ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC SÀI GÒN

HUỲNH BÁ BẢO

THUẬT TOÁN K-MEANS GIẢI BÀI TOÁN PHÂN CỤM DỮ LIỆU

TIỂU LUẬN MÔN SEMINAR CHUYÊN ĐỀ

Người hướng dẫn khoa học: Ths. Phan Tấn Quốc

Thành phố Hồ Chí Minh, năm 2021

LỜI MỞ ĐẦU

Chúng ta đang ở trong thời đại được gọi là thời đại thông tin. Trong thời đại thông tin này, bởi vì chúng ta tin rằng thông tin dẫn đến sức mạnh và thành công, và nhờ vào các công nghệ tinh vi như máy tính, vệ tinh,… chúng ta đã thu thập được một lượng lớn thông tin. Ban đầu, với sự ra đời của máy tính và các phương tiện lưu trữ kỹ thuật số hàng loạt, chúng tôi bắt đầu thu thập và lưu trữ tất cả các loại dữ liệu, dựa vào sức mạnh của máy tính để giúp phân loại thông tin hỗn hợp này. Thật không may, những tập hợp dữ liệu khổng lồ được lưu trữ trên các cấu trúc khác nhau rất nhanh chóng trở thành quá tải. Sự hỗn loạn ban đầu này đã dẫn đến việc tạo ra cơ sở dữ liệu có cấu trúc và hệ quản trị cơ sở dữ liệu. Hệ thống quản lý cơ sở dữ liệu hiệu quả là tài sản rất quan trọng để quản lý một kho dữ liệu lớn và đặc biệt là để truy xuất hiệu quả các thông tin cụ thể từ một tập hợp lớn bất cứ khi nào cần. Sự gia tăng của các hệ thống quản lý cơ sở dữ liệu cũng đã góp phần vào việc thu thập ồ ạt tất cả các loại thông tin gần đây.

Ngày nay, chúng ta có nhiều thông tin hơn chúng ta có thể xử lý: từ các giao dịch kinh doanh và dữ liệu khoa học, đến ảnh vệ tinh, báo cáo văn bản và tình báo quân sự. Việc truy xuất thông tin chỉ đơn giản là không còn đủ cho việc ra quyết định. Đối mặt với những tập hợp dữ liệu khổng lồ, giờ đây chúng tôi đã tạo ra những nhu cầu mới để giúp chúng tôi đưa ra các lựa chọn quản lý tốt hơn. Những nhu cầu này là tự động tóm tắt dữ liệu, trích xuất "bản chất" của thông tin được lưu trữ và khám phá các mẫu trong dữ liệu thô.

Bài tiểu luận này khảo sát về các thuật toán khai phá dữ liệu. Các thuật toán này giúp phân loại các khách hàng, sản phẩm, địa điểm,… theo một hoặc nhiều thuộc tính cho trước từ đó đưa ra các quyết định để giải quyết bài toán phân cụm dữ liệu.

Bài tiểu luận này gồm 4 chương: Chương 1 giới thiệu tổng quan về phân cụm dữ liệu. Thuật toán phân cụm phân hoạch “K-means” được trình bày ở chương 2. Tiếp đến là thuật toán phân cụm phân cấp “BIRCH” ở chương 3. Và thuật toán phân cụm theo mật độ “DBSCAN” được mô tả ở chương 4.

MỤC LỤC

DANH MỤC CÁC HÌNH

DANH MỤC CÁC BẢNG

CHƯƠNG 1: GIỚI THIỆU

1. Mục tiêu:

- Tìm hiểu các công thức để tính toán khoảng cách tới tâm cụm để đưa ra quyết định cho thuật toán K-means
- Hiểu được được cách chạy thuật toán K-means và cách ứng dụng K-means vào coding.
- Biết được cách ứng dụng thuật toán K-means vào các dữ liệu thực tế.

2. Đối tượng nghiên cứu:

- Bài toán phân cụm dữ liệu trong khai phá dữ liệu
- Công thức tính khoảng cách tới tâm cụm
- Thuật toán K-means

3. Phân cụm dữ liệu:

3.1. Định nghĩa:

Phân cụm dữ liệu là một kỹ thuật trong khai phá dữ liệu, nhằm tìm kiếm, phát hiện các cụm, các mẫu dữ liệu tiềm ẩn, quan trọng trong tập dữ liệu lớn, từ đó cung cấp thông tin tri thức hữu ích cho ra quyết định.

3.2. Bài toán phân cụm dữ liệu:[7]

Bài toán phân cụm dữ liệu thường được hiểu là bài toán học không có giám sát và được phát biểu như sau:

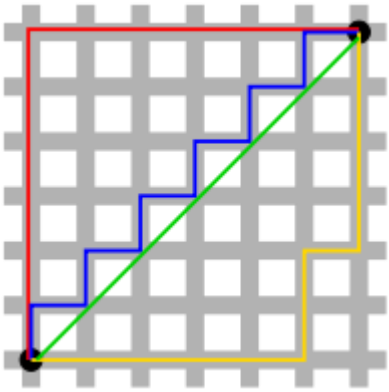
Cho tập $X=\{x_1,\dots x_n\}$ gồm n đối tượng dữ liệu trong không gian p-chiều, $x_i \in R^p$. Ta cần chia X thành k cụm đôi một không giao nhau: $\bigcup_{i=1}^{k} C_i$, $C_i \cap C_j = \emptyset$, $i \neq j$, sao cho các đối tượng trong cùng một cụm thì tương tự nhau và các đối tượng trong các cụm khác nhau thì khác nhau hơn theo một cách nhìn nào đó.

Số lượng k cụm có thể được cho trước hoặc xác định nhờ phương pháp phân cụm. Để thực hiện phân cụm ta cần xác định được mức độ tương tự giữa các đối tượng, tiêu chuẩn để phân cụm, trên cơ sở đó xây dựng mô hình và các thuật toán phân cụm theo

nhiều cách tiếp cận. Mỗi cách tiếp cận cho ta kết quả phân cụm với ý nghĩa sử dụng khác nhau.

3.3. Một số độ đo trong phân cụm dữ liệu:

-Manhattan: Công thức tính khoảng cách Manhattan là công thức dùng để tính toán khoảng cách giữa hai điểm dữ liệu trong không gian dạng lưới theo đường thẳng. Ở đây, ta sẽ sử dụng lại công thức tính Minkowski với p = 1 để tính toán. Ngoài ra công thức tính khoảng cách Manhattan còn được biết đến với các dạng bài toán như Taxicab Geometry, City Block Distance. Ở hình 2.2 mọi khoảng cách giữa hai điểm đều có kết quả như nhau khi tính chúng bằng công thức Manhattan.



Hình 1.1. Minh hoạ cách tính khoảng cách trong công thức Manhattan

Công thức : $d = \sum_{i=1}^{n} |x_i - y_i|$

Trong đó :

d : là khoảng cách giữa hai điểm

n : là số lượng thuộc tính của một dữ liệu

i : là thuộc tính thứ i của hai điểm dữ liệu

x_i: là thuộc tính của dữ liệu thứ i trong tập dữ liệu huấn luyện

y_i: là thuộc tính của dữ liệu thứ i trong tập kiểm thử

-Euclidean **(Euclid)**: Công thức tính khoảng cách Euclidean là một trong những công thức được sử dụng phổ biến. Nó được tính toán dựa vào công thức của Minkowski khi ta cho giá trị p = 2. Công thức:

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Trong đó:

d : là khoảng cách giữa hai điểm

n : là số lượng thuộc tính của một dữ liệu

i : là dữ liệu thứ i

x_i: là thuộc tính của dữ liệu thứ i trong tập dữ liệu

y_i: là thuộc tính của dữ liệu thứ i trong tập kiểm thử.

Công thức Cosine **(Cosine Similarity)**: Công thức tính góc cosine của hai vector được dùng tìm điểm tương đồng giữa nhiều dữ liệu khác nhau. Trong công thức, ta sẽ đo độ của góc giữa hai vector và để thực hiện được công thức này thì hai vector phải khác với $\vec{0}$. Công thức này sẽ có kết quả nằm từ [-1, 1].

Gọi A và B là hai vector ta có công thức tính góc cosine như sau :

$$Simularity(A,B) = \cos(\Theta) = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}|.|\vec{B}|}$$

Khi ta tính được góc cosine của hai loại dữ liệu từ công thức trên, ta sẽ dễ dàng đưa ra ba kết luận như sau:

- Cos 0’ = 1: Tức là hai dữ liệu đang có điểm tương đồng
- Cos 90’ = 0: Tức là hai dữ liệu không hề có điểm tương đồng
- Cos 180’ = -1: Tức là hai dữ liệu có những điểm đối nghịch nhau

3.4. Thế nào là một phân cụm tốt:[7]

- Một phương pháp tốt sẽ tạo ra các cụm có chất lượng cao theo nghĩa có sự tương tự cao trong một lớp, tương tự thấp giữa các lớp.
- Chất lượng của kết quả gom cụm phụ thuộc vào: độ đo tương tự sử dụng và việc cài đặt độ đo tương tự.
- Chất lượng của phương pháp gom cụm cũng được đo bởi khả năng phát hiện các mẫu bị che.

3.5. Ứng dụng:

Phân cụm dữ liệu đã được ứng dụng trong các lĩnh vực khác nhau như phân đoạn ảnh, nhận dạng đối tượng, ký tự và các chuyên ngành cổ điển như tâm lý học, kinh doanh,… Một số ứng dựng cơ bản của phân cụm dữ liệu bao gồm:

- Thương mại: phân cụm dữ liệu giúp các thương nhân khám phá ra các nhóm khách hàng quan trọng có các đặc điểm tương đồng nhau và đặc tả họ từ các mẫu mua bán trong cơ sở dữ liệu về khách hàng
- Sinh học: phân cụm dữ liệu dùng để xác định các loài sinh vật phân loại các mẫu Gen tương đồng và thu được cấu trúc của các mẫu
- Phân tích dữ liệu không gian: Do sự đồ sộ của dữ liệu không gian như dữ liệu thu được từ các hình ảnh chụp từ vệ tinh, các thiết bị y học hoặc hệ thống thông tin địa lý (GIS),… người dùng rất khó để kiểm tra các dữ liệu không gian một cách chi tiết. phân cụm dữ liệu có thể trợ giúp người dùng tự động phân tích và xử lý các dữ liêu không gian như nhận dạng và chiết xuất các đặc tính hoặc các mẫu dữ liệu quan tâm có thể tồn tại trong cơ sở dữ liệu không gian.
- Lập quy hoạch đô thị: Nhận dạng các nhóm nhà theo kiểu, vị trí địa lý, loại nhà,… để thuận tiện cho việc quy hoạch đô thị
- Địa lý: Phân lớp các động thực vật theo vị trí địa lý từ đó đưa ra các đặc trưng của giống loài mà ta tìm hiểu
- Khai phá web: phân cụm dữ liệu có thể khám phá ra các nhóm tài liệu quan trọng, có nhiều ý nghĩa trong môi trường web. Các lớp tài liệu này trợ giúp cho việc khám phá các tri thức dữ liệu

3.6. Các yêu cầu đối với phân cụm dữ liệu:[1]

Việc xây dựng, lựa chọn một thuật toán phân cụm là bước then chốt để giải bài toán phân cụm dữ liệu. Sự lựa chọn này hoàn toàn phụ thuộc vào đặc tính của dữ liệu cần phân cụm, mục đích của ứng dụng thực tế hoặc xác định độ ưu tiên giữa các cụm hay tốc độ thực hiện thuật toán. Đa phần các nghiên cứu phát triển thuật toán phân cụm dữ liệu đều nhằm vào các yêu cầu cơ bản gồm có:

- Có khả năng mở rộng: Một số thuật toán có thể ứng dụng tốt cho tập dữ liệu nhỏ (khoảng 200 bản ghi dữ liệu) nhưng không hiệu quả khi áp dụng cho tập dữ liệu lớn (khoảng 1 triệu bản ghi).

