****

**BÁO CÁO MÔN HỌC**

**Trí Tuệ Nhân Tạo**

**ĐỀ TÀI: THUẬT TOÁN K-NEAREST NEIGHBORS**

**TPHCM T10/2023**

****

**Trí Tuệ Nhân Tạo**

**ĐỀ TÀI: THUẬT TOÁN K-NEAREST NEIGHBORS**

**GV hướng dẫn:** Trần Đình Toàn

|  |  |  |
| --- | --- | --- |
| STT | Họ và Tên | MSSV |
| 1 | Nguyễn Thị Thùy Linh | 2001210927 |
| 2 | Nguyễn Ngọc Quân | 2001210779 |
| 3 | Phùng Huỳnh Thanh Ngân | 2001210235 |

**BẢNG PHÂN CÔNG LÀM VIỆC NHÓM**

|  |  |  |  |
| --- | --- | --- | --- |
| **MSSV** | **Tên** | **Nhiệm vụ** | **Đánh giá** |
| 2001210927 | Ngô Thị Thùy Linh | Tìm hiểu code, ưu nhược điểm, ứng dụng, demo thuật toán | 100% |
| 2001210779 | Nguyễn Ngọc Quân | Cơ chế hoạt động của thuật toán, ví dụ | 100% |
| 2001210235 | Phùng Huỳnh Thanh Ngân | Làm powerpoint, word, tìm hiểu về khái niệm định nghĩa thuật toán | 100% |

**LỜI CAM ĐOAN**

Báo cáo tiểu luận môn Trí Tuệ Nhân Tạo này là công trình nghiên cứu của nhóm 5 gồm sinh viên gồm:

* Ngô Thị Thùy Linh : Trưởng nhóm
* Nguyễn Ngọc Quân : Thành viên
* Phùng Huỳnh Thanh Ngân: Thành viên

Các số liệu trong bài tiểu luận này được nhóm nghiên cứu thu thập, phân tích một cách khách quan, trung thực, có nguồn gốc rõ ràng và có sự chọn lọc kỹ càng, với sự giúp đỡ nhiệt tình của Gv Trần Đình Toàn.

Chúng em xin cam đoan rằng:

* Nội dung trong bài là trung thực và không có bất kỳ sự sao chép hay sử dụng để bảo vệ một học vị nào.
* Tất cả những sự giúp đỡ cho việc xây dựng cơ sở lý luận cho bài luận đều được trích dẫn đầy đủ và ghi rõ nguồn gốc rõ ràng và được phép công bố.
* Báo cáo tiểu luận này được xây dựng dựa trên cơ sở lý luận và thực tiễn, với sự tham khảo của các tài liệu khoa học có liên quan. Danh mục tài liệu tham khảo được liệt kê đầy đủ trong phần cuối của báo cáo.

**LỜI CẢM ƠN**

Nhóm 5 xin gửi lời cảm ơn sâu sắc đến khoa Công Nghệ Thông Tin trường Đại Học Công Thương TPHCM đã tạo điều kiện thuận lợi cho nhóm được học tập và hoàn thành đề tài “Thuật toán K- Nearest Neighbor” cho bài báo cáo. Đặc biệt, chúng em xin gửi lời cảm ơn chân thành nhất đến thầy/cô Gv.Trần Đình Toàn đã tận tình chỉ dạy, truyền đạt kiến thức cho nhóm chúng em trong quá trình làm bài.

Nhóm chúng em đã cố gắng vận dụng những kiến thức đã học được và tìm tòi thêm nhiều thông tin để hoàn thành bài tiểu luận này. Tuy nhiên, do kiến thức còn hạn chế và không có nhiều kinh nghiệm trên thực tiễn nên khó tránh khỏi những thiếu sót trong bài làm. Rất kinh mong thầy cho nhóm thêm những góp ý để bài tiểu luận của em được hoàn thiện hơn. Chúng em xin chân thành cảm ơn!

