BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA TOÁN - CƠ - TIN

----- 🕮 -----

BÁO CÁO CUỐI KỲ

ĐỀ TÀI: CHẨN ĐOÁN UNG THƯ SỬ DỤNG CÁC THUẬT TOÁN PHÂN CỤM VỚI CNN

Môn học: Khai phá Dữ liệu

Nhóm 2 thực hiện:

Họ và tên - Mã sinh viên	Nhiệm vụ	Ghi chú
Đỗ Thanh Lâm - 20002064	Mean-Shift, CNN	
Lê Thế Cường - 20002035	Mean-Shift, Davis-Bouldin, Running Time	
Hán Duy Khánh - 20002059	K-Means, Hierachical	
Nguyễn Thị Vân Anh - 20002031	K-Means, Hierachical	
Nguyễn Văn Tùng - 20002098	DBSCAN, Web	
Trần Diệu Minh - 20002075	DBSCAN, CNN	

Giảng viên hướng dẫn: PGS. TS. Lê Hoàng Sơn

Hà Nội - 2023

MỤC LỤC

LỜI MỞ ĐẦU

Trong vài thập niên gần đây, cùng với sự thay đổi và phát triển không ngừng của ngành công nghệ thông tin nói chung và trong các ngành công nghệ phần cứng, phần mềm, truyền thông và hệ thống các dữ liệu phục vụ trong các lĩnh vực kinh tế - xã hội nói riêng, thì việc thu thập thông tin cũng như nhu cầu lưu trữ thông tin càng ngày càng lớn.

Bên cạnh đó, việc tin học hóa một cách ồ ạt và nhanh chóng các hoạt động sản xuất, kinh doanh cũng như nhiều lĩnh vực hoạt động khác đã tạo ra cho chúng ta một lượng dữ liệu lưu trữ khổng lồ. Hàng triệu Cơ sở dữ liệu đã được sử dụng trong các hoạt động sản xuất, kinh doanh, quản lý,… trong đó có nhiều Cơ sở dữ liệu cực lớn cỡ Gigabyte, thậm chí là Terabyte.

Sự bùng nổ này đã dẫn tới một yêu cầu cấp thiết là cần có những kỹ thuật và công cụ mới để tự động chuyển đổi lượng dữ liệu khổng lồ kia thành các tri thức có ích, trích chọn được những thông tin có ý nghĩa từ những tập dữ liệu lớn để từ đó có thể giải quyết được các yêu cầu của thực tế như trợ giúp ra quyết định, dự đoán,… Từ đó, một trong những kỹ thuật quan trọng của Khai phá dữ liệu (Data Mining) đã ra đời để giải quyết các yêu cầu đó: phân cụm dữ liệu (Data Clustering).

Đặc biệt, trong y học, tuy đã có những thiết bị chuyên dụng để chụp lại các bộ phận bên trong cơ thể con người như máy chụp X - Quang, máy chụp cắt lớp,… nhưng những thiết bị này thường cho ra những hình ảnh không được rõ nét hoặc không tập trung vào vùng cần quan tâm, gây khó khăn cho việc chẩn đoán tình hình bệnh. Vì vậy, việc ứng dụng phân cụm dữ liệu vào lĩnh vực y học để giúp tiền xử lý những dữ liệu y tế như vậy là rất cần thiết.

Do đó, trong đề tài lần này, chúng em sẽ nghiên cứu về việc ứng dụng các thuật toán phân cụm với mô hình CNN trong lĩnh vực y học, cụ thể đó là xử lý những hình ảnh về khoang miệng, sử dụng mô hình CNN để chẩn đoán xem đó có phải ung thư hay không. Chúng em đặt tên cho đề tài này là: “Chẩn đoán ung thư sử dụng các thuật toán phân cụm với CNN”.

ĐẶT VẤN ĐỀ

Hiện nay các bệnh ung thư nói chung cũng như bệnh ung thư trên khoang miệng nói riêng đang là một vấn đề rất nghiêm trọng với tỉ lệ ngày càng cao trên toàn thế giới. Theo ước tính trên thế giới, tỉ lệ ung thư khoang miệng khác nhau tuỳ theo khu vực địa lý. Ở Hoa Kỳ, ung thư vùng đầu cổ chiếm 15% tổng số ung thư các loại với tỷ lệ mắc là 9,5 ca trên 100.000 dân. Trong đó, tỷ lệ các khối u ác tính vùng khoang miệng là 30% tổng số ung thư đầu cổ và 5% tổng số các ung thư nói chung. Tính đến năm 2008, ung thư khoang miệng là một trong mười ung thư nam giới phổ biến nhất Việt Nam.