- Thích nghi với các kiểu dữ liệu khác nhau: Thuật toán có thể áp dụng hiệu quả cho việc phân cụm các tập dữ liệu với nhiều kiểu dữ liệu khác nhau như dữ liệu kiểu số, kiểu nhị phân, dữ liệu định danh, hạng mục,... và thích nghi với kiểu dữ liệu hỗn hợp.
- Khám phá ra các cụm với hình thù bất kỳ: Do hầu hết các cơ sở dữ liệu có chứa nhiều cụm dữ liệu với các hình thù khác nhau như: hình lõm, hình cầu, hình que,… Vì vậy, để khám phá được các cụm có tính tự nhiên thì các thuật toán phân cụm cần phải có khả năng khám phá ra các cụm dữ liệu có hình thù bất kỳ.
- Tối thiểu lượng tri thức cần cho xác định các tham số vào: Do các giá trị đầu vào thường ảnh hưởng rất lớn đến thuật toán phân cụm và rất phức tạp để xác định các giá trị vào thích hợp đối với các cơ sở dữ liệu lớn.
- Ít nhạy cảm với thứ tự của dữ liệu vào: Cùng một tập dữ liệu, khi đưa vào xử lý cho thuật toán phân cụm dữ liệu với các thứ tự vào của các đối tượng dữ liệu ở các lần thực hiện khác nhau thì không ảnh hưởng lớn đến kết quả phân cụm.
- Khả năng thích nghi với dữ liệu nhiễu cao: Hầu hết các dữ liệu phân cụm trong khai phá dữ liệu đều chứa đựng các dữ liệu lỗi, dữ liệu không đầy đủ, dữ liệu rác.
- Thuật toán phân cụm không những hiệu quả đối với các dữ liệu nhiễu mà còn tránh dẫn đến chất lượng phân cụm thấp do nhạy cảm với nhiễu.
- Ít nhạy cảm với các tham số đầu vào: Nghĩa là giá trị của các tham số đầu vào khác nhau ít gây ra các thay đổi lớn đối với kết quả phân cụm.
- Thích nghi với dữ liệu đa chiều: Thuật toán có khả năng áp dụng hiệu quả cho dữ liệu có số chiều khác nhau.
- Dễ hiểu, dễ cài đặt và khả thi.

Các yêu cầu này đồng thời là các tiêu chí để đánh giá hiệu quả của các phương pháp phân cụm dữ liệu, đây là những thách thức cho các nhà nghiên cứu trong lĩnh vực phân cụm dữ liệu.

3.7. Một số kỹ thuật phân cụm dữ liệu:

Có rất nhiều các phương pháp gom cụm khác nhau. Việc lựa chọn phương pháp nào tuỳ thuộc vào kiểu dữ liệu, mục tiêu và ứng dụng cụ thể. Nhìn chung, có thể chia thành các phương pháp sau:

3.7.1. Phân cụm phân hoạch:[2][3][6]

Ý tưởng chính của kỹ thuật này là phân một tập dữ liệu có n phần tử cho trước thành k nhóm dữ liệu sao cho mỗi phần tử dữ liệu chỉ thuộc về một nhóm dữ liệu và mỗi nhóm dữ liệu có tối thiểu ít nhất một phần tử dữ liệu. Các thuật toán phân hoạch có độ phức tạp rất lớn khi xác định nghiệm tối ưu toàn cục cho vấn đề phân cụm dữ liệu, vì nó phải tìm kiếm tất cả các cách phân hoạch có thể được. Chính vì vậy, trên thực tế người ta thường đi tìm giải pháp tối ưu cục bộ cho vấn đề này bằng cách sử dụng một hàm tiêu chuẩn để đánh giá chất lượng của các cụm cũng như để hướng dẫn cho quá trình tìm kiếm phân hoạch dữ liệu. Với chiến lược này, thông thường người ta bắt đầu khởi tạo một phân hoạch ban đầu cho tập dữ liệu theo phép ngẫu nhiên hoặc theo heuristic và liên tục tinh chỉnh nó cho đến khi thu được một phân hoạch mong muốn, thoả mãn các điều kiện ràng buộc cho trước. Các thuật toán phân cụm phân hoạch cố gắng cải tiến tiêu chuẩn phân cụm bằng cách tính các giá trị đo độ tương tự giữa các đối tượng dữ liệu và sắp xếp các giá trị này, sau đó thuật toán lựa chọn một giá trị trong dãy sắp xếp sao cho hàm tiêu chuẩn đạt giá trị tối thiểu. Như vậy, ý tưởng chính của thuật toán phân cụm phân hoạch tối ưu cục bộ là sử dụng chiến lược ăn tham để tìm kiếm nghiệm.

Lớp các thuật toán phân cụm phân hoạch bao gồm các thuật toán đề xuất đầu tiên trong lĩnh vực khai phá dữ liệu cũng là các thuật toán được áp dụng nhiều trong thực tế như k-means, k-medoid, PAM, CLARA, CLARANS.

3.7.2. Phân cụm phân cấp: [2][4] [6]

Phân cụm phân cấp sắp xếp một tập dữ liệu đã cho thành một cấu trúc có dạng hình cây, cây phân cấp này được xây dựng theo kỹ thuật đệ quy. Cây phân cụm có thể được xây dựng theo hai phương pháp tổng quát: phương pháp "trên xuống" (Top down) và phương pháp "dưới lên" (Bottom up).

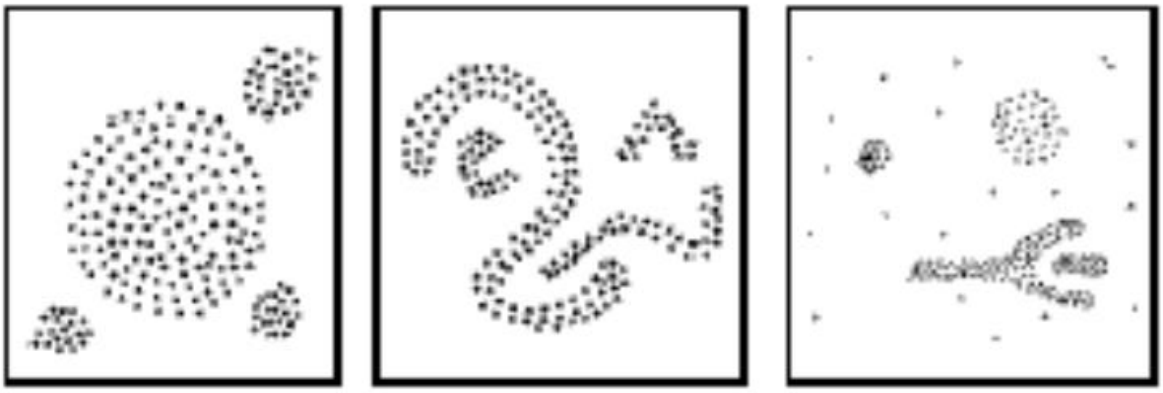
Phương pháp Bottom up: Phương pháp này bắt đầu với mỗi đối tượng được khởi tạo tương ứng với các cụm riêng biệt, sau đó tiến hành nhóm các đối tượng theo một độ đo tương tự (như khoảng cách giữa hai trung tâm của hai nhóm), quá trình này được thực hiện cho đến khi tất cả các nhóm được hòa nhập vào một nhóm (mức cao nhất của cây phân cấp) hoặc cho đến khi các điều kiện kết thúc thoả mãn. Như vậy, cách tiếp cận này sử dụng chiến lược ăn tham trong quá trình phân cụm.

Phương pháp Top Down: Bắt đầu với trạng thái là tất cả các đối tượng được xếp trong cùng một cụm. Mỗi vòng lặp thành công, một cụm được tách thành các cụm nhỏ hơn theo giá trị của một phép đo độ tương tự nào đó cho đến khi mỗi đối tượng là một cụm hoặc cho đến khi điều kiện dừng thoả mãn. Cách tiếp cận này sử dụng chiến lược chia để trị trong quá trình phân cụm.

Trong thực tế áp dụng, có nhiều trường hợp người ta kết hợp cả hai phương pháp phân cụm phân hoạch và phương phân cụm phân cấp, nghĩa là kết quả thu được của phương pháp phân cấp có thể cải tiến thông quan bước phân cụm phân hoạch. Phân cụm phân hoạch và phân cụm phân cấp là hai phương pháp phân cụm dữ liệu cổ điển, hiện nay đã có nhiều thuật toán cải tiến dựa trên hai phương pháp này đã được áp dụng phổ biến trong khai phá dữ liệu. Một số thuật toán phân cụm phân cấp điển hình như CURE, BIRCH, Chemeleon, AGNES, DIANA,...

3.7.3. Phân cụm dựa trên mật độ: [2][5] [6]

Phương pháp này nhóm các đối tượng theo hàm mật độ xác định. Mật độ được định nghĩa như là số các đối tượng lân cận của một đối tượng dữ liệu theo một ngưỡng nào đó. Trong cách tiếp cận này, khi một cụm dữ liệu đã xác định thì nó tiếp tục được phát triển thêm các đối tượng dữ liệu mới miễn là số các đối tượng lân cận của các đối tượng này phải lớn hơn một ngưỡng đã được xác định trước. Phương pháp phân cụm dựa vào mật độ của các đối tượng để xác định các cụm dữ liệu và có thể phát hiện ra các cụm dữ liệu với hình thù bất kỳ. Tuy vậy, việc xác định các tham số mật độ của thuật toán rất khó khăn, trong khi các tham số này lại có tác động rất lớn đến kết quả phân cụm dữ liệu. Hình minh hoạ về các cụm dữ liệu với các hình thù khác nhau dựa trên mật độ được khám phá từ 3 cơ sở dữ liệu khác nhau:



Hình 1.2. Một số hình dạng khám phá bởi phân cụm dựa trên mật độ

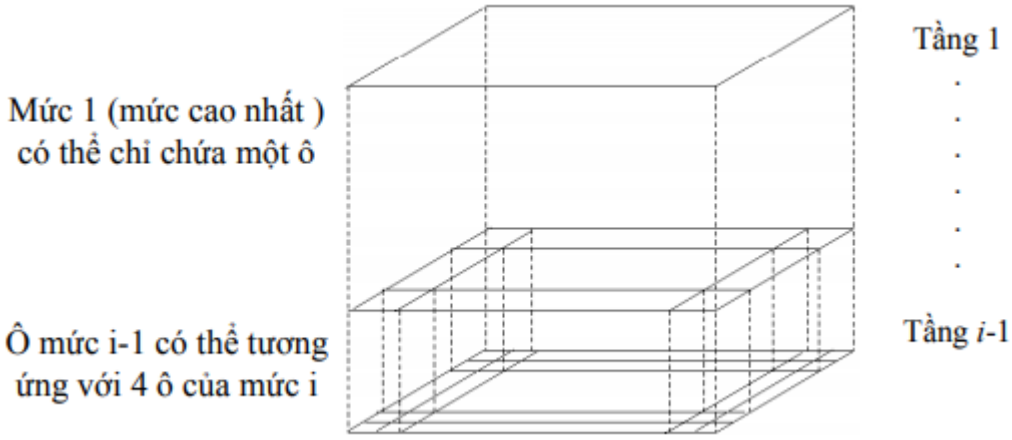
Các cụm có thể được xem như các vùng có mật độ cao, được tách ra bởi các vùng không có hoặc ít mật độ. Khái niệm mật độ ở đây được xem như là các số các đối tượng láng giềng.

