TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

TIỂU LUẬN CUỐI KỲ
HỌC PHẦN: KHOA HỌC DỮ LIỆU

TÊN ĐỀ TÀI

Phân cụm và phân loại thị trường cầu thủ bóng đá

Nhóm	13
Họ và tên sinh viên	
Lê Xuân Tiến Nhật	Nhóm: 21.10
Nguyễn Phan Bảo Lộc	Nhóm: 21.10
Lê Phước Duy	Nhóm: 21.10

ĐÀ NẴNG, 05/2024

TÓM TẮT

Trong tiểu luận khoa học dữ liệu này, nhóm chúng em sẽ tiến hành Phân tích thị trường cầu thủ bóng đá trên thế giới với mục tiêu tìm hiểu về các thuộc tính, chỉ số của cầu thủ và nhóm hóa các cầu thủ thành các nhóm có tính chất tương đương. Kết hợp sử dụng các phương pháp phân tích dữ liệu, kỹ thuật phân cụm và phân loại dữ liệu. Kết quả thu được là các cụm cầu thủ có tính chất riêng biệt và có thể phân loại cầu thủ vào nhóm tương ứng dựa trên các thuộc tính, chỉ số của cầu thủ.

BẢNG PHÂN CÔNG NHIỆM VỤ

Sinh viên thực hiện	Nhiệm vụ	Đánh giá
Lê Phước Duy	- Tìm hiểu DBSCAN và mô hình hóa. - Tìm hiểu KNN và mô hình hóa. - Trực quan hóa dữ liệu đơn biến.	Đã hoàn thành
Lê Xuân Tiến Nhật	- Crawl dữ liệu. - Mã hóa, chuẩn hóa dữ liệu. - Tìm hiểu Random Forest và mô hình hóa. - Trực quan hóa mối quan hệ đa biến. - Nhận xét.	Đã hoàn thành
Nguyễn Phan Bảo Lộc	- Tìm hiểu Hierachical clustering và mô hình hóa. - Tìm hiểu Logistic Regression và mô hình hóa. - Trực quan hóa mối quan hệ đa biến. - Clean data.	Đã hoàn thành

MỤC LỤC

DANH MỤC HÌNH VẼ

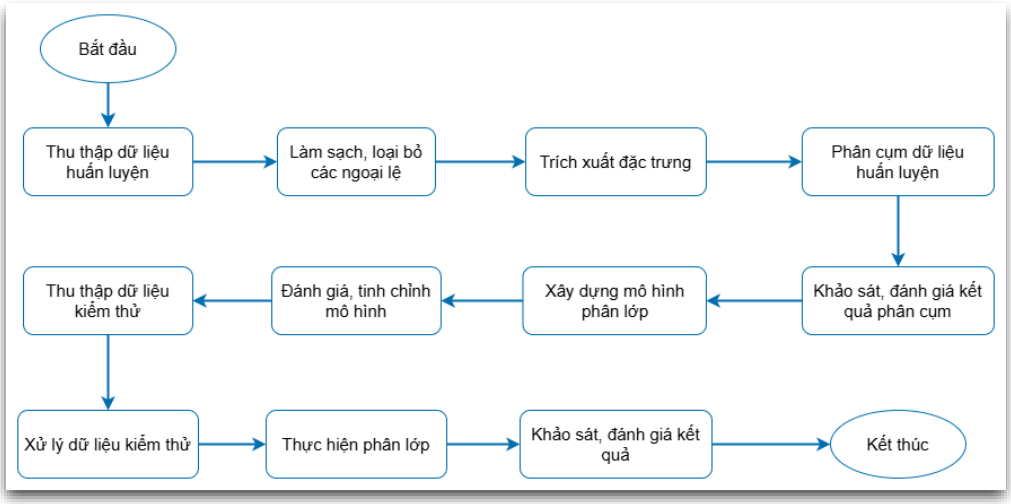
1. Giới thiệu

Bài toán: Phân cụm và phân loại thị trường cầu thủ bóng đá thế giới dựa trên thông tin và chỉ số thi đấu của cầu thủ.

Mục tiêu:

- Phân cụm thị trường cầu thủ thành các phân khúc dựa trên đặc điểm thi đấu.
- Phân loại cầu thủ vào các phân khúc thị trường đã xác định.
- Hiểu rõ hơn về thị trường cầu thủ bóng đá thế giới và đưa ra các khuyến nghị cho các đội bóng, nhà môi giới cầu thủ...

Phương pháp:



Hình 1. Sơ đồ khối giải pháp tổng quan về phân cụm và phân loại.

- Thu thập dữ liệu: Thu thập dữ liệu về các cầu thủ bóng đá thế giới từ nguồn trang uy tín SofaScore.
- Tiền xử lý dữ liệu: Xử lý thiếu dữ liệu, chuyển đổi dữ liệu và chuẩn hóa dữ liệu.
- Phân cụm: Sử dụng hai phương pháp phân cụm Hierarchical Clustering và DBSCAN để nhóm các cầu thủ có đặc điểm tương đồng vào các nhóm.
- Phân loại: Sử dụng ba phương pháp phân loại Logistic Regression, Random Forest và K-Nearest Neighbors để dự đoán phân khúc thị trường của mỗi cầu thủ.
- Đánh giá và tinh chỉnh:
    - Đánh giá hiệu suất của các mô hình phân loại bằng các độ đo như độ chính xác và F1 score.
    - Tinh chỉnh các tham số của các mô hình để cải thiện hiệu suất.
- Đánh giá và diễn giải kết quả: Phân tích kết quả phân cụm và phân loại để hiểu rõ hơn về thị trường cầu thủ bóng đá thế giới.

2. Thu thập và mô tả dữ liệu

2.1. Thu thập dữ liệu

2.1.1. Nguồn dữ liệu

Thu thập dữ liệu về các cầu thủ bóng đá thế giới từ trang https://www.sofascore.com.

2.1.2. Công cụ thu thập

Sử dụng một kết hợp giữa thư viện requests và BeautifulSoup trong Python để thu thập dữ liệu từ các trang web chứa thông tin về cầu thủ bóng đá.

2.1.3. Cách thức sử dụng công cụ

Sử dụng thư viện requests để gửi các yêu cầu HTTP đến các URL và API của các trang web cần thu thập dữ liệu. Một số thông tin có được từ API ở dạng JSON và một số thông tin sử dụng BeautifulSoup để phân tích cú pháp HTML của trang web và trích xuất thông tin cần thiết từ các thẻ HTML.

Các thông tin được lấy về sau đó được lưu trữ trong các tệp dữ liệu có định dạng phù hợp như CSV.

2.1.4. Đầu vào và đầu ra của quá trình thu thập

Đầu vào của quá trình thu thập là các đường dẫn URL và API trang web SofaScore chứa thông tin và chỉ số thi đấu liên quan về cầu thủ bóng đá.

Đầu ra của quá trình thu thập là các tập dữ liệu có cấu trúc, chứa thông tin chi tiết về các cầu thủ bóng đá, bao gồm các quốc gia, giải đấu và các câu lạc bộ bóng đá, các chỉ số và thông tin về các trận đấu mà họ tham gia.

2.1.5. Ví dụ cách thực hiện

Đầu tiên, nhóm đã viết một số hàm Python để crawl thông tin về các cầu thủ đang thi đấu trong các giải đấu hàng đầu thế giới từ trang web chính thức của SofaScore. Các hàm chức năng này tự động duyệt qua các đường dẫn URL và API của trang web SofaScore, trích xuất thông tin về các cầu thủ như tên, quốc tịch, tuổi, vị trí thi đấu, điểm số, số trận đấu tham gia, và các chỉ số kỹ thuật khác như chỉ số tấn công, phòng thủ, kỹ thuật... và các thông tin về các quốc gia, giải đấu và các câu lạc bộ mà các cầu thủ tham gia. Sau đó, lưu trữ thông tin này vào các tập dữ liệu CSV để sử dụng cho quá trình phân cụm và phân loại sau này.

