1.5. PHÂN LOẠI HỌC MÁY

Học máy được phân thành các loại chính sau đây:

* Học có giám sát (Supervised Learning)
* Học không giám sát (Unsupervised Learning)
* Học củng cố (Reinforcement Learning)

A diagram of a diagram

Description automatically generated

HỌC CÓ GIÁM SÁT (SUPERVISED LEARNING)

* Học có giám sát là một nhóm thuật toán sử dụng dữ liệu được gán nhãn nhằm mô hình hóa mối quan hệ giữa biến đầu vào x và biến đầu vào y.
* Hai nhóm bài toán cơ bản trong học có giám sát là classification (phân loại) và regression (hồi quy), trong đó có các giá trị rời rạc trong khi biến đầu ra của bài toán hồi quy có các giá trị liên tục.
* Với Supervised Learning, bên cạnh xây dựng các mô hình mạnh, việc thu thập và gán nhãn dữ liệu tốt và hợp lý cũng đóng vai trò then chốt để giải quyết các bài toán trong thực tế.

HỌC KHÔNG GIÁM SÁT (UNSUPERVISED LEARNING)

* Ngược lại Supervised Learning, Unsupervised Learning là một nhóm thuật toán sử dụng dữ liệu không có nhãn.
* Các thuật toán theo cách tiếp cận này hướng đến việc mô hình hóa được cấu trúc hay thông tin ẩn trong dữ liệu.
* Hay nói các khác, sử dụng các phương pháp này thiên về việc mô tả tính chất hay đắc tính của dữ liệu.
* Thông thường, các thuật toán này dựa trên những thông tin sau:
* Mối quan hệ tương tự (similarity) giữa các ví dụ (được gọi là instance) trong dữ liệu như trong các thuật toán clustering (phân cụm)
* Xác suất đồng xuất hiện của các đối tượng như trong Association mining.
* Các phép biến đôi ma trận để trích xuất các đựac trưng như PCA, SVD.
* Ứng dụng chính của học không giám sát:
* Chia phân khúc sản phẩm
* Chia phân khúc khách hàng
* Phát hiện sự tương đồng
* Hệ thông khuyến nghị
* Gắn nhãn các tập dữ liệu không được gắn nhãn
* ...

HỌC CỦNG CỐ (REINFORCEMENT LEARNING)

* Học củng cố là một lĩnh vực quan trọng trong trí tuệ nhân tạo (AI), nơi máy tính học hành và ra quyết định thông qua việc tương tác với môi trường.
* Trong học tăng cường, một hệ thống hoặc “agent” (người chơi) tương tác với một môi trường, thực hiện các hành động và nhận được phản hổi trong hình thức phần thưởng hoặc hình phạt.
* Mục tiêu của học tăng cường là học cách lựa chọn hành động để tối ưu hóa một mục tiêu cụ thể theo thời gian.

A diagram of a agent and environment

Description automatically generated

1.6. CÁC GUỒN DỮ LIỆU MIỄN PHÍ CHO HỌC MÁY

Nguồn dữ liệu mở thông dụng:

* UC Irvine Machine Learning Reponsitory
* Kaggle datasets
* Amazon’s AWS datasets
* Data Portals
* OpenDataMonitor
* U.S Government’s Open Data
* Quandl

Danh sách datasets và công cụ tìm datasets

* ...

# CHƯƠNG II: TỔNG QUAN VỀ BỘ CÔNG CỤ HỖ TRỢ PHÂN TÍCH DỮ LIỆU ANACONDA

THUẬT NGỮ CƠ BẢN

* Environment: một Environment thường bao gồm một phiên bản cụ thể của ngôn ngữ Python hoặc R và một số package.
* Package: là tập hợp các modules, các thư viện có sẵn của Python. Các modules có liên quan đến nhau thường được đặt trong cùng 1 packages. VD: Tensorflow, OpenCV-Python, NumPy, Pandas, Pytorch,...

GIỚI THIỆU ANACONDA

* Anaconda là nền tảng (platform) phân phối các ngôn ngữ lập trình Python và R về KHDL thông dụng nhất hiện nay.
* Nhằm mục đích đơn giản hóa việc quản lý và triển khai các package của Python và R.
* Bản phân phối này bao gồm các gói KHDL phù hợp với Windows, Linux và macOS.
* Conda hoạt động trên giao diện dòng lệnh, chẳng hạn như Anaconda Prompt trên Windows và terminal trên macOS và Linux.
* Navigator là một giao diện đồ họa (GUI) cho phép khởi chạy các ứng dụng và quản lý các gói (package), môi trường (environment) và kênh (channel) mà không cần sử dụng các lệnh dòng lệnh.

Anaconda Navigator là một ưng dụng có giao diện người dùng (GUI) hỗ trợ quản lý các bộ môi trường phát triển tích hợp (Integrated Development – IDE)

QUẢN LÝ MÔI TRƯỜNG (ENVIRONMENT)

Navigator sử dụng conda để tạo các môi trường riêng biệt chứa các tệp (files), gói (package) và các phần phụ thuộc của

# CHƯƠNG III: TRỰC QUAN HÓA DỮ LIỆU

3.1 GIỚI THIỆU VỀ TRỰC QUAN HÓA DỮ LIỆU

DỮ LIỆU LÀ GÌ?