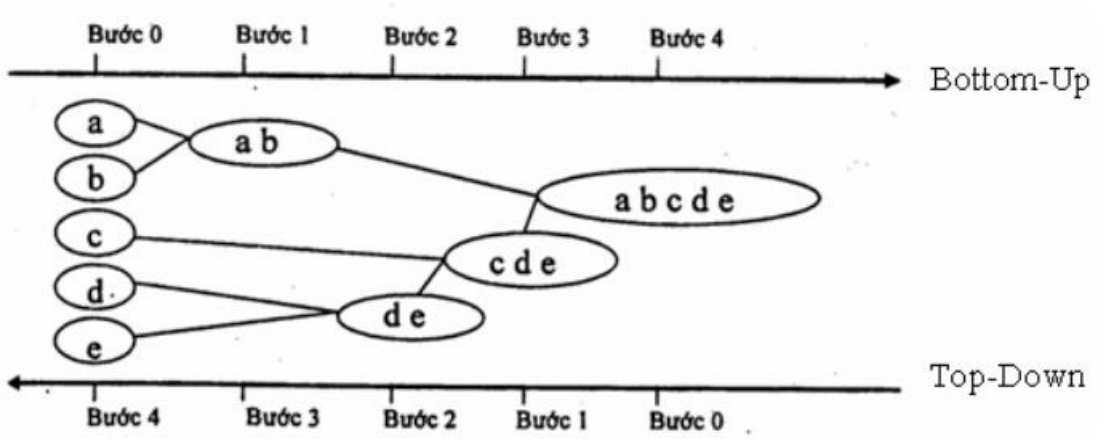
Do đó, bài toán chúng em hướng đến vừa giúp tìm hiểu, nghiên cứu về ung thư khoang miệng, áp dụng các kỹ thuật phân cụm để khoanh vùng vị trí cần quan tâm, đồng thời xây dựng một mô hình, tạo dựng một trang web giúp chẩn đoán ung thư khoang miệng dựa trên các ảnh chụp thông thường. Mong rằng dự án này có thể nhận được những góp ý từ thầy cô và mọi người để chúng em có thể hoàn thiện, đặc biệt có thể đưa vào thực tế giúp đỡ nền y học.

CHƯƠNG 1: CÁC HƯỚNG TIẾP CẬN CỦA BÀI TOÁN PHÂN CỤM

Có nhiều hướng tiếp cận và các ứng dụng trong thực tế nhưng tất cả đều hướng tới hai mục tiêu chung đó là chất lượng của các cụm khám phá được và tốc độ thực hiện các thuật toán. Các kỹ thuật đó có thể phân loại theo các cách tiếp cận chính sau:

I. Phương pháp phân cụm phân cấp

Cấu trúc phân cụm phân cấp xây dựng trên một hệ thống phân cấp cụm. Các cụm chứa các nút cụm con. Các cụm ngang hàng được phân chia thành các điểm cùng cụm cha. Cách tiếp cận này cho phép tìm hiểu chi tiết dữ liệu ở các cấp độ khác nhau. Phương pháp phân cụm được chia làm hai loại là phân cụm phân cấp hội tụ Bottom - Up và phân cụm phân cấp chia nhóm Top-Down.



Phân cụm phân cấp hội tụ khởi đầu với một điểm cụm và kết hợp đệ quy với hai hoặc nhiều cụm thích hợp nhất. Phân cụm phân cấp chia nhóm đầu tiên là một cụm chia tách bắt đầu với một cụm của tất cả các điểm dữ liệu và đệ quy chia tách các cụm thích hợp nhất. Quá trình này tiếp tục cho đến khi đạt được một tiêu chí dừng lại được. Phân cụm phân cấp dựa trên kết quả thống kê kết quả liên kết trong cụm.

II. Phương pháp phân cụm phân hoạch

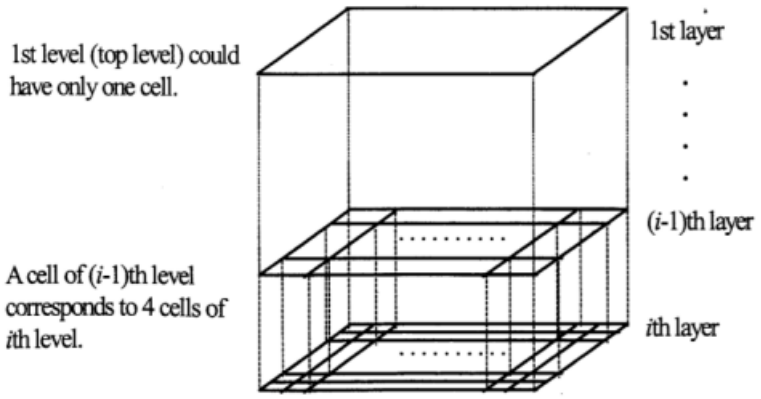
Phân cụm phân hoạch phân chia dữ liệu thành các tập số. Kiểm tra tất cả các hệ thống tập hợp con có thể là tính toán không khả thi. Di chuyển lặp đi lặp lại các điểm trong cụm. Sau khi các cụm được xây dựng phương pháp phân cụm phân hoạch sẽ xem xét lại các cụm để cải thiện các cụm tốt hơn. Với dữ liệu thích hợp sẽ đem lại hiệu quả cao trong phân cụm.

