particularly for teams that prioritize high pressing intensity from wide players.

These insights further validate the model's ranking, demonstrating its ability to differentiate not just positional roles but also stylistic tendencies. While Guðmundsson remains a potential alternative, Sávio's closer alignment in playstyle makes him the most natural replacement for Kvaratskhelia.

8.1.2 Case Study: Replacing Olivier Giroud at AC Milan

Another realistic case study involves Olivier Giroud, who, at 37, left AC Milan after the 2023/24 season to continue his career in Los Angeles. Giroud played a pivotal role for Milan, excelling in aerial duels, link-up play, and penalty-box presence. His departure posed a major challenge for Milan's scouting team, which needed to identify a worthy replacement, a striker capable of maintaining the team's attacking structure. The model's recommendations are now analyzed to identify the strikers with the highest statistical similarity to Giroud. The *Figure 45* presents the top 20 most comparable forwards, ranked by similarity score.

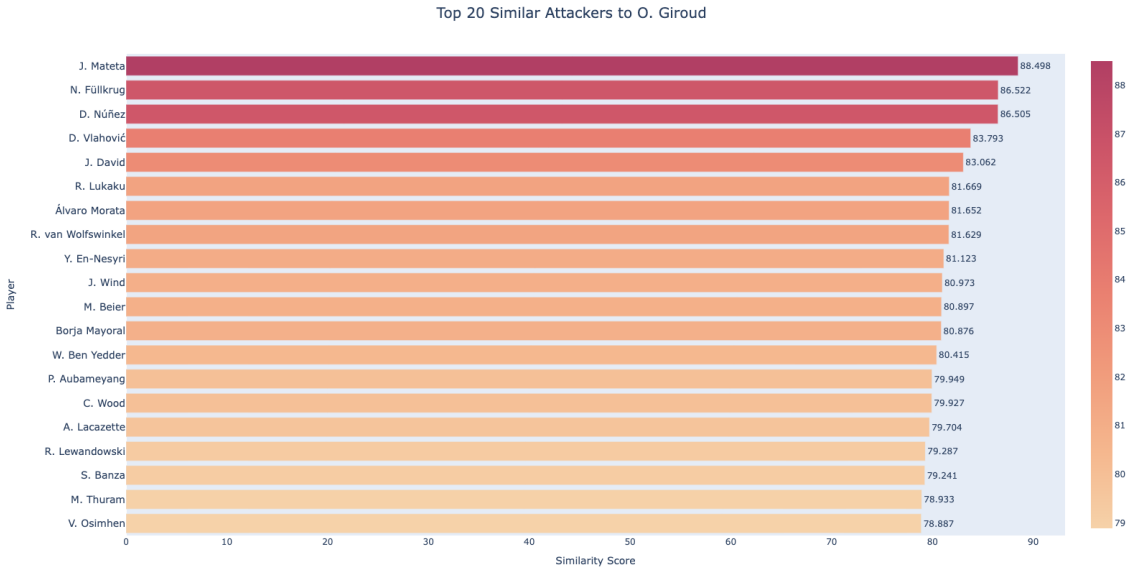


Figure 45 - Top 20 Similar Attackers to O. Giroud by the KNN Model

Notably, all suggested players were pure center-forwards[18], reinforcing the model's ability to capture positional and stylistic traits without explicit position labels. However, what makes this case study particularly interesting is that among the results, Álvaro

[18] For further details on specific player roles, Transfermarkt is recommended again as a reference.

Morata ranked seventh, aligning with Milan's actual summer 2024 transfer decision. The Spanish striker shares key attributes with Giroud, particularly in hold-up play, tactical intelligence, and team-oriented attacking contributions.

While on-field attributes are central to this model, transfer decisions also depend on financial constraints, including fees, wages, and long-term investment. Dusan Vlahović and Darwin Núñez, both highly ranked in the model, may have been stronger replacements for Giroud, but their €60M+ market values made them unattainable for Milan. Instead, Morata was signed for €13M, offering a cost-effective alternative with a similar playing profile.

A direct radar plot comparison (*Figure 46*) highlights key similarities and differences between Giroud and Morata.

- Offensive Contribution – Both strikers exhibit comparable offensive impact.
- Finishing Ability – Neither is a prolific goal scorer, but both contribute beyond scoring, excelling in build-up play and creating space.
- Physicality – Giroud dominates this category, as expected. His ability to hold up play, win aerial duels, and outmuscle defenders has been one of his defining traits. Morata, while physically capable, is not as strong in direct physical contests.
- Morata appears slightly more agile and mobile, performing better in explosiveness, suggesting that he is more suited to quick transitions and counterattacks, whereas Giroud has relied more on positional play.
- Giroud outperforms Morata in clutch performance, reinforcing his reputation as a big-game player who delivers in decisive moments.
- Shooting and Passing Efficiency – Giroud exhibits greater precision in both metrics, making him a more reliable focal point in attack.

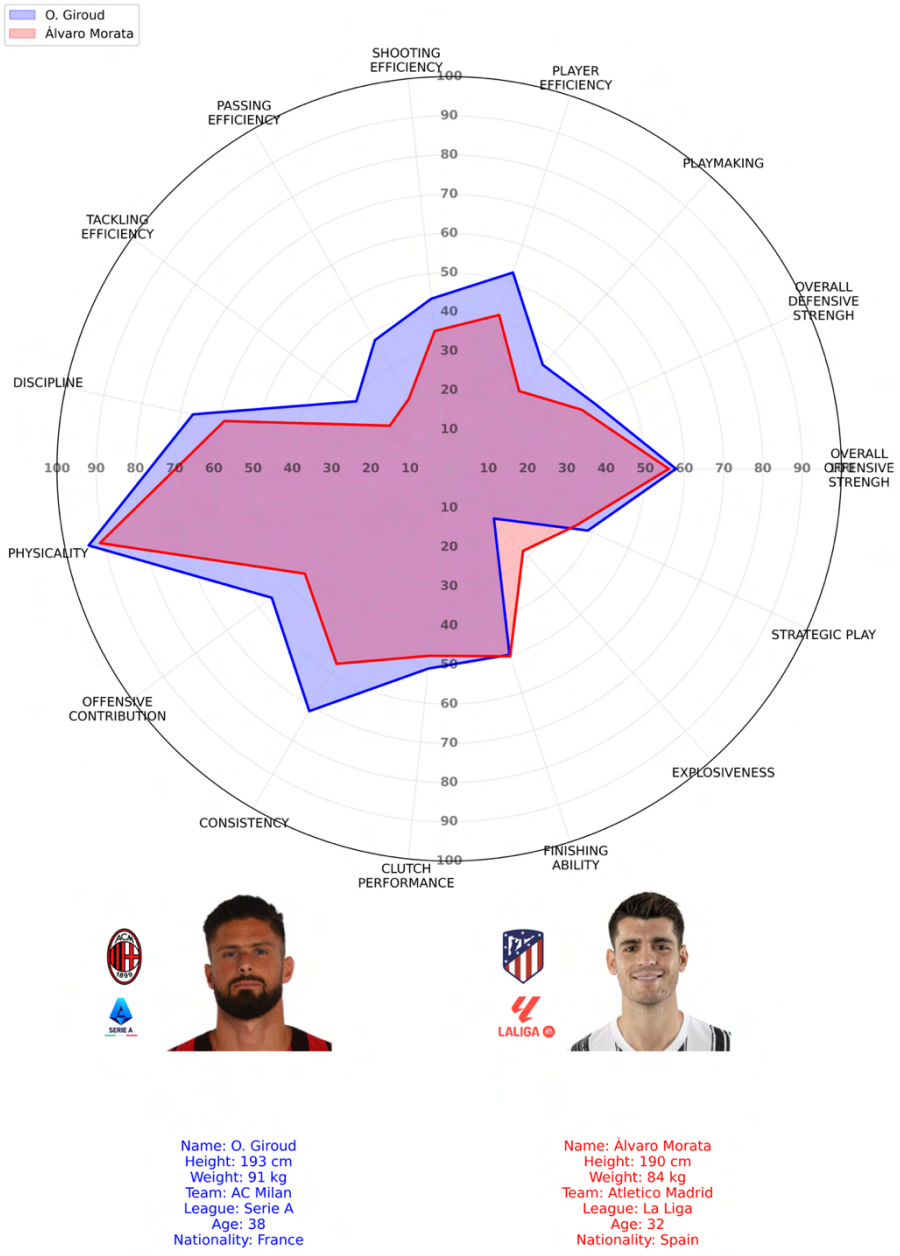


Figure 46 - Overlaid Radar Plot comparison: Giroud and Morata

However, in the 2024/25 season, AC Milan underwent two managerial changes, leading to tactical shifts that impacted both the team and Morata. In the first half of the season, he played 28 matches, recording just 6 goals and 1 assist, an inadequate return for a starting striker. While goal-scoring depends on team dynamics, Milan's xG ranked second in Serie A, yet they were 18th in goal conversion rate, highlighting a good creation of offensive opportunities but a struggle to capitalize on high-quality chances. Milan's xG-to-goal conversion gap placed pressure on the attacking finishers, particularly the

central striker. With Morata struggling to convert, the club took action in January 2025, sending him on loan to Galatasaray and searching for a proven goal-scorer to lead the attack.

This situation can underscore a key strength of the scouting model: it goes beyond simply identifying direct replacements, allowing clubs to target players based on specific attributes. While teams often look for similar profiles when replacing a player, certain situation (such as Milan's need for a more clinical striker) require a different approach. This is still where performance indices and comparative analysis become essential tools in scouting. Instead of searching for a "Morata-type" attacker, Milan's objective was to find a striker with exceptional finishing ability, composure in front of goal, and a high conversion rate. Consequently, another valuable application of this algorithm is Benchmarking Against an Elite Player in specific abilities. Haaland ranks third overall in the finishing ability index and consistently appears among the top 20 players in several key offensive metrics, including Overall Offensive Strength, Shooting Efficiency, Offensive Contribution, and Clutch Performance. His dominance in these areas makes him an ideal benchmark for identifying a high-quality finisher. However, Haaland is not a realistic transfer target for Milan. As of 2025, he remains a key player for Manchester City, with an estimated market value exceeding €200 millions and a reported salary of €18 million per season—figures well beyond Milan's financial reach. Given this, Haaland serves as a performance benchmark rather than a viable signing, helping to identify players with similar attributes. The results of this comparison are presented in *Figure 47*.

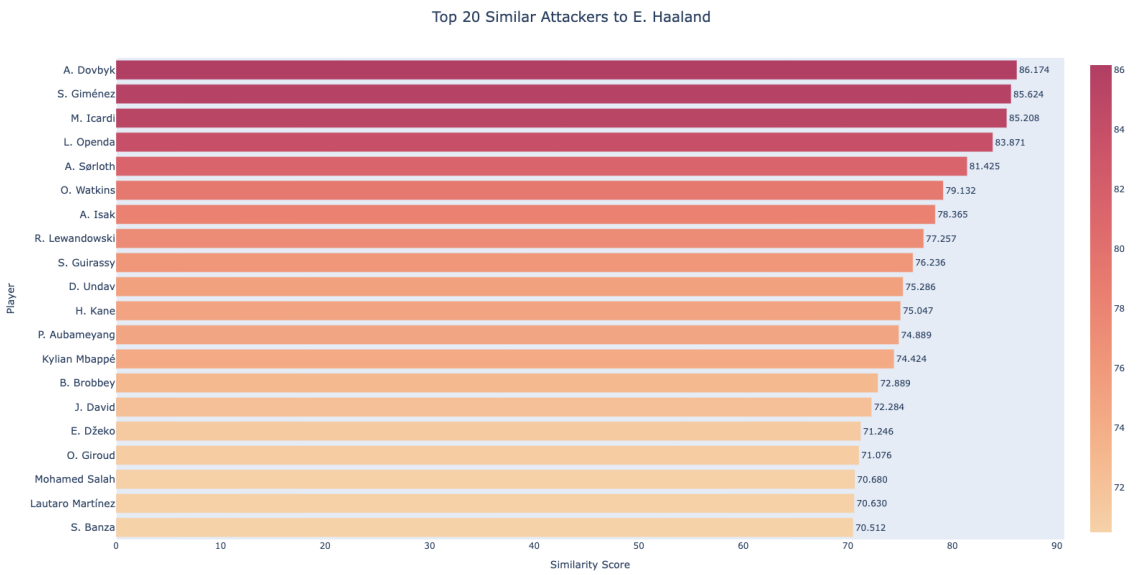


Figure 47 - Top 20 Similar Attackers to E. Haaland by the KNN Model

The model identifies Artem Dovbyk, Santiago Giménez, and Mauro Icardi as the three most similar players to Erling Haaland.

- Dovbyk recently joined AS Roma, making him unavailable.
- Icardi, while a strong finisher, is over 30 and does not fit Milan's long-term squad strategy.
- Other high-profile strikers like Watkins, Openda and Lewandowski appear in the rankings but are either too expensive or too old to justify a major investment.

Given these constraints, is understandable why Milan decided to sign Santiago Giménez for €32 million and it aligns with the model's recommendations. This consideration was not only driven by reasoning but was also explicitly validated by AC Milan's Senior Advisor, Zlatan Ibrahimović, who, in a recent press conference, emphasized the club's need for a decisive goal-scorer:

> *"When we started the season, we believed in what we had built. But we were not satisfied with the results, so we decided to make changes. […] He (referring to Gimenez) has great quality, he has the hunger to score goals […] This is something you either have or you don't, it's an innate talent. His best qualities are showcased in the penalty area, where he excels in finishing."*

Therefore, the strategy of using a benchmark top player to identify specific attributes proves to be highly effective in a real-world scenario. By selecting Haaland as a reference

point, the model successfully identified Giménez. As illustrated in *Figure 48*, Giménez' radar plot closely mirrors Haaland's, confirming his suitability as a high-scoring dominant striker. In contrast, Morata's profile differs significantly, highlighting his distinct playing style and clarifying why he was not the ideal solution for Milan's needs.

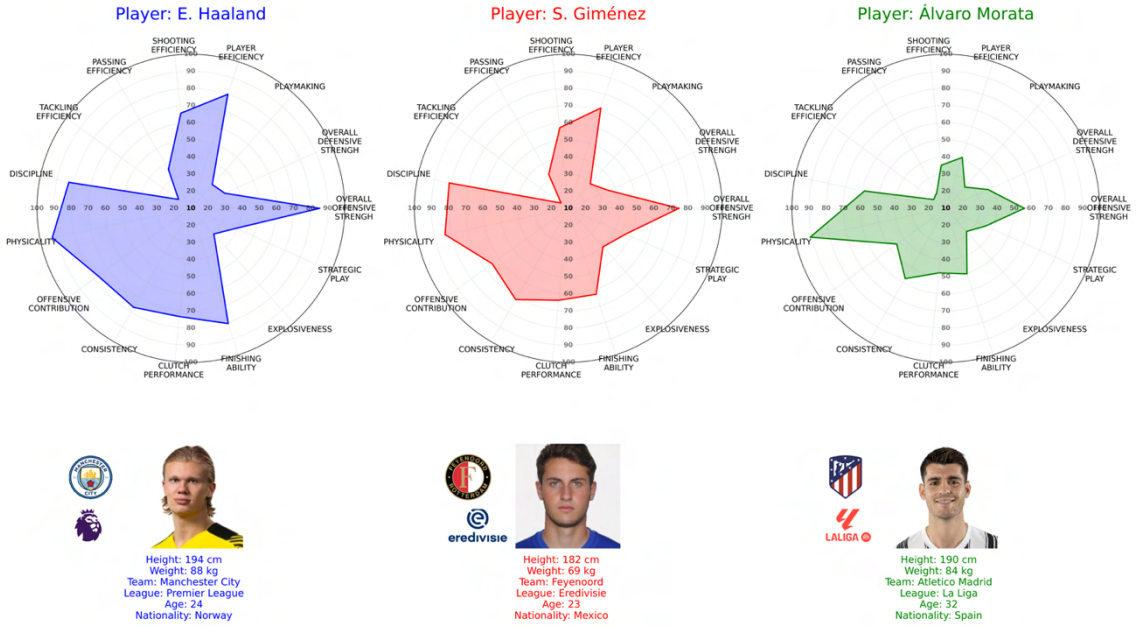


Figure 48 - Side-By-Side Radar Plot comparison: Haaland, Giménez and Morata

8.2 Testing the Model on Midfielfers

8.2.1 Case Study: Rodri

After testing the model's performance on attackers, the next step is to evaluate its effectiveness for other positions, starting with midfielders, considering both defensive and offensive profiles.