* Dữ liệu là thông tin hoặc sự tập hợp của các yếu tố, sự kiện, số liệu, sự mô tả hay bất kì thứ gì có khả năng được thu thập, ghi lại hoặc biểu diễn.
* Dữ liệu có thể tồn tại ở nhiều dạng khác nhau, và nó có giá trị khi được xem xét, phân tích, xử lý hoặc sử dụng để hỗ trợ ra quyết định hoặc hiểu về hiện tượng nào đó.

GIỚI THIỆU VỀ TRỰC QUAN HÓA DỮ LIỆU

* Trực quan hóa dữ liệu là quá trình biểu diễn dữ liệu bằng các hình ảnh, biểu đồ, đồ thị hoặc các phương tiện trực quan khác để giúp hiểu rõ dữ liệu một cách dễ dàng và nhanh chóng.
* Trực quan hóa dữ liệu giúp biến dữ liệu trừu tượng và khó hiểu thành hình ảnh hoặc biểu đồ dễ hiểu và hấp dẫn.
* Trực quan hóa dữ liệu là nghệ thuật và khoa học kể những câu chuyện hấp dẫn bằng dữ liệu (Story Telling).
* Các nhà phát triển và khoa học dữ liệu ngày nay, bất kể lĩnh vực hoạt động, đều động ý rằng việc truyền đạt thông tin chi tiết một cách hiệu quả bằng cách sử dụng trực quan hóa dữ liệu là rất quan trọng.

3.2 CÁC LOẠI BIỂU ĐỒ CƠ BẢN

BIỂU ĐỒ CỘT (BAR CHART)

* Biểu đồ cột/thanh là sự thể hiện trực quan các giá trị ở dạng thanh hình chữ nhật ngang hoặc dọc.
* Chiều cao (dài) tỷ lệ thuận với các giá trị được biểu diễn. Được hiển thị theo hai trục, một trục hiển thị phần tử và trục còn lại hiển thị giá trị của phần tử.
* Biểu đồ thanh được hiển thị theo chiều dọc còn được gọi là biểu đồ cột và biểu đồ thanh ngang được gọi là biểu đồ thanh trong một số công cụ như Microsoft Excel.

A close-up of a graph

Description automatically generated

* Biểu đồ thanh cũng có thể được kết hợp cho nhiều giá trị của một phần tử theo thời gian để hiển thị mối tương quan tương đối về hiệu suất (chẳng hạn như doanh thu hằng năm của các bộ phận khác nhau trong công ty/công ty đối thủ cạnh tranh theo thời gian).

BIỂU ĐỒ ĐƯỜNG (LINE CHART)

* Biểu đồ đường là biểu đồ hai chiều của các giá trị được kết nối theo thứ tự. Các giá trị được hiển thị (hoặc phân tán) một cách có trật tự và dược kết nối.
* Biểu đồ đường cho thấy xu hướng của một hoặc nhiều phần tử so với thời gian.

A graph of a number of people

Description automatically generated with medium confidence

Biểu đồ đường có thể là:

* Biểu đồ đường đơn giản (hiển thị giá trị của một phần tử trên một tham chiếu – chẳng hạn như thời gian).
* Nhiều biểu đồ đường – hiển thị nhiều giá trị trên một điểm tham chiếu tương tự - chẳng hạn như giá trị cổ phiếu của nhiều công ty theo thời gian (hiển thị bằng các màu khác nhau).
* Splines – biểu đồ đường biểu diễn đường nối cong của các điểm thay vì đường thẳng
* Biểu đồ đường bậc thang – nơi kết nối giữa các điểm được hiển thị trong một bước.

BIỂU ĐỒ TRÒN (PIE CHART)

* Biểu đồ hình tròn hiển thị tỷ lệ hoặc phần trăm của phần tử dữ liệu ở định dạng hình tròn. Biểu đồ hình tròn được chia thành nhiều phần khác nhau dựa tren giá trị/phần trăm của phần tử dữ liệu cần làm nổi bật.
* Tổng của toàn bộ phần tử tương ứng với 100% giá trị củ dữ liệu được hiển thị.

BIỂU ĐỒ ĐIỂM (SCATTER PLOT)

* Biểu đồ phân tán là biểu đồ hai chiều thể hiện suẹ so sánh của hai biến theo tọa độ toán học.
* Trong đó, trục tung Oy là các giá trị của biến được dự đoán (biến phụ thuộc) và trục hoành Ox là các giá trị của biến dùng để đưa ra dự đoán (biến độc lập).
* Đây là một đồ thị biểu hiện mối tương quan giữa nguyên nhân và kết quả.

BIỂU ĐỒ BONG BÓNG (BUBBLE CHART)

* Bubble chart là một biểu đồ phổ biến để biểu thị dữ liệu có mối quan hệ giữa ba chiều: hoành độ, tung độ và kích cỡ.
* Biểu đồ bong bóng cho phép giải thích mối quan hệ giữa hai biến và thể hiện cả mức độ sự liên quan giữa hai biến đó. Kích cỡ của các điểm trên biểu đồ có thể được sử dụng để biểu thị một giá trị thứ ba.

BIỂU ĐỒ PHÂN PHỐI TẦN SUẤT (HISTOGRAM)

* Biểu đồ Histogram là một dạng biểu đồ thể hiện tần suất theo dạng cột. Dữ liệu được biểu thị bằng các cột trên biểu đồ có độ cao khác nhau tùy thuộc vào tấn suất (bao nhiều lần) phạm vi dữ liệu cụ thể xảy ra.