NHÓM 5

**MỤC LỤC**

[*Mở đầu* 6](#_Toc149089346)

[Nội dung 7](#_Toc149089347)

[I Giới thiệu thuật toán K-Nearest Neighbors 7](#_Toc149089348)

[**1.Định nghĩa** 7](#_Toc149089349)

[2. Ý tưởng 8](#_Toc149089350)

[II.Cách thức hoạt động của KNN 9](#_Toc149089351)

[**1.** **Cách hoạt động của KNN** 9](#_Toc149089352)

[2.Ví dụ minh họa 9](#_Toc149089353)

[III.Demo: 11](#_Toc149089354)

[IV Ưu điểm và nhược điểm 14](#_Toc149089355)

[V ỨNG DỤNG 15](#_Toc149089356)

[VI. SO SÁNH KNN & K MEANS 15](#_Toc149089357)

[VII. Cài đặt thuật toán 16](#_Toc149089358)

[Tài Liệu Tham Khảo 16](#_Toc149089359)

**MỤC LỤC HÌNH ẢNH**

[Hình 1.1:Minh họa 1 9](#_Toc149398636)

[Hình 2.1:Ví dụ 1 12](file:///C:\Users\Admin\Downloads\Báo%20cáo%20l2.docx#_Toc149398637)

[Hình 2.2:Ví dụ 2 13](file:///C:\Users\Admin\Downloads\Báo%20cáo%20l2.docx#_Toc149398638)

[Hình 3.1: Đồ thị mẫu 14](#_Toc149398639)

[Hình 3.2:Tạo lớp hàng xóm 14](#_Toc149398640)

[Hình 3.3:Tính toán khoảng cách 15](#_Toc149398641)

[Hình 3.4:Công thức tính khoảng cách 15](#_Toc149398642)

[Hình 3.5:Lấy hàng xóm gần 16](#_Toc149398643)

[Hình 3.6:Tìm và lấy nhã phổ biến nhất 16](#_Toc149398644)

[Hình 3.7:Phân loại nhãn 17](#_Toc149398645)

# ***Mở đầu***

***Lý do chọn đề tài:***

**Đơn giản và dễ triển khai:** Thuật toán K-NN là một thuật toán đơn giản và dễ hiểu. Không có quá nhiều giả định phức tạp hoặc tham số cần điều chỉnh. Do đó, nó có thể được triển khai một cách nhanh chóng và dễ dàng, đặc biệt là đối với các bài toán đầu tiên trong lĩnh vực học máy.

**Tính linh hoạt và không giả định mô hình**: K-NN không giả định bất kỳ mô hình hay phân phối cụ thể nào về dữ liệu. Điều này giúp nó phù hợp với nhiều loại dữ liệu và bài toán khác nhau. Nó có thể được sử dụng cả cho bài toán phân loại và dự đoán và không yêu cầu kiến thức chuyên sâu về lĩnh vực đang xử lý.

**Khả năng xử lý dữ liệu phi cấu trúc**: K-NN không yêu cầu dữ liệu có cấu trúc đặc biệt hay tiền xử lý phức tạp. Nó có thể xử lý dữ liệu có nhiễu, dữ liệu thiếu, và các biến số có thể có ý nghĩa khác nhau. Điều này làm cho K-NN trở thành một lựa chọn hợp lý khi bạn có dữ liệu phi cấu trúc hoặc không biết trước về tính chất của dữ liệu.

**Tính nhất quán và ổn định:** K-NN cho kết quả nhất quán và ổn định đối với các bộ dữ liệu lớn. Với số lượng hàng xóm lớn, ước lượng của K-NN trở nên ổn định hơn và ít bị ảnh hưởng bởi nhiễu. Điều này làm cho K-NN phù hợp khi làm việc với các tập dữ liệu lớn hoặc tập dữ liệu có tính biến động cao.

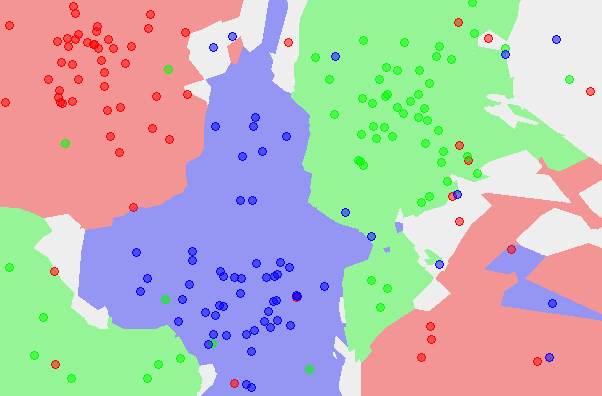
Vì những lí do trên, Nhóm chúng em quyết định chọn KNN làm đề tài cho báo cáo này.