A particularly challenging case is Rodri, Manchester City's world-class midfielder, who played a pivotal role in both City's Premier League title and Spain's European Championship victory, where he was named best player of the tournament. His Ballon d'Or win further solidified his status as the best player in the world for the 2023/24 season, highlighting his exceptional influence in midfield and making him an irreplaceable figure for Manchester City. Finding a true like-for-like replacement for Rodri is no easy task. His unique skill set and influence make him one of the most irreplaceable players in world football. This became even more evident when a long-term

injury sidelined him, exposing a major weakness in Manchester City's squad. While no single player can fully replicate his impact, the model aims to identify the closest possible alternatives—midfielders who share key attributes and could, at least in part, fill the void. *Figure 49* presents the top recommendations generated by the model.

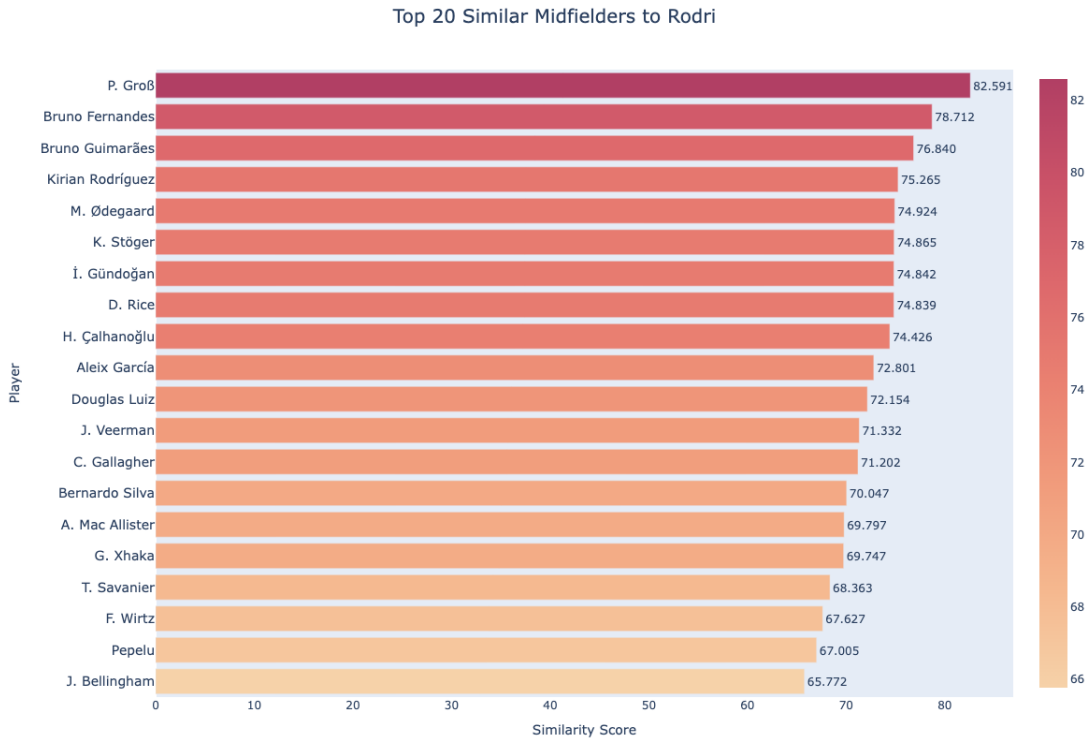


Figure 49 - Top 20 Similar Midfielders to Rodri by the KNN Model

This initial result highlights a limitation of the model when applied to a player like Rodri. While many suggested players are top-tier defensive playmakers—such as Bruno Guimarães, Granit Xhaka, Hakan Çalhanoğlu, and Declan Rice—the list also includes attacking midfielders like Bruno Fernandes, Florian Wirtz, Jude Bellingham, and Martin Ødegaard, who operate in a completely different role.

Before assuming the model has misclassified midfielders, it's essential to analyze the performance indices in detail. To do so, *Figure 50* compares Rodri's radar plot with Granit Xhaka, a true deep-lying playmaker, and Bruno Fernandes, an advanced attacking midfielder.

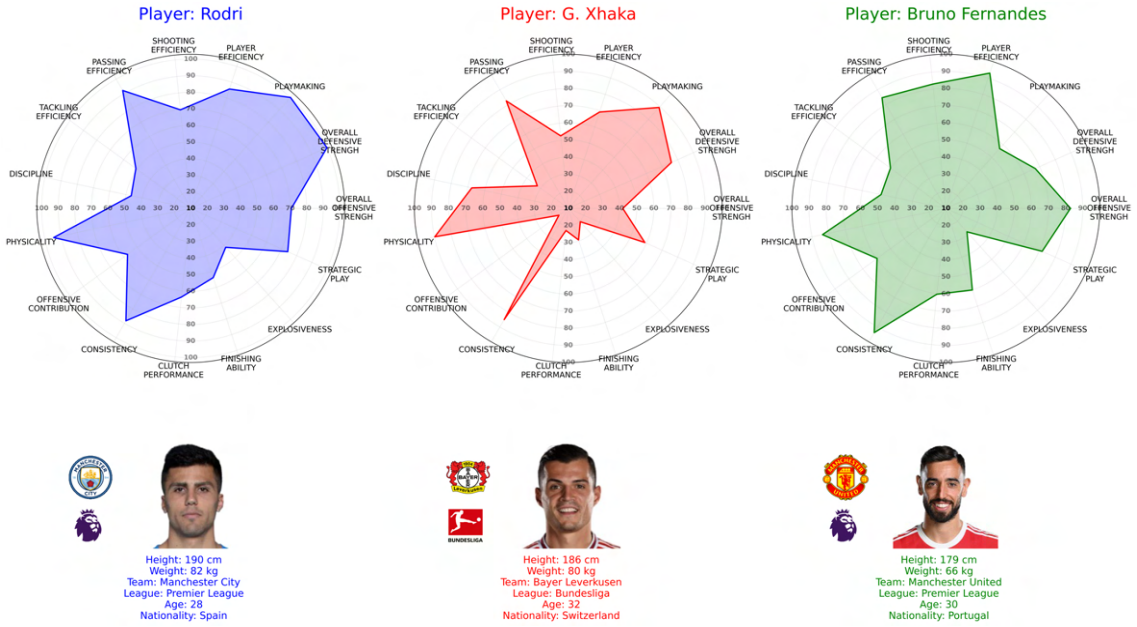


Figure 50 - Side-By-Side Radar Plot comparison: Rodri, Xhaka and Bruno Fernandes

The comparison reveals that Xhaka's profile closely resembles Rodri's, confirming his role as a deep-lying midfielder:

- Playmaking & Passing Efficiency – Both excel in dictating tempo and distributing the ball, though Rodri remains superior.
- Defensive Strength – Both rank highly in duels and ball recovery.
- Physicality & Discipline – Both are imposing midfielders who provide defensive stability.

However, Rodri significantly outperforms Xhaka in offensive metrics:

- Shooting Efficiency & Finishing Ability – Rodri registers much higher values, unusual for a defensive midfielder.
- Clutch Performance & Offensive Contribution – Unlike Xhaka, Rodri frequently directly impacts the scoreboard.

Interestingly, when focusing on offensive metrics, Rodri aligns more closely with Bruno Fernandes than with Xhaka. As expected, Bruno Fernandes dominates in attacking contributions, ranking higher in overall offensive strength, shooting efficiency, and finishing ability. However, Rodri remains surprisingly close, despite playing in a much deeper role. Rodri's statistical impact further reinforces this. In the analyzed season, he

played 34 league matches, scoring 8 goals and providing 9 assists—extraordinary numbers for a defensive midfielder. By comparison, Bruno Fernandes, regarded as one of the best attacking midfielders, played 23 matches, contributing 5 goals and 6 assists. This highlights Rodri's unique ability to influence the final third, setting him apart from traditional defensive midfielders.

This comparison clearly highlights Rodri as a statistical outlier—his ability to excel in both defensive and attacking phases makes him an atypical defensive midfielder. As a result, the model identifies similarities with both defensive and offensive midfielders. This is further reflected in the similarity scores, where only one player exceeds 80, while several in the top 20 fall below 70, reinforcing the challenge of finding a true like-for-like replacement.

8.2.2 Evaluating the Model on Defensive Midfielders

To determine whether Rodri's classification challenge stems from his unique skill set rather than a model limitation, the next test uses Granit Xhaka as the target. While still a defensive midfielder, Xhaka has a more traditional profile. If the model consistently returns players with similar tactical roles, this would confirm that Rodri is an outlier rather than exposing a flaw in the model. The bar plot shows in *Figure 51* presents the results generated by the model.

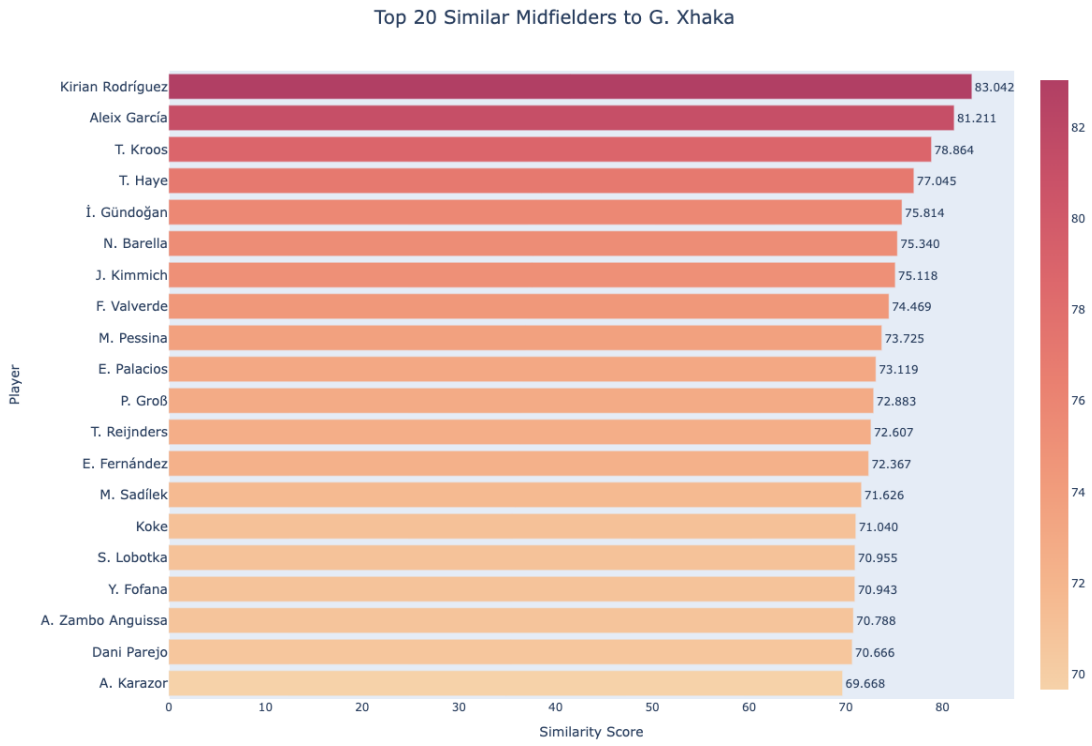


Figure 51 - Top 20 Similar Midfielders to G. Xhaka by the KNN Model

The model's output consists exclusively of defensive and central midfielders, reinforcing its accuracy in recognizing role-specific similarities. The list includes:

- Deep-lying playmakers such as Enzo Fernández, Lobotka, and Fofana.
- Hybrid central midfielders like Barella, Gündogan, and Reijnders, who balance defensive duties with playmaking responsibilities.

To further validate the model's ability to differentiate elite midfield profiles, *Figure 52* compares Xhaka to Toni Kroos and Joshua Kimmich, two midfielders identified as stylistically similar.

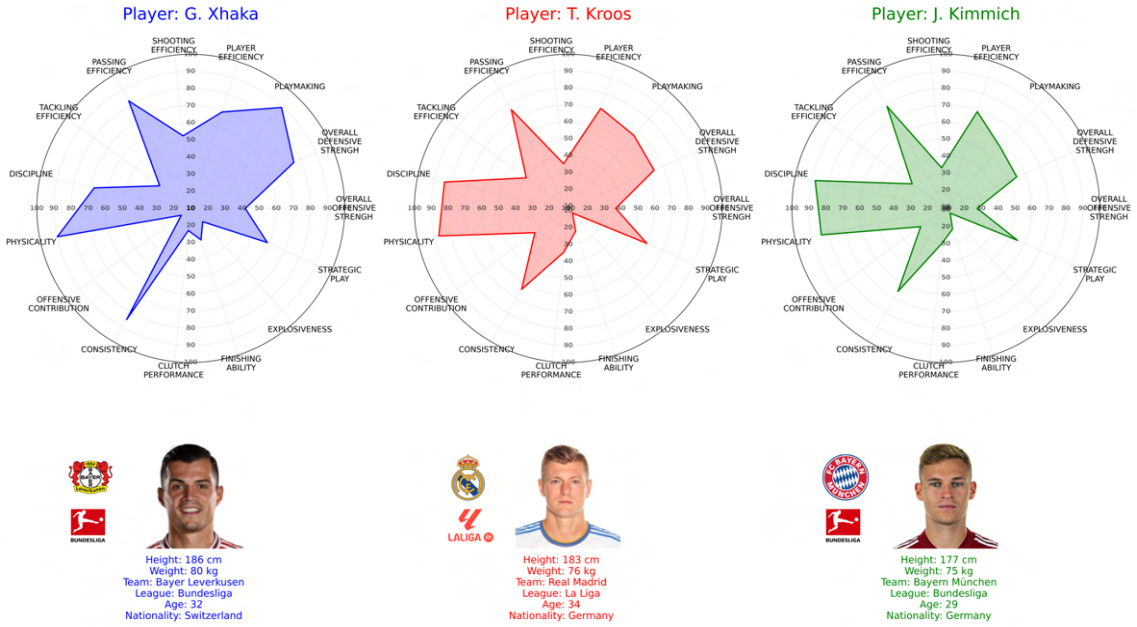


Figure 52 - Overlaid Radar Plot comparison: Xhaka, Kroos and Kimmich

Their radar plots highlight strong similarities in key attributes such as playmaking, passing efficiency, and strategic play, which are crucial for midfielders responsible for dictating tempo and orchestrating possession. This confirms that, unlike Rodri, Xhaka aligns more closely with traditional deep-lying playmakers, making his profile more directly comparable in a scouting analysis.

To further investigate the defensive midfield role, the model is now tested with a ball-winning midfielder rather than a center playmaker. Casemiro, currently at Manchester United, is an ideal example: his game revolves around ball recovery, tackling, and disrupting opposition play rather than orchestrating possession. *Figure 53* presents the top 20 most similar players to Casemiro.