BIỂU ĐỒ DOUGHNUT (DOUGHNUT CHART)

* Biểu đồ bánh doughnut có ứng dụng tương tự biểu đồ tròn.
* Tuy nhiên, bới chứa được nhiều hơn một dải dữ liệu, doughnut chart có thể biểu diễn được sự thay đổi của các phần dữ liệu.
* Với mỗi thay đổi, một vòng tròn mới sẽ được thêm vào.

BIỂU ĐỒ VÙNG (AREA CHART)

* Biểu đồ vùng thể hiện mối quan hệ giữa các giá trị và thời gian hoặc các giá trị khác nhau. Nó biểu thị dữ liệu về tăng giảm và suẹ thay đổi của các giá trị theo thời gian hoặc theo một trục xác định.
* Các giá trị được biểu diễn bằng một chuỗi điểm được kết nối bởi một đường bằng màu sắc cho phép bạn theo dõi dữ liệu dễ dàng.

# CHƯƠNG IV: TIỀN XỬ LÝ DỮ LIỆU

4.1 GIỚI THIỆU TIỀN XỬ LÝ DỮ LIỆU

* Dữ liệu thong thế giới thực bị ảnh hưởng nhiều bởi các yếu tố tiêu cực như nhiễu (noise), thiếu giá trị (missing value), dữ liệu không nhất quán và không cần thiết.

**DỮ LIỆU CHÁT LƯỢNG THẤP**

**⇓**

**HIỆU SUẤT HỌC MÁY CHẤT LƯỢNG THẤP**

* Dữ liệu chất lượng thấp sẽ dẫn đến kết quả khai thác chất lượng thấp. Cần phải nỗ lực rất nhiều để tiền xử lý dữ liệu nhằm nâng cao chất lượng dữ liệu để khác thác hiệu quả.

4.2 LÀM SẠCH DỮ LIỆU (DATA CLEANING)

* Dữ liệu trong thế giới thực có xu hướng không đầy đủ, nhiễu và khôn nhất quán.
* Làm sạch dữ liệu (Data Cleaning) bao gồm:
* Xử lý các giá trị còn thiếu (Missing Value)
* Xử lý dữ liệu nhiễu (Noise Data)
* Xử lý dữ liệu bị thiếu là một phần quan trọng của tiền xử lý dữ liệu và đôi khi dữ liệu thất sự có thể bị thiếu do nhiều lý do khác nhau như lỗi tỏng quá trình thu thập hoặc lỗi kỹ thuật.
* Các phương pháp xử lý giá trị còn thiếu:
* Loại bỏ các dòng hoặc cột có dữ liệu bị thiếu (Ignore the tupple)
* Điền giá trị còn thiếu theo cách thủ công
* Sử dụng hằng số chung để điền giá trị còn thiếu
* Sử dụng các phương pháp thống kê cơ bản để điền các giá trị còn thiếu (trung bình, trung vị,...)
* Sử dụng mô hình dự báo để điền vào các giá trị còn thiếu

# CHƯƠNG V: CÁC THUẬT TOÁN HỌC TẬP CÓ GIÁM SÁT (SUPERVISED LEARNING)

4.1 Giới thiệu học tập có giám sát (Supervised Learning)

Quy trình học của mô hình trong Supervised Learning có thể được mô tả như sau:

1. Tập dữ liệu huấn luyện: Chia thành hai phần chính – dữ liệu huấn luyện và dữ liệu kiểm thử. Dữ liệu huấn luyện chữa các mẫu dữ liệu đã được gán nhãn.

2. Chọn mô hình: Chọn một mô hình học mát phù hợp với các bài toán cụ thể. Các mô hình thông thường bao gồm các thuật toán như Support Vector Machines, Decision Trees, Random Forests, Neural Networks và nhiều thuật toán khác.

3. Huấn luyện mô hình: Áp dụng dữ liệu huấn luyện vào mô hình để nó học cách ánh xạ từ dữ liệu đầu vào đến đầu ra. Trong quá trình này, mô hình điều chỉnh các tham số của nó để giảm sai số giữa đầu ra dự đoán và đầu vào thực tế.

4. Kiểm thử mô hình: Sử dụng dữ liệu kiểm thử để đánh giá hiệu suất của mô hình trên các dữ liệu mới mà nó chưa từng thấy trước đó.

5. Tinh chỉnh mô hình: Nếu mô hình không hoạt động tốt trên dữ liệu kiểm thử, có thể cần phải điều chỉnh các tham số hoặc chọn một kiến trúc mô hình khác để cải thiện hiệu suất.

ỨNG DỤNG CỦA HỌC CÓ GIÁM SÁT (SUPERVISED LEARNING)

Nhận dạng hình ảnh và video (Computer Vision). Sử dụgn AI để nhận dạng và phân tích nội dung từ việc nhận biết khuôn mặt cho đến phân loại đối tượng.

Dịch ngôn ngữ: Dùng để dịch tự động giữa các ngôn ngữ khác nhau dựa trên dữ liệu đã được gán nhãn với các cặp câu tương ứng.