Thời gian thu thập: Tập dữ liệu được thu thập trong giai đoạn từ tháng 3/2024 đến tháng 5/2024. Tập huấn luyện đã được thu thập và xác nhận có hơn 10000 mẫu, và tập kiểm thử có

kích thước tương ứng với 10% đến 30% tập huấn luyện đã được thu thập sau ngày 8/5/2024, như yêu cầu của đề tài.

```
2024-05-26 16:10:32.216698
Start scraping data...
Current date and time :  2024-05-26 16:59:23
Start processing ...
Processing 0 : 329601 ...
Processing 1 : 1482434 ...
Processing 2 : 927308 ...
Processing 3 : 1482381 ...
```

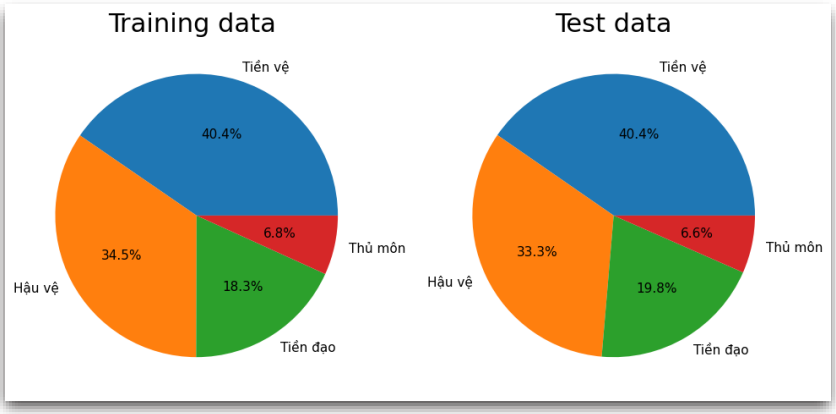
Hình 2. Minh chứng thời gian thu thập tập huấn luyện.

2.2. Mô tả và trực quan hóa dữ liệu

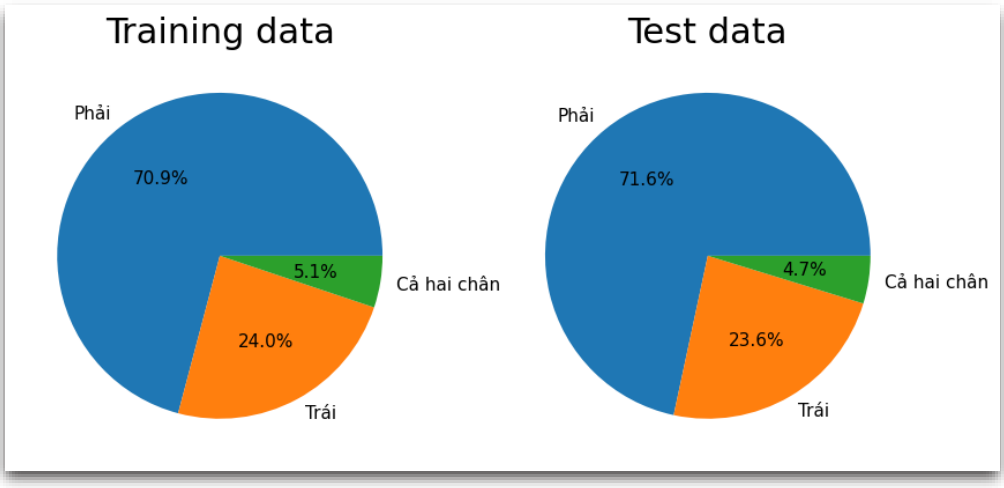
Thống kê tổng quan về tập dữ liệu:

- Gồm 3 tập dữ liệu chính: Cầu thủ bóng đá (21 đặc trưng), Câu lạc bộ (4 đặc trưng), Quốc gia (5 đặc trưng).
- Dữ liệu huấn luyện thô:
  - Quốc gia: 209 mẫu, 5 đặc trưng.
  - Câu lạc bộ: 2089 mẫu, 4 đặc trưng.
  - Cầu thủ: 11651 mẫu, 21 đặc trưng.
- Dữ liệu kiểm thử thô:
  - Quốc gia: 210 mẫu, 5 đặc trưng.
  - Câu lạc bộ: 1012 mẫu, 4 đặc trưng.
  - Cầu thủ: 2979 mẫu, 21 đặc trưng.

Trực quan hóa dữ liệu:



Hình 3. Biểu đồ thống kê vị trí cầu thủ trong tập huấn luyện và kiểm thử.

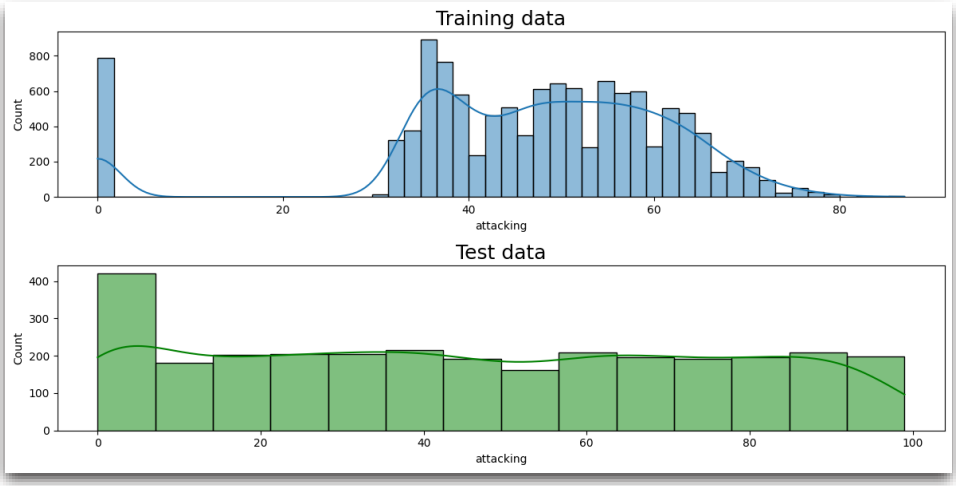


Hình 4. Biểu đồ thống kê chân sở trường trong tập huấn luyện và kiểm thử.

Hình 5. Biểu đồ phân bố chiều cao cầu thủ trong tập huấn luyện và kiểm thử.

Hình 6. Biểu đồ phân bố số trận thi đấu trong tập huấn luyện và kiểm thử.

Hình 7. Biểu đồ phân bố chỉ số tactical trong tập huấn luyện và kiểm thử.



Hình 8. Biểu đồ phân bố chỉ số attacking trong tập huấn luyện và kiểm thử.

Tập dữ liệu kiểm thử nhìn chung vẫn giữ tính chất phân bố của tập huấn luyện ở các biến thông tin cơ bản. Ở các biến đặc trưng về chỉ số thi đấu ở tập kiểm thử do được thu thập ngẫu nhiên nên có phân bố trải đề, khác với tập huấn luyện có biến phân phối chuẩn và cả phân phối không chuẩn.