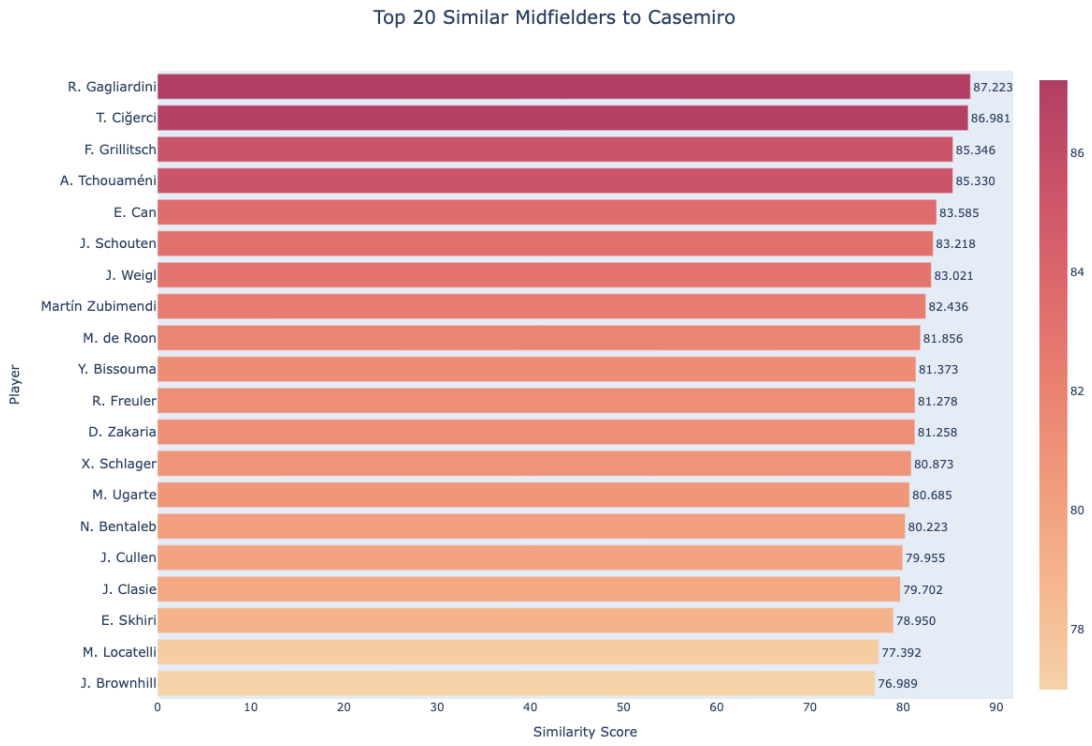


Figure 53 - Top 20 Similar Midfielders to Casemiro by the KNN Model

Notably, none of the players retrieved for Casemiro overlap with those identified for Xhaka, despite both are central defensive midfielders. This confirms that while they share a position on the pitch, their tactical roles and responsibilities are fundamentally different. During the 2024 summer transfer window, Manchester United sought a replacement for Casemiro, exploring options to move him either in the summer or the following winter market. Ultimately, they signed Manuel Ugarte from PSG, a move that perfectly aligns with the model's recommendations, as Ugarte appears among the most similar players to Casemiro.

The radar plot in *Figure 54* highlights their near-identical skill distributions, reinforcing the idea that Ugarte is an ideal replacement for Casemiro in both playing style and tactical role.

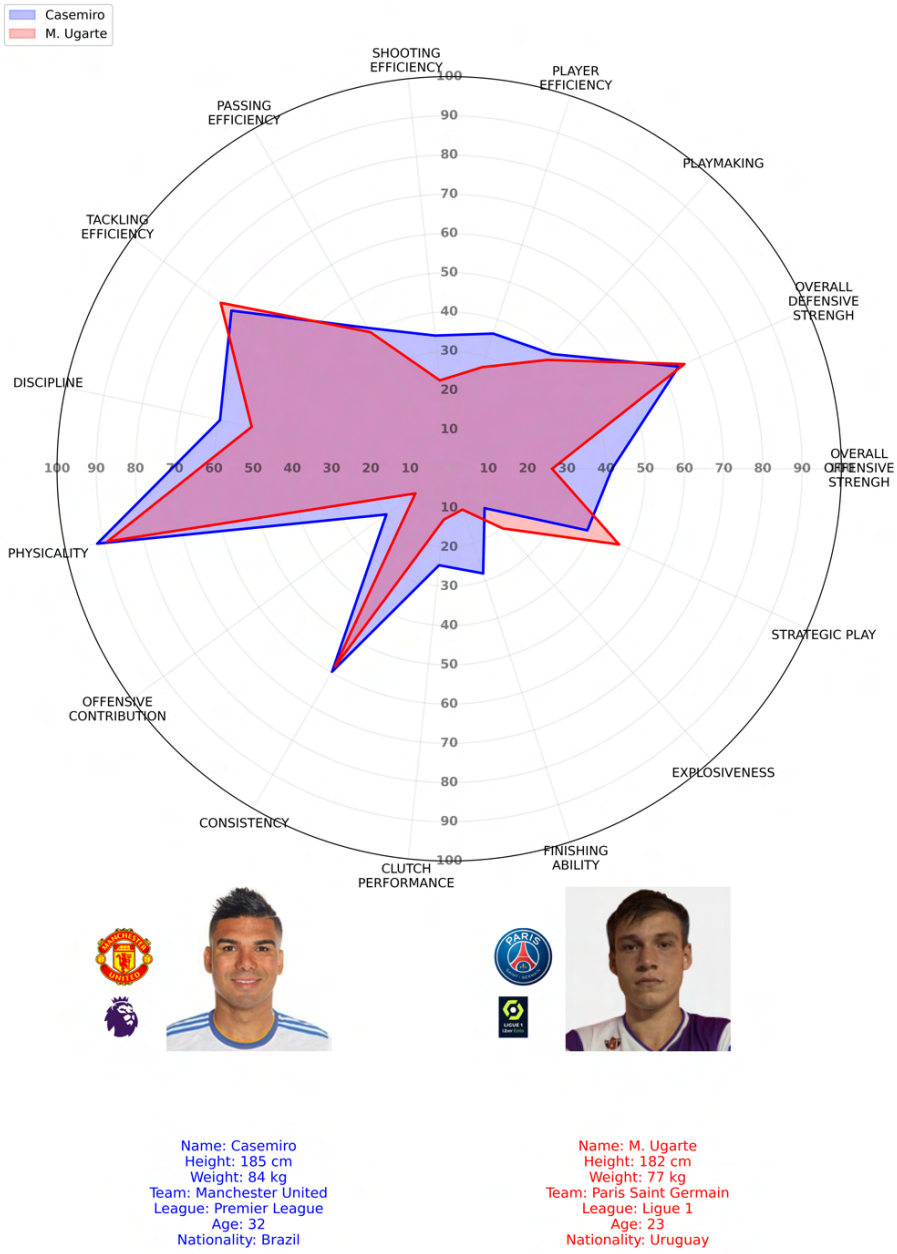


Figure 54 - Overlaid Radar Plot comparison: Casemiro and Ugarte

To further validate the model's ability to distinguish between different midfield roles, we can compare Toni Kroos, a player identified as similar to Xhaka, with Manuel Ugarte, one of Casemiro's closest matches. If their radar plots show distinctly different distributions, it confirms that the model can accurately differentiate between creative defensive midfielders and defensive ball-winning midfielders, even if they play in the exact same position on the field. Conversely, if their shapes align closely, it suggests a limitation in the model's ability to separate these roles effectively.

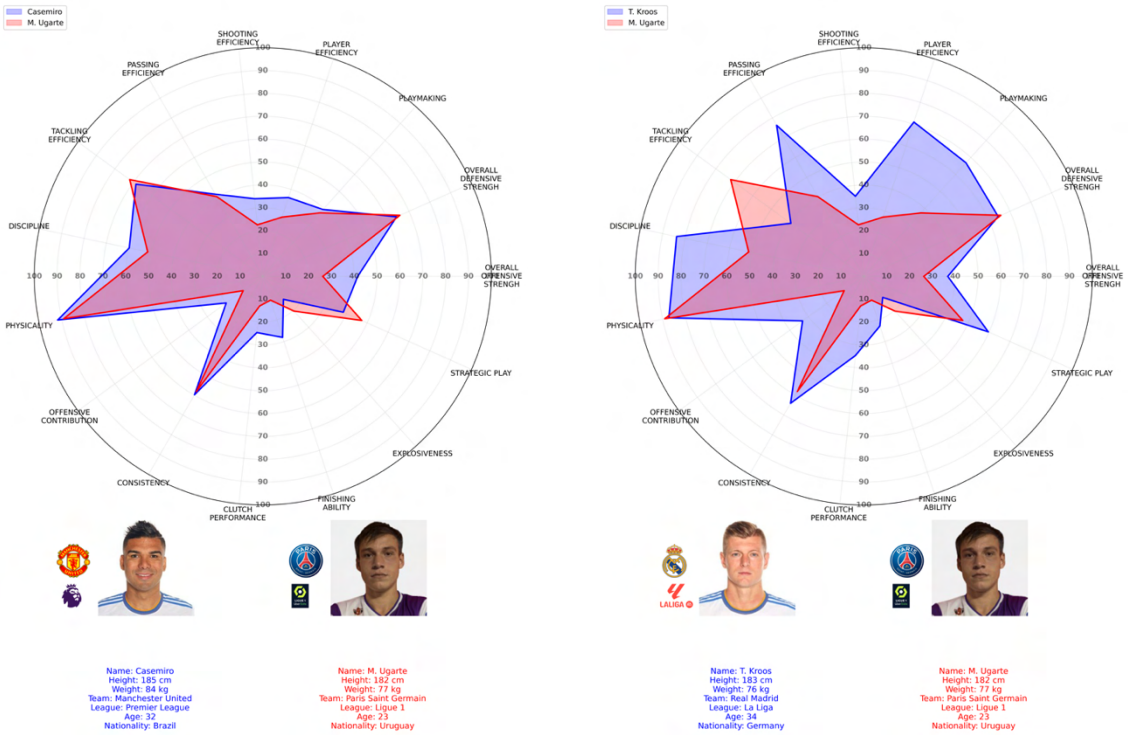


Figure 55 - Overlaid Radar Plot comparison: Kroos and Ugarte vs Casemiro and Ugarte

As expected, the comparison in *Figure 55* reveals a complete contrast: Kroos and Ugarte's skill distributions differ significantly, whereas Ugarte and Casemiro show near-identical radar plot structures. This further supports the model's capability to classify and categorize midfielders based on their actual playing style.

8.2.3 Evaluating the Model on Attacking Midfielders

Attacking midfielders generally possess more distinct offensive traits, making them an interesting category to test within the model.

Case Study: Andrea Colpani

Rather than conducting an exhaustive analysis for each player, this section focuses on key examples to assess the model's effectiveness. Andrea Colpani, an emerging talent at Monza, is a fitting test case due to his attacking offensive playmaking skills, goal-scoring ability, and creative vision. Operating primarily as a *trequartista*, he serves as a

benchmark to evaluate whether the model accurately identifies players with similar offensive profiles. *Figure 56* presents the top 20 recommended midfielders for Colpani according to the model.

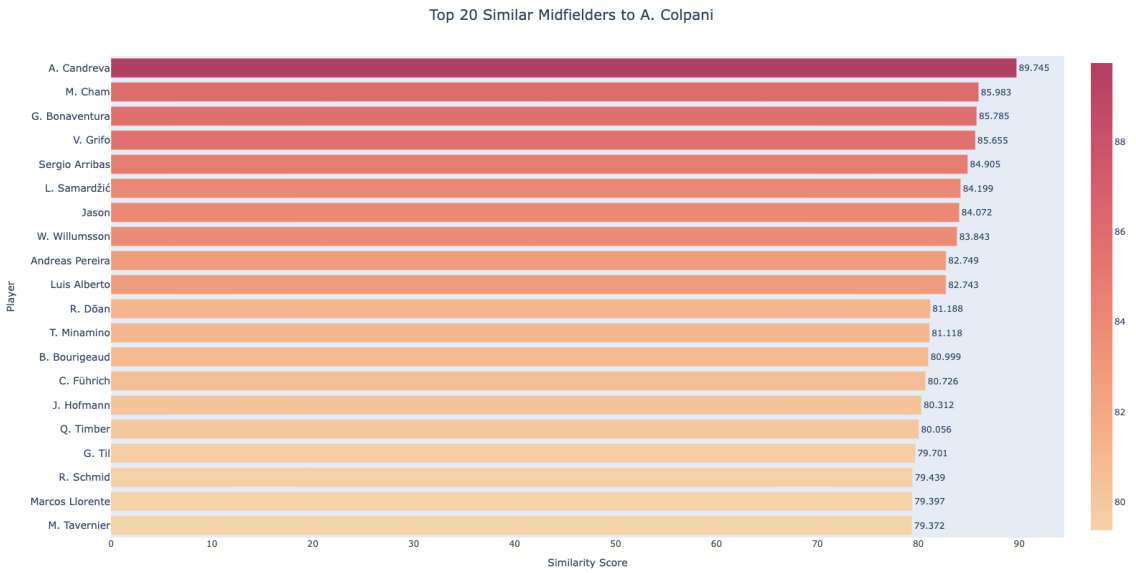


Figure 56 - Top 20 Similar Midfielders to A. Colpani by the KNN Model

A review of the model's output against Transfermarkt assessments confirms that nearly all suggested players are highly offensive midfielders, primarily central attacking midfielders (*trequartisti*). A few exceptions include offensive wingers who occasionally operate centrally, but overall, the model effectively captures Colpani's attacking profile. From a scouting perspective, clubs seeking a Colpani-like replacement might prioritize players with Serie A experience to reduce adaptation risks. Four notable candidates from the model's suggestions are: Antonio Candreva, Giacomo Bonaventura, Lazar Samardžić and Luis Alberto. All are technically gifted, attack-minded midfielders who excel in goal creation and linking play in advanced areas. *Figure 57* compares Colpani against Samardžić and Luis Alberto to further assess the accuracy of these matches.