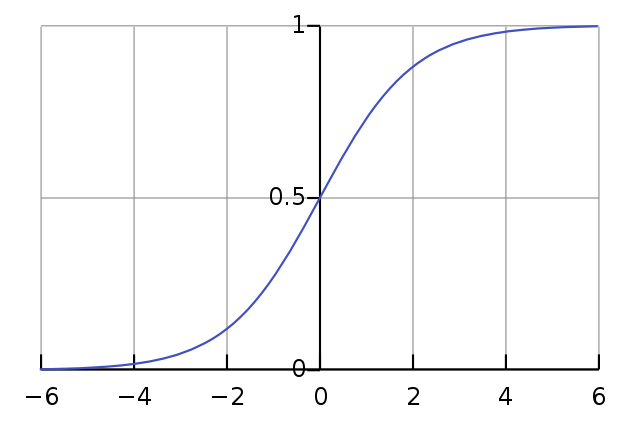
4.2 Giới thiệu hai bài toán phân loại (Classification) và bài toán hồi quy (Regresstion)

4.3

HỒI QUY LOGISTIC

Hồi quy logistic là một phương pháp trong thống kê và máy học được sử dụng để mô hình hóa và dự đoán xác suất của một biến phụ thuộc nhị phân (binary outcome), có nghĩa là biến phụ thuộc chỉ nhận một trong hai giá trị có thể: 0 hoặc 1, “có” hoặc “không”, “thành công” hoặc “thất bại”.

Hồi quy logistic sử dụng hàm logistic (hoặc hàm sigmoid) để chuyển đổi đầu ra của một hà, tuyến tính thành một giá trị nằm trong khoảng (0, 1). Công thức của hàm sigmoid như sau:



Công thức tổng quát của hồi quy logistic:

Trong đó:

: Xác suất của sự kiện Y xảy ra

: số Euler (khoảng 2.71828)

: các tham số mô hình

: các biến độc lập

Ưu điểm:

Dễ hiểu và triển khai: Có thể áp dụng một cách nhanh chóng mà không đòi hỏi nhiều công sức trong việc triển khai.

Tính linh hoạt: Linear Regression có thể được áp dụng cho nhiều dạng dữ liệu và có thể mở rộng để xử lý các vấn đề phức tạp hơn thông qua các biến độc lập tuyến tính hoặc phi tuyến tính.

Dễ hiểu và giải thích: Kết quả của mô hình Linear Regression thường dễ giải thích cho người không chuyên môn.

Hiệu suất tốt khi dữ liệu đơn giản: Khi mối quan hệ giữa biến độc lập và biến phụ thuộc là tuyến tính, Linear Regression thường cho kết quả tốt và ổn định.

Nhược điểm:

Nhạy cảm với dữ liệu nhiễu: Logistic Regression có thể nhạy cảm với dữ liệu nhiễu, điều này có thể làm ảnh hưởng đến hiệu suất.

Mối quan hệ phi tuyến tính: Nếu mối quan hệ giữa biến độc lập và biến phụ thuộc là phi tuyến tính. Logistic Regression không thể mô hình hóa hiệu quả.

Chỉ áp dụng cho mối quan hệ tuyến tính đơn biến: Logistic Regression chủ yếu được sử dụng để mô hình hóa mối quan hệ giữa một biến độc lập và một biến phụ thuộc. Đối với mối quan hệ phức tạp hơn, nó có thể không đủ linh hoạt.

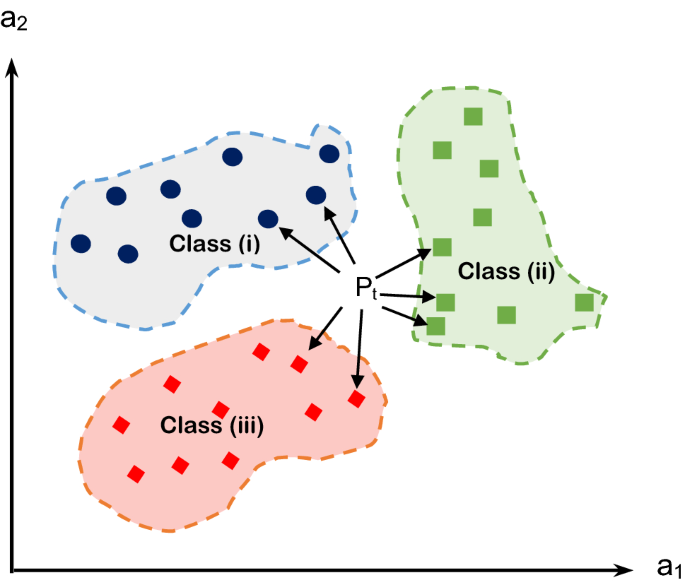
Assumption về độc lập của các biến: Logistic Regression đặt ra giả định về độc lập giữa các biến độc lập, điều này có thể không đáp ứng trong thực tế.

Không xử lý tốt với biến độc lập có phản hổi không đồng đều: Khi các biến độc lập không được phân phối đồng đều trong không gian giá trị. Logistic Regression có thể cho kết quả không chính xác.

GIỚI THIỆU VỀ K-NEAREST NEIGHBOR

Thuật toán K-Nearest Neighbor (KNN) là một kỹ thuật học máy phân loại học có giám sát phổ biến được sử dụng cho các nhiệm vụ phân loại và hồi quy.

KNN dựa trên ý tưởng rằng các điểm dữ liệu tương tự có xu hướng có nhãn hoặc giá trị tương tự.



Trong giai đoạn huấn luyện, thuật toán KNN lưu trữ toàn bộ tập dữ liệu huấn luyện làm tài liệu tham khảo.