# **Nội dung**

# I Giới thiệu thuật toán K-Nearest Neighbors

## **1.Định nghĩa**

Trong lý thuyết học máy và xác suất thống kê, thuật toán k - mẫu gần nhất (k-nearest neighbors) là một thuật toán phi tham số (non - parametric), học giám sát (supervised learning), lười (lazy learning), sử dụng cho cả hai mục đích là hồi quy và phân lớp.



Hình 1.1:Minh họa 1

Thuật toán được Evelyn Fix và Joseph Hodges công bố đầu tiên vào năm 1951 và được Thomas Cover mở rộng sau này vào năm 1967.

Tính "lười" của k-NN được thể hiện ở 2 tính chất: hàm mục tiêu được xấp xỉ tại các địa phương, và k-NN không có giai đoạn huấn luyện. Mọi tính toán được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới.

Bản chất của k-NN chính là tính toán khoảng cách từ điểm cần xét đến k điểm lân cận, nên nếu các features có scale khác nhau, hoặc có đơn vị khác nhau, ta có thể thực hiện chuẩn hoá (normalize) dữ liệu để có thể đẩy mạnh hiệu suất của thuật toán.

Tuỳ thuộc vào mục đích phân loại hay hồi quy mà đầu ra của k-NN sẽ khác nhau:

Với bài toán phân loại, output của k-NN là lớp của bản ghi cần phân loại. Một bản ghi được xác định lớp (class) theo nguyên tắc biểu quyết đa số. Theo đó, lớp của bản ghi được xác định theo k bản ghi gần nhất với bản ghi đang được xét theo nguyên tắc số đông. k đảm bảo điều kiện là số nguyên dương, thường đủ nhỏ.

Với bài toán hồi quy, output của k-NN chính là giá trị dự đoán của của hàm mất mát tại bản ghi đang được xét. Giá trị dự đoán này là trung bình nhãn của k mẫu gần nhất.

Dữ liệu huấn luyện là tập hợp những điểm dữ liệu đã được gắn nhãn (với bài toán phân lớp), và tập các điểm dữ liệu với giá trị tương ứng (với bài toán hồi quy). Một đặc trung của thuật toán k-NN chính là sự nhạy cảm với hình dạng của dữ liệu.

# **2. Ý tưởng**

Thuật toán KNN cho rằng những dữ liệu tương tự nhau sẽ tồn tại **gần nhau** trong một không gian, từ đó công việc của chúng ta là sẽ tìm k điểm gần với dữ liệu cần kiểm tra nhất. Việc tìm khoảng cách giữa 2 điểm củng có nhiều công thức có thể sử dụng, tùy trường hợp mà chúng ta lựa chọn cho phù hợp

**3. Phương pháp**

Phương pháp K-Nearest Neighbors (K-NN) là một thuật toán học máy không parametric, tức là nó không giả định bất kỳ mô hình hay phân phối cụ thể nào về dữ liệu. Thay vào đó, nó dựa vào việc so sánh và lựa chọn các điểm dữ liệu gần nhất để thực hiện phân loại hoặc dự đoán.

Dưới đây là các bước cơ bản để triển khai thuật toán K-NN:

1. Chuẩn bị dữ liệu huấn luyện: Thuật toán K-NN yêu cầu một tập dữ liệu huấn luyện có nhãn, trong đó mỗi điểm dữ liệu đã được gán nhãn. Nếu bạn đang làm việc trên bài toán phân loại, nhãn sẽ là các nhãn lớp (class labels). Nếu bạn đang làm việc trên bài toán dự đoán, nhãn có thể là các giá trị số (ví dụ: dự đoán giá trị của một thuộc tính).
2. Xác định số hàng xóm (K): K là một số nguyên dương được xác định trước, thường là một số lẻ. Giá trị của K quyết định số lượng hàng xóm sẽ được lựa chọn để thực hiện phân loại hoặc dự đoán cho một điểm dữ liệu mới.
3. Đo độ tương đồng: Để xác định các điểm dữ liệu gần nhất, chúng ta cần đo độ tương đồng (hoặc khoảng cách) giữa các điểm dữ liệu trong tập huấn luyện và điểm dữ liệu mới. Khoảng cách Euclide là một phương pháp phổ biến để tính toán độ tương đồng, nhưng có thể sử dụng các phương pháp đo khác như khoảng cách Manhattan, khoảng cách Cosine, hoặc hàm kernel.
4. Lựa chọn hàng xóm: Dựa trên độ tương đồng tính toán, chúng ta chọn K điểm dữ liệu gần nhất với điểm dữ liệu mới. Các điểm dữ liệu này được xem là "hàng xóm" trong thuật ngữ K-NN.
5. Phân loại hoặc dự đoán: Cuối cùng, chúng ta sử dụng nhãn của các hàng xóm để phân loại hoặc dự đoán cho điểm dữ liệu mới. Đối với bài toán phân loại, chúng ta thực hiện đa số phiếu bầu (voting) để xác định nhãn cuối cùng dựa trên nhãn của K hàng xóm. Đối với bài toán dự đoán, chúng ta có thể tính trung bình giá trị của các hàng xóm hoặc sử dụng các phương pháp khác tùy thuộc vào bài toán cụ thể.

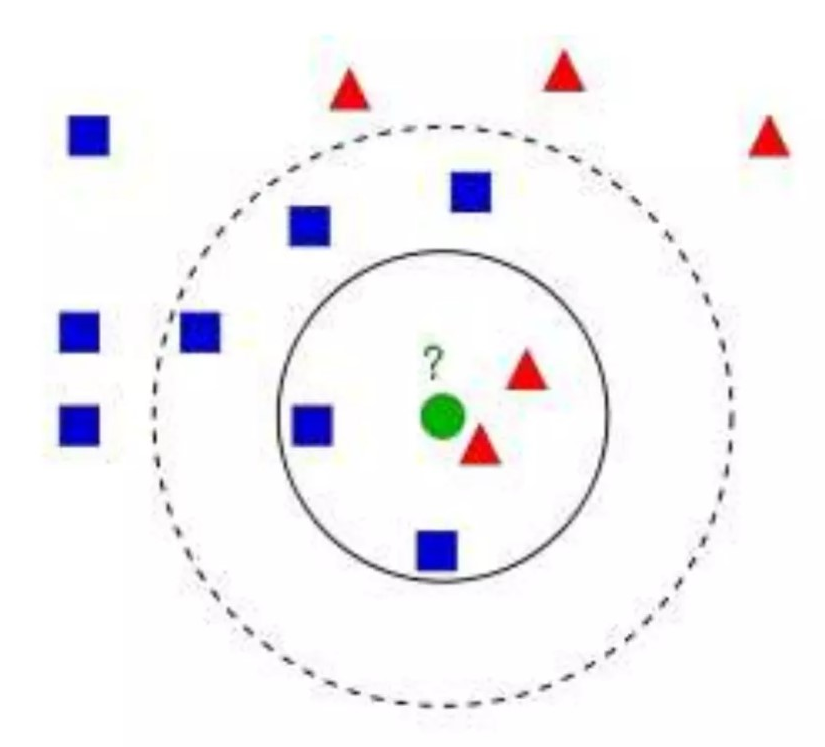
Điều quan trọng khi triển khai thuật toán K-NN là lựa chọn đúng giá trị K và phương pháp đo độ tương đồng phù hợp với bài toán. Ngoài ra, việc chuẩn bị và tiền xử lý dữ liệu cũng có thể ảnh hưởng đáng kể đến kết quả của thuật toán.