3. Trích xuất đặc trưng

3.1. Lựa chọn đặc trưng

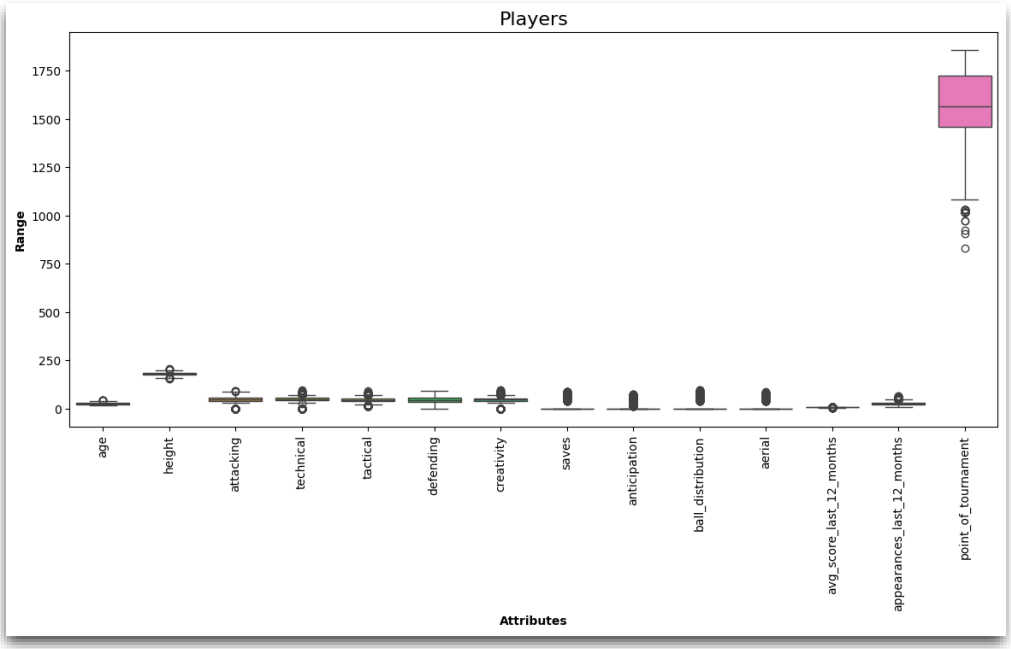
Lựa chọn các đặc trưng có giá trị thông tin cao, có sự dao động giá trị như thông tin cơ bản, chỉ số thi đấu, điểm số, số lần ra sân... và loại bỏ các đặc trưng không cần thiết như tên cầu thủ, tên câu lạc bộ, số áo...

Các đặc trưng được lựa chọn bao gồm: Giá trị thị trường (Market_value), Tuổi (Age), Chiều cao (Height), Vị trí thi đấu (Position), Chân sở trường (Preferred Foot), Tấn công (Attacking), Kỹ thuật (Technical), Chiến thuật (Tactical), Phòng thủ (Defending), Sáng tạo (Creativity), Cứu thua (Saves), Dự đoán bóng (Anticipation), Phân phối bóng (Ball

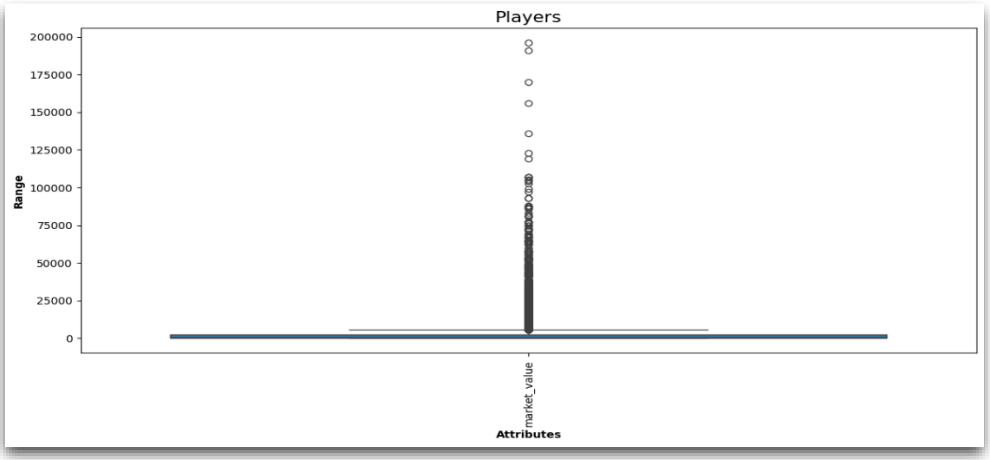
distribution), Kiểm soát trên không (Aerial), Điểm trung bình (Avg score last 12 months), Số trận đấu tham gia (Appearances last 12 months), Hệ số giải (Point of tournament).

3.2. Làm sạch, loại bỏ ngoại lệ

Loại bỏ các lỗi định dạng, xoá các ký tự không liên quan với giá trị của các biến đặc trưng. Kết quả thu được:



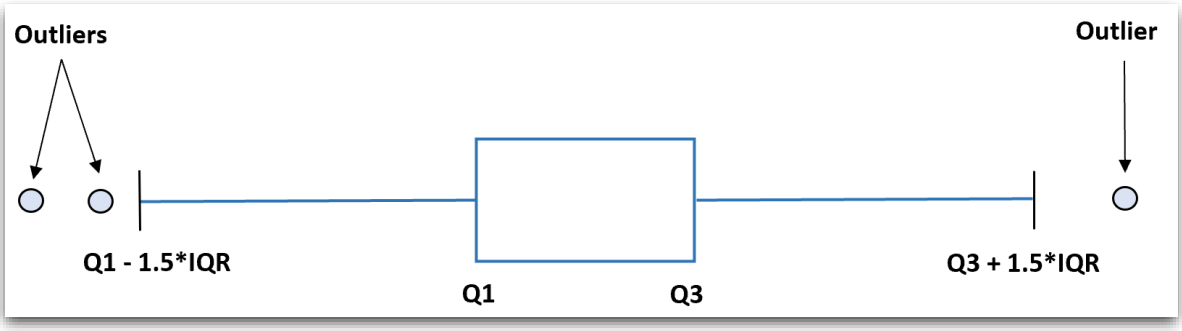
Hình 9. Kết quả các biến đặc trưng sau khi làm sạch và loại bỏ ngoại lệ.



Hình 10. Kết quả các biến giá trị thị trường sau khi làm sạch và loại bỏ ngoại lệ.