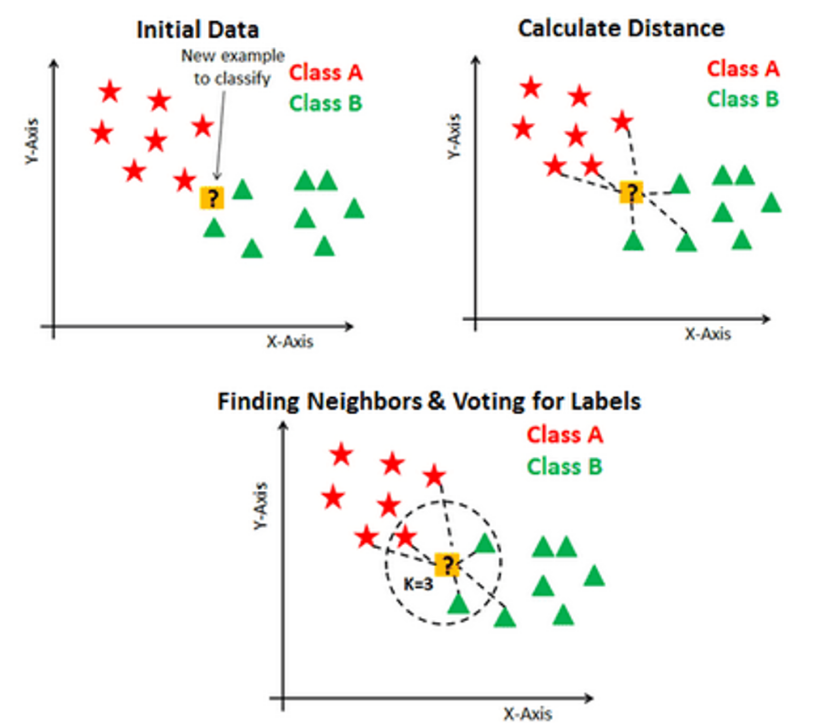
Khi đưa ra dự đoán, nó sẽ tính toán khoảng cách giữa điểm dữ liệu đầu vào và tất cả các mẫu huấn luyện bằng cách sử dụng thước đo khoảng cách đã chọn, chẳng hạn như khoảng cách Euclide.

KNN xác định K hàng xóm gần nhất với điểm dữ liệu đầu và dựa trên khoảng cách của chúng.

Đối với hồi quy, nó tính toán giá trị trung bình hoặc trung bính có trong số của các giá trị mục tiêu của K lân cận để dự đoán giá trị cho điểm dữ liệu đầu vào.

Với KNN, trong bài toán Classification, label của một điểm dữ liệu mới (hay kết quả của câu hỏi trong bài thi) được suy ra trực tiếp từ K điểm dữ liệu gần nhất trong training set.

Nhãn lớp của 1 test data có thể được quyết định bằng bỏ phiếu hoặc đa số (major voting) giữa các điểm gần nhất hoặc nó có thể được suy ra bằng cách đánh trọng số khác nhau cho mỗi trong các điểm gần nhất đó rồi suy ra nhãn lớp.

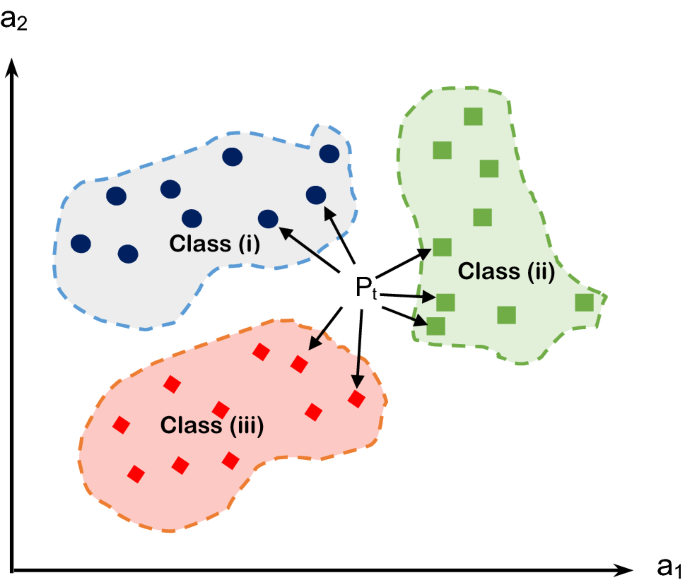


Sự khác biệt giữa các thuật ngữ này là về mặt kỹ thuật kỹ thuật, “bỏ phiếu đa số” yêu cầu đa số lớn hơn 50%, điều này chủ yếu có hiệu quả khi chỉ có hai loại.

Khi có nhiều lớp, VD: ba loại không nhất thiết cần 50% phiếu bầu để đưa ra kết luận về một lớp, có thể chỉ định nhãn lớp với số phiếu bầu lớn nhất và lớn hơn 33,31%.

K-Nearest Neighbor là một trong những thuật toán Supervised Learning đơn giản nhất (mà hiệu quả trong một vài trường hợp) trong Machine Learning.