# **II.Cách thức hoạt động của KNN**

## **Cách hoạt động của KNN**

Bước 1: Xác định tham số K= số láng giềng gần nhất.  
Bước 2: Tính khoảng cách đối tượng cần phân lớp với tất cả các đối tượng trong  
training data.  
Bước 3: Sắp xếp khoảng cách theo thứ tự tăng dần và xác định K láng giềng gần  
nhất với đối tượng cần phân lớp  
Bước 4: Lấy tất cả các lớp của K láng giềng gần nhất.  
Bước 5: Dựa vào phần lớn lớp của K để xác định lớp cho đối tượng cần phân lớp.

# **2.Ví dụ minh họa**



Hình 2.1:Ví dụ 1

**Vidu 1**

Giả sử bài toán được đặt ra: mình mới quen một người bạn, tuy nhiên mình là fan của Us-Uk vậy nên mình cần biết người bạn này có phải là fan của K-Pop hay không. Qua thời gian tìm hiểu mình đã thu thập được một số dữ liệu và đã biểu hiện dưới dạng hình vẽ trên.

Ta dễ dàng nhìn thấy có hai loại: hình vuông màu xanh biểu diễn cho những người là fan của K-pop, tam giác màu đỏ biểu diễn cho những người không là fan của K-pop, hình tròn màu xanh là người bạn mình muốn biết có phải là fan K-pop hay không, khoảng cách giữa chấm tròn và các điểm còn lại biểu diễn độ thân thiết của bạn đó với những người bạn.  
 Phương pháp đơn giản nhất để kiểm tra xem bạn đó chơi thân với người bạn nào nhất, tức là tìm xem điểm gần chấm xanh thuộc class nào (hình vuông hay tam giác). Từ hình trên ta dễ dàng nhận thấy điểm gần chấm xanh nhất là hình tam giác màu đỏ, do đó nó sẽ được phân vào lớp tam giác màu đỏ.

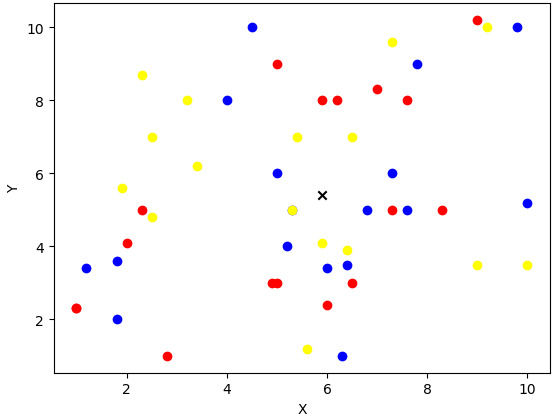
Có một vấn đề trong phương pháp trên, xung quanh cấm xanh xuất hiện rất nhiều hình vuông màu xanh nên việc xét điểm gần nhất là chưa khả thi. Vì vậy, ta sẽ xét k điểm gần nhất. Giả sử, ta lấy K=3, dựa theo hình trên ta dễ dàng nhận ra có hai  
hình tam giác đỏ và một hình vuông xanh có khoảng cách gần chấm xanh nhất, do đó chấm xanh được phân vào lớp tam giác đỏ. Lấy K=7, ta có năm hình vuông xanh và hai hình tam giác đỏ, lúc này chấm xanh được xếp vào lớp hình vuông xanh. Trường hợp lấy K=4, ta nhận thấy sẽ có hai hình vuông xanh và hai hình tam giác đỏ, đây là trường hợp có điểm bằng nhau, với trường hợp này KNN sẽ xử lý bằng cách so sánh tổng khoảng cách của các hình gần nhất với điểm ta đang xét.  
 Do xuất hiện trường hợp có điểm bằng nhau, vì vậy người ta thường chọn k là số lẻ. Đó cũng là ý tưởng của KNN.