Một số thuật toán phân cụm dữ liệu dựa trên mật độ điển hình như: DBSCAN, OPTICS, DENCLUE, SNN,…

3.7.4. Phân cụm dựa trên lưới:[6]

Kỹ thuật phân cụm dựa trên mật độ không thích hợp với dữ liệu nhiều chiều, để giải quyết cho đòi hỏi này, người ta đã sử dụng phương pháp phân cụm dựa trên lưới. Đây là phương pháp dựa trên cấu trúc dữ liệu lưới để phân cụm dữ liệu, phương pháp này chủ yếu tập trung áp dụng cho lớp dữ liệu không gian. Thí dụ như dữ liệu được biểu diễn dưới dạng cấu trúc hình học của đối tượng trong không gian cùng với các quan hệ, các thuộc tính, các hoạt động của chúng. Mục tiêu của phương pháp này là lượng hoá tập dữ liệu thành các ô (Cell), các ô này tạo thành cấu trúc dữ liệu lưới, sau đó các thao tác phân cụm dữ liệu làm việc với các đối tượng trong từng ô này. Cách tiếp cận dựa trên lưới này không di chuyển các đối tượng trong các ô mà xây dựng nhiều mức phân cấp của nhóm các đối tượng trong một ô. Trong ngữ cảnh này, phương pháp này gần giống với phương pháp phân cụm phân cấp nhưng chỉ có điều chúng không trộn các ô. Do vậy các cụm không dựa trên độ đo khoảng cách (hay còn gọi là độ đo tương tự đối với các dữ liệu không gian) mà nó được quyết định bởi một tham số xác định trước. Ưu điểm của phương pháp phân cụm dữ liệu dựa trên lưới là thời gian xử lý nhanh và độc lập với số đối tượng dữ liệu trong tập dữ liệu ban đầu, thay vào đó là chúng phụ thuộc vào số ô trong mỗi

chiều của không gian lưới. Một thí dụ về cấu trúc dữ liệu lưới chứa các ô trong không gian như hình sau:



Hình 1.3. Mô hình cấu trúc dữ liệu lưới

3.7.5. Phân cụm dữ liệu dựa trên mô hình:[6]

Phương pháp này cố gắng khám phá các phép xấp xỉ tốt của các tham số mô hình sao cho khớp với dữ liệu một cách tốt nhất. Chúng có thể sử dụng chiến lược phân cụm phân hoạch hoặc chiến lược phân cụm phân cấp, dựa trên cấu trúc hoặc mô hình mà chúng giả định về tập dữ liệu và cách mà chúng tinh chỉnh các mô hình này để nhận dạng ra các phân hoạch.

Phương pháp phân cụm dữ liệu dựa trên mô hình cố gắng khớp giữa dữ liệu với mô hình toán học, nó dựa trên giả định rằng dữ liệu được tạo ra bằng hỗn hợp phân phối xác suất cơ bản. Các thuật toán phân cụm dựa trên mô hình có hai tiếp cận chính: Mô hình thống kê và Mạng Nơron. Một số thuật toán điển hình như EM, COBWEB,...

3.7.6. Phân cụm dữ liệu có ràng buộc:[7]

Sự phát triển của phân cụm dữ liệu không gian trên cơ sở dữ liệu lớn đã cung cấp nhiều công cụ tiện lợi cho việc phân tích thông tin địa lý, tuy nhiên hầu hết các thuật toán này cung cấp rất ít cách thức cho người dùng để xác định các ràng buộc trong thế giới thực cần phải được thoả mãn trong quá trình phân cụm dữ liệu. Để phân cụm dữ liệu không gian hiệu quả hơn, các nghiên cứu bổ sung cần được thực hiện để cung cấp cho người dùng khả năng kết hợp các ràng buộc trong thuật toán phân cụm.

Thực tế, các phương pháp trên đã và đang được phát triển và áp dụng nhiều trong phân cụm dữ liệu. Đến nay, đã có một số nhánh nghiên cứu được phát triển trên cơ sở của các phương pháp tiếp cận trong phân cụm dữ liệu đã trình bày ở trên như sau:

- Phân cụm thống kê : Dựa trên các khái niệm phân tích thống kê, nhánh nghiên cứu này sử dụng các độ đo tương tự để phân hoạch các đối tượng, nhưng chúng chỉ áp dụng cho các dữ liệu có thuộc tính số.
- Phân cụm khái niệm : Các kỹ thuật phân cụm được phát triển áp dụng cho dữ liệu hạng mục, chúng phân cụm các đối tượng theo các khái niệm mà chúng xử lý.
- Phân cụm mờ : Sử dụng kỹ thuật mờ để phân cụm dữ liệu, trong đó một đối tượng dữ liệu có thể thuộc vào nhiều cụm dữ liệu khác nhau. Các thuật toán thuộc loại này chỉ ra lược đồ phân cụm thích hợp với tất cả hoạt động đời sống hàng ngày, 31 chúng chỉ xử lý các dữ liệu thực không chắc chắn. Thuật toán phân cụm mờ quan trọng nhất là thuật toán FCM (Fuzzy c-means).
- Phân cụm mạng Kohonen : loại phân cụm này dựa trên khái niệm của các mạng nơ ron. Mạng Kohnen có tầng nơ ron vào và các tầng nơ ron ra. Mỗi nơ ron của tầng vào tương ứng với mỗi thuộc tính của bản ghi, mỗi một nơ ron vào kết nối với tất cả các nơ ron của tầng ra. Mỗi liên kết được gắn liền với một trọng số nhằm xác định vị trí của nơ ron ra tương ứng.

Các kỹ thuật phân cụm dữ liệu trình bày ở trên đã được sử dụng rộng rãi trong thực tế, thế nhưng hầu hết chúng chỉ nhằm áp dụng cho tập dữ liệu với cùng một kiểu thuộc tính. Vì vậy, việc phân cụm dữ liệu trên tập dữ liệu có kiểu hỗn hợp là một vấn đề đặt ra trong khai phá dữ liệu trong giai đoạn hiện nay. Phần nội dung tiếp theo của luận văn sẽ trình bày tóm lược về các yêu cầu cơ bản làm tiêu chí cho việc lựa chọn, đánh giá kết quả cho các phương pháp phân cụm phân cụm dữ liệu.

4. Kết luận:

Như vậy, với phần giới thiệu này đã cung cấp cho bạn câu trả lời cho câu hỏi về khai phá dữ liệu, thuật toán phân cụm dữ liệu. Mình cũng đã trình bày các ứng dụng của phân cụm dữ liệu. Qua đây, hi vọng rằng các bạn đã tìm được câu trả lời cho thuật toán phân cụm dữ liệu là gì. Qua đó để thấy được tầm quan trọng của việc khai phá dữ liệu để phục vụ cho công việc trong tương lai.

CHƯƠNG 2: THUẬT TOÁN K-MEANS

1. Lịch sử:

Thuật ngữ " *k* -means" được James MacQueen sử dụng lần đầu tiên vào năm 1967, mặc dù ý tưởng này quay trở lại Hugo Steinhaus vào năm 1956. Thuật toán tiêu chuẩn được đề xuất lần đầu tiên bởi Stuart Lloyd của Bell Labs vào năm 1957 như một kỹ thuật cho điều chế mã xung, mặc dù nó không được xuất bản dưới dạng một bài báo cho đến năm 1982. Năm 1965, Edward W. Forgy đã công bố về cơ bản cùng một phương pháp, đó là lý do tại sao nó đôi khi được gọi là Lloyd-Forgy.

2. Thuật toán chi tiết:

Thuật toán phân cụm k-means do MacQueen đề xuất trong lĩnh vực thống kê năm 1967, mục đích của thuật toán k-means là sinh ra k cụm dữ liệu $\{C_1, C_2, \ldots, C_k\}$ từ một tập dữ liệu ban đầu gồm n đối tượng trong không gian d chiều $X_i = (x_{i1}, x_{i2}, \ldots, x_{id})$ ($i = \overline{1,n}$), sao cho hàm tiêu chuẩn: $\sum_{i=1}^{k} \sum_{x \in C_i} D^2 (x - m_i)$ đạt giá trị tối thiểu. Trong đó: m_i là trọng tâm của cụm C_i , D là khoảng cách giữa hai đối tượng.

Trọng tâm của một cụm là một vector, trong đó giá trị của mỗi phần tử của nó là trung bình cộng các thành phần tương ứng của các đối tượng vector dữ liệu trong cụm đang xét. Tham số đầu vào của thuật toán là số cụm k, tập cơ sở dữ liệu gồm n phần tử và tham số đầu ra của thuật toán là các trọng tâm của các cụm dữ liệu. Độ đo khoảng cách D giữa các đối tượng dữ liệu thường được sử dụng dụng là khoảng cách Euclide, bởi vì đây là mô hình khoảng cách dễ để lấy đạo hàm và xác định các cực trị tối thiểu. Hàm tiêu chuẩn và độ đo khoảng cách có thể được xác định cụ thể hơn tuỳ vào ứng dụng hoặc các quan điểm của người dùng. Thuật toán k-means bao gồm các bước cơ bản như sau:

Input: Một cơ sở dữ liệu gồm n đối tượng và số các cụm *k*

Output: Các cụm C_i ($i=\overline{1,\ k}$) sao cho hàm tiêu chuẩn E đạt giá trị tối thiểu.

Begin

Bước 1: Khởi tạo

Chọn *k* đối tượng m_j ($j= \overline{1,\ k}$) là trọng tâm ban đầu của *k* cụm từ tập dữ liệu(Việc lựa chọn này có thể là ngẫu nhiên hoặc theo kinh nghiệm)

Bước 2: Tính toán khoảng cách

Đối với mỗi điểm X_i ($i=\overline{1,\ n}$), tính toán khoảng cách của nó tới mỗi trọng tâm m_j ($j= \overline{1,\ k}$). Sau đó tìm trọng tâm gần nhất đối với mỗi điểm.

Bước 3: Cập nhật lại trọng tâm

Đối với mỗi $1 \le j \le k$, cập nhật trọng tâm cụm m_j bằng cách xác định trung bình cộng các vectơ đối tượng dữ liệu.

Điều kiện dừng:

Lặp lại các bước 2 và 3 cho đến khi các trọng tâm của cụm không thay đổi.