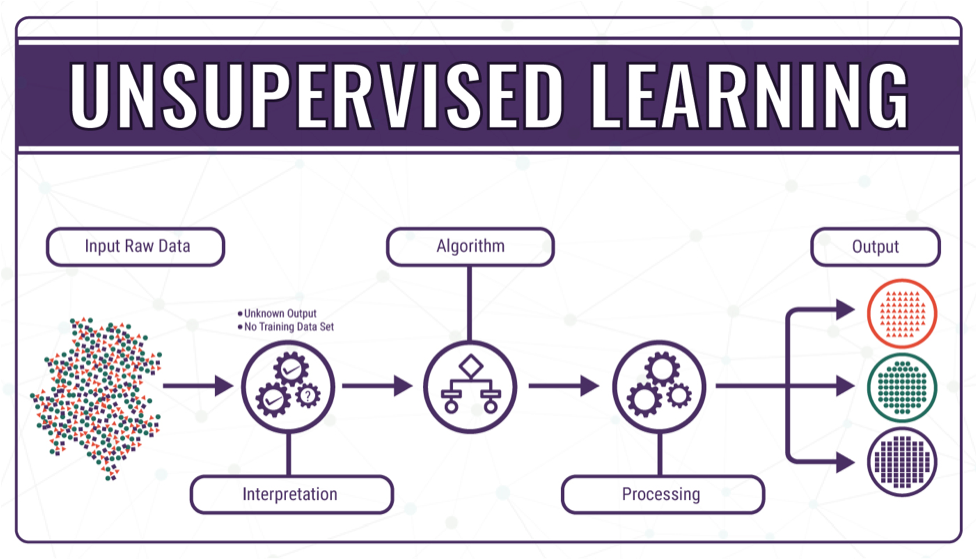
Khi training, thuật toán này không học một điều gì từ dữ liệu training (đây cũng là lý do thuật toán này được xếp vào loại lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới.



# CHƯƠNG VI: CÁC THUẬT TOÁN HỌC TẬP KHÔNG GIÁM SÁT (UNSUPERVISED LEARNING)

Ngược lại Supervised Learning, Unsupervised Learning (Học không giám sát) là một nhóm các thuật toán sử dụng dữ liệu không có nhãn.

Trong học không giám sát, các máy tính sẽ sử dụng dữ liệu không gán nhãn và tự học, mà không cần có sự giám sát nào bên ngoài.

Các thuật toán theo cách tiếp cận này hướng đến việc mô hình hóa được cấu trúc hay thông tin ẩn trong dữ liệu.

Hay nói cách khác, sử dụng các phương pháp này thiên về việc mô tả tính chất hay đặc tính của dữ liệu.

Thông thường, các thuật toán này dựa trên những thông tin sau:

Mối quan hệ tương tự (similarity) giữa các ví dụ (được gọi là instance) trong dữ liệu như trong các thuật toán clusting (phân cụm).

Xác suất đồng xuất hiện của các đối tượng như trong Association mining.

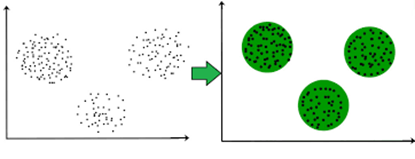
Các phép biến đổi ma trận để trích xuất các đựac trưng như PCA, SVD.

Phân cụm (Clustering):

Quảng cá và tiếp thị: Phân loại khách hàng thành các nhóm có sở thích hoặc hành vi mua hàng tương đồng.

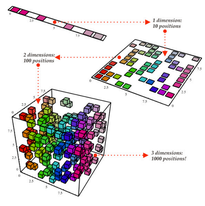
Chăm sóc sức khỏe: Phân nhóm bệnh nhân dựa trên các biểu hiện lâm sàng để đưa ra điều trị cá nhân hóa.

Xử lý hình ảnh: Nhận diện cụm đối tượng trong ảnh mà không cần gán nhãn trước.



Giảm chiều dữ liệu (Dimensionality Reduction):  
Xử lý dữ liệu: Giảm kích thước của dữ liệu giúp tăng tốc quá trình đào tạo mô hình và giảm độ phức tạp của mô hình.

Hiểu biểu đồ đám đông: Giảm chiều dữ liệu để hiểu rõ hơn về cấu trúc và quy luật của dữ liệu.



Phát hiện bất thường (Anomaly Detection):

Bảo mật mạng: Phát hiện các hoạt động bất thường trên mạng để nhận diện tấn công mạng.

Quản lý rủi ro tài chính: Nhận diện giao dịch tài chính không bình thường trong ngân hàng hoặc bảo hiểm.

Xử lý ngôn ngữ tự nhiên:  
Phân loại chủ đề: Phân loại văn bản vào các chủ đề không cần nhãn trước.

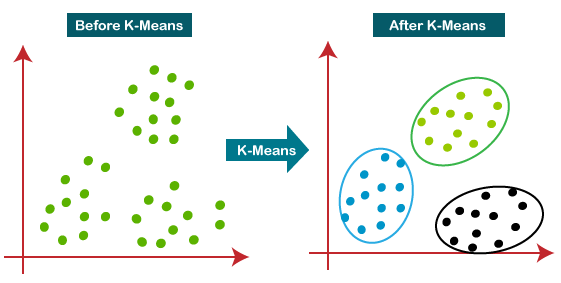
Phân loại ý kiến: Nhận xét và phân loại ý kiến từ người dùng. Học biểu diễn (Representation Learning)

Tạo biểu diễn tự động: Học biểu diễn của dữ liệu mà không cần sự giám sát trước, hữu ích trong việc rút trích đặc trưng và tạo biểu diễn tự đọng cho dữ liệu đầu vào.

6.3 THUẬT TOÁN K-MEANS

Thuật toán K-means được giưới thiệu bởi J.MacQueen vào năm 1967.