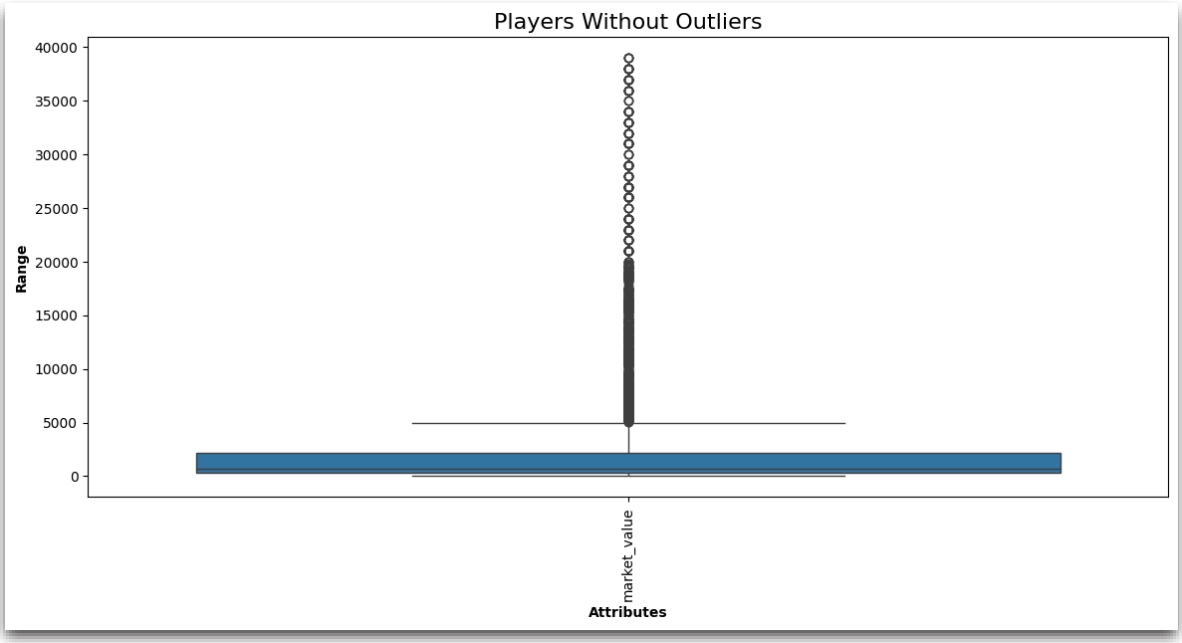
Tiếp theo, loại bỏ hoặc thay thế bằng giá trị trung bình cho các mẫu dữ liệu có giá trị thiếu hoặc trống cho các đặc trưng quan trọng. Ở đây, Hệ số giải đấu (point_of_tournament) sau khi được ghép vào là có giá trị trống.

Sau đó, loại bỏ các giá trị ngoại lệ sử dụng phương pháp IQR (Interquartile Range) để phát hiện và loại bỏ các giá trị ngoại lệ.



Hình 11. Mô tả phương pháp IQR.

Với tập dữ liệu cầu thủ, giá trị ngoại lệ xuất hiện nhiều ở Market value. Sau khi áp dụng phương pháp IQR, ta được kết quả:



Hình 12. Kết quả các biến giá trị thị trường sau khi sử dụng IQR.

3.3. Mã hóa đặc trưng danh mục

Với dữ liệu "Position" và "Preferred_foot" là kiểu dữ liệu danh mục danh nghĩa, cần chuyển đổi thành các biến nhị phân và không tạo ra mối quan hệ thứ bậc giữa các giá trị

Sử dụng One - Hot Encoding để mã hóa các biến danh mục 'Position' và 'Preferred Foot'.

Cấu trúc tập dữ liệu sau khi mã hóa đặc trưng danh mục:

```
 20  pos_D                      11651 non-null  bool
 21  pos_F                      11651 non-null  bool
 22  pos_G                      11651 non-null  bool
 23  pos_M                      11651 non-null  bool
 24  foot_Both                  11651 non-null  bool
 25  foot_Left                  11651 non-null  bool
 26  foot_Right                 11651 non-null  bool
dtypes: bool(7), float64(2), int64(13), object(5)
```

Hình 13. Các biến danh mục.

pos_D	pos_F	pos_G	pos_M	foot_Both	foot_Left	foot_Right
False	True	False	False	False	True	False
False	True	False	False	False	False	True
False	True	False	False	False	False	True

Hình 14. Các biến danh mục sau khi áp dụng OneHotEncoding.

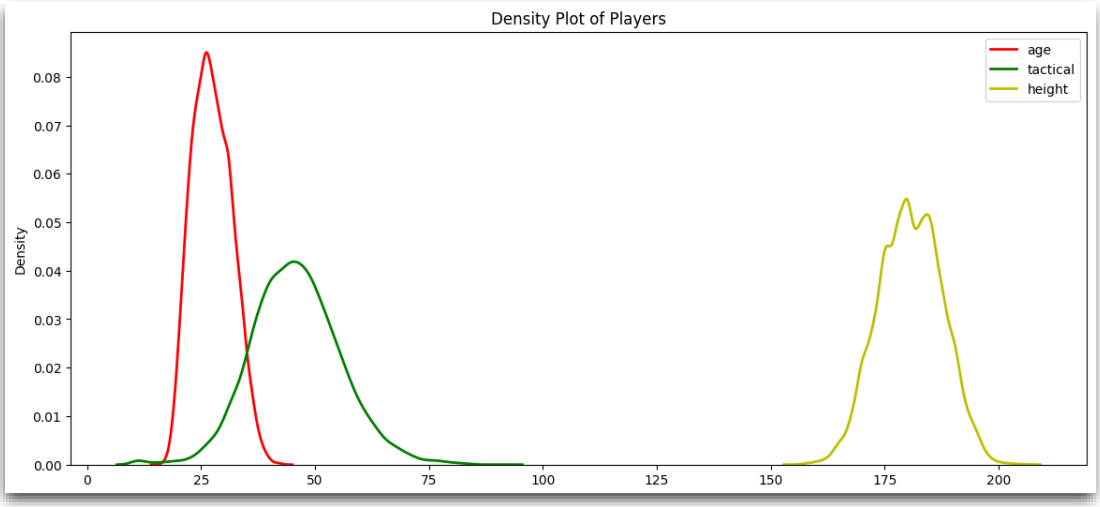
3.4. Biến đổi đặc trưng

3.4.1. Chuẩn hóa (standardization)

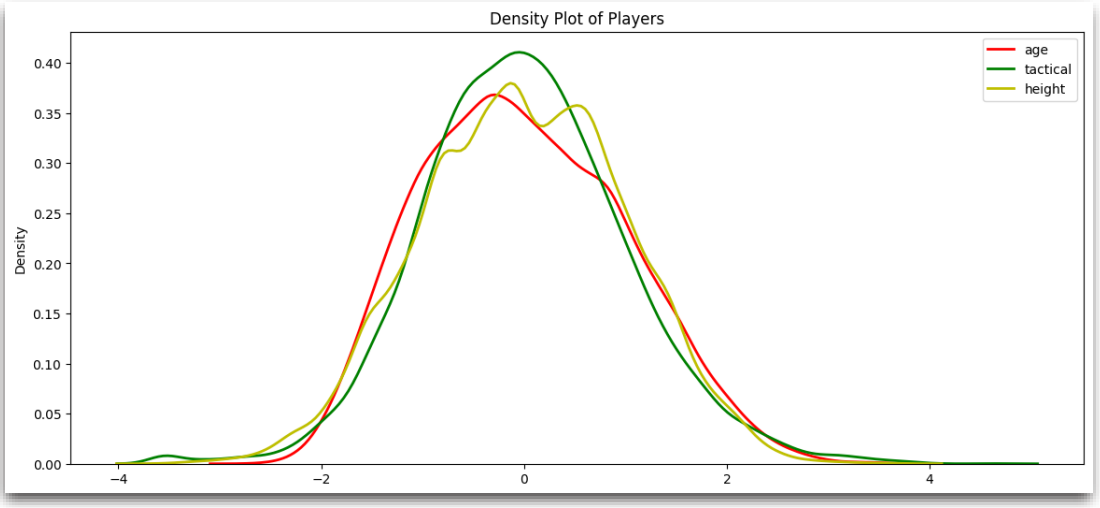
Kĩ thuật chuẩn hoá được áp dụng đối với những biến có phân phối chuẩn. Biến được biến đổi theo kì vọng và độ lệch chuẩn như sau: $x' = \frac{x-\bar{x}}{\sigma(x)}$

Các biến sau khi được chuẩn hoá sẽ có cùng một dạng phân phối chuẩn hoá với trung bình bằng 0 và phương sai bằng 1. Nhờ đó quá trình huấn luyện sẽ trở nên ổn định và hội tụ tới nghiệm tối ưu nhanh hơn.