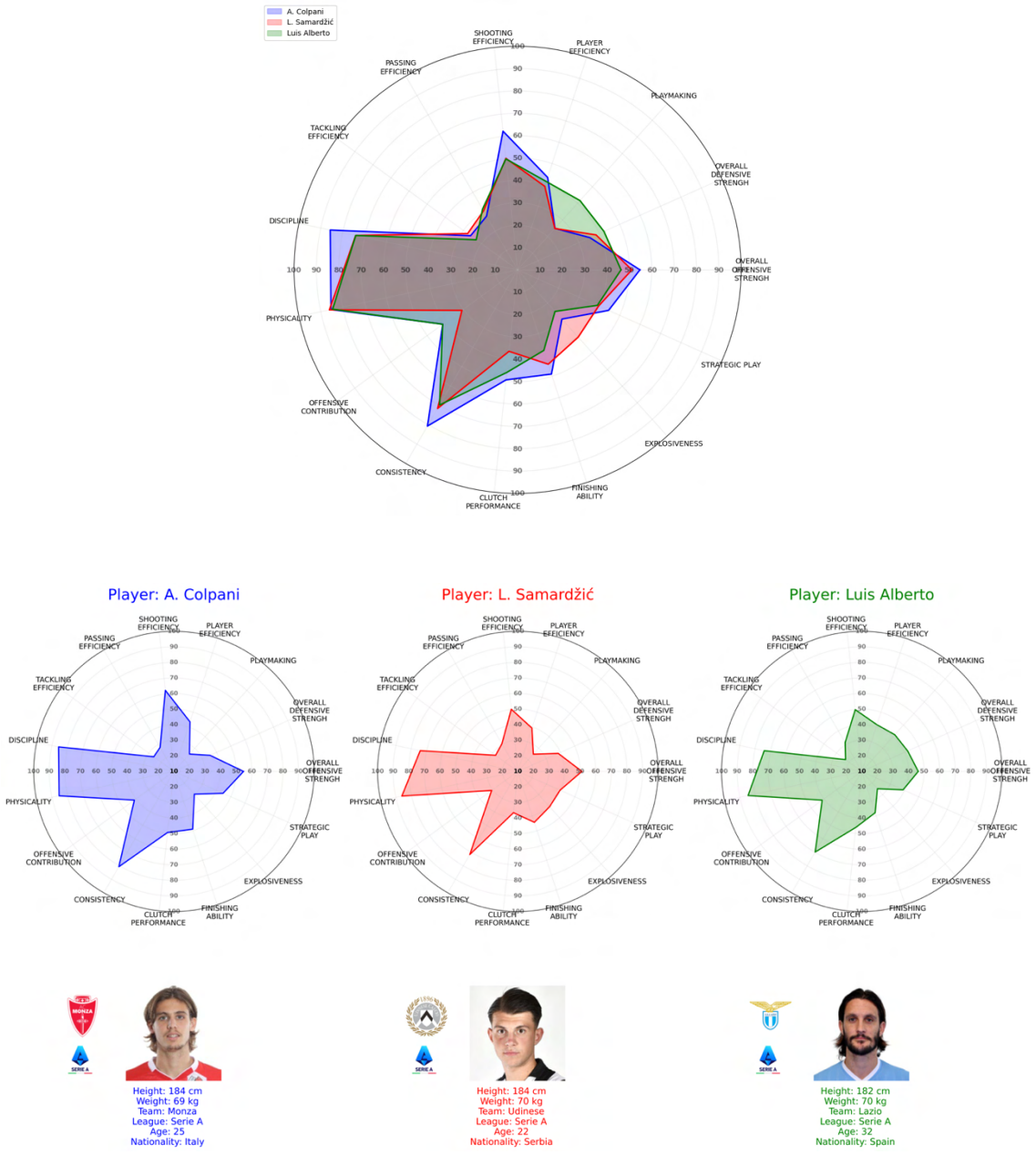


Figure 57 - Overlaid Radar Plot and Side-By-Side Radar Plot comparisons: Colpani, Samardžić, Luis Alberto

As seen in *Figure 57*, the radar plots confirm that their skill distributions are highly similar, validating the model's ability to accurately identify comparable attacking midfielders.

Case Study: Jude Bellingham

Now, can be interesting to move the focus on an elite international profile who, like Rodri, might function as an outlier within his role. Jude Bellingham, known for his versatility and goal-scoring, is an exceptional case. During the season, he scored 19 goals in 28 league matches, numbers more typical of a striker than a midfielder. Given his unique profile, the model may struggle to find direct midfield comparisons.

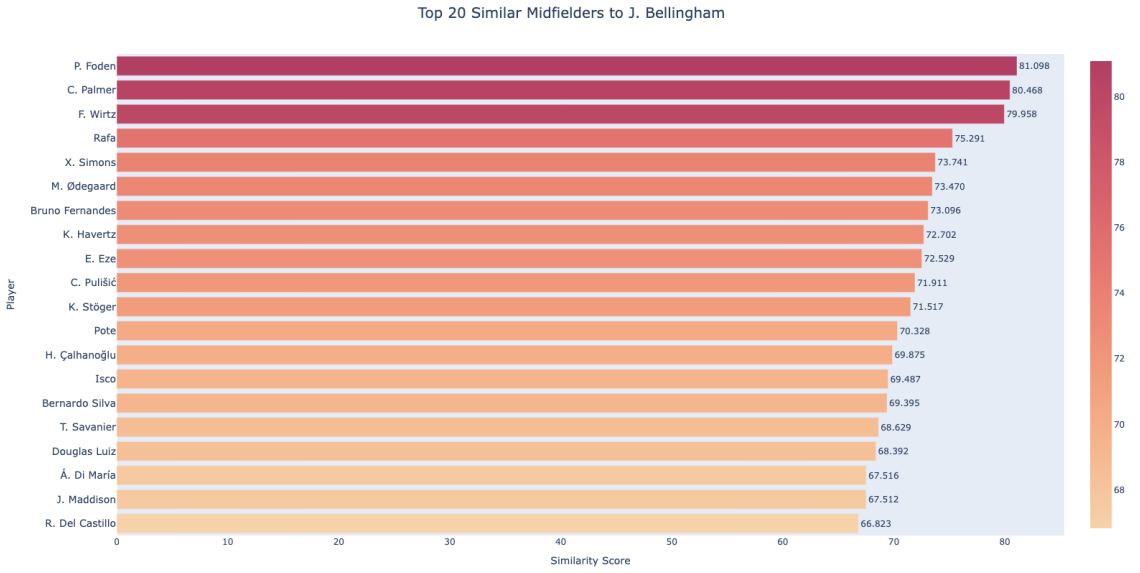


Figure 58 - Top 20 Similar Midfielders to J. Bellingham by the KNN Model

Unlike with Colpani, where nearly all the top 20 results were clear stylistic matches, Bellingham's recommendations are more varied (*Figure 58*), with only two or three attacking midfielders surpassing the 80% similarity threshold. This highlights a fundamental truth in football scouting: *The stronger a player is, the harder he is to replace*.

Bellingham's estimated €180 million market value reflects this reality—there are few players in world football with his exact skill set. Given this, it is insightful to examine his suggested comparisons, selecting:

- One player with a high similarity score (above 80%).
- One player with a lower, but still notable, similarity score (around 70%).

A suitable comparison is with Cole Palmer, another outlier who finished the season with 14 goals and 6 assists, and Kai Havertz, an attacking midfielder who has also played as a forward, recording 13 goals and 7 assists (*Figure 59*).

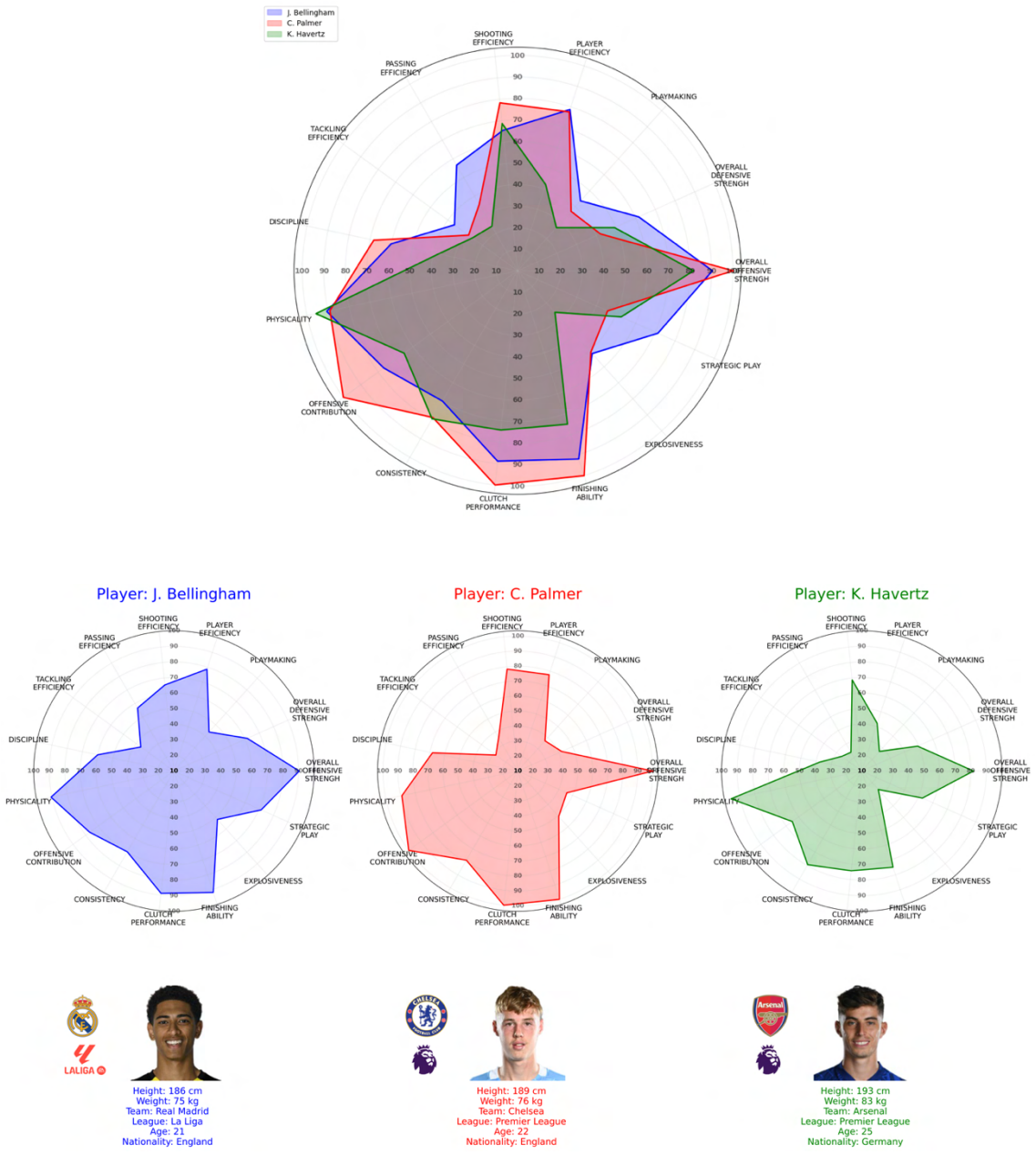


Figure 59 - Overlaid Radar Plot and Side-By-Side Radar Plot comparisons: Colpani, Samardžić, Luis Alberto

8.2.4 Final Validation: Comparing Different Midfield Profiles Across the Model

An important aspect to examine is how similar player profiles vary based on the level of the reference player. Since Colpani and Bellingham are both attacking midfielders but at vastly different levels, their comparable players should share structural similarities. If the model functions correctly, a player similar to Colpani should have a radar plot shape that mirrors that of a player similar to Bellingham, but with lower values across all attributes. This would indicate that while Bellingham is an elite version of the role, the underlying stylistic elements remain consistent. To test this, a player similar to Colpani is compared to a player similar to Bellingham. If their radar plots overlap proportionally, it confirms that the difference lies in performance level rather than playing style.

Conversely, comparing an attacking midfielder to a player similar to Xhaka (a defensive midfielder) should produce a completely different radar plot shape, rather than just higher or lower values. Since Xhaka represents a defensive profile, his skill distribution should extend in different directions, confirming the model's ability to distinguish between midfield roles rather than just overall ability.

This hypothesis is tested using:

- Cole Palmer (Bellingham's close match).
- Lazar Samardžić (Colpani's close match).
- Toni Kroos (Xhaka's close match).

Figure 60 visualizes this comparison through radar plots.

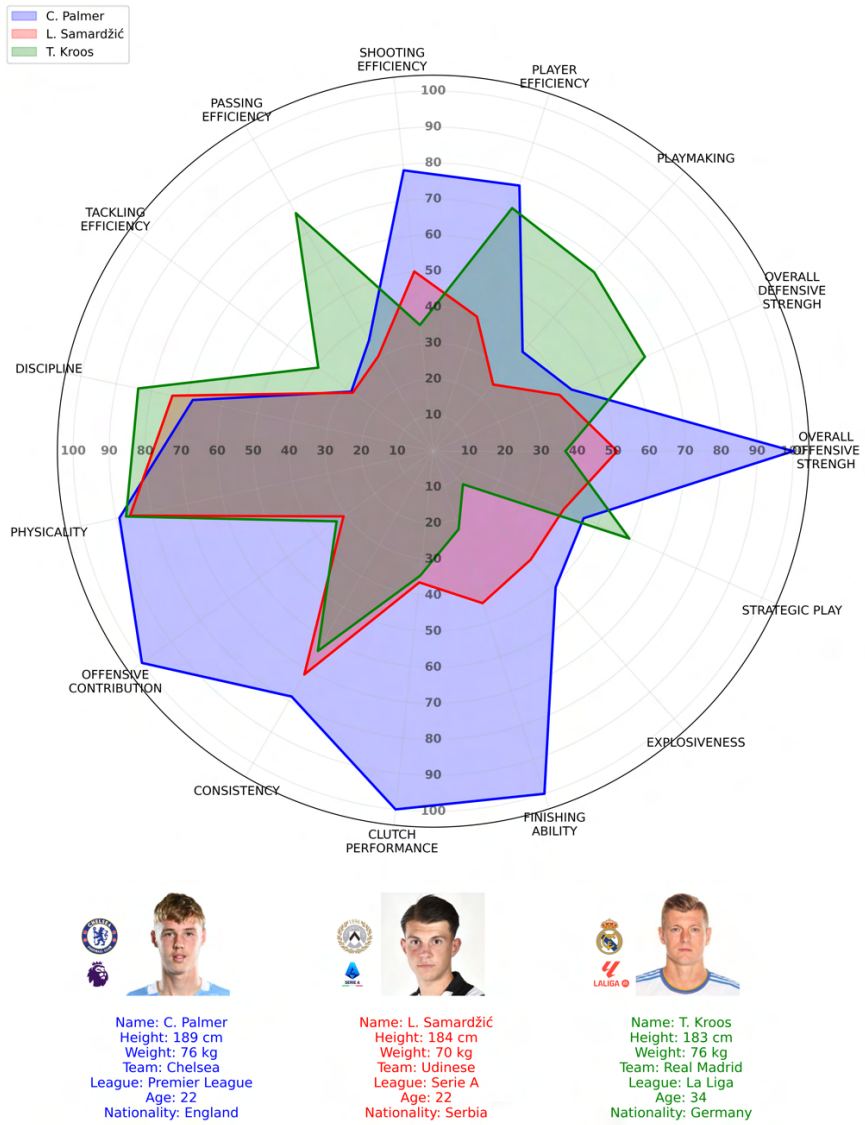


Figure 60 - Overlaid Radar Plot and Side-By-Side Radar Plot comparisons: Palmer, Samardžić, Kroos

The results confirm the hypothesis:

- Samardžić's radar plot fits almost entirely within Palmer's, indicating that while Palmer operates at a higher level, both share a similar playing style.
- Kroos's radar plot, however, has a completely different shape, emphasizing strengths in playmaking, passing efficiency, and strategic play, while scoring much lower in offensive metrics, finishing, and explosiveness.
- Unlike Samardžić and Palmer, whose differences are quantitative, the contrast between Kroos and the attacking midfielders is qualitative, reinforcing that they excel in entirely different aspects of the game.

8.3 Testing the Model on Defenders

8.3.1 Testing the model on Center-Backs

For center-backs, it was adopted a methodology not yet explored in this case studies: selective variable inclusion. As previously explained, one of the key strengths of this model is its flexibility, allowing the user to include or exclude specific variables based on the desired player profile. Several offensive indices are present in the model. However, when evaluating central defenders, goal-scoring and assisting abilities could not be a priority. While a center-back contributing offensively can be a valuable asset, their primary role remains defensive solidity. Of course, especially in modern football, defenders are often required to contribute to the build-up phase and even generate attacking opportunities. However, to establish a baseline, it was started with a more traditional approach before considering a more modern interpretation. By excluding attributes related to playmaking, passing efficiency, shooting, and finishing ability, the model will focus solely on defensive qualities. This approach assumes that a club seeking a center-back prioritizes defensive strength, tackling ability, and physicality, rather than technical skills on the ball. While this may be less common in today's total football philosophy, it remains relevant for teams in need of a physically dominant, rather than ball-playing player.