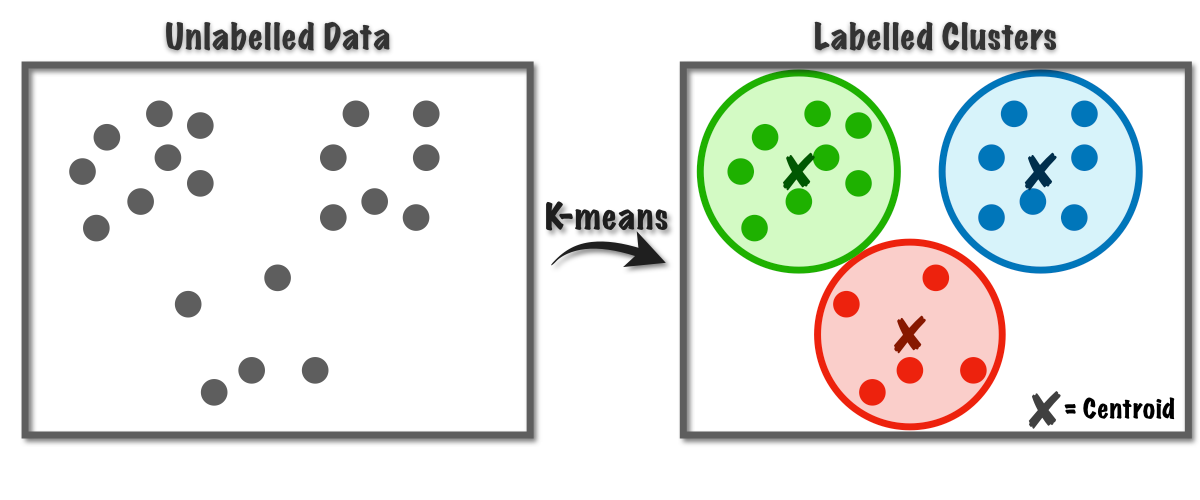
Thuật toán này giúp phân loại các điểm dữ liệu thành các nhóm không trùng lặp, gọi là các cụm, dựa trên sự tương tự giữa chúng.



Trong thuật toán K-means mỗi cụm dữ liệu được đặc trưng bỏi một tâm (centroid)

Tâm là điểm đại diện nhất cho một cụm và có giá trị bằng trung bìn của toàn bộ các quan sát nằm trong cụm.

Chúng ta sẽ dựa vào khoảng cách từ mỗi quan sát tới các tâm để xác định nhãn cho chúng trùng thuộc về tâm gần nhất.



Ban đầu thuật toán sẽ khởi tạo ngẫu nhiên một số lượng xác định trước tâm cụm

Sau đó tiến hành xác định nhãn cho từng điểm dữ liệu và tiếp tục cập nhật lại tâm cụm

Thuật toán sẽ dừng cho tới khi toàn bộ các điểm dữ liệu được phân về đúng cụm hoặc số lượt cập nhật tâm chạm ngưỡng

Nguyên tắc hoạt động:

1. Chọn số cụm (k): Bước đầu tiên của thuật toán là chọn số lượgn cụm mà bạn muốn tạo (k)
2. Khởi tạo điểm trung tâm: Chọn ngẫu nhiên k điểm từ tập dữ liệu làm điểm trung tâm ban đầu của các cụm
3. Phân loại điểm dữ liệu: Gán mỗi điểm dữ liệu vào cụm của điểm trung tâm gần nhất. Điều này thường dựa trên khoảng cách Euclidean giữa các điểm.
4. Cập nhật điểm trung tâm: Tính trung bình của tất cả các điểm trong mỗi cụm và đặt điểm trung tâm mới
5. Lặp lại bước 3 và bước 4: Lặp lại các bước phân loại và cập nhật điểm trung tâm cho đến khi không có sự thay đổi đáng kể nào trong phân loại hoặc đạt đến một số lần tối đa

THUẬT TOÁN DBSCAN

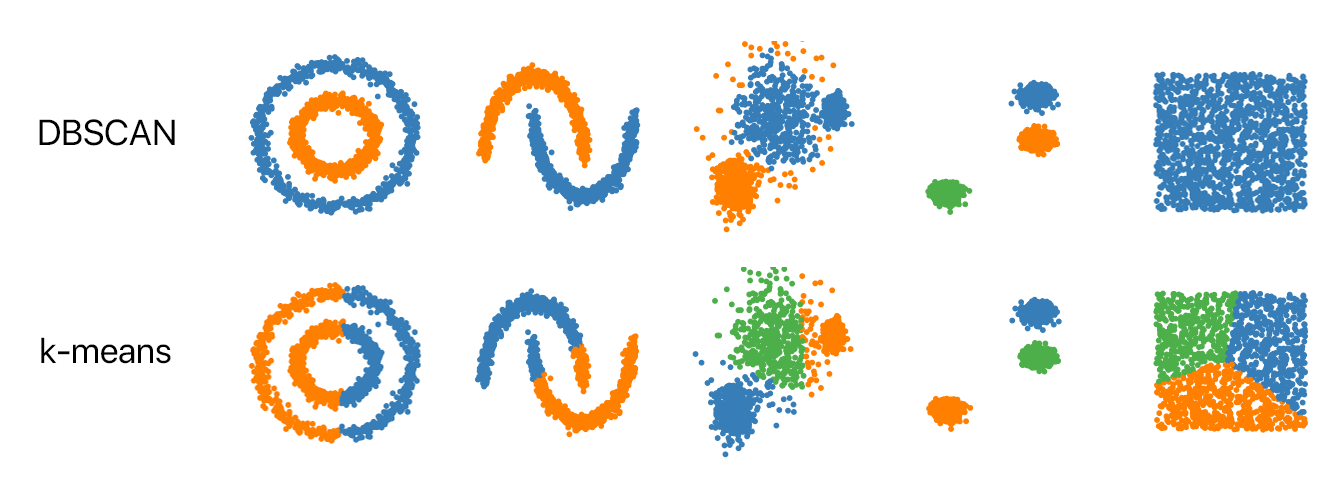
Điểm chung của những thuật toán phân cụm đều là dựa vào khoảng cách để xsc định cụm cho từng quan sát, cập nhật lại cụm dần dần qua các vòng lặp