End.

Thuật toán k-means được chứng minh là hội tụ và có độ phức tạp tính toán là: $O((n\cdot k\cdot d)\cdot\tau\cdot T^{flop})$. Trong đó: n là số đối tượng dữ liệu, k là số cụm dữ liệu, d là số chiều, τ là số vòng lặp, T^{flop} là thời gian để thực hiện một phép tính cơ sở như phép tính nhân, chia, …Như vậy, do k-means phân tích phân cụm đơn giản nên có thể áp dụng đối với tập dữ liệu lớn. Tuy nhiên, nhược điểm của k-means là chỉ áp dụng với dữ liệu có thuộc tính số và khám phá ra các cụm có dạng hình cầu, k-means còn rất nhạy cảm với nhiễu và các phần tử ngoại lai trong dữ liệu.

Đến nay, đã có rất nhiều thuật toán kế thừa tư tưởng của thuật toán k-means áp dụng trong khai phá dữ liệu để giải quyết tập dữ liệu có kích thước rất lớn đang được áp dụng rất hiệu quả và phổ biến như thuật toán k-medoid, PAM, CLARA, CLARANS, …

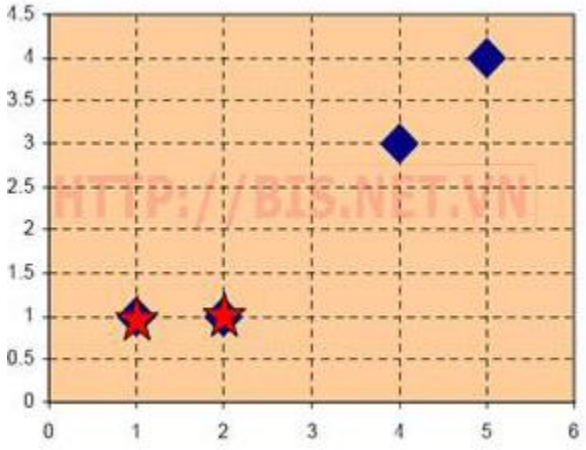
3. Ví dụ:

Giả sử ta có 4 loại ván A,B,C,D, mỗi loại thuộc được biểu diễn bởi 2 đặc trưng chiều dài và chiều rộng như sau. Mục đích của ta là nhóm các miếng ván đã cho vào 2 nhóm (K=2) dựa vào các đặc trưng của chúng.

Bảng 2.1. Miếng ván cửa hàng SMY

Ván	Dài(m)	Rộng(m)
A	1	1
B	2	1
C	4	3
D	5	4

Bước 1. Khởi tạo tâm cho 2 nhóm. Giả sử ta chọn A là tâm của nhóm thứ nhất (tọa độ tâm nhóm thứ nhất $C_1(1,1)$) và B là tâm của nhóm thứ 2 (tọa độ tâm nhóm thứ hai C_2 (2,1)).



Hình 2.1. Khởi tạo các điểm và chọn tâm cụm là $C_1(1,1)$ và $C_2(2,1)$

Bước 2. Tính khoảng cách từ các đối tượng đến tâm của các nhóm (Khoảng cách Euclidean) và nhóm các đối tượng vào nhóm gần nhất

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

d(A,C_1)= $\sqrt{(1-1)^2 + (1-1)^2} = 0$

d(B,C_1)= $\sqrt{(2-1)^2 + (1-1)^2} = 1$

d(C,C_1)= $\sqrt{(4-1)^2 + (3-1)^2} = 3.61$

d(D,C_1)= $\sqrt{(5-1)^2 + (4-1)^2} = 5$

$d(A,C_2)= \sqrt{(1-2)^2 + (1-1)^2} = 1$

$d(B,C_2)= \sqrt{(2-2)^2 + (1-1)^2} = 0$

$d(C,C_2)= \sqrt{(4-2)^2 + (3-1)^2} = 2.83$

$d(D,C_2)= \sqrt{(5-2)^2 + (4-1)^2} = 4.24$

Bảng 2.2. Nhóm các đối tượng vào cụm gần nhất(lần 1)

Ván	Khoảng cách tới C1	Khoảng cách tới C2	Nhóm
A	0	1	1
B	1	0	2
C	3.61	2.83	2
D	5	4.24	2

Bước 3. Cập nhật lại vị trí trọng tâm[8]

$$\mu_k = \frac{1}{n}(X_{k1} + X_{k2} + \ldots + X_{kn})$$

Trong đó:

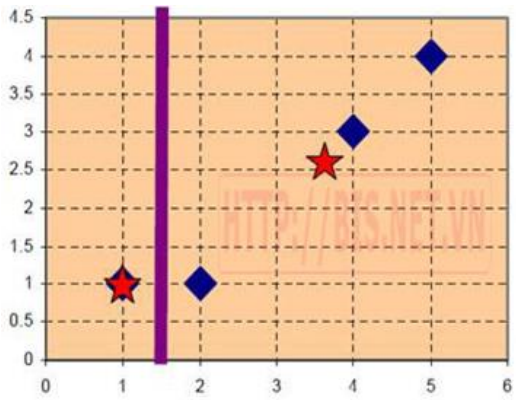
μ_k: chỉ số dữ liệu của trung tâm k

n: tổng số điểm cụm k

$X_{ki}(i=\overline{1,\ n})$: chỉ số các dữ liệu thứ k_i thuộc cụm k

Cụm 1 có một đối tượng A nên tâm cụm $C_1(1,1)$ không thay đổi. Cụm 2 có 3 đối tượng B,C,D tâm cụm C_2 được thay đổi như sau:

$$C_2=(\frac{2+4+5}{3}, \frac{1+3+4}{3}) = (\frac{11}{3}, \frac{8}{3})$$



Hình 2.2. Chọn lại tâm cụm (lần 1)

Bước 4. Lặp lại các bước 2, 3 cho đến khi trọng tâm của cụm không thay đổi

4-1. Tính khoảng cách từ các đối tượng đến tâm của các nhóm (Khoảng cách Euclidean) và nhóm các đối tượng vào nhóm gần nhất

$d(A,C_1)= \sqrt{(1-1)^2 + (1-1)^2} = 0$

$d(B,C_1)= \sqrt{(2-1)^2 + (1-1)^2} = 1$

$d(C,C_1)= \sqrt{(4-1)^2 + (3-1)^2} = 3.61$

$d(D,C_1)= \sqrt{(5-1)^2 + (4-1)^2} = 5$

$d(A,C_2)= \sqrt{\left(1-\frac{11}{3}\right)^2 + \left(1-\frac{8}{3}\right)^2} = 3.14$

$d(B,C_2)= \sqrt{\left(2-\frac{11}{3}\right)^2 + \left(1-\frac{8}{3}\right)^2} = 2.36$

$d(C,C_2)= \sqrt{\left(4-\frac{11}{3}\right)^2 + \left(3-\frac{8}{3}\right)^2} = 0.47$

$d(D,C_2)= \sqrt{\left(5-\frac{11}{3}\right)^2 + \left(4-\frac{8}{3}\right)^2} = 1.89$

Bảng 2.3. Nhóm các đối tượng vào cụm gần nhất(lần 2)

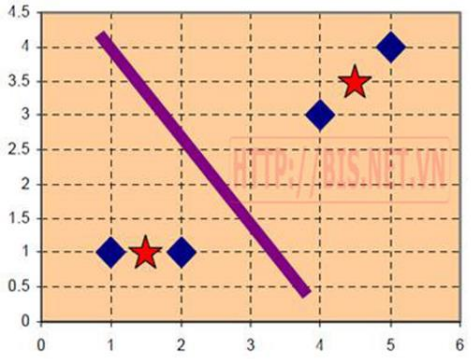
Ván	Khoảng cách tới C1	Khoảng cách tới C2	Nhóm
A	0	3.14	1
B	1	2.36	1
C	3.61	0.47	2
D	5	1.89	2

4-2 Cập nhật lại vị trí trọng tâm

Cụm 1 có hai đối tượng A,B, cụm 2 có 3 đối tượng C,D được thay đổi như sau:

$$C_1=\left(\frac{1+2}{2},\frac{1+1}{2}\right) = \left(\frac{3}{2},1\right)$$

$$C_2=\left(\frac{4+5}{2},\frac{3+4}{2}\right) = \left(\frac{9}{2},\frac{7}{2}\right)$$



Hình 2.3. Chọn lại tâm cụm (lần 2)

4-3 Tính khoảng cách từ các đối tượng đến tâm của các nhóm (Khoảng cách Euclidean) và nhóm các đối tượng vào nhóm gần nhất

$d(A,C_1)= \sqrt{\left(1-\frac{3}{2}\right)^2 + (1-1)^2} = 0{,}5$

$d(B,C_1)= \sqrt{\left(2-\frac{3}{2}\right)^2 + (1-1)^2} = 0.5$

$d(C,C_1)= \sqrt{\left(4-\frac{3}{2}\right)^2 + (3-1)^2} = 3.2$

$d(D,C_1)= \sqrt{\left(5-\frac{3}{2}\right)^2 + (4-1)^2} = 4.61$

$d(A,C_2)= \sqrt{\left(1-\frac{9}{2}\right)^2 + \left(1-\frac{7}{2}\right)^2} = 4.3$

$d(B,C_2)= \sqrt{\left(2-\frac{9}{2}\right)^2 + \left(1-\frac{7}{2}\right)^2} = 3.54$

$d(C,C_2)= \sqrt{\left(4-\frac{9}{2}\right)^2 + \left(3-\frac{7}{2}\right)^2} = 0.71$

$d(D,C_2)= \sqrt{\left(5-\frac{9}{2}\right)^2 + \left(4-\frac{7}{2}\right)^2} = 0.71$

4-4 Cập nhật lại vị trí trọng tâm

Cụm 1 có hai đối tượng A,B, cụm 2 có 3 đối tượng C,D được thay đổi như sau:

$$C_1=\left(\frac{1+2}{2},\frac{1+1}{2}\right) = \left(\frac{3}{2},1\right)$$

$$C_2=\left(\frac{4+5}{2},\frac{3+4}{2}\right) = \left(\frac{9}{2},\frac{7}{2}\right)$$

Ta thấy vị trí trọng tâm như khi chọn lại tâm cụm (lần 2) nên thuật toán dừng và kết quả phân nhóm như sau:

Bảng 2.4 Kết quả bài ví dụ

Ván	Dài(m)	Rộng(m)	Nhóm
A	1	1	1
B	2	1	1
C	4	3	2
D	5	4	2

4. Đánh giá thuật toán:

4.1. Ưu điểm:

- Độ phức tạp (K.N.l) l là số lần lặp
- Có khả năng mở rộng, có thể dễ dàng sửa đổi với những dữ liệu mới
- Bảo đảm hội tụ sau một số bước lặp hữu hạn
- Luôn có K cụm dữ liệu
- Luôn có ít nhất một điểm trong một cụm dữ liệu
- Các cụm không phân cấp và không bị chồng chéo dữ liệu lên nhau
- Mọi thành viên của một cụm là gần chính với cụm đó hơn so với bất cứ cụm khác

4.2. Khuyết điểm:

- Không có khả năng tìm ra các cụm không lồi và các cụm có hình dạng phức tạp
- Khó khăn trong việc xác định tâm các cụm ban đầu
    - Chọn ngẫu nhiên các trung tâm cụm lúc khởi tạo
    - Độ hội tụ của thuật toán phụ thuộc vào khởi tạo các vectơ trung tâm cụm
- Khó để chọn ra số cụm tối ưu ngay từ đầu, mà phải trải qua nhiều lần thử để tìm ra số lượng cụm tối ưu
- Rất nhạy cảm với nhiễu và các phần tử ngoại lai trong dữ liệu
- Không phải lúc nào mỗi đối tượng cũng chỉ thuộc về 1 cụm, chỉ phù hợp giữa đường biên giữa các cụm rõ

5. Kết luận:

Chương này đã trình bày chi tiết về thuật toán K-means và nêu ra các ví dụ ưu, nhược điểm của thuật toán này từ đó bạn có thể hiểu sơ lược về cách mà nó hoạt động. Như bạn

thấy ở trên k-means là một thuật toán nhanh và độ chính xác cao nhưng nó vẫn còn một số sai sót do dữ liệu bị nhiễu nên phải chọn một tâm cụm hoặc số k cụm khác để tính toán hiệu quả hơn. Hiện nay có nhiều thuật toán cải thiện điều đó như thuật toán Elbow, Silhoutte,… rất nhanh và hiệu quả.