Hình 2.2:Ví dụ 2

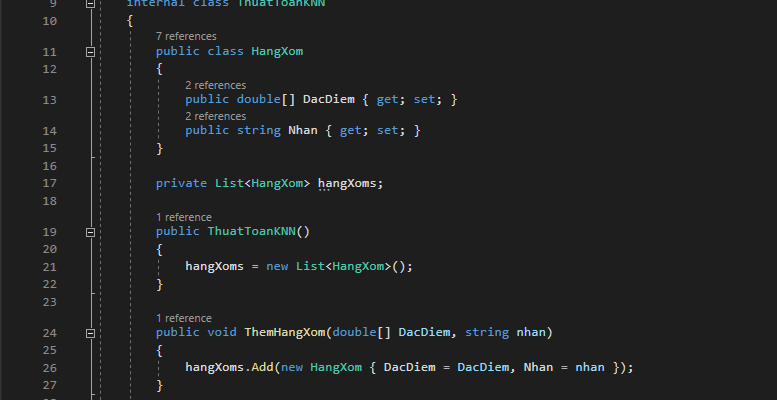
Hình trên là bài toán phân lớp với ba lớp: đỏ, lam, lục. Mỗi điểm dữ liệu mới sẽ được gán nhãn theo màu của điểm đó mà nó thuộc về. Trong hình này, chú ý vùng khoanh tròn màu vàng, ta nhận thấy rằng điểm màu lục nằm giữa hai vùng lớn với nhiều dữ liệu đỏ và lam, điểm này rất có thể là nhiễu dẫn đến việc dữ liệu test nếu rơi vào vùng này sẽ có nhiều khả năng cho kết quả sai lệch.

# **III.Demo:**



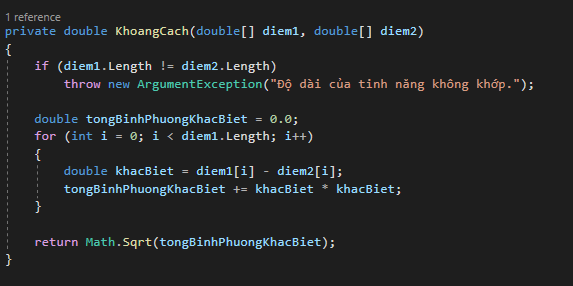
Hình 3.1: Đồ thị mẫu

Vẽ đồ thị có 50 điểm gồm 3 lớp xanh, đỏ, vàng với các tọa độ (x,y) như hình bằng ngôn ngữ python và một tọa độ x bất kỳ minh họa cho thuật toán K-nn.



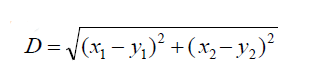
Hình 3.2:Tạo lớp hàng xóm

Đầu tiên tạo lớp Hàng Xóm với các thuộc tính Đặc điểm(là các bộ dữ liệu dùng để so sánh, phân biệt các điểm hàng xóm) và Nhãn(đặt tên , cho biết tên của các nút để phân loại) cho từng hàng xóm. Khởi tạo lớp và khai báo danh sách hàng xóm . Thêm các hàng xóm vào danh sách với các đặc điểm và nhãn của chúng.



Hình 3.3:Tính toán khoảng cách

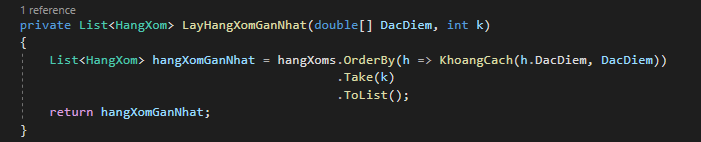
Các hàng xóm đã có trong danh sách ta tiến hành tính khoảng cách độ dài hoặc chênh lệch từ điểm mới đến tất các hàng xóm có trong danh sách bằng công thức sau:



Hình 3.4:Công thức tính khoảng cách

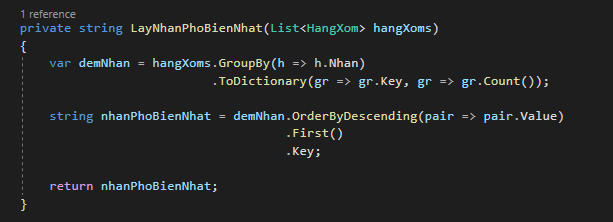
Trước hết ta cần kiểm tra xem độ dài của điểm 1 và điểm 2 có bằng nhau không, nếu không bằng có nghĩa là 2 điểm không có cùng số chiều và ném ra 1 ngoại lệ. Không thỏa điều kiện trên thì ta đến bước tiếp theo là tính khoảng cách Euclidean(tên Tiếng Anh :Euclidean distance, giữa hai điểm trong không gian Euclid là độ dài của đoạn thẳng nối hai điểm đó).