Kết quả trước và sau khi áp dụng chuẩn hóa lên các biến có phân phối chuẩn:



Hình 15. Biểu đồ mật độ các biến age, tactical, height



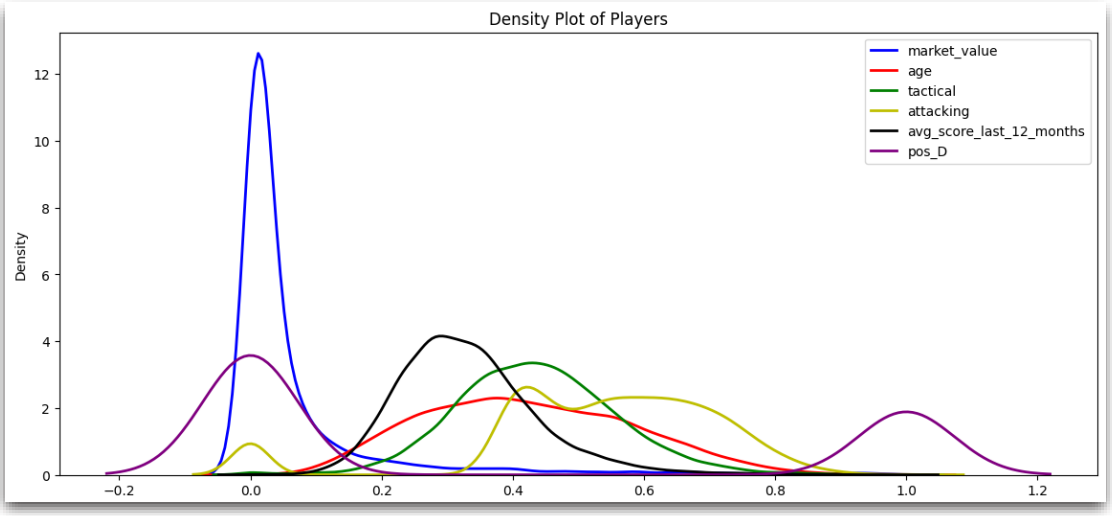
Hình 16. Biểu đồ mật độ các biến age, tactical, height sau khi chuẩn hóa.

3.4.2. Kỹ thuật scaling

Kĩ thuật scaling thường áp dụng trên những biến không tuân theo phân phối chuẩn. Thông qua scaling, toàn bộ giá trị của biến sẽ được đưa về một miền giá trị bị giới hạn trong khoảng [0,1]. Đối với dữ liệu cầu thủ, các biến không có phân phối chuẩn đã được xử lý ngoại lệ từ trước, do đó chúng ta sử dụng kỹ thuật Minmax Scaling.

Biến được đưa về các range [0,1] theo công thức: $x' = \frac{x - min(x)}{max(x) - min(x)}$

Sử dụng StandardScaler cho các biến có phân phối chuẩn và MinMaxScaler cho các biến không phân phối chuẩn. Cuối cùng, sử dụng MinMaxScaler một lần nữa cho toàn bộ dữ liệu để đồng nhất trước khi phân cụm.



Hình 17. Biểu đồ mật độ các biến market_value, age, tactical, attacking, score, pos_D.

	market_value	age	height	attacking	technical	tactical	defending	creativity	saves	anticipation	...	avg_score_last_12_months	appearances_last_12_months	point_of_tournament	pos_D	pos_F	pos_G	pos_M	foot_Both	foot_Left	foot_Right
0	0.184531	0.222222	0.66	0.781609	0.617978	0.5000	0.358696	0.526882	0.0	0.0	...	0.438424	0.410714	1.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0
1	0.061961	0.333333	0.46	0.712644	0.539326	0.5375	0.347826	0.505376	0.0	0.0	...	0.300493	0.482143	1.0	0.0	1.0	0.0	0.0	0.0	0.0	1.0
2	0.029442	0.222222	0.56	0.735632	0.573034	0.4250	0.315217	0.516129	0.0	0.0	...	0.270936	0.428571	1.0	0.0	1.0	0.0	0.0	0.0	0.0	1.0

3 rows × 22 columns

Hình 18. Dữ liệu sau khi scaling.

4. Mô hình hóa dữ liệu

4.1. Bài toán phân cụm

4.1.1. Hierachical Clustering

Hierarchical Clustering là một phương pháp phân cụm trong lĩnh vực học máy và khai phá dữ liệu, được sử dụng để tổ chức các điểm dữ liệu thành các cụm phân biệt dựa trên mức độ tương đồng hoặc khoảng cách giữa chúng. Phương pháp này xây dựng cấu trúc phân cấp của dữ liệu, trong đó mỗi điểm dữ liệu được coi là một cụm và các cụm này được liên kết với nhau theo một cách có tổ chức.

Tham số chính được sử dụng:

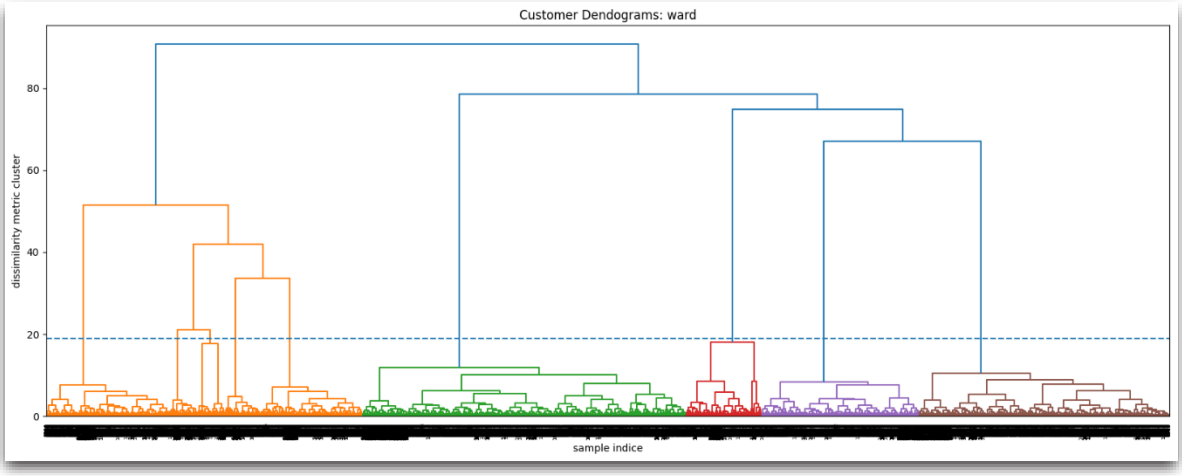
- Linkage method: Tham số sử dụng để liên kết giữa các điểm dữ liệu. Có 4 cách liên kết dữ liệu là “Complete linkage”, “Single linkage”, “Average linkage” và “Ward linkage”.
- N_cluster: Số lượng cụm cần phân chia.