Thuật toán K-Means khởi tạo ngẫu nhiên các centroids và sau đó cập nhật cụm bằng cách cập nhật lại centroids

Nhược điểm của thuật toán này là phải xác định số lượng cầm cần phân chia, tâm của cụm sẽ bị ảnh hưởng bởi các điểm khưởi tạo tâm cụm (centroids) đầu tiên

Trong DBSCAN thì vấn đề này được khắc phục nhờ cơ chế hình thành cụm đặc biệt mà ở đố các điểm dữ liệu nhiễu sẽ được tách thành một phần riêng

Đối với những phân phối có hình dạng đặc biệt mà k-Means không phân cụm tốt thì DBSCAN cũng có thể phân cụm được



DBSCAN (Density-Based Spatial Clustering of Applications with Noise) là một thuật toán phân loại dựa trên mật độ, thường được sử dụng trong học máy để phân loại các điểm dữ liệu thành các cụm dựa trên mật độ của chúng.

Nó có khả năng xác định cụm có hình dạng và kích thước linh hoạt và có khả năng xử lý dữ liệu chứa nhiễu.

Nguyên tắc hoạt động:

1. Chọn một điểm bất kỳ từ tập dữ liệu làm điểm bắt đầu
2. Xác định xem có đủ điểm xung quanh nó trong một kích thước (epsilon) và trong một khu vực mật độ (rho) hay không. Nếu có, chúng ta tạo một cụm mới và thêm tất cả các điểm trong khu vực mật độ và cụm đó.
3. Lặp lại quá trình trên cho tất cả các điểm mới thêm vào cụm. Nếu có thêm điểm nào trong khu vực mật độ; Ngược lại, chúng ta chuyển sang điểm tiếp theo.
4. Lặp lại quá trình cho tất cả các điểm trong tập dữ liệu. Cứ như vậy, chúng ta xác định tất cả các cụm trên mật độ của chúng

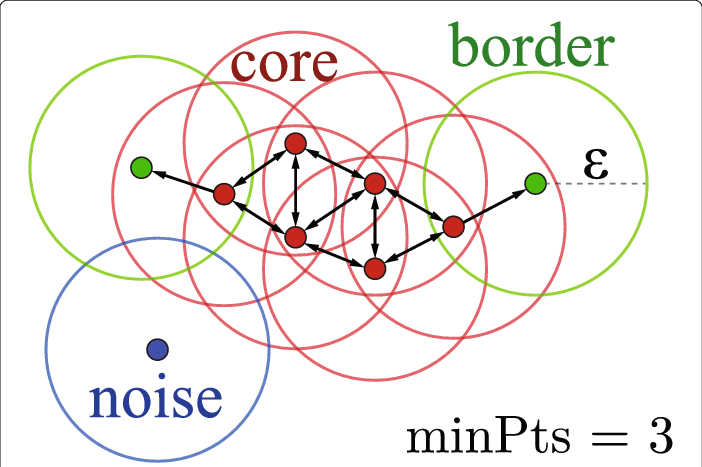
Các định nghĩa trong DBSCAN:

Epsilon (): đây là tham số trong DBSCAN quy định bán kính của một vùng lân cận xung quanh mỗi điểm dữ liệu. Một điểm được coi là thuộc vào một cụm nếu có ít nhất một điểm khác trong vùng lân cận bán kính

MinPts (Minimum Points): là số lượgn tối thiểu các điểm trong vùng lân cận bán kính để một điểm được coi là làm trung tâm của một cụm. Nếu số điểm trong vùng lân cận của một điểm là ít hơn MinPts, điểm đó sẽ được coi là nhiễu (Noise)

Core Points (Điểm trung tâm): Một điểm được xem là làm trung tâm của một cụm nếu có ít nhát MinPts điểm khác trong vùng lân cận bán kính của nó

Border Point (Điểm biên): Một điểm là điểm biên nếu nằm trong vùng lân cận của một điểm trung tâm, nhưng không đủ điểm để trở thành một điểm trung tâm



Noise (Nhiễu): Là các điểm không thuộc vào bất kì cụm nào và không có đủ số điểm xung quanh để trở thành một điểm trung tâm.

Reachability Distance (Khoảng cách có thể đạt được): Đây là một độ đo khoảng cách giữa hai điểm, tính theo cách mà DBSCAN xác định nó. Nếu điểm P có thể đạt được từ điểm Q, tức là có một chuỗi các điểm từ Q đến P sao cho mỗi điểm trong chuỗi nằm trong vùng lân cận bán kính của điểm trước đó

Density Reachability:

Core-Edge Density: Độ do mật độ của một cụm được tính bằng cách đếm số điểm trong vùng lân cận bán kính của một điểm trung tâm

Core-Density: Độ do mật đọ của một điểm trung tâm được tính bằng cách đếm số điểm trong vùng lân cận bán kính của nó

Connected Components: Là tập hợp các điểm được nối với nhau thông qua các liên kết Reachability

Nhước điểm:

Cần lựa chọn đúng các tham số epsilon và rho, điều này có thẻ khá khó quá

Có thể gặp khó khăn khi xử lý dữ liệu có mật độ đều.