CHƯƠNG 3: THUẬT TOÁN BIRCH

1. Khái niệm: [2][7]

BIRCH (Balanced Iterative Reducing and Clustering Using Hierarchies) do Tian Zhang, amakrishnan và Livny đề xuất năm 1996, là thuật toán phân cụm phân cấp sử dụng chiến lược Top down. Ý tưởng của thuật toán là không cần lưu toàn bộ các đối tượng dữ liệu của các cụm trong bộ nhớ mà chỉ lưu các đại lượng thống kê. Đối với mỗi cụm dữ liệu, BIRCH chỉ lưu một bộ ba (n, LS, SS), với n là số đối tượng trong cụm, LS là tổng các giá trị thuộc tính của các đối tượng trong cụm và SS là tổng bình phương các giá trị thuộc tính của các đối tượng trong cụm. Các bộ ba này được gọi là các đặc trưng của cụm CF=(n, LS, SS) (Cluster Features - CF) và được lưu giữ trong một cây được gọi là cây CF.

2. Đặc trưng cụm – Clustering Feature(CF): [7]

2.1 Định nghĩa:

Là một bộ 3 lưu giữ thông tin của cụm. Cho n điểm dữ liệu p chiều trong cụm C_ivới i=1,2,…n, đặc trưng cụm (CF) được định nghĩa là 1 bộ 3: CF (n,LS,SS).

Trong đó:

n: là số đối tượng (điểm) trong cụm.

LS: là tổng các giá trị thuộc tính của các đối tượng trong cụm

SS: là tổng bình phương của các giá trị thuộc tính của các đối tượng trong cụm

2.2 Lý thuyết cộng CF:

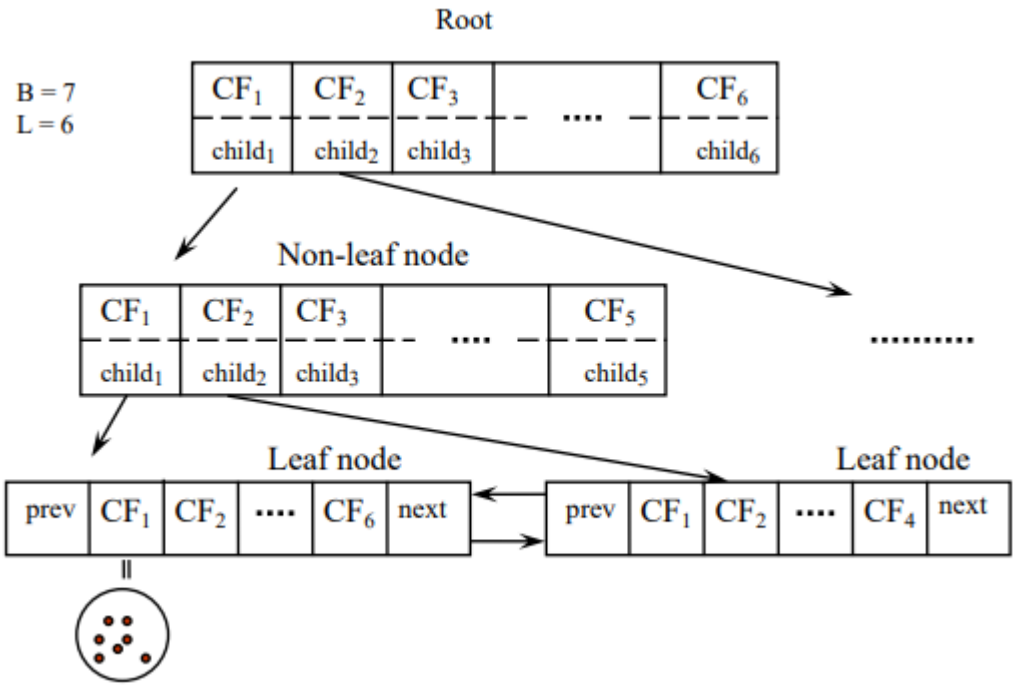
Giả sử có CF1 =(n_1, LS_1, SS_1), và CF_2=(n_2, LS2 ,SS_2) là các đặc trưng cụm của 2 cụm rời nhau. Khi đó CF gộp của 2 cụm này được tính là:

$$CF_1 + CF_2 (n_1+n_2 ,LS_1+ LS_2 ,SS_1+SS_2)$$

Từ định nghĩa CF và lý thuyết cộng, chúng ta biết rằng ứng với C_0, R, D, D_0, D_1,D_2, D_3, và D_4 đều có thể tính một cách dễ dàng.

2.3 Cây CF:

Các đặc trưng của cụm (Cluster Features – CF) được lưu giữ trong một cây gọi là cây CF (CF tree). Người ta đã chứng minh rằng, các đại lượng thống kê chuẩn như độ đo khoảng cách, có thể xác định cây CF. Hình dưới đây biểu thị một ví dụ về cây CF. Chúng ta thấy rằng tất cả các nút trong lưu tổng các đặc trưng cụm CF của nút con, trong khi đó các nút lá lưu trữ các đặc trưng của cụm dữ liệu.



Hình 3.1. Cây CF được sử dụng trong thuật toán BIRCH

Cây CF là cây cân bằng, nhằm để lưu trữ các đặc trưng của cụm (CF). Cây CF chứa các nút trong và nút lá, nút trong là nút chứa các nút con và nút là thì không có con. Nút trong lưu trữ tổng các đặc trưng cụm (CF) của các nút con của nó.

Một cây CF được đặc trưng bởi 2 tham số:

- Yếu tố phân nhánh (Branching Factor-B): Nhằm xác định tối đa các nút con của một nút trong. Mỗi nút trong chữa nhiều nhất B mục dạng [CF_i, $child_i$], với i=1,2…B, $child_i$ là một con trỏ tới nút con thứ i, và CF_i là CF của cụm con thể hiện bởi child. Nút lá chứa nhiều nhất L mục dạng [CF_i], với i=1,2…L.
- Ngưỡng (Threshloid-T): Khoảng cách tối đa giữ bất kì một cặp đối tượng trong nút lá của cây, khoảng cách này còn được gọi là bán kính hoặc đường kính của các cụm con được lưu tại các nút lá.

Hai tham số này có ảnh hưởng đến kích thước của cây CF.

Cây CF được xây dựng tự động như là việc chèn thêm đối tượng dữ liệu mới. Nó được sử dụng cho việc định hướng chèn mới chính xác vào các cụm con với mục đích phân cụm. Thực tế, tất cả các cụm được tạo thành như mỗi điểm dữ liệu được đưa vào cây CF.

Cây CF là biểu diễn rất nhỏ của bộ dữ liệu vì mỗi mục trong một nút lá không phải là một điểm dữ liệu duy nhất mà là một cụm con.

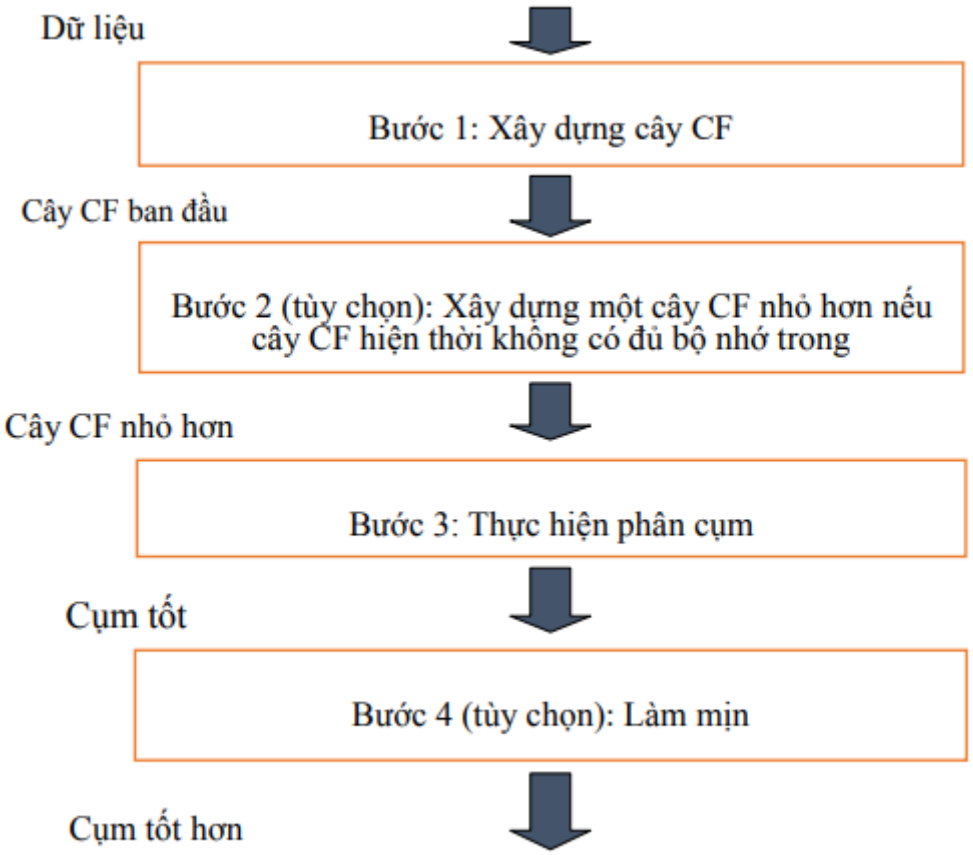
3. Các bước cơ bản thuật toán: [2]

Bước 1: Các đối tượng dữ liệu lần lượt được chèn vào cây CF, sau khi chèn hết các đối tượng ta thu được cây CF khởi tạo. Mỗi một đối tượng được chèn vào nút lá gần nhất tạo thành cụm con. Nếu đường kính của cụm con này lớn hơn T thì nút lá được tách. Khi một đối tượng thích hợp được chèn vào nút lá thì tất cả các nút trở tới gốc của cây được cập nhật với các thông tin cần thiết.