Các bước thực hiện thuật toán:

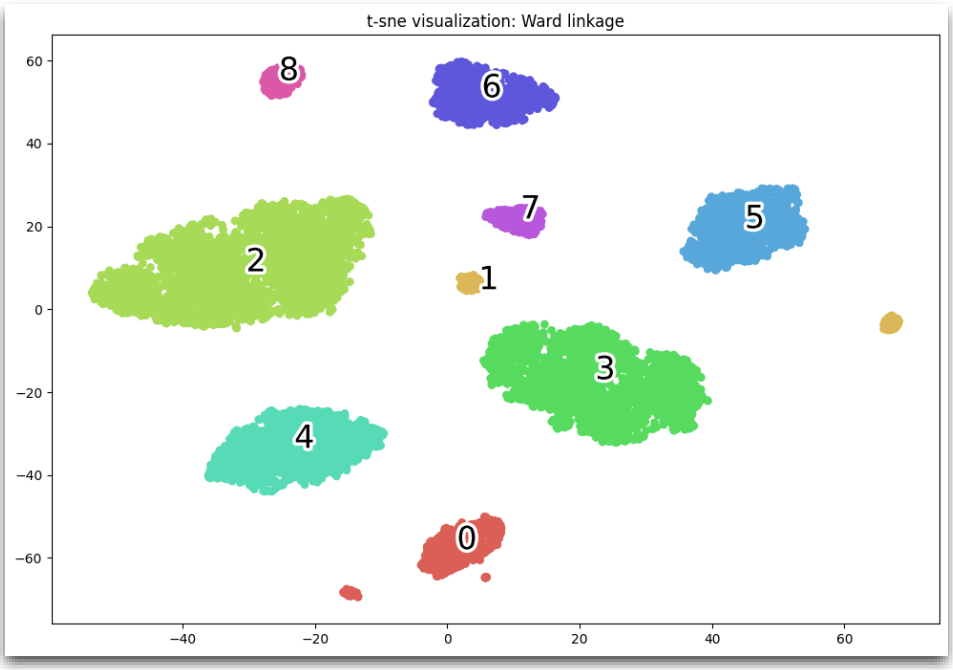
- Sử dụng chiến lược hợp nhất(agglomerative) kết hợp biểu đồ dendrogram.
- Bắt đầu, xem mỗi điểm dữ liệu là một cụm. Gộp dần các cụm theo chiều từ dưới lên theo phương pháp đo liên kết đã chọn.

- Vẽ biểu đồ dendrogram và khảo sát chọn số lượng cụm bằng cách vẽ đường thẳng cắt ngang qua biểu đồ dendrogram.

Kết quả: Sau khi khảo sát qua bằng biểu đồ dendrogram thì xác định được 9 cụm.



Hình 19. Biểu đồ dendrogram.



Hình 20. Biểu đồ kết quả phân cụm bằng Hierarchical - Ward linkage.

Các cụm thu được cùng với các tính chất:

- Cụm 0: là thủ môn, cao, lớn tuổi, tactical cao, chân phải|số ít chân trái hoặc cả 2 chân, giá trị trung bình thấp.
- Cụm 1: là tiền đạo/hậu vệ, attacking| technical| creativity cao, defending| tactical thấp, thuận 2 chân, lớn tuổi, giá trị trung bình thấp, giải đấu nhỏ.

- Cụm 2: là tiền vệ, hay ra sân, giải đấu lớn, defending| tactical khá cao, chân phải, giá trị trung bình cao.
- Cụm 3: là hậu vệ, khá lớn tuổi, hay ra sân, defending| tactical trung bình cao, giải đấu lớn,chân phải, giá trị trung bình cao.
- Cụm 4: là tiền đạo, attacking| technical| creativity trung bình cao, chân phải, giá trị cao.
- Cụm 5: là hậu vệ, trẻ, hay ra sân, attacking| technical| creativity trung bình cao, giải đấu lớn, chân trái, giá trị trunh bình cao.
- Cụm 6: là tiền vệ, hay ra sân, giải đấu lớn, attacking| technical| creativity trung bình, chân trái, giá trị trung bình cao.
- Cụm 7: là tiền đạo, attacking| technical| creativity trung bình cao, chân trái, giá trị trung bình.
- Cụm 8: là tiền vệ, ít ra sân, attacking| technical| creativity trung bình, giải đấu nhỏ, thuận 2 chân, giá trị thấp.

4.1.2. DBSCAN

DBSCAN là một thuật toán cơ sở để phân nhóm dựa trên mật độ. Nó có thể phát hiện ra các cụm có hình dạng và kích thước khác nhau từ một lượng lớn dữ liệu chứa nhiễu.

Các tham số chính trong thuật toán DBSCAN:

- MinPts: là một ngưỡng số điểm dữ liệu được nhóm lại với nhau nhằm xác định vùng lân cận epsilon.
- Epsilon: một giá trị khoảng cách được sử dụng để xác định vùng lân cận epsilon của bất kỳ điểm dữ liệu nào.

Các loại điểm trong DBSCAN:

- Core point: là điểm có ít nhất minPts điểm trong vùng lân cận epsilon.
- Border point: là điểm có ít nhất 1 core point ở vùng lân cận epsilon nhưng mật độ không đủ minPts điểm.
- Noise point: là điểm không phải là điểm lõi hay điểm biên.

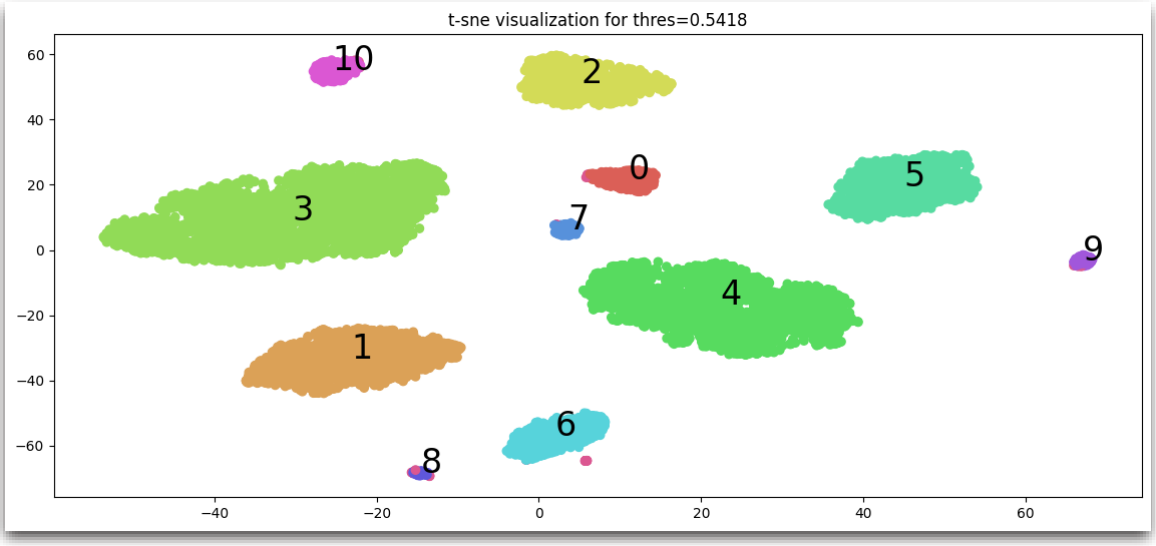
Các bước thực hiện thuật toán:

- Lựa chọn 1 điểm bất kỳ. Sau đó xác định các core point và border point.
- Chọn ngẫu nhiên core point (không nằm trong cụm nào cả) gán vào 1 cụm.
- Tìm các điểm lân cận của core point.

- Nếu điểm đó là core point thì thêm vào cụm, tiếp tục mở rộng cụm từ core point này.
- Nếu điểm đó là border point thì thêm vào cụm nhưng không mở rộng cụm từ điểm này.

- Lặp lại đệ qui toàn bộ quá trình để xác định cụm mới.

Với giá trị của epsilon là 0.5418, thuật toán DBSCAN chia dữ liệu thành 11 cụm:



Hình 21. Biểu đồ kết quả phân cụm bằng DBSCAN (eps = 0.5418).