Vòng lặp for để duyệt các phẩn tử có trong điểm 1 và điểm 2. Tại mỗi lần lặp, phương thức tính sự khác biệt được tính bằng hiệu của chúng và sự khác biệt đó bằng tổng bình phương của 2 điểm, cuối cùng thì căn bậc 2 lên ta thu được khoảng cách giữa 2 điểm.



Hình 3.5:Lấy hàng xóm gần

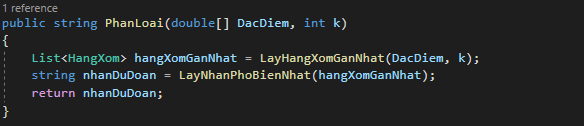
Sau khi có được khoảng cách ta sắp xếp các hàng xóm theo thứ tự tăng dần bằng phưng thức OrderBy.K (là số hàng xóm gần nhất ) lấy k số hàng xóm gần nhất bằng phương thức Take(k).Trả về danh sách hàng xóm gần nhất sau khi tuyển chọn



Hình 3.6:Tìm và lấy nhã phổ biến nhất

Sau khi đã có danh sách các hàng xóm gần nhất ta tiến hành tìm các nhãn phổi biến nhất. Đầu tiên, ta nhóm lại thành một tập hợp dựa trên giá trị của nhãn định danh. Sau đó, chúng ta dùng phương thức ToDictionary để chuyển đổi các nhóm thành một từ điển với Nhãn là khóa và giá trị là số lượng các điểm hàng xóm có trong từ điển theo từng nhóm.

Sắp xếp từ điển đếm Nhãn theo danh sách tăng dần để điểu này đảm bảo nhãn có số lượng đối tượng nhiều nhất nằm ở trên đầu.Tiếp đó dùng phưng thức First() để lấy cặp khóa đầu tiên trong từ điển hay còn gọi là nhãn phổ biến nhất và phương thức Key() dùng để truy cập vào khóa của từ điển. Sau cùng trả về nhãn mà có số lượng phổ biến nhất



Hình 3.7:Phân loại nhãn

Từ danh sách các hàng xóm gần nhất, số lượng hàng xóm lấy theo K . Tạo biến string nhanDuDoan để lưu nhãn dự đoán cho điểm mới bằng cách lấy nhân phổ biến nhất trong danh sách các hàng xóm gần nhất và trả về tên của loại nhan đó

# **IV Ưu điểm và nhược điểm**

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| **Đơn giản và dễ triển khai**: Thuật toán KNN đơn giản và dễ hiểu, không yêu cầu quá nhiều trọng số hay tham số phức tạp. Điều này làm cho việc triển khai thuật toán trở nên dễ dàng và nhanh chóng. | **Yếu tố tính toán**: Một trong những hạn chế của KNN là yếu tố tính toán. Với mỗi điểm dữ liệu mới cần phân loại, thuật toán phải tính toán khoảng cách đến tất cả các điểm dữ liệu huấn luyện đã biết. Điều này đòi hỏi một lượng tính toán đáng kể, đặc biệt là với tập dữ liệu lớn. |
| **Linh hoạt trong việc xử lý dữ liệu**: KNN có khả năng phân loại dữ liệu với nhiều đặc trưng và nhãn khác nhau. Nó không giới hạn bởi các giả định về phân phối của dữ liệu và có thể áp dụng cho nhiều loại bài toán khác nhau. | **Nhạy cảm với dữ liệu nhiễu**: KNN có thể nhạy cảm với dữ liệu nhiễu và các điểm dữ liệu outlier. Điều này có thể ảnh hưởng đến độ chính xác của thuật toán, vì nó có thể gây ra các dự đoán sai lệch. |
| **Không cần quá trình huấn luyện phức tạp**: KNN không yêu cầu quá trình huấn luyện phức tạp như các thuật toán học máy khác. Thay vào đó, nó sử dụng các điểm dữ liệu huấn luyện đã biết để phân loại điểm dữ liệu mới. | **Lựa chọn K và phương pháp tính khoảng cách**: Độ chính xác của thuật toán KNN phụ thuộc vào giá trị của K (số lượng điểm gần nhất) và phương pháp tính khoảng cách giữa các điểm dữ liệu. Lựa chọn K và phương pháp tính khoảng cách phù hợp là quan trọng để đạt được kết quả tốt. |
| **Dễ dàng mở rộng:** KNN có thể được mở rộng để xử lý dữ liệu lớn và đa chiều bằng cách sử dụng các cấu trúc dữ liệu như cây KD-Tree hoặc cấu trúc dữ liệu băm. Điều này giúp cải thiện hiệu suất tính toán của thuật toán. |  |