To test this approach, Dayot Upamecano is chosen as a benchmark[19]. Known for his strength, aggressiveness, and one-on-one defensive ability, Upamecano exemplifies a physically imposing center-back rather than a refined ball-playing defender.

Given the focus on defensive attributes, non-defensive variables are excluded, ensuring the search prioritizes players excelling in core defensive metrics.

The selected variables for this search are (those marked with # are excluded from the search):

[19] A former coach, Peter Zeidler, once noted: "He anticipated almost everything. He wasn't the biggest, fastest, or most technical player, but eight times out of ten, he was first to the ball."

```
index_columns = [
        #'overall_offensive_strengh_index',
        'overall_defensive_strengh_index',
        #'playmaking_index',
        'player_efficiency_index',
        #'shooting_efficiency_index',
        #'passing_efficiency_index',
        'tackling_efficiency_index',
        'discipline_index',
        'physicality_index',
        #'offensive_contribution_index',
        'consistency_index',
        #'clutch_performance_index',
        #'finishing_ability_index',
        #'explosiveness_index',
        'strategic_play_index']
```

This refinement ensures that the model does not prioritize defenders who may have recorded a similar but marginal offensive metrics, while potentially ranking lower those who are more defensively similar to Upamecano. Thus, *Figure 61* analyze now the result for Upamecano with this features selection:

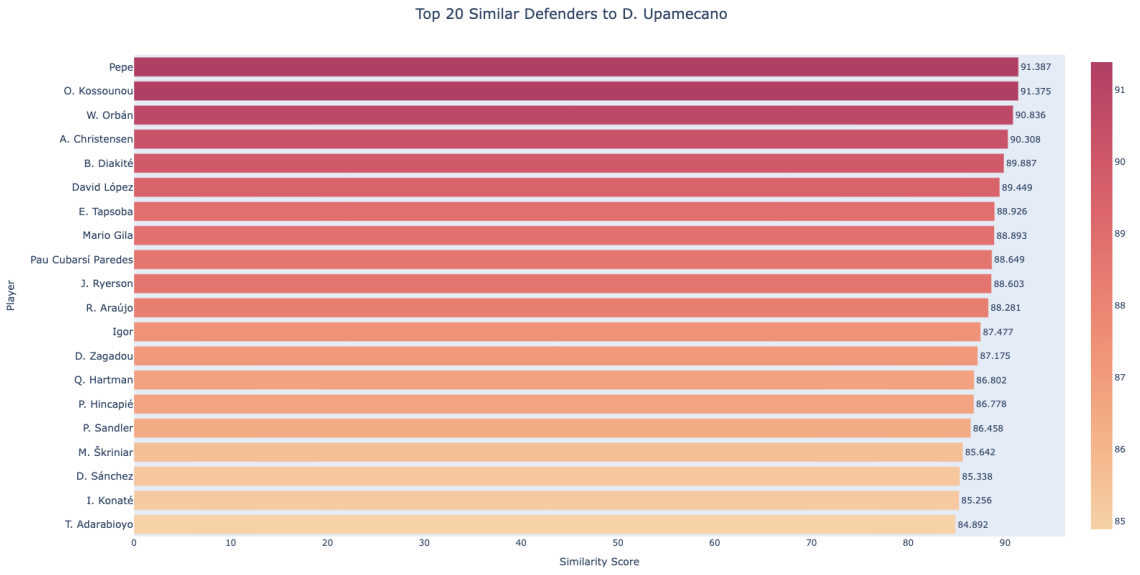


Figure 61 - Top 20 Similar Defenders to D. Upamecano by the KNN Model (Filtered Variables)

The results confirm that all suggested players are central defenders known for their physical strength, and aggressive defensive approach, rather than technical ability. None of them are particularly renowned for technical elegance, which aligns with the search criteria. Numerically, the model has produced an even higher degree of similarity, with

the top-ranked players exceeding 90% similarity and all others above 85%. This aligns with expectations: by reducing the number of input variables, fewer distinguishing features are considered, increasing the likelihood of highly similar matches.

To further investigate, two particularly interesting results are selected for comparison:

- Pepe, ranked as the most similar player, a defender notorious for his extreme defensive aggression rather than technical finesse.
- Milan Škriniar, another top-tier center-back well known for his aggressive defensive approach.

Figure 62 shows the radar plots to assess how closely their skill sets align.

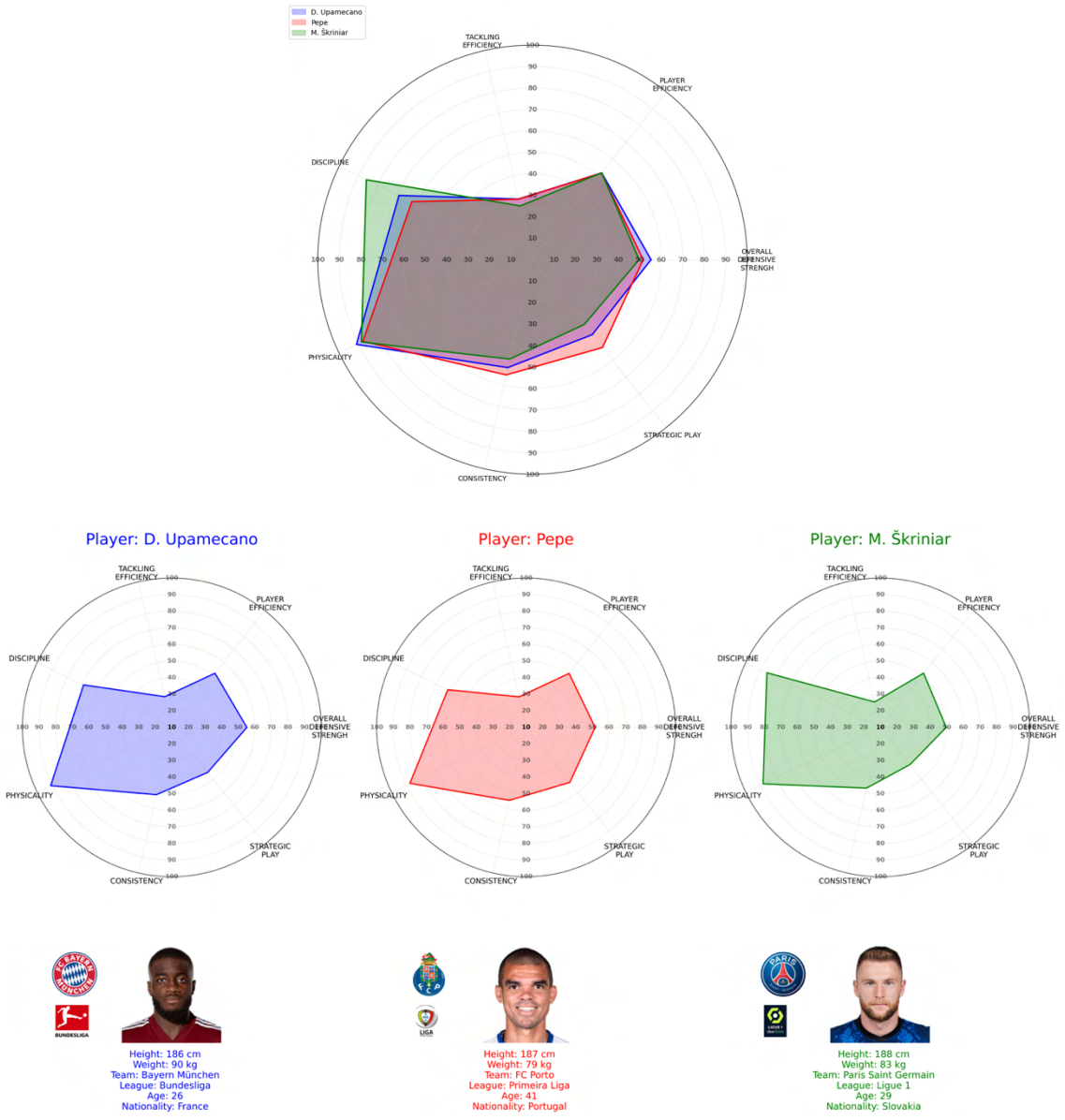


Figure 62 - Overlaid Radar Plot and Side-By-Side Radar Plot comparisons: Upamecano, Pepe, Škriniar

As seen in the radar plot, the skill set distributions of Upamecano, Pepe, and Škriniar are almost perfectly overlapping, confirming the accuracy of the model's recommendations. Their profiles align across all selected defensive attributes, reinforcing their shared playing style as physically dominant, aggressive center-backs.

When reintegrating all variables, a noticeable shift occurs in the similarity rankings.

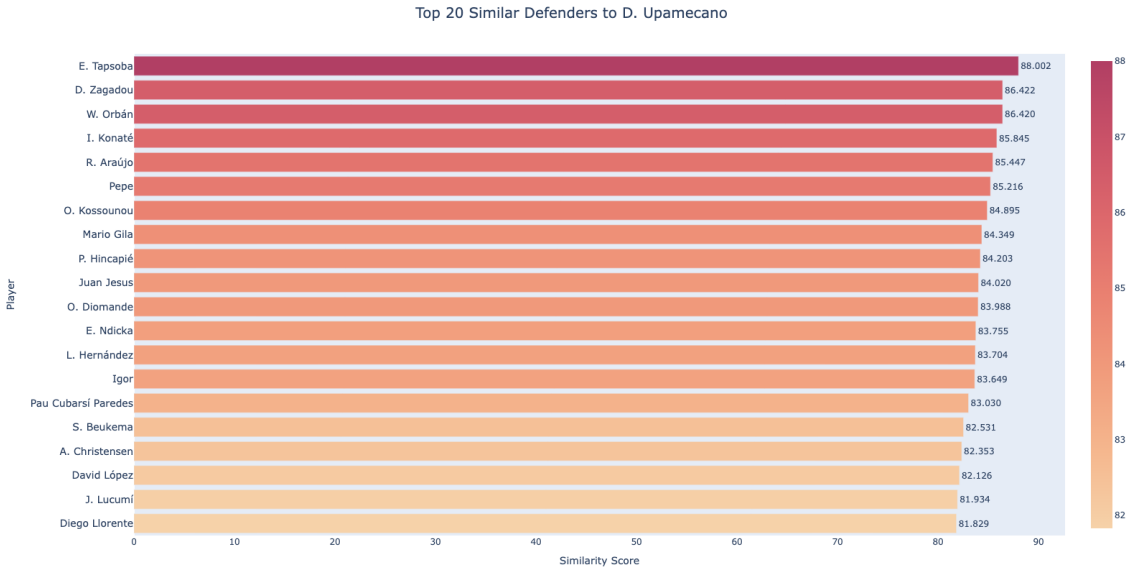


Figure 63 - Top 20 Similar Defenders to D. Upamecano by the KNN Model (All Variables)

As can be seen in *Figure 63*, The overall similarity scores have slightly decreased, confirming that the more variables included, the more differentiation occurs between players, as two players are never exact copies of one another. Notably, Pepe's similarity score drops from 91% to 85%, moving him from the top position down to sixth place, while Škriniar no longer appears among the top 20 most similar players. This suggests that while Pepe was the closest match defensively, considering his overall skill set—including his ability with the ball—Tapsoba emerges as the most similar player to Upamecano. This approach aligns more with the modern tactical approach in football, where defenders are increasingly expected to contribute to build-up play rather than simply clearing the ball. To visualize these changes can be plotted Pepe alongside Tapsoba instead of Škriniar (*Figure 64*).

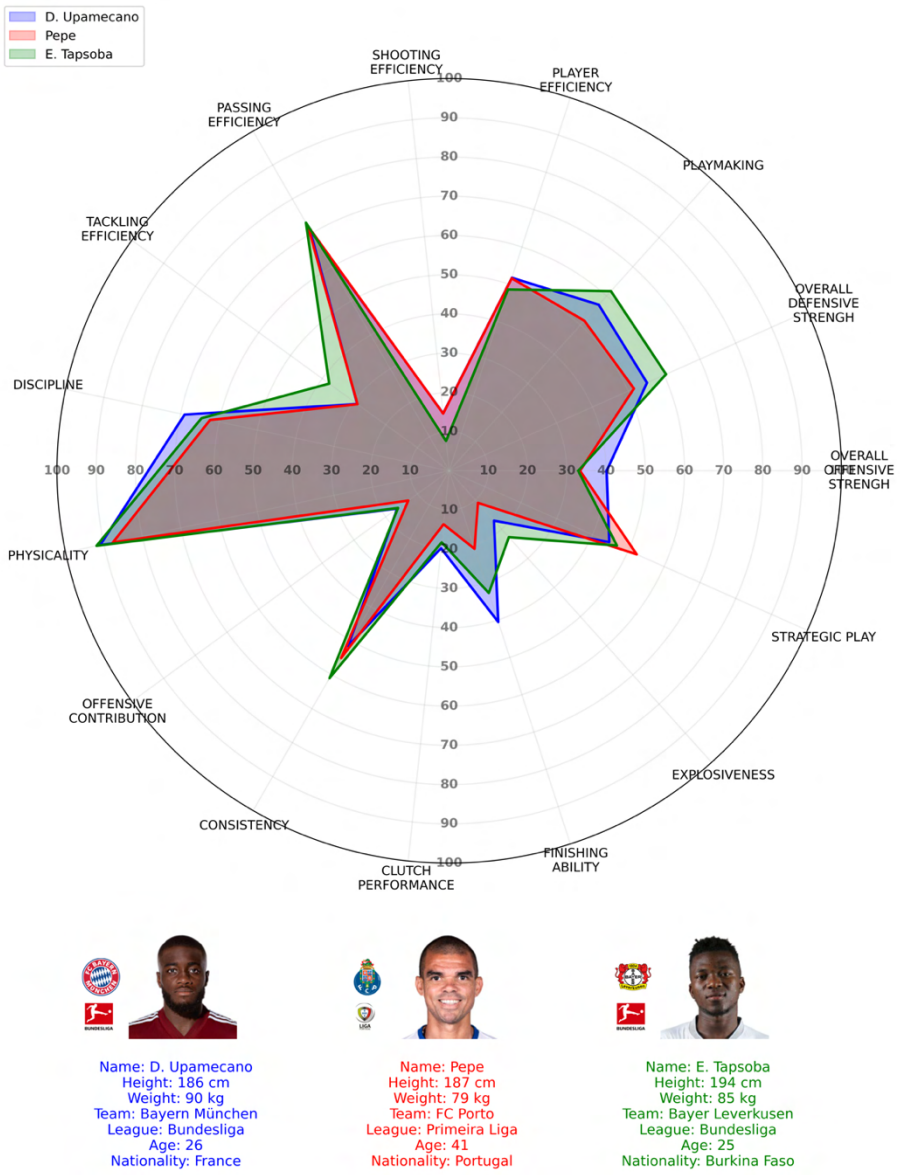


Figure 64 - Overlaid Radar Plot comparison: Upamecano, Pepe and Tapsoba

The updated radar plot reveals that, while Pepe remains highly similar to Upamecano, Tapsoba aligns more closely when considering overall attributes, particularly those related to ball progression like playmaking, passing efficiency and explosiveness.

This analysis highlights the importance of tailored variable selection:

- For a purely defensive search, Pepe and Škriniar emerge as ideal comparisons.
- For a modern center-back with ball-playing ability, Tapsoba proves to be the closest match.

By adjusting the included attributes, the model adapts to different tactical needs.