Bước 2: Nếu cây CF hiện thời không có đủ bộ nhớ trong thì tiến hành dựng cây CF nhỏ hơn: kích thước của cây CF được điều khiển bởi tham số T và vì vậy việc chọn một giá trị lớn hơn cho nó sẽ hòa nhập một số cụm con thành một cụm, điều này làm cho cây CF nhỏ hơn. Bước này không cần yêu cầu bắt đầu đọc dữ liệu lại từ đầu nhưng vẫn đảm bảo hiệu chỉnh cây dữ liệu nhỏ hơn.

Bước 3:Thực hiện phân cụm: các nút lá của cây CF lưu giữ các đại lượng thống kê của các cụm con. Trong bước này, BIRCH sử dụng các đại lượng thống kê này để áp dụng một số kĩ thuật phân cụm ví dụ như k-means và tạo ra một khởi tạo cho phân cụm.

Bước 4: Phân phối lại các đối tượng dữ liệu bằng cách dùng các đối tượng trọng tâm cho các cụm đã được khám phá từ bước 3: đây là một bước tùy chọn để duyệt lại tập dữ liệu và gán nhãn lại cho các đối tượng dữ liệu tới các trọng tâm gần nhất. Bước này nhằm để gán nhãn cho các dữ liệu khởi tạo và loại bỏ các đối tượng ngoại lai.



Hình 3.2. Khái quát thuật toán BIRCH

4. Các vấn đề cần quan tâm ở bước 1: [7]

4.1. Xây dựng lại cây CF:

Trong bước này ta sử dụng tất cả các mục lá của cây CF cũ để xây dựng lại một cây CF mới với ngưỡng lớn hơn. Trong quá trình xây dựng lại ta cần điều chỉnh đường đi tới nút lá. Đường đi tới nút lá tương ứng với một đường đi duy nhất tới nút lá. Thuật toán xây dựng lại là các thuật toán quét và giải phóng đường đi cây cũ, và tạo ra đường đi cho cây mới. Kích thước của cây mới phải nhỏ hơn cây trước. Việc chuyển từ cây cũ sang cây mới cần ít nhất thêm h trang bộ nhớ, trong đó h là chiều cao cây cũ.

4.2. Giá trị ngưỡng T:

Để tăng ngưỡng ta sử dụng phương pháp heuristic. Lựa chọn giá trị ngưỡng mới sao cho số lượng các điểm dữ liệu được quét dưới giá trị ngưỡng mới:

Phương pháp 1: Tìm nút lá đông nhất và hai mục gần nhất trên lá có thể được sáp nhập dưới ngưỡng mới.

Phương pháp 2: Giả sử lượng chiếm đóng các cụm lá tăng tuyến tính với điểm dữ liệu. một loạt các cặp giá trị: số lượng các điểm dữ liệu và khối lượng=> khối lượng mới (một điểm dữ liệu mới, sử dụng tối thiểu hồi quy tuyến tính) => ngưỡng mới. Sử dụng một số phương pháp heuristic để điều chỉnh hai ngưỡng trên và chọn một.

4.3. Outlier-handling option:

Outlier là là một giá trị ngoại lai hay nhiễu, trong cây CF nó đóng vai trò là một mục lá mật độ thấp, nó được đánh giá là quan trọng đối với mô hình phân nhóm tổng thể. Sử dụng một số không gian đĩa để xử lý giá trị ngoại lai.

Khi xây dựng lại các cây CF, một mục lá cũ chỉ được ghi vào đĩa nếu nó được coi là một outlier tiềm năng. Điều này có thể làm giảm kích thước của cây CF.

Một outlier không đủ tiêu chuẩn khi:

- Tăng trong giá trị ngưỡng;
- Sự thay đổi trong việc phân phối do nhiều dữ liệu được đọc.

Quét các outlier tiềm năng để hấp thu mà không gây ra phát triển quá kích thước cây:

- Hết không gian đĩa.
- Tất cả các điểm dữ liệu đã được quét.

4.4. Delay-Split option:

Khi hết bộ nhớ. Có thể có nhiều hơn các điểm dữ liệu phù hợp trong cây CF hiện tại. Chúng ta có thể tiếp tục đọc dữ liệu điểm và ghi những điểm dữ liệu cần chia một nút vào đĩa cho đến khi hết không gian đĩa. Ưu điểm của phương pháp này là nhiều hơn các điểm dữ liệu phù hợp trong cây trước khi chúng ta phải xây dựng lại.

5. Đánh giá thuật toán:[2]

5.1. Ưu điểm:

- Với cấu trúc cây CF, BIRCH có tốc độ thực hiện phân cụm dữ liệu nhanh.
- Tốt với tập dữ liệu lớn.
- Hiệu quả khi tập dữ liệu tăng trưởng theo thời gian.
- Chỉ duyệt toàn bộ dữ liệu một lần với một lần quét thêm tuỳ chọn.
- Độ phức tạp là O(n), với n là số đối tượng dữ liệu.

5.2. Khuyết điểm:

- Chất lượng của các cụm không được tốt.
- Nếu dùng khoảng cách Euclide, nó chỉ tốt với các dữ liệu số.
- Tham số T có ảnh hưởng rất lớn tới kích thước và tính tự nhiên của cụm.
- Không thích hợp với dữ liệu đa chiều.

6. Kết luận:

Chương này đã cho chúng ta hiểu thế nào là thuật toán BIRCH. Với thuật toán này ta giúp chia nhỏ lượng dữ liệu ra để ta phân cụm một cách dễ dàng hơn với tốc độ nhanh hơn. Đối với trường hợp sử dụng khoảng cách Euclide để đo khoảng cách thì chỉ thích hợp với dữ liệu số nếu dữ liệu là chữ sẽ tạo nhiễu làm cho chất lượng cụm không được tốt.

CHƯƠNG 4: THUẬT TOÁN DBSCAN

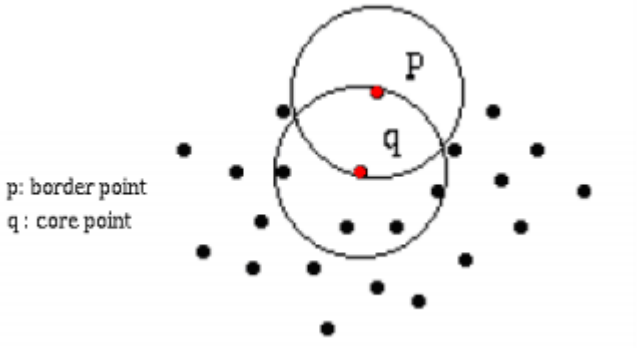
1. Khái niệm:[2][9]

Thuật toán DBSCAN (Density Based Spatial Clustering of Applications with Noise) do Martin Ester và các tác giả khác đề xuất là thuật toán gom tụm dựa trên mật độ, hiệu quả với cơ sở dữ liệu lớn, có khả năng xử lý nhiễu.

Ý tưởng chính của thuật toán là vùng lân cận mỗi đối tượng trong một cụm có số đối tượng lớn hơn ngưỡng tối thiểu. Hình dạng vùng lân cận phụ thuộc vào hàm khoảng cách giữa các đối tượng (nếu sử dụng khoảng cách Manhattan trong không gian 2 chiều thì vùng lân cận có hình chữ nhật, nếu sử dụng khoảng cách Eucler trong không gian 2 chiều thì vùng lân cận có hình tròn).

Tìm tất cả các đối tượng mà các láng giềng của nó thuộc về lớp các đối tượng đã xác định ở trên. Một cụm được xác định bằng một tập tất cả các đối tượng liên thông mật độ với các láng giềng của nó. DBSCAN có thể tìm ra các cụm với hình thù bất kỳ.

Các đối tượng trong mỗi cụm được phân làm 2 loại: đối tượng bên trong cụm (core point: điểm nhân) và đối tượng nằm trên đường biên của cụm (border point: điểm biên).



Hình 4.1. Điểm nhân và điểm biên

Khoảng cách Euclide được dùng đo sự tương tự giữa các đối tượng nhưng không hiệu quả đối với DL đa chiều. Vậy nên DBSCAN dựa trên các khái niệm mật độ có thể áp dụng cho các tập DL không gian lớn đa chiều.

DBSCAN yêu cầu xác định bán kính Eps của các láng giềng và số các láng giềng tối thiểu MinPts, thường xác định bằng ngẫu nhiên hoặc kinh nghiệm.

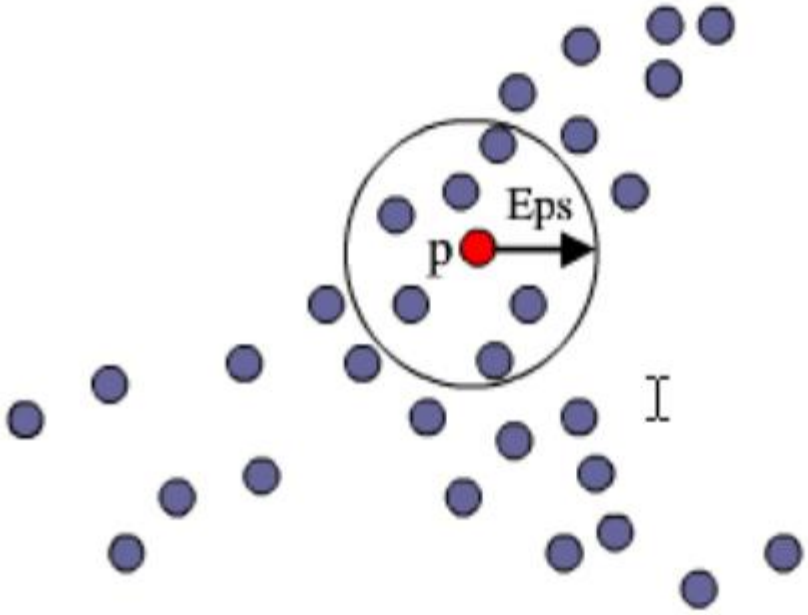
2. Định nghĩa:[2][9]

<u>Định nghĩa 1</u> : Lân cận của một điểm p với ngưỡng Eps (Eps - Neighborhood of a point)

Vùng lân cận Eps của đối tượng p, ký hiệu $N_{Eps}(p)$ là tập hợp các đối tượng q sao cho khoảng cách giữa p và q dist(p,q) nhỏ hơn Eps.

$$N_{Eps}(p) = \{q \in D \mid dist(p,q) \leq Eps\}.$$

Tính chất: Nói chung vùng lân cận của điểm biên có số đối tượng ít hơn đáng kể so hơn điểm nhân.



Hình 4.2. Các đối tượng nằm trong điểm nhân p với bán kính Eps

Điểm p muốn nằm trong cụm C thì $N_{Eps}(p)$ phải có tối thiểu MinPts điểm.

Số điểm tối thiểu được chọn là bao nhiêu cũng là bài toán khó, vì: Nếu số điểm tối thiểu lớn thì chỉ những điểm nằm thực sự trong cụm C mới đạt đủ tiêu chuẩn, trong khi đó những điểm nằm ngoài biên của cụm không thể đạt được điều đó. Ngược lại, nếu số điểm tối thiểu là nhỏ thì mọi điểm sẽ rơi vào một cụm.