# **V ỨNG DỤNG**

Thuật toán K-Nearest Neighbors (KNN) có nhiều ứng dụng trong lĩnh vực của học máy và khai phá dữ liệu. Dưới đây là một số ví dụ về ứng dụng của thuật toán KNN:

1. Phân loại văn bản: KNN có thể được sử dụng để phân loại văn bản vào các nhóm khác nhau, chẳng hạn như phân loại email vào hộp thư đến hay hộp thư rác, phân loại tin tức vào các chủ đề khác nhau, hoặc phân loại bài viết trên mạng xã hội.
2. Nhận diện chữ viết tay: KNN có thể được sử dụng để nhận diện chữ viết tay, ví dụ như phân loại chữ số viết tay thành các số từ 0 đến 9.
3. Phân loại hình ảnh: KNN có thể được sử dụng để phân loại hình ảnh vào các nhóm khác nhau, chẳng hạn như phân loại hình ảnh chó và mèo, hoặc phân loại các loại hoa dựa trên hình ảnh.
4. Gợi ý sản phẩm: KNN có thể được sử dụng để gợi ý sản phẩm cho người dùng dựa trên sự tương đồng giữa các người dùng khác nhau. Ví dụ, khi một người mua hàng trực tuyến xem một sản phẩm, KNN có thể được sử dụng để gợi ý các sản phẩm tương tự cho người dùng đó.
5. Phân loại bệnh lý y tế: KNN có thể được sử dụng để phân loại các bệnh lý y tế dựa trên các đặc trưng của bệnh nhân, chẳng hạn như các chỉ số sinh lý, kết quả xét nghiệm và lịch sử bệnh án.

# **VI. SO SÁNH KNN & K MEANS**

**KNN và K-means là hai thuật toán học máy được sử dụng phổ biến trong phân loại và phân cụm dữ liệu.** Cả hai thuật toán đều có ưu điểm là đơn giản, dễ hiểu và dễ triển khai. Tuy nhiên, giữa hai thuật toán này cũng có một số điểm khác biệt đáng chú ý.

|  |  |  |
| --- | --- | --- |
| **Tính chất** | **KNN** | **K-Means** |
| **Mục đích** | Phân loại | Phân cụm |
| **Cách thức hoạt động** | Dựa trên ý tưởng rằng các điểm dữ liệu tương tự nhau sẽ có xu hướng nằm trong cùng một lớp | Dựa trên ý tưởng rằng các điểm dữ liệu trong cùng một cụm sẽ có xu hướng gần nhau hơn so với các điểm dữ liệu trong các cụm khác |
| **Các tham số** | K, số lượng láng giềng gần nhất | K, số lượng cụm mong muốn; phương thức tính khoảng cách |
| **Ưu điểm** | Có thể xử lý dữ liệu có dạng bất kỳ, không cần phải chuẩn hóa dữ liệu; hiệu quả với số lượng điểm dữ liệu nhỏ | Có thể phân cụm dữ liệu hiệu quả với số lượng điểm dữ liệu lớn |
| **Nhược điểm** | Có thể bị ảnh hưởng bởi các điểm dữ liệu ngoại lai | Có thể bị mắc kẹt trong các giải pháp cục bộ |

# **VII. Cài đặt thuật toán**

# **Tài Liệu Tham Khảo**

<https://tek4.vn/khoa-hoc/machine-learning-co-ban/thuat-toan-k-nearest-neighbors-knn>

<https://codelearn.io/sharing/thuat-toan-k-nearest-neighbors-knn>