8.3.2 Testing the model on Full-Backs

Full-backs are among the most tactically complex roles in modern football. Traditionally considered defenders with primarily defensive responsibilities, their role has evolved significantly. Today, full-backs are often key contributors in both defensive and attacking phases, frequently acting as complete box-to-box players. Some, like Achraf Hakimi and Theo Hernández, excel in offensive transitions with their speed and direct attacking runs, while others, such as Trent Alexander-Arnold and Federico Dimarco, contribute more through playmaking and creative passing to the offensive phase.

Given this dual responsibility, testing the model on a modern full-back like Hakimi provides a valuable case study.

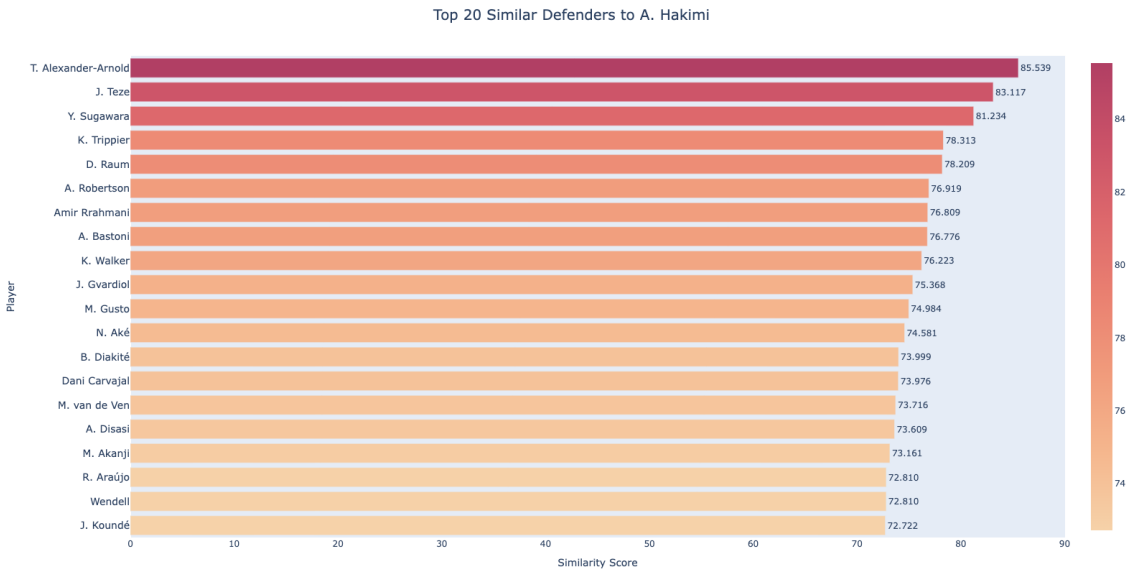


Figure 65 - Top 20 Similar Defenders to A. Hakimi by the KNN Model (All Variables)

The results shown in *Figure 65* reveal an interesting mix of similar players. Many offensive-minded full-backs appear, including Alexander-Arnold, Trippier, and Van de Ven. However, some more defensively oriented full-backs like Carvajal and Walker are also included, along with a few ball-playing center-backs. While at first glance this may seem unexpected, a closer look suggests that the central defenders retrieved are not traditional, physically dominant stoppers (like Pepe or Upamecano shown above) but rather progressive defenders comfortable advancing into midfield.

To further investigate, we compare Hakimi with two selected players:

- Trent Alexander-Arnold, a full-back with strong playmaking and offensive capabilities.
- Alessandro Bastoni, a ball-playing center-back who frequently advances forward in possession.

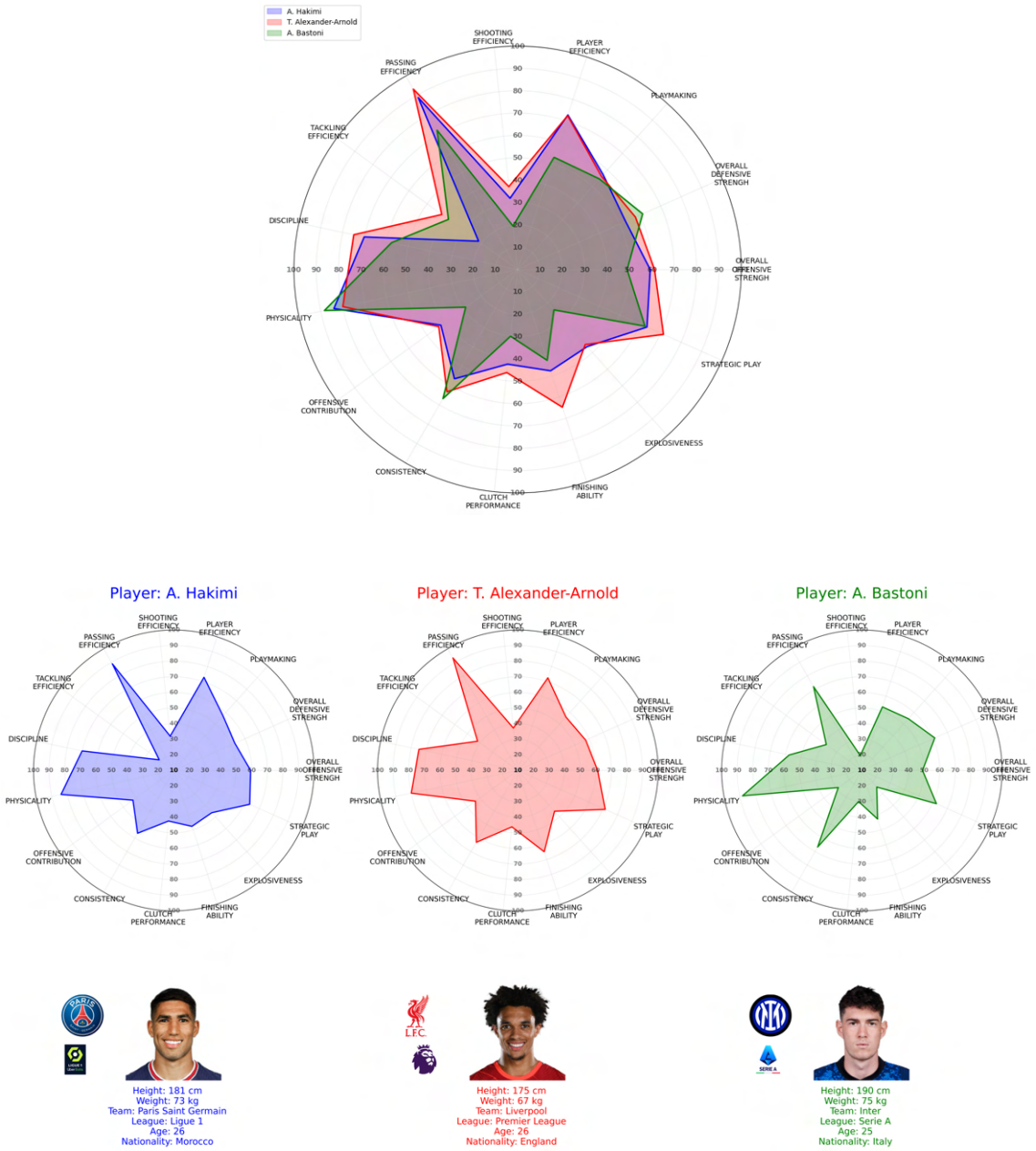


Figure 66 - Overlaid Radar Plot and Side-By-Side Radar Plot comparisons: Hakimi, Alexander-Arnold and Bastoni

The radar plots in *Figure 66* provides clear insights. As expected, Alexander-Arnold's profile aligns much more closely with Hakimi's, whereas Bastoni's shape differs significantly. However, when overlaid, the distinctions become even more apparent.

- Both Hakimi and Alexander-Arnold excel in offensive attributes, particularly in finishing ability, shooting efficiency, overall offensive strength, offensive contribution, and clutch performance.
- Bastoni, by contrast, scores higher in consistency, physicality, and overall defensive strength, emphasizing his role as a defensive-minded player rather than an attacking outlet.

This highlights a key takeaway: full-backs inherently possess a hybrid nature, engaging heavily in both defensive and offensive actions. Because of this, their positional versatility may cause some overlap with progressive center-backs in data-driven models. However, this does not indicate an issue with the model but rather emphasizes the importance of refining selection criteria based on scouting needs. While feature filtering is less crucial for attackers, in this case, given the highly hybrid nature of the role, the model struggles more with full-backs.

For a team specifically searching for an attacking full-back, filtering by defensive attributes alone may not be ideal. Instead, focusing on offensive indicators would produce more relevant results. Through extensive testing, this pattern has emerged more frequently among full-backs compared to other roles. The model consistently distinguishes midfielders, wingers, strikers, and center-backs with high accuracy, while full-backs, due to their tactical duality, occasionally overlap with progressive defenders. This nuance reinforces the importance of contextual scouting judgment alongside data-driven insights.

CHAPTER 9:
CONCLUSIONS

9.1 Machine Learning for Football Scouting: Summary and Applications

This study has demonstrated how a data-driven approach can enhance player scouting and recruitment, addressing the increasing complexity of modern football. Traditional scouting, while still fundamental, faces significant limitations in terms of scalability, consistency, and resource efficiency. In response to these challenges, the machine learning model developed in this study wanted to support clubs in identifying, comparing, and evaluating players.

Performance indices were constructed and validated to quantify key attributes, with adjustments for league difficulty ensuring cross-league comparability and enabling player analysis on a standardized scale. A K-Nearest Neighbors (KNN) model was implemented to create an efficient and effective scouting tool, capable of identifying statistically similar players based not only on performance metrics but also on stylistic and tactical tendencies. The model is adaptable, allowing users to tailor queries by customizing the k value and selecting specific variables, user-friendly, featuring an intuitive player search system, and highly interpretable, integrating quantitative similarity scores with graphical analysis for clearer insights. The results confirm the effectiveness of this approach. Across multiple positions—from offensive strikers, wingers and midfielders to defensive midfielders, center backs and full-backs—the model successfully retrieved players with highly comparable playing styles, even without relying on explicit positional labels.

As outlined initially, this scouting algorithm should be seen as a starting point rather than a definitive decision-making tool. Its primary value lies in its ability to objectively filter large player databases, identifying those with similar performance profiles. However, the scouting process remains multi-layered, requiring further refinement before reaching a final decision. A club looking to reinforce its squad can start by broadening the shortlist, analyzing not just the top 20, but potentially 50, 100 or more players who statistically match the desired profile. Using visualization tools, analysts can compare key attributes, assess tactical compatibility, and investigate performance tendencies, progressively

refining the selection. This structured process allows for a more efficient and objective scouting approach, ensuring that only the most suitable candidates are considered for further evaluation and in-depth scouting.

Once the shortlist has been refined, the final step involves direct scouting, where the selected targets are closely monitored through video analysis and in-person evaluations. While data-driven insights streamline the initial selection, human judgment remains essential in assessing intangibles attributes that data cannot fully capture or may misrepresent. Returning to the problem initially posed, this approach is particularly valuable for smaller clubs that do not have the human and financial resources to maintain a global scouting network. Rather than deploying scouts universally, they can analyze player performance at scale, compare profiles, and narrow the focus to a select group ensuring that financial and human investments are directed toward the most promising candidates. By combining data-driven analysis with targeted scouting, clubs can reduce costs, improve decision-making, and enhance their competitiveness, even with limited budgets.

9.2 Limitations and Future Steps

9.2.1 Enhancing Data Availability

A potential improvement to the study would be the inclusion of additional player attributes, such as preferred foot. Wingers and full-backs, for instance, are often selected based on their dominant foot to fit specific tactical roles on either the left or right flank. While the model correctly identifies similar players based on positional role and responsibilities, having this additional variable would further refine comparisons. Currently, this limitation can be mitigated by cross-referencing with external sources like Transfermarkt, which provide this information.

Similarly, while the model effectively recognizes players without relying on explicit positional labels, having access to their official registered positions (not for model input, but for validation purposes) would be beneficial. As previously discussed, the strength of this approach lies in going beyond rigid position labels, since players with the same designation can exhibit vastly different styles, while those with different labels may share significant similarities. However, access to official positional data would reduce the

reliance on external tools like Transfermarkt for manual verification. The same applies to contract values and salary data, which, while not integrated into the model to avoid bias, remain valuable contextual factors for real-world scouting decisions. Unfortunately, the API used in this study does not provide this information, and Transfermarkt strictly prohibits data extraction through both scraping and API calls, significantly limiting access to these metrics.

A major step forward would be the inclusion of spatial performance data, which would significantly improve the accuracy of player comparisons. This would allow the model to move beyond aggregated statistics and incorporate key variables such as Total distance in Km covered during a match, Directionality of passes (forward vs. backward, rather than just pass completion), Shot location and (missing a chance from close range in front of goal is far more significant than missing a long-range shot).
These additions would enable both the creation of new metrics, such as the famous xG, to include new indices or refine existing ones, making them even more representative of a player's true contribution. However, the limitation is not methodological but rather data-related—the approach remains valid and applicable, but access to richer data would enhance empirical quality. Unfortunately, spatial data is not publicly available. Major providers such as Opta, StatsBomb and Wyscout, sell these datasets directly to professional clubs and broadcasters, making open access extremely challenging. Football, as a highly competitive industry, restricts public data availability since clubs gain a strategic advantage from exclusive access to proprietary analytics. Even in cases where public databases exist, many impose strict limitations on API usage or data scraping, further restricting access.

Even within the current dataset constraints, future work could broaden the scope of analysis by:

- Including lower-ranked leagues in the UEFA coefficient system, while acknowledging that player performances in these leagues would carry significantly less weight due to the lower competitive level.
- Incorporating performance in other competitions like European and domestic cups or national team tournaments

By integrating richer datasets and expanding the coverage of leagues and competitions, this model could evolve into an even more comprehensive and practical scouting tool.

9.2.2 Optimizing Index Weighting

An aspect that could be further improved in future developments concerns the mathematical construction of performance indices. While the methodology provides an objective framework for comparing players and delivers realistic and consistent results, the weights assigned to each index were determined subjectively. These weights were based on detailed analysis and evaluation but were not derived through a purely mathematical optimization process.

As extensively discussed, the indices were validated using machine learning models, which confirmed their mathematical consistency and practical relevance. However, a key future step would be to develop an optimization model or leverage machine learning techniques to fine-tune and optimize the assigned weights. One possible approach could involve for example evolutionary heuristic metrics[20], which function by iteratively adjusting and optimizing parameter weights based on results.

Among the various future developments considered so far, automating the optimization of index weights appears less immediately impactful than other potential improvements. While refining weight distributions through mathematical optimization techniques is a compelling avenue, it differs significantly from player performance analysis, which relies on objective data and structured comparisons. A crucial distinction must be made between quantifying individual player attributes (a process that can be effectively modeled through actual data) and interpreting the game itself, which remains a far more complex challenge. For a weighting system to be fully optimized, it would not only mean to adjust numerical values but learn to understand football itself, recognizing which aspects of the game within each index hold greater importance in different contexts. This understanding can only be derived from data, but the real challenge and fundamental question is: how much data would be required for AI to develop this understanding? More importantly, how

[20] In this context, an evolutionary algorithm could explore different weight distributions. By simulating multiple iterations and selecting the most effective configurations, the system could gradually refine the weight structure of performance indices, reducing reliance on manual calibration.

much data would need to be collected? Practically speaking, every single movement on the pitch, even standing still, is a data point that could influence the events.