Các cụm thu được cùng với các tính chất:

- Cụm 0: là tiền đạo, thuận chân trái, giá trị trung bình cao, đang thi đấu ở giải đấu lớn.
- Cụm 1: là tiền đạo, thuận chân phải, giá trị trung bình thấp.
- Cụm 2: là tiền vệ, thuận chân trái, có các chỉ số attacking| technical| creativity trung bình cao.
- Cụm 3: là tiền vệ, thuận chân phải, có chỉ số defending trung bình cao.
- Cụm 4: là hậu vệ, thuận chân phải, có các chỉ số tactical|defending trung bình cao.
- Cụm 5: là hậu vệ, thuận chân trái, có chỉ số attacking trung bình cao.
- Cụm 6: là thủ môn, thuận chân phải, có các chỉ số tactical| ball_distribution| saves| aerial trung bình cao, đang thi đấu ở giải đấu lớn.
- Cụm 7: là tiền đạo, thuận 2 chân, giá trị trung bình thấp, đang thi đấu ở giải đấu nhỏ.
- Cụm 8: là thủ môn, thuận chân trái, có các chỉ số tactical| ball_distribution| saves| aerial trung bình thấp, đang thi đấu ở giải đấu nhỏ.

- Cụm 9: là hậu vệ, thuận 2 chân, đang thi đấu ở giải đấu nhỏ.
- Cụm 10: là tiền vệ, thuận 2 chân, có các chỉ số attacking| technical| creativity trung bình cao, đang thi đấu ở giải đấu nhỏ.

4.2. Bài toán phân loại

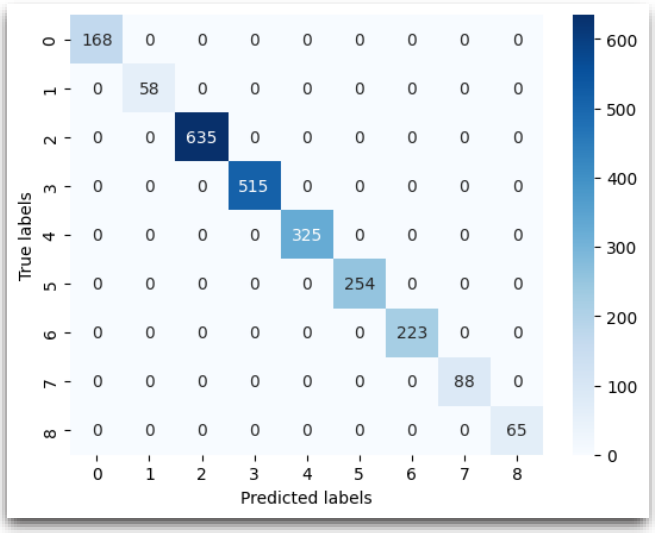
4.2.1. Mô hình phân loại Logistic Regression

Mô hình Logistic Regression là một mô hình thống kê được sử dụng trong bài toán phân loại, nơi mục tiêu là dự đoán lớp(class) của dữ liệu dựa trên một tập hợp các biến đầu vào (features).

Bộ tham số chính của mô hình:

- Penalty: Hảm sử dụng làm thành phần điều chuẩn.
- C: Hệ số nhân thành phần điều chuẩn.
- Solver: Chọn thuật toán tối ưu hóa sử dụng huấn luyện.
- Max_iter: Số lần lặp tối đa để huấn luyện mô hình.

Đánh giá kết quả trên tập huấn luyện, kiểm thử:



Hình 22. Ma trận nhầm lẫn khi phân loại bằng Logistic Regression

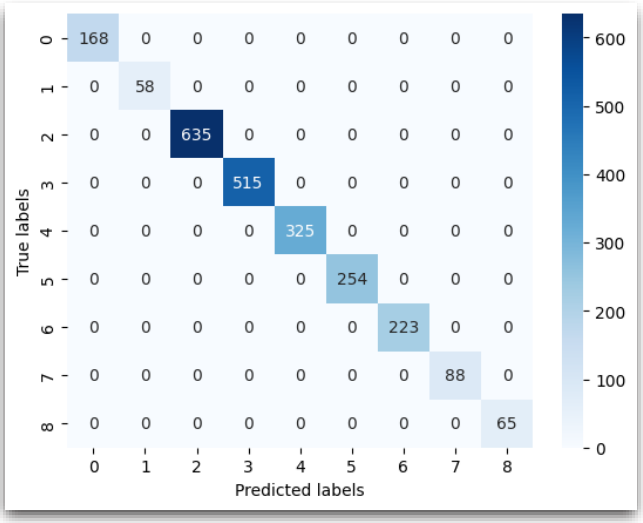
4.2.2. Mô hình phân loại Random Forest

Random Forest là mô hình thuật toán cải tiến từ Decision Tree. Random Forest xây dựng nhiều cây quyết định bằng thuật toán Decision Tree với mỗi cây quyết định là khác nhau. Sau đó tổng hợp lại kết quả dự đoán từ các cây quyết định.

Bộ tham số chính của mô hình:

- n_estimators: số lượng cây quyết định sử dụng.
- max_features: số lượng feature được chọn ngẫu nhiên tại mỗi nút chia. Việc chọn số lượng feature thích hợp có thể tăng tính đa dạng, giảm overfitting.
- max_depth: độ sâu tối đa của mỗi cây quyết định. Giới hạn độ sâu giúp tránh overfitting, nhưng cũng có thể làm mất đi sự linh hoạt của mô hình.
- min_samples_split: số lượng mẫu tối thiểu cần để chia một nút. Giúp kiểm soát độ phức tạp của mô hình.
- min_samples_leaf: số lượng mẫu tối thiểu để tạo ra một lá (leaf) mới. Giúp kiểm soát kích thước của cây.

Đánh giá kết quả trên tập huấn luyện, kiểm thử:



Hình 23. Ma trận nhầm lẫn khi phân loại bằng Random Forest.

4.2.3. Mô hình phân loại K-Nearst Neighbors (KNN)

K-Nearst Neighbors (KNN) là phương pháp phân lớp dựa vào khoảng cách gần nhất. Nhãn của một điểm dữ liệu mới được suy ra từ K điểm dữ liệu gần nhất trong tập training set.

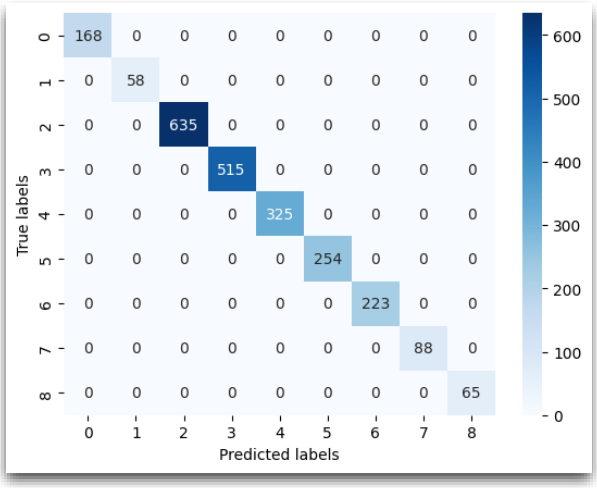
Các công thức tính khoảng cách:

- Euclidean: $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$
- Manhattan: $\sum_{i=1}^{k}|x_i - y_i|$
- Minkowski: $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

Các tham số chính trong mô hình phân loại KNN:

- N_neighbors: xác định số lượng điểm láng giềng gần nhất cần xem xét khi đưa ra dự đoán.
- Weights: xác định cách trọng số của các điểm láng giềng được tính toán khi đưa ra dự đoán. Cụ thể:
  - Weights = uniform (mặc định), tất cả các láng giềng được xem xét đều có trọng số như nhau.
  - Weights = distance, các láng giềng gần hơn có trọng số cao hơn, có nghĩa là các láng giềng gần điểm dự đoán hơn sẽ có ảnh hưởng lớn hơn đến dự đoán cuối cùng.