Theo ĐN trên, chỉ những điểm thực sự nằm trong cụm mới thoả điều kiện là điểm thuộc vào cụm. Những điểm nằm ở biên của cụm không thoả.

Để tránh được điều này, có thể đưa ra một tiêu chuẩn khác để định nghĩa: Nếu một điểm p muốn thuộc một cụm C phải tồn tại một điểm q mà p ∈ NEps(q) và số điểm trong NEps(q) phải lớn hơn số điểm tối thiểu như định nghĩa sau:

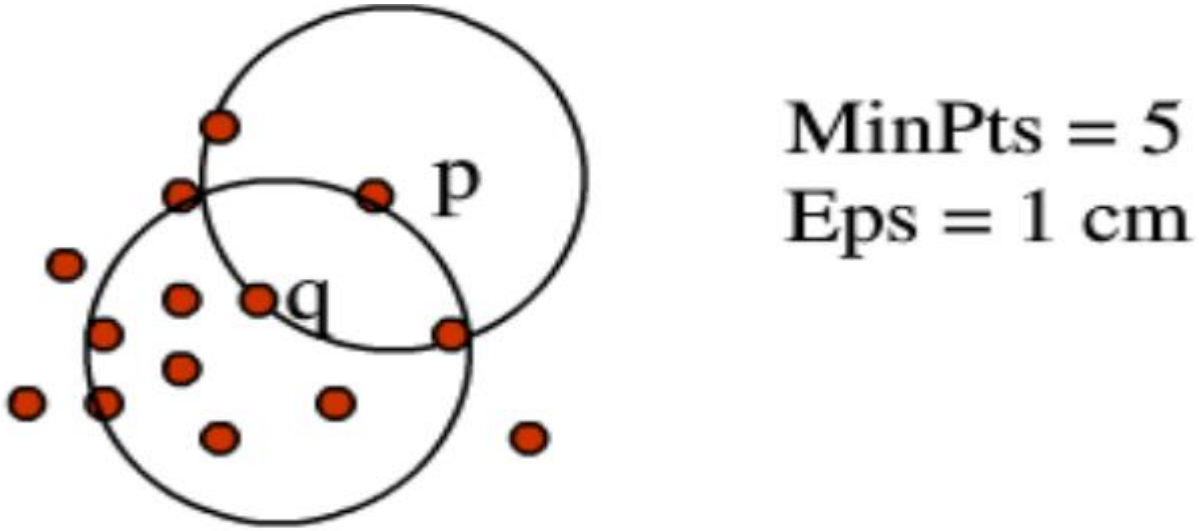
<u>Định nghĩa 2</u> : Đến được trực tiếp theo mật độ (Directly Density - reachable)

Một điểm p được gọi là đến được trực tiếp từ điểm q với ngưỡng Eps nếu :

- p ∈NEps(q) q
- || NEps(q)|| ≥ MinPts (Điều kiện nhân)

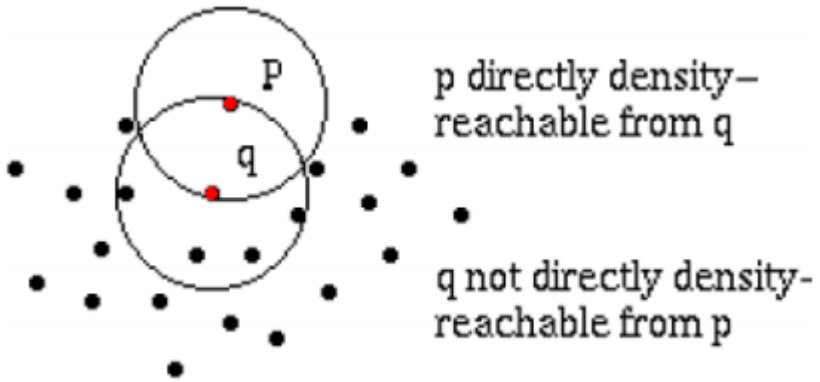
Tính chất:

- Nếu p, q đều là điểm nhân quan hệ directly density-reachable đối xứng nghĩa là p tới được trực tiếp theo mật độ từ q và ngược lại.



Hình 4.3 Điểm nhân có quan hệ quan hệ directly density-reachable đối xứng

- Nếu trong p, q có một điểm nhân (core point), một điểm biên như hình dưới thì chỉ điểm biên có tới được trực tiếp theo mật độ từ điểm nhân mà không có chiều ngược lại (bất đối xứng).



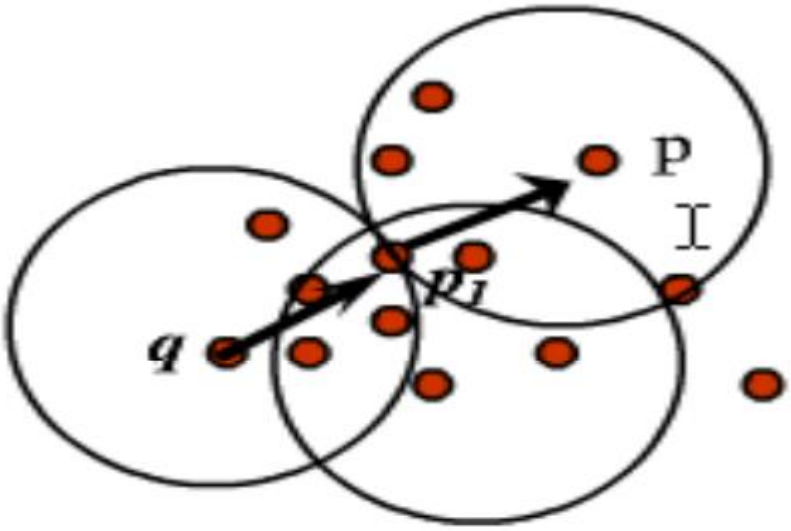
Hình 4.4 Điểm nhân có quan hệ quan hệ directly density-reachable không đối xứng

<u>Định nghĩa 3</u> : Đến được mật độ (Density - Reachable)

Đối tượng p tới được theo mật độ (density-reachable) thỏa Eps, MinPts từ đối tượng q nếu tồn tại một dãy p_1, p_2, ..., p_n (p_1 =q, pn= p) sao cho p_{i+1} tới được theo mật độ trực tiếp từ p_i.

Tính chất:

- Quan hệ density-reachable là sự mở rộng của directly density-reachable.
- Quan hệ density-reachable có tính bắt cầu
- Nếu p, q đều là điểm nhân (core point) thì quan hệ density-reachable là đối xứng nghĩa là p tới được theo mật độ từ q và ngược lại.
- Nếu p, q đều là điểm biên (border point) thì p không tới được theo mật độ từ q và ngược lại.
- Nếu trong p, q có một điểm nhân (core point), một điểm biên như hình dưới thì chỉ điểm biên có tới được theo mật độ từ điểm nhân mà không có chiều ngược lại (bất đối xứng).



Hình 4.5. Quan hệ đến được theo mật độ

Hai điểm biên của một cụm C có thể không đến được nhau vì cả hai có thể đều không thoả mãn điều kiện nhân.

Phải tồn tại một điểm nhân trong C mà cả hai điểm đều có thể đến được từ điểm đó.

Để cho thuận tiện chúng ta có định nghĩa liên thông mật độ.

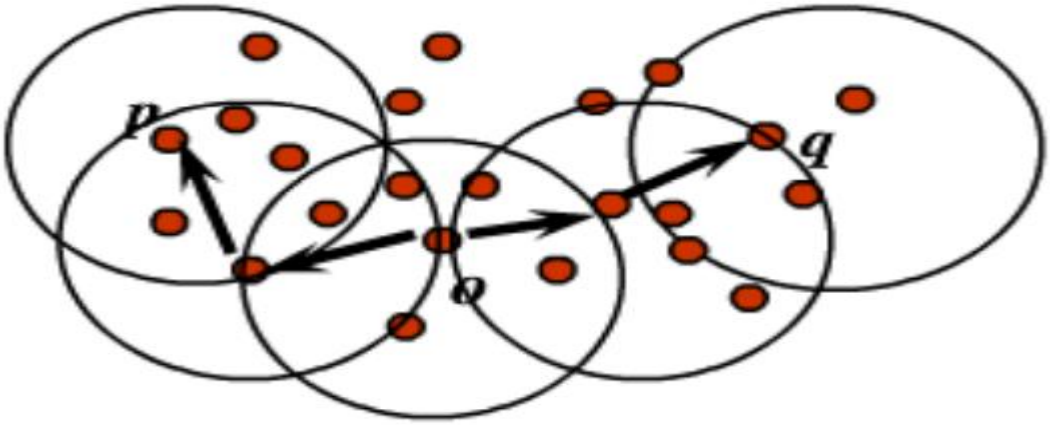
<u>Định nghĩa 4</u> : Liên thông mật độ (Density - Reachable)

Đối tượng p kết nối theo mật độ (density-connected) thỏa Eps, MinPts với đối tượng q tồn tại đối tượng O sao cho cả p và q đều tới được theo mật độ từ O.

Tính chất:

- Đối với các đối tượng tới được theo mật độ với đối tượng khác, quan hệ density-connected có tính phản xạ

- Quan hệ density-connected có tính đối xứng.



Hình 4.6 Quan hệ liên thông theo mật độ

Định nghĩa 5 : Cụm (Clustering)

Cho cơ sở dữ liệu D, cụm C thỏa Eps và MinPts là tập con khác rỗng của D thỏa 2 điều kiện sau:

- ∀p,q: nếu p ∈C và q liên hệ theo mật độ từ p thỏa Eps và MinPts thì q∈C.

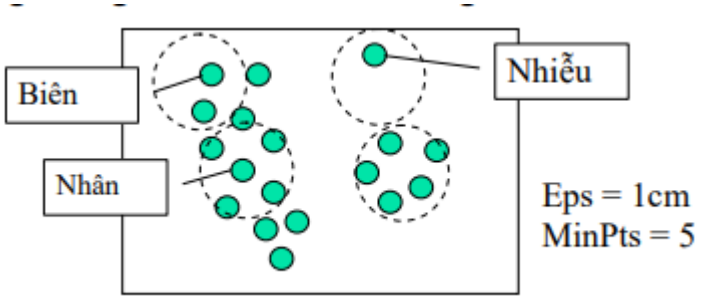
- ∀p,q∈C: p kết nối theo mật độ với p thỏa Eps và MinPts.

Cụm C thỏa định nghĩa trên sẽ có ít nhất MinPts đối tượng vì lý do sau: C phải có ít nhất một đối tượng p (C khác rỗng), p phải liên hệ mật độ với bản thân nó thông qua một đối tượng o (điều kiện 2 của định nghĩa 5). Vì vậy, O là điểm nhân và vùng lân cận Eps của O có ít nhất MinPts đối tượng (do p có liên hệ mật độ từ O).

Định nghĩa 6 : Dữ liệu nhiễu (Noise)

Cho các cụm C_1 ,. . ., C_k của cơ sở dữ liệu D với các tham số Eps_i and $MinPts_i$, (i = 1, . . ., k). Tập nhiễu là tập các đối tượng thuộc D nhưng không thuộc bất kỳ cụm C_i nào.

$$noise = \{p \in D \mid \forall i: p \notin Ci\}.$$



Hình 4.7 Cụm và nhiễu

3. Thuật toán DBSCAN:[9]

Thuật toán DBSCAN là thuật toán gom tụm các đối tượng trong cơ sở dữ liệu không gian thỏa mãn định nghĩa 5 và 6.