III. Phương pháp phân cụm dựa trên mật độ

Một tập mở trong không gian Euclide có thể được chia thành một tập hợp các thành phần kết nối. Việc thực hiện ý tưởng này cho phân vùng của một tập hợp hữu hạn các điểm đòi hỏi phải có khái niệm về kết nối, mật độ, ranh giới. Chúng liên quan đến điểm lân cận gần nhất. Một cụm quy định như là một thành phần kết nối dày đặc, phát triển ở bất kỳ hướng nào mà mật độ cao nhất. Dựa trên các thuật toán mật độ có khả năng phát hiện các cụm với hình dạng bất kỳ điều này giúp loại bỏ các giá trị ngoại lai hoặc nhiễu.

IV. Phân cụm dữ liệu dựa trên lưới

Phương pháp phân cụm dựa trên lưới đã được sử dụng trong một số nhiệm vụ khai thác dữ liệu của cơ sở dữ liệu lớn. Trong phân cụm dữ liệu dựa trên lưới, không gian đặc trưng được chia thành một số hữu hạn các ô hình chữ nhật hình thành lên lưới. Trên cấu trúc của lưới quá trình phân cụm được thực hiện. Quá trình đa phân tích thay đổi kích thước của ô hình chữ nhật có thể hình thành lên lưới. Trong không gian đa chiều *d*, lưới có dạng một hình lập phương với kích thước *d* tương ứng với các ô. Trong cấu trúc lưới phân cấp kích thước ô có thể được giảm để đạt được một cấu trúc ô chính xác hơn. Cấu trúc phân cấp có thể được chia thành nhiều cấp độ giải quyết. Mỗi ô ở mức độ cao hơn k sẽ được phân chia thành các ô có cấp độ thấp hơn *k+1*. Các ô ở mức độ thấp *k+1* sẽ được hình thành bởi việc chia tách các ô *k* vào các ô nhỏ hơn.



CHƯƠNG 2: CÁC THUẬT TOÁN TRONG PHÂN CỤM DỮ LIỆU

I. Thuật toán phân cụm phân hoạch (Thuật toán K-means)

K-means phân vùng tập dữ liệu thành K nhóm cụm riêng biệt không chồng chéo được xác định trước, trong đó mỗi điểm dữ liệu chỉ thuộc về một nhóm. Nó cố gắng làm cho các điểm dữ liệu trong cụm càng giống nhau càng tốt đồng thời giữ cho các cụm càng khác biệt(càng xa) càng tốt. Nó chỉ định các điểm dữ liệu cho một cụm sao cho tổng bình phương khoảng cách giữa các điểm dữ liệu và trung tâm của cụm (trung bình cộng của tất cả các điểm dữ liệu thuộc cụm đó)) là nhỏ nhất.

1. Thuật toán K-means

Bước 1: Khởi tạo ngẫu nhiên k tâm cụm $\mu_1, \dots \mu_k$.

Bước 2: Lặp lại quá trình cập nhật tâm cụm cho tới khi dừng:

a. Xác định nhãn cho từng điểm điểm liệu c_i dựa vào khoảng cách Euclide tới từng tâm cụm:

$$c_i = argmin_j ||x_i - \mu_j||^2$$

b. Tính toán lại tâm cho từng cụm theo trung bình toàn bộ các điểm dữ liệu trong một cụm:

$$\mu_j := \frac{\sum_{i=1}^{n} 1(c_i = j) x_i}{\sum_{i=1}^{n} 1(c_i = j)}$$

Hàm 1(.) trả về 1 nếu nhãn của điểm dữ liệu c_i được dự báo thuộc về cụm j, trái lại trả về 0.

Sau mỗi vòng lặp thì tâm của cụm sẽ dịch chuyển và thuật toán sẽ hội tụ khi tâm cụm ngừng thay đổi vị trí.

2. Sự hội tụ của thuật toán K-means

Xét hàm biến dạng (distortion function) dạng MSE:

$$\tau(c, \mu) = \sum_{i=1}^{n} ||x_i - 1(c_i = j)\mu_j||^2$$

$$\begin{aligned}
\frac{\delta \mathcal{L}(\mathbf{c}, \mu)}{\delta \mu_j} &= \sum_{i=1}^{n} \sum_{j=1}^{k} \frac{\delta \left\| \mathbf{x}_i - \mathbf{1}(c_i = j)\mu_j \right\|_2^2}{\delta \mu_j} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{k} \frac{\delta \left[\mathbf{x}_i - \mathbf{1}(c_i = j)\mu_j \right]^\intercal \left[\mathbf{x}_i - \mathbf{1}(c_i = j)\mu_j \right]}{\delta \mu_j} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{k} -\mathbf{1}(c_i = j)[\mathbf{x}_i - \underbrace{\mathbf{1}(c_i = j)}_{1}\mu_j] \\
&= \sum_{i=1}^{n} \sum_{j=1}^{k} -\mathbf{1}(c_i = j)[\mathbf{x}_i - \mu_j]
\end{aligned}$$