While data analysis is an irreplaceable and highly impactful tool, football is not an exact science. Unlike purely mathematical systems, it remains partially unpredictable, ss the saying goes, “the ball is round”, reminding us that no model can entirely encapsulate the unpredictability and variability of the game. This does not diminish the value of the model developed in this study. As demonstrated, the model is also capable of recognizing certain tactical patterns, effectively identifying players with similar playing styles, but its purpose is not to interpret all the football’s complexities. As data collection and AI capabilities continue to advance, build a fully optimizing football-specific weight distributions become a more viable direction, but at present and with the available data, it remains a long-term prospect rather than an immediately actionable step.

9.2.3 Improving Accessibility

An additional and more practical future step would be to further enhance user accessibility by developing a web application that makes the model even more intuitive and seamless to use. This would provide a more interactive and streamlined experience, integrating all the visualizations and functionalities of the model while allowing clubs, analysts, and scouts to explore and compare players even more easily.

BIBLIOGRAPHY

Tanapatpiboon, K., Chalidabhongse, T. H., & Kamal, M. M. (2024). "Research on Video Target Detection and Tracking in Sports Analytics: A Comparative Study of YOLO Implementations."

Zou, X., Huang, Y., Zhou, N., & Fang, Z. (2022). Research on video target detection and tracking in football matches.

Mead, J., O'Hare, A., & McMenemy, P. (2023). Expected goals in football: Improving model performance and demonstrating value.

Pei, Y., De Silva, V., & Caine, M. (2023). Passing Heatmap Prediction Based on Transformer Model Using Tracking Data For Football Analytics.

Freitas, D. N., Mostafa, S. S., Caldeira, R., Santos, F., Fermé, E., Gouveia, É. R., & Morgado-Dias, F. (2025). Predicting noncontact injuries of professional football players using machine learning

Satvedi, R., & Pyne, D. B. (2022). Injury Prediction for Soccer Players Using Machine Learning. Proceedings of the 7th International Conference on Signal and Image Processing

Jauhiainen, S., Äyrämö, S., Forsman, H., & Kauppi, J.-P. (2019). Talent identification in soccer using a one-class support vector machine. International Journal of Computer Science in Sport.

Abhinav, B. V., Nandan, A. J., Gurikar, A. S., Joshi, A., Pandharkar, A., & Prajwala, T. R. (2024). An xG Based Football Scouting System Using Machine Learning Techniques.

Gómez-Rubio, V., Lagos, J., & Palmí-Perales, F. (2024). Spatial similarity index for scouting in football. Preprint.

Sayeed, H. (2023). A Machine Learning Framework to Scout Football Players. MSc Research Project, National College of Ireland.

IBM. (n.d.). What is an API? https://www.ibm.com/think/topics/api

Shafranovich, Y. (2005). Common Format and MIME Type for Comma-Separated Values (CSV) Files. Internet Engineering Task Force (IETF).

McKinney, W. (2010). Data structures for statistical computing in Python. In Proceedings of the 9th Python in Science Conference.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning with Applications in R.

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. Informatica.

Zhang, C. (2000). Supervised Learning.

Google Developers (n.d.). Classification: Accuracy, Precision, Recall. https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall?&hl=it

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.).

Abu Saa, S., Al-Zoubi, H., & Abu-Naser, S. S. (2023). Investigation of the Role of Test Size, Random State, and Dataset in the Accuracy of Classification Algorithms. ResearchGate.

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). Applied Logistic Regression (3rd ed.). Wiley.

Breiman, L. (2001). Random forests. Machine Learning.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine Learning.

Quinlan, J. R. (1986). Induction of decision trees. Machine Learning.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics.

Scikit-learn Developers. (n.d.). Glossary of Common Terms. https://scikit-learn.org/stable/glossary.html#term-random-state

Bishop, C. M. (2006). Pattern Recognition and Machine Learning.

Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

IBM. (n.d.). K-Nearest Neighbors (KNN). IBM Think. https://www.ibm.com/think/topics/knn

Atilim Cetin. (2019). KNN - K-Nearest Neighbors (1). Towards Data Science. https://medium.com/towards-data-science/knn-k-nearest-neighbors-1-a4707b24bd1d.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. IEEE Transactions on Information Theory.

Biau, G., & Devroye, L. (2015). Lectures on the Nearest Neighbor Method.

Müller, C., Rein, R., & Memmert, D. (2023). Visual analytics of soccer player performance using objective ratings.

Smit, R., van den Berg, R., & van Lingen, R. (2014). Radar plots for visualizing multiple patient-related outcome measures in clinical practice.

Stanojevic, J., & Stanojevic, M. (2022). Violin-boxplot and enhanced radar plot as components of effective graphical dashboards: An educational example of sports analytics.

Navarro, G. (2001). A guided tour to approximate string matching. ACM computing surveys (CSUR), 33(1), 31-88.

APPENDIX

A. K-Nearest Neighbors (KNN): Theoretical Foundations

The K-Nearest Neighbors (KNN) algorithm is a fundamental instance-based learning method widely used in classification and similarity search tasks. Unlike traditional machine learning models that require explicit training, KNN operates as a lazy learner, meaning it does not build a generalization model during a training phase. Instead, it stores all training instances and makes predictions dynamically based on the similarity between new data points and existing samples (*IBM, n.d.*).

KNN operates on a simple yet effective principle: a new data point is analyzed based on the majority features of its K closest neighbors in the feature space. The model follows three core steps:

I. *Feature Space Representation*:

Each instance in the dataset is represented as a point in a multidimensional space, where each axis corresponds to a feature (in this study, the player performance indices).

II. *Distance Calculation*:

To determine the similarity between a query point and existing instances, KNN computes a distance metric. The most commonly used metric is the *Euclidean distance*:

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$$

where p and q are two data points (players in this study), and n is the number of features (performance indices).

The Euclidean distance ensures that instances with similar attribute values are positioned closer in the feature space, allowing the KNN algorithm to effectively identify the most comparable data points.

III. *Neighbor Selectio*n:

*A*fter computing the distances between the query instance and all points in the dataset, the algorithm selects the K nearest neighbors.

In the context of this research, KNN is used as a similarity search algorithm rather than a classification or regression model. The key objective is to retrieve players with the most comparable performance attributes to a given reference player. The parameter K—representing the number of retrieved players—can be adjusted based on the desired level of similarity. A lower K provides highly specific recommendations, whereas a higher K includes a broader range of comparable players.

In classification problems, increasing K can negatively impact accuracy, as the model may incorporate more distant and potentially less relevant neighbors, leading to misclassification. However, in this specific similarity search context, K does not affect accuracy since the goal is not classification but rather the retrieval of similar players. Instead, K simply determines the number of nearest neighbors considered, allowing for finer or broader comparisons based on the selection criteria.

Having established the theoretical foundation of the K-Nearest Neighbors algorithm, its specific implementation within this study can now be examined. The following section provides a detailed explanation of the KNN implementation in this scouting system.

B. KNN Application in Player Scouting

The theoretical foundations of K-Nearest Neighbors, including its underlying principles, are detailed in the appendix, allowing the focus here to remain on its application within this scouting model. The objective is to leverage KNN as a similarity search tool, identifying players with the closest performance profiles to a given reference

player. The model ranks players based on their closeness to the selected individual, allowing for an intuitive and flexible comparison.

The following visual examples serve to illustrate the model's logic. As an initial example, the representation is limited to two dimensions, specifically the Explosiveness Index and the Player Efficiency Index are taken as example, to provide a clear and intuitive understanding of how the KNN model identifies similar players.

I. Mapping Players in the Feature Space

Every player in the dataset is represented as a point in a high-dimensional space, where each axis corresponds to one of the computed performance indices.

The following visualization (*Figure 67*) provides an example of this distribution in the two dimensions, with each point representing a player positioned by his scores in the selected indices.

Red points correspond to attackers, blue points represent midfielders, green points denote defenders.

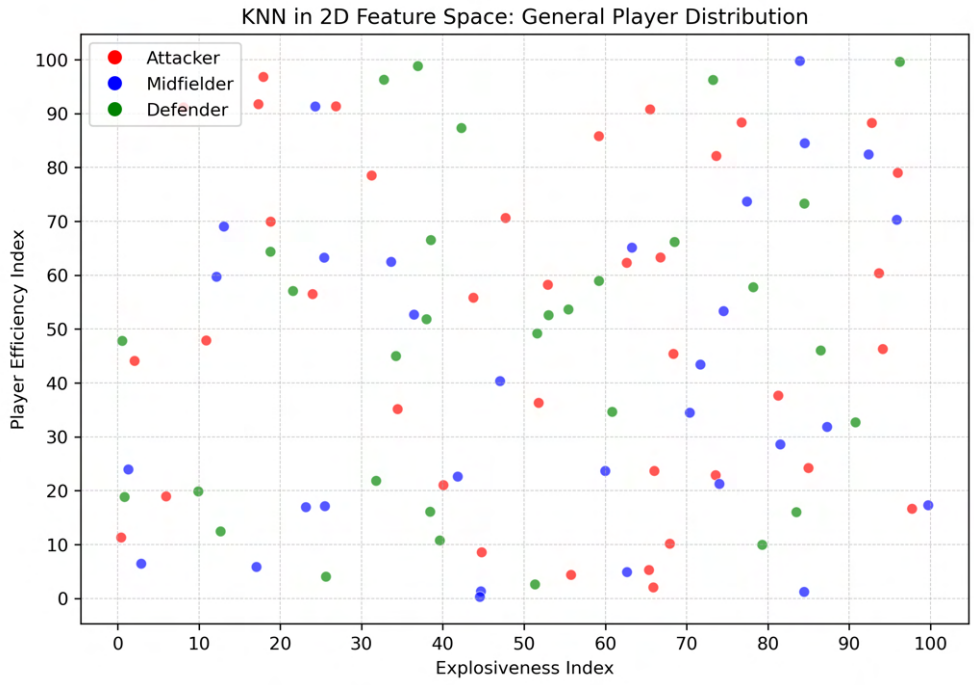


Figure 67 - KNN in 2D Feature Space: General Player Distribution

This spatial representation is important and highly interpretable because players with similar attributes will be positioned closer together, while those with significantly different playstyles will be farther apart. Although this example uses only two indices for visualization purposes, the actual model operates in a multi-dimensional space, incorporating all selected performance metrics to compute similarity.

II. Selecting a Target Player

Once a player of interest is chosen, the model searches for the most similar players within the feature space. In the second visualization (*Figure 68*), taken as an example, the target player is a midfielder (represented by a blue point) and is highlighted with a red circle.

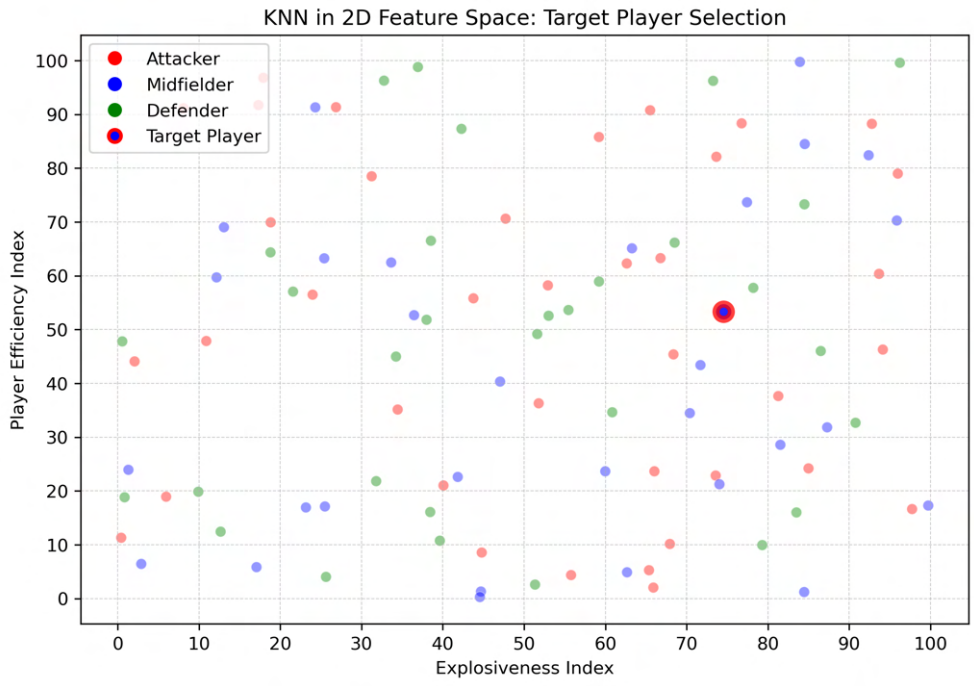


Figure 68 - KNN in 2D Feature Space: Target Player Selection

At this stage, the algorithm prepares to compare this player against all others.

III. Neighbor Selection and Retrieving the Nearest Neighbors

The KNN model then identifies the K most similar players. These players are the closest in terms of statistical profile, meaning their performance attributes are numerically similar to those of the target player.

In the third visualization (*Figure 69*), an example is shown where the KNN model is set to K=5, meaning it searches for the five most similar players to the selected target player. These nearest neighbors are marked in yellow.

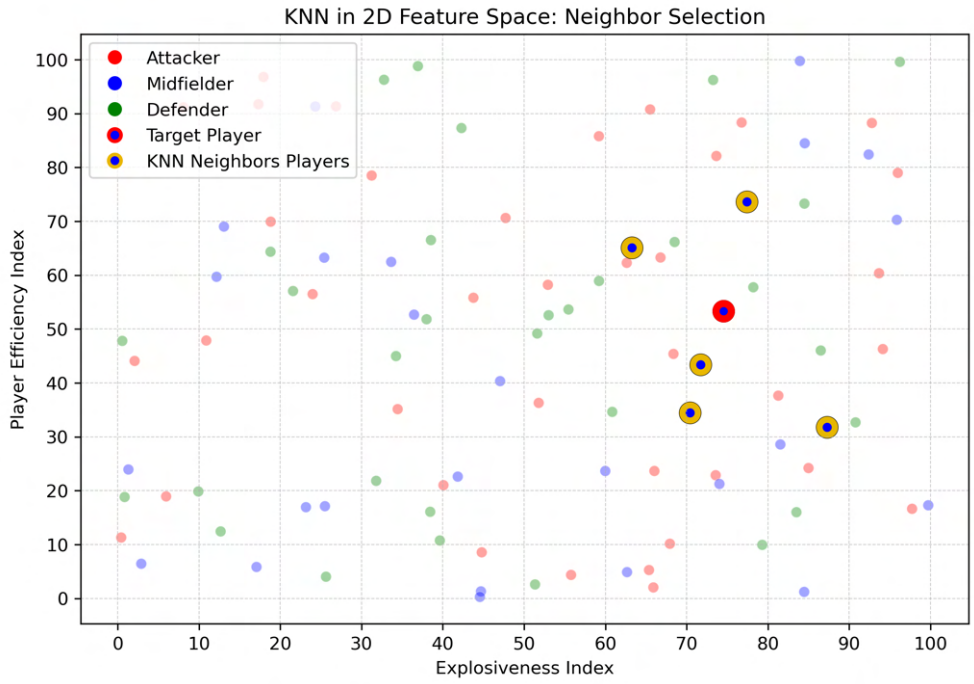


Figure 69 - KNN in 2D Feature Space: Neighbor Selection

A crucial aspect of this implementation is that the model only searches for players of the same role. This constraint ensures that comparisons remain meaningful—since all indices were normalized within each role, the similarity search must be role-specific to maintain logical consistency. Comparing a defender to an attacker, for example, would be misleading due to differences in index scaling but also completely illogical in football-specific and real-world scouting scenarios. To ensure both statistical validity and football realism, the model compares each player only with others in the same role. This guarantees that similarity assessments align with real-world football logic, preserving the

integrity and usefulness of the analysis. In this case, since the target player is a midfielder, the model retrieves the five closest midfielders in the feature space.