Ứng với thông số Eps, MinPts cho trước, DBSCAN xác định một cụm thông qua 2 bước:

1) chọn đối tượng bất kỳ thỏa mãn điều kiện điểm nhân làm đối tuợng hạt giống;

2) tìm các đối tượng tới được theo mật độ từ đối tượng hạt giống. Trong trường hợp lý tưởng thì ứng với mỗi cụm, cần phải xác định được thông số Eps, MinPts ít nhất một đối tượng thuộc cụm; sau đó, tìm tất cả các đối tượng cho từng cụm. Tuy nhiên, không dễ gì xác định được các thông tin trên nhanh chóng và chính xác nên DBSCAN sử dụng

thông số Eps, MinPts của cụm có mật độ ít dày đặc nhất làm thông số chung cho tất cả các cụm. Các thông số này được xác định thông qua thuật toán heuristic.

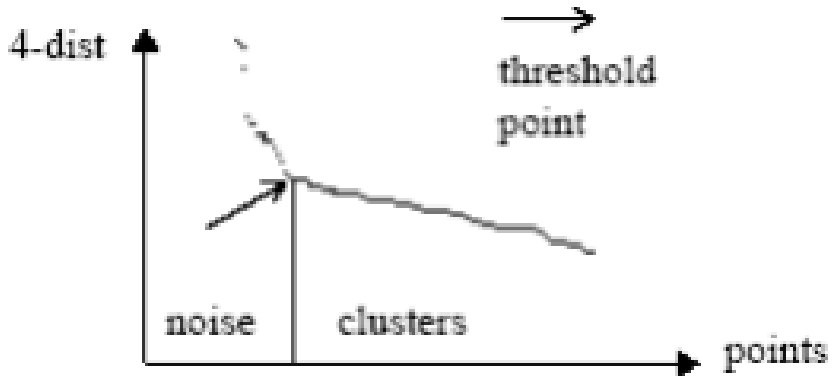
Để tìm một cụm, thuật toán DBSCAN bắt đầu từ một đối tượng bất kỳ và sau đó tìm tất cả các đối tượng tới được theo mật độ thỏa Eps and MinPts từ đối tượng p. Nếu p là điểm nhân, bước trên sinh ra một cụm thỏa Eps and MinPts. Nếu p là điểm biên thì không tìm được đối tượng tới được theo mật độ từ p, DBSCAN duyệt đối tượng tiếp theo trong cơ sở dữ liệu. Vì sử dụng chung thông số Eps và MinPts cho tất cả các cụm nên DBSCAN có thể kết hợp 2 cụm thỏa định nghĩa 5 thành một cụm nếu hai cụm gần nhau. Khoảng cách giữa 2 tập đối tượng S_1 và S_2, ký hiệu dist (S_1, S_2) là giá trị khoảng cách nhỏ nhất của 2 đối tượng p, q thuộc S_1, S_2: dist (S_1, S_2) = min{dist(p,q) | p∈S_1, q∈S_2}. Hai tập đối tượng trong cùng một cụm có thể tách nhau nếu khoảng cách giữa chúng lớn hơn Eps. Vì vậy, thuật toán DBSCAN có thể cần phải được gọi đệ qui để loại trừ cụm có số đối tượng lớn. Vấn đề này không khó giải quyết vì DBSCAN cung cấp thuật toán cở bản rất hiệu quả.

4. Xác định thông số Eps and MinPts:

Thông số Eps và MinPts cho thuật toán DBSCAN có thể được xác định bằng tay hoặc thông qua thuật toán heuristics xác định thông số Eps và MinPts cho cụm có mật độ ít dày đặc nhất. Thuật toán này dựa trên 2 quan sát sau: Gọi d là khoảng cách giữa đối tượng p và đối tượng gần nhất thứ k thì vùng lân cận d của đối tượng p chứa k+1 đối tượng (hoặc nhiều hơn k+1 đối tượng khi nhiều đối tượng có cùng khoảng cách đến p). Thay đổi giá trị k không dẫn đến thay đổi lớn giá trị của d trừ khi k đối tượng này cùng nằm xấp xỉ trên một đường thẳng.

Với giá trị k cho trước, hàm k-dist là khoảng cách từ một đối tượng đến đối tượng gần nhất thứ k. Tạo đồ thị sorted k-dist bằng cách sắp xếp các đối tượng theo giá trị kdist giảm dần. Nếu chọn một đối tượng bất kỳ p, đặt thông số Eps là k-dist(p) và MinPts là k, các đối tượng có khoảng cách với p nhỏ hơn hoặc bằng giá trị k-dist sẽ thuộc về cụm tạo bởi đối tượng p. Nếu tìm được đối tượng ngưỡng với giá trị k-dist lớn nhất ở trong cụm mỏng nhất của D, ta sẽ tìm được giá trị thông số mong muốn. Đối tượng ngưỡng này là đối tượng đầu tiên trong vùng lõm đầu tiên của đồ thị sorted k-dist (xem hình 5). Tất cả

các đối tượng với giá trị k-dist cao hơn (bên trái đối tượng ngưỡng) được xem là nhiễu. Các đối tượng còn lại (bên phải đối tượng ngưỡng) sẽ thuộc về một cụm nào đó.



Hình 4.8 Đồ thị sorted 4-dsit

Nói chung, khó xác định tự động được vùng lõm đầu tiên nhưng với người dùng có thể xác định được khá dễ dàng bằng cách quan sát trên đồ thị.

DBSCAN cần hai thông số: Eps và MinPts. Tuy nhiên, kết quả thí nghiệm cho thấy đồ thị k-dist với k > 4 không khác biệt nhiều so với đồ thị sorted 4-dist nhưng chi phí tính toán lại tăng đáng kể. Vì vậy, ta có thể loại trừ thông số MinPts bằng cách cho MinPts là 4.

Tóm lại, thông số Eps và MinPts cho thuật toán DBSCAN có thể xác định qua các bước sau:

- Hệ thống tính toán và hiển thị đồ thị sorted 4-dist.

- Nếu người dùng có thể ước tính số phần trăm nhiễu thì hệ thống sẽ đề nghị đối tượng ngưỡng theo số phần trăm nhiễu do người dùng nhập vào.

- Người dùng có thể chấp nhận đối tượng ngưỡng được đề nghị hoặc chọn đối tượng khác làm đối tượng ngưỡng. Giá trị 4-dist của đối tượng ngưỡng được sử dụng làm thông số Eps cho thuật toán DBSCAN.

5. Đánh giá thuật toán:[2]

5.1. Ưu điểm:

- Có thể phát hiện các cluster có hình dạng bất kỳ
- Chỉ yêu cầu một hàm đo khoảng cách và hai tham số đầu vào: Eps và MinPts.
- Cho ra kết quả tốt và thực thi hiệu quả trên nhiều tập dữ liệu.

5.2. Khuyết điểm:

- Không thích hợp cho việc tìm các cluster trong CSDL cực lớn.
- Nếu tập dữ liệu có mật độ thay đổi lớn, thuật toán quản lý kém hiệu quả.
- Độ phức tạp trung bình của mỗi truy vấn là O(logn).
- Độ phức tạp của thuật toán là O(nlogn), n là kích thước tập dữ liệu.

6. Kết luận:

Như vậy chương này đã trả lời được cho câu hỏi thuật toán DBSCAN là gì. Chương này mình đã nêu chi tiết về thuật toán này bao gồm các định nghĩa liên quan. Mình còn trình bày một thuật toán dùng để xác định các thông số chung Eps và MinPts

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Phân cụm dữ liệu là nhiệm vụ quan trọng trong khai phá dữ liệu, thu hút sự quan tâm của nhiều nhà nghiên cứu. Các kỹ thuật phân cụm đã và đang được ứng dụng thành công trong nhiều lĩnh vực khoa học, đời sống xã hội. Hiện nay, do sự phát triển không ngừng của công nghệ thông tin và truyền thông, các hệ thống cơ sở dữ liệu ngày càng đa dạng, và tăng trưởng nhanh cả về chất lẫn về lượng. Hơn nữa, nhu cầu về khai thác các tri thức từ các cơ sở dữ liệu này ngày càng lớn. Vì vậy, việc nghiên cứu các mô hình dữ liệu mới, áp dụng các phương pháp khai phá dữ liệu, trong đó có kỹ thuật phân cum dữ liệu là việc làm rất cần thiết có nhiều ý nghĩa.

Trong tiểu luận này, trước tiên em đã trình bày những hiểu biết của mình về phân cụm dữ liệu sau đó là phần nội dung chính của đồ án: Các thuật toán dùng để phân cụm dữ liệu. Ở phần nội dung em đã trình bày được thế nào là bài toán phân cụm dữ liệu , các ứng dụng, các kiểu dữ liệu có thể phân cụm, các độ đo độ tương tự. Đặc biệt, em tập trung đi sâu nghiên cứu về kỹ thuật phân cụm dữ liệu phân hoạch với thuật toán K-means, kỹ thuật phân cụm dữ liệu phân cấp với thuật toán điển hình của kỹ thuật này là BIRCH và kỹ thuật phân cụm dữ liệu dựa trên mật độ với thuật toán DBSCAN.

Hướng phát triển:

-Tiếp tục nghiên cứu các kỹ thuật khác để phân cụm dữ liệu nhằm nâng cao hiệu suất khai phá dữ liệu trên hệ thống dữ liệu lớn phân tán

-Áp dụng các kỹ thuật phân cụm dữ liệu vào lĩnh vực thương mại điện tử, chính phủ điện tử,…

DANH MỤC TÀI LIỆU THAM KHẢO

Tiếng nước ngoài:

[1] Jiawei Han and Micheline Kamber. “Data Mining: Concepts and Techniques, 2nd edition”. The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor, Morgan Kaufmann Publishers, ISBN 1-55860-901-6, 2006.

Tiếng Việt:

[2] Đặng Thị Thu Hiền, Cluster Analysis, Viện nghiên cứu cao cấp về toán, 2016

[3] Đào Minh Tùng, Phân cụm đa mức web bằng k-means dựa trên chủ để ẩn và thực nghiệm đánh giá, Trường đại học Công nghệ, 2011

[4]Lương Bá Hợp, Thuật toán phân cụm dữ liệu phân cấp BIRCH, 2014

[5] Nguyễn Huỳnh Anh Duy, Giải thuật DBSCAN cải tiến cho phân cụm các tập dữ liệu lớn, Trường đại học Cần Thơ, 2014

[6]Hoàng Văn Dũng, Khai phá dữ liệu web bằng kĩ thuật phân cụm, Trường đại học sư phạm Hà Nội, 2007.

[7]Phạm Ngọc Sâm, Phân cụm dữ liệu Bài toán và các giải thuật theo tiếp cận phân cấp, Trường Đại học Hải Phòng, 2013

[8]Đạt Hoàng, Thuật toán K-means Clustering, Đại học Bách khoa Hà Nội, 2020

[9]Nguyễn Hoàng Phương, Thuật toán DBSCAN, 2018