Đánh giá kết quả trên tập huấn luyện, kiểm thử:



Hình 24. Ma trận nhầm lẫn khi phân loại bằng KNN.

4.2.3. Tham số huấn luyện

Cv: Số lượng fold trong cross-validation. Chia dữ liệu thành k fold và sử dụng mỗi fold một lần để kiểm tra.

Verbose: điều này quy định mức độ thông tin xuất ra trong quá trình huấn luyện, nghĩa là xuất ra một lượng lớn thông tin.

N_jobs: số lượng công việc thực hiện song song trong quá trình tìm kiếm siêu tham số.

Param_grid: Từ điển chứa các tham số và thử nghiệm. Mục đích tìm ra bộ tham số tối ưu nhất.

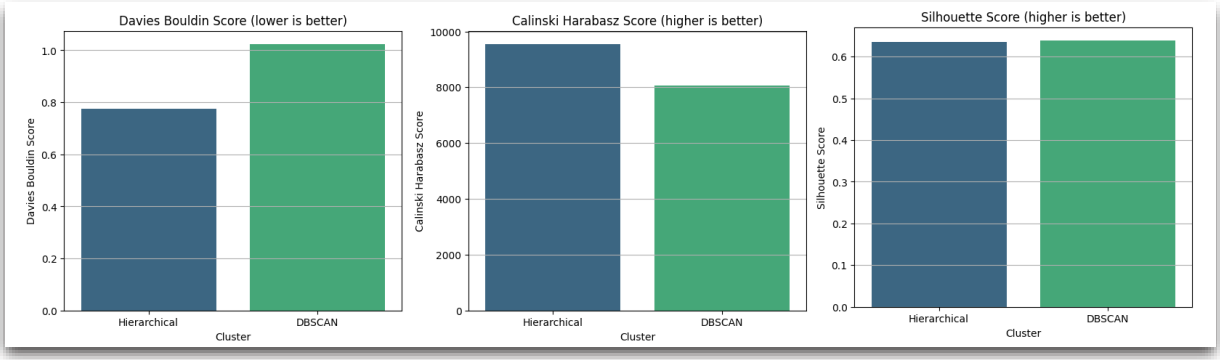
4.3. Đánh giá các mô hình

4.3.1. Các mô hình phân cụm

Davies-Bouldin Score đo lường độ tương đồng giữa các cụm. Điểm số này dựa trên tỷ lệ của khoảng cách nội cụm (intra-cluster distance) và khoảng cách liên cụm (inter-cluster distance).

Calinski-Harabasz Score đo lường tỷ lệ giữa tổng độ phân tán giữa các cụm (between-cluster dispersion) và tổng độ phân tán trong các cụm (within-cluster dispersion).

Silhouette Score đánh giá mức độ tương đồng của một điểm dữ liệu với cụm của nó (cohesion) so với các cụm khác (separation).



Hình 25. Biểu đồ đánh giá kết quả phân cụm giữa DBSCAN và Hierarchical.

Nhìn chung, Hierarchical Clustering cho kết quả phân cụm rõ ràng hơn DBSCAN.

Cả 2 phương pháp đều hỗ trợ phân cụm dữ liệu mà không cần xác định trước số cụm. **Hierarchical Clustering** mất thời gian tính toán trên tập dữ liệu lớn nhưng cho ra kết quả phân cụm thể hiển tốt mỗi quan hệ giữa các cụm. Trong khi **DBSCAN** có thể xử lý các điểm nhiễu hiệu quả hơn nhưng khó khăn trong xử lý dữ liệu có nhiều điểm lân cận dẫn đến hiệu suất phân cụm không được tốt với tập dữ liệu có các biến đặc trưng trải đều, ít tách biệt.

4.3.2. Các mô hình phân loại

Model	F1 Score	Accuracy	Thời gian thực thi (2331 mẫu)
Logistic Regression	100.0%	100.0%	1.5 ms
Random Forest	100.0%	100.0%	12.9 ms
K-Nearest Neighbors	100.0%	100.0%	116.8 ms

Bảng thống kê kết quả thực hiện các mô hình phân loại.

Cả 3 mô hình phân loại đều cho kết quả dự đoán cao (100%) với thời gian thực thi có sự chênh lệch khá lớn giữa:

- Logistic Regression có thời gian thực thi nhanh nhất (1.5ms).
- K-Nearest Neighbors có thời gian thực thi chậm nhất (116.8ms).

So sánh kết quả phân loại trên tập kiểm thử với 3 mô hình đều cho cùng kết quả phân loại.

Độ chính xác trên tập xác thực cao (100%) là do ở quá trình phân cụm dữ liệu cho tập huấn luyện và xác thực. Do các biến đặc trưng có giá trị phân bố trải đều, ít tách biệt dẫn đến ít sự tác động đến kết quả phân cụm, làm cho sự ảnh hưởng của các biến đặc trưng danh mục là rất lớn (vị trí đá và chân sở trường). Do đó, dựa vào giá trị của các biến đặc trưng danh mục sẽ nhanh chóng được phân loại vào kết quả cụm chính xác.

5. Kết luận

Bài toán 1 - Phân cụm thị trường cầu thủ bóng đá:

Với tập dữ liệu cầu thủ bóng đá với các biến đặc trưng phân bố giá trị trải đều và ít tách biệt, lựa chọn mô hình phân cụm Hierarchical Clustering cho hiệu suất tốt hơn.

Tập dữ liệu cầu thủ được phân thành 9 cụm với các tính chất riêng biệt.

Bài toán 2 - Phân loại cầu thủ bóng đá

Cả 3 mô hình Logistic Regression, Random Forest, KNN cho độ chính xác trên tập xác thực và kiểm thử.

Nguyên nhân: Do các biến đặc trưng có giá trị phân bố trải đều, ít tách biệt dẫn đến ít sự tác động đến kết quả phân cụm, làm cho sự ảnh hưởng của các biến đặc trưng danh mục là rất lớn. Do đó, dựa vào giá trị của các biến đặc trưng danh mục sẽ nhanh chóng được phân loại vào kết quả cụm chính xác.

Hướng phát triển:

Thu thập thêm dữ liệu cho tập huấn luyện

Điều chỉnh phân bộ dữ liệu cho 2 tập huấn luyện và kiểm thử

Hiểu chỉnh các tham số mô hình

6. Tài liệu tham khảo

[1] Phạm Huy Thành, Các độ đo trong phân cụm và áp dụng vào phát hiện mô hình tổ chức trong khai phá quá trình. *K55_Pham_Huy_Thanh_Thesis.pdf (vnu.edu.vn)*

[2] Feature Engineering, 11.1. Feature Engineering — Deep AI KhanhBlog (phamdinhkhanh.github.io)

[3] Hierarchical Clustering (phân cụm phân cấp), 14. Hierarchical Clustering (phân cụm phân cấp) — Deep AI KhanhBlog (phamdinhkhanh.github.io)

[4] DBSCAN, 15. DBSCAN — Deep AI KhanhBlog (phamdinhkhanh.github.io)

[5] Random Forest, Random Forest algorithm — Machine Learning cho dữ liệu dạng bảng (machinelearningcoban.com)

[6] KNN, Machine Learning cơ bản (machinelearningcoban.com)

[7] LogisticRegression, LogisticRegression — scikit-learn 1.5.0 documentation