IV. Measuring Similarity with Euclidean Distance

To determine which players are most similar, the algorithm computes the Euclidean distance between the target player and every other player in the dataset. The fourth visualization illustrates this by showing dashed lines connecting the target player to its K nearest neighbors, with numerical distance values displayed.

- A smaller distance indicates a higher degree of similarity.
- A larger distance means the player is relatively less similar.

Therefore, the players with the smallest distance values are selected as the closest matches. As shown in the following visualization (*Figure 70*), the model connects the target player to its nearest neighbors same-role players with dashed lines, displaying the calculated distances.

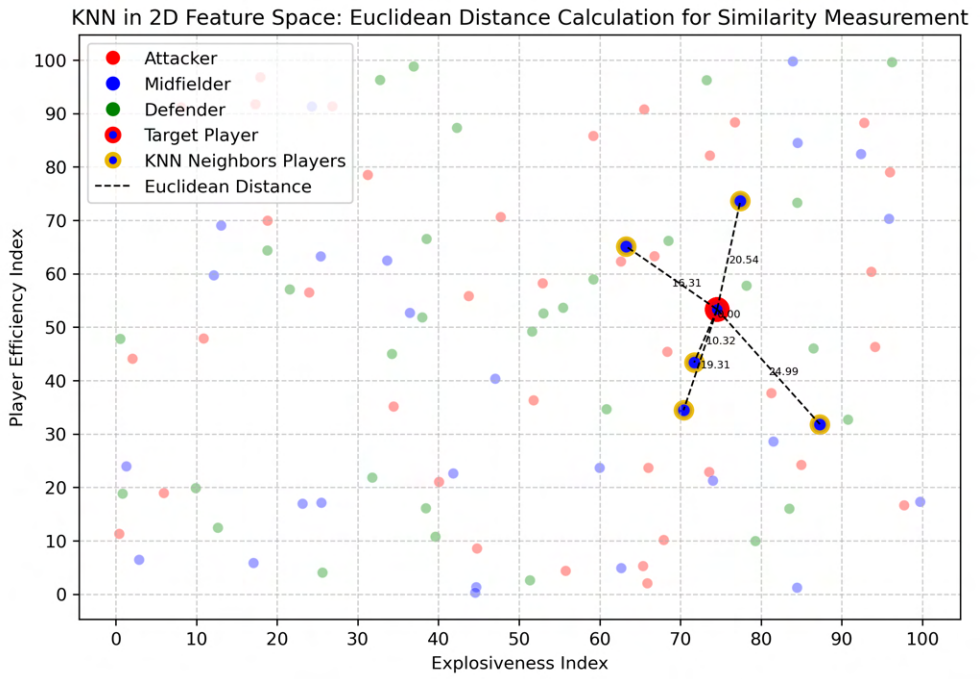


Figure 70 - KNN in 2D Feature Space: Euclidean Distance Calculation for Similarity Measurement

As previously mentioned, the model does not operate in just two dimensions but in a high-dimensional feature space, depending on the number of indices selected by the user. While it is relatively simple to interpret distances between points in two dimensions visually, the complexity increases significantly as the number of dimensions grows. In a two-dimensional space, it is even possible to visually identify the closest points simply by observing their relative positions on a graph, as demonstrated in the previous visualizations. However, as the number of dimensions increases, direct human analysis becomes impractical. This is where machine learning plays a necessary role, as it enables the computation of distances in a multi-dimensional space that would otherwise be impossible to analyze manually.

Since human perception is limited to three dimensions, higher-dimensional spaces cannot be visualized directly. Therefore, the visualization library used (*Matplotlib*) does not support direct representation of spaces beyond three dimensions. However, for illustration purposes, we can extend the previous two-dimensional representations to a three-dimensional space by adding an additional feature as example (Shooting Efficiency Index) to better approximate the actual multi-dimensional nature of the model.

I. Mapping Players in a 3D Feature Space

The first 3D visualization (*Figure 71*) shows the general distribution of players, now with an added third axis representing the shooting efficiency. The color scheme remains the same: red for attackers, blue for midfielders, and green for defenders. Each player is positioned according to their values in these three indices, demonstrating how they are distributed based on their indices level in a higher-dimensional space.

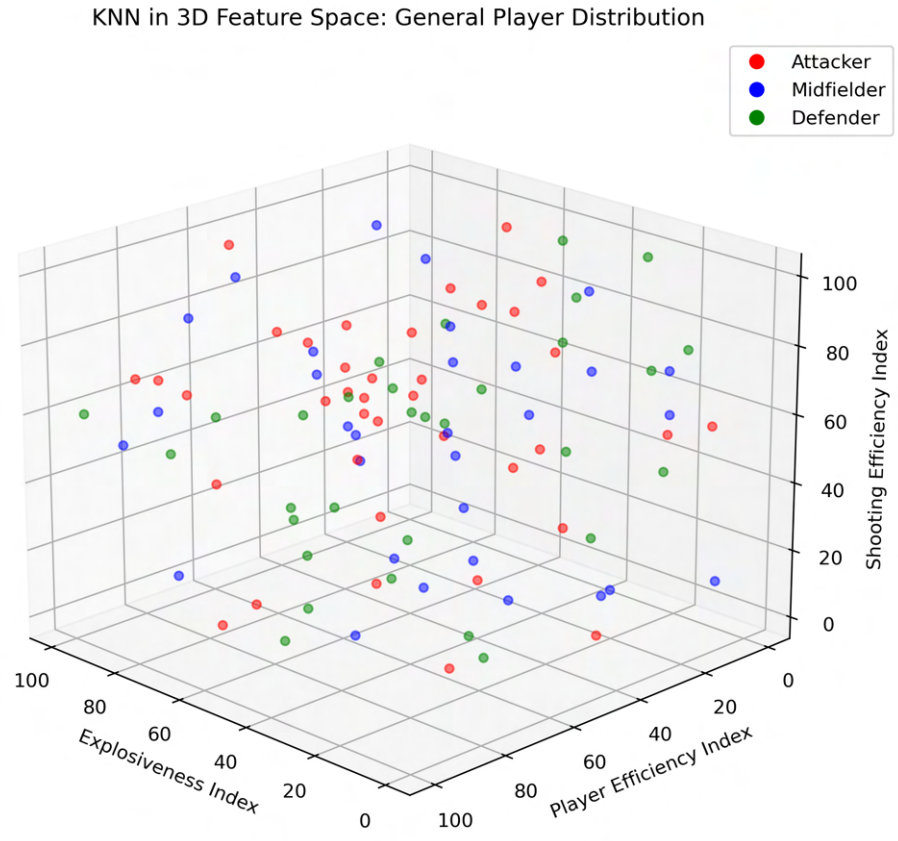


Figure 71 - KNN in 3D Feature Space: General Player Distribution

II. Selecting a Target Player in a 3D Feature Space

In the second 3D visualization (*Figure 72*), a target midfielder has been selected, highlighted with a red outline. This step remains identical to the 2D case but now occurs in a more complex feature space. The presence of a third feature allows for a more refined search, as players are now compared based on a higher range of attributes.

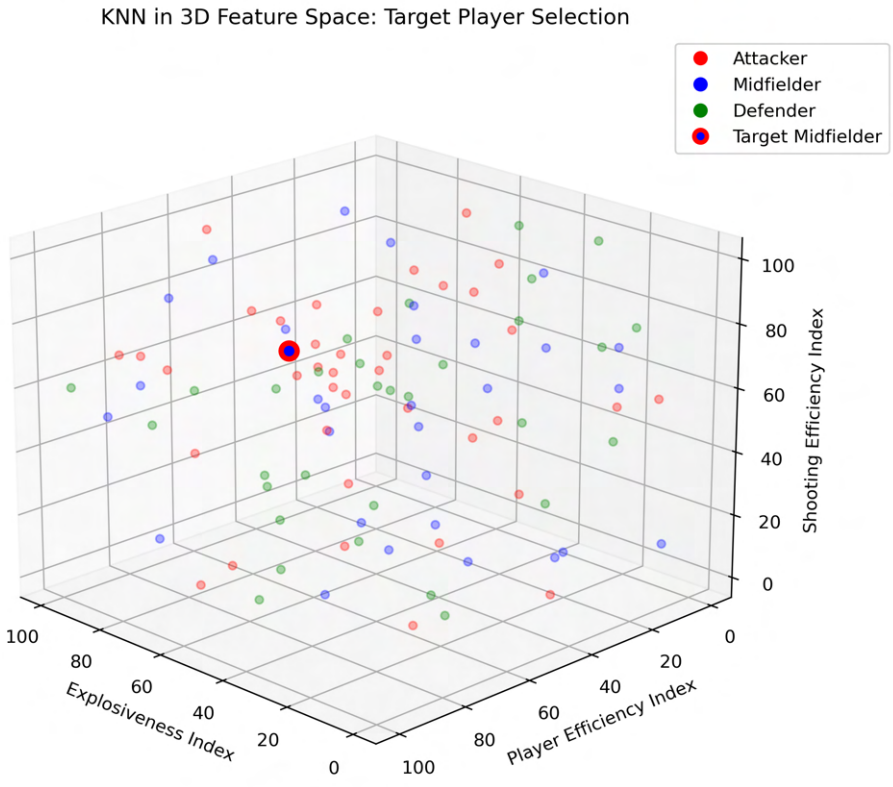


Figure 72 - KNN in 3D Feature Space: Target Player Selection

III. Neighbor Selection and Retrieving the Nearest Neighbors in a 3D Feature Space

Figure 73 illustrates the selection of the five most similar midfielders, highlighted with yellow markers. The retrieval process follows the same principle as before but now accounts again for three performance metrics simultaneously. Therefore, players closer to the target in this space are those with the most similar balance of explosiveness, overall efficiency, and shooting proficiency.

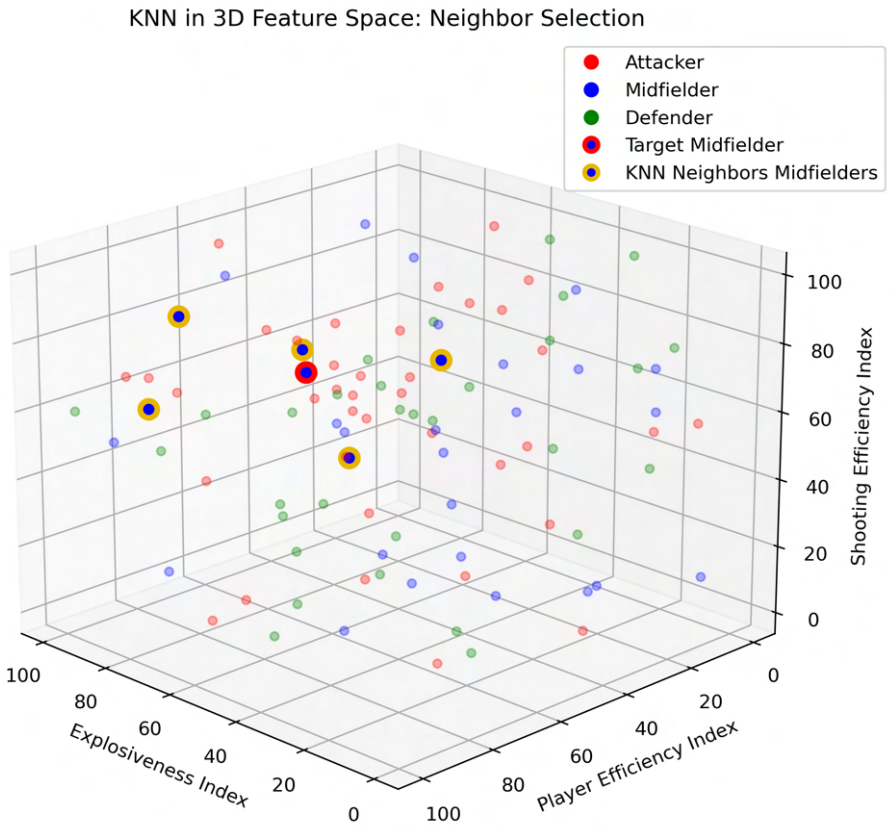


Figure 73 - KNN in 3D Feature Space: Neighbor Selection

Even at just three dimensions, visually determining the closest players becomes noticeably less intuitive compared to the two-dimensional case. While patterns can still be observed, accurately measuring distances between points is no longer straightforward. This demonstrates how increasing the number of features complicates manual interpretation, reinforcing the necessity of a machine learning model to systematically compute and rank similarities.

IV. Measuring Similarity with Euclidean Distance in a 3D Feature Space

Finally, *Figure 74* displays the Euclidean distances between the target player and its nearest neighbors in this expanded feature space. Dashed lines indicate the computed distances as before, with numerical values included to illustrate the exact similarity score. Again, the smaller the distance, the greater is the similarity between players.

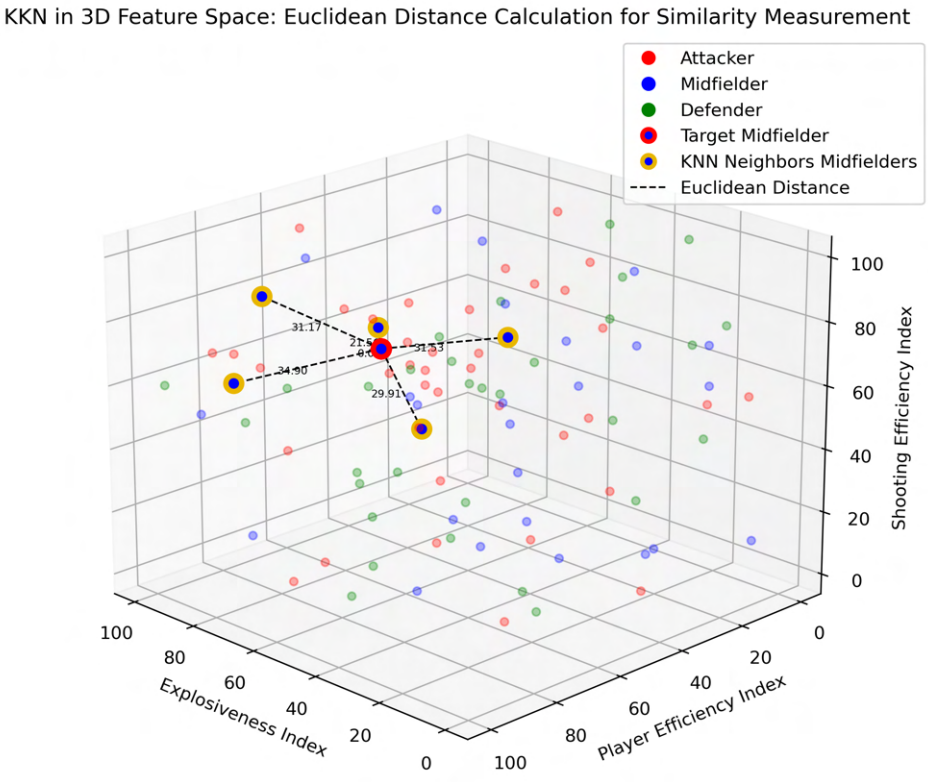


Figure 74 - KKN in 3D Feature Space: Euclidean Distance Calculation for Similarity Measurement

These 3D visualizations provide a more realistic representation of how the model operates, although in practice the actual computations extend beyond three dimensions, incorporating several performance indices. While a true multi-dimensional representation is beyond human visualization capabilities, the fundamental principles remain consistent, ensuring accurate similarity retrieval across a broad range of attributes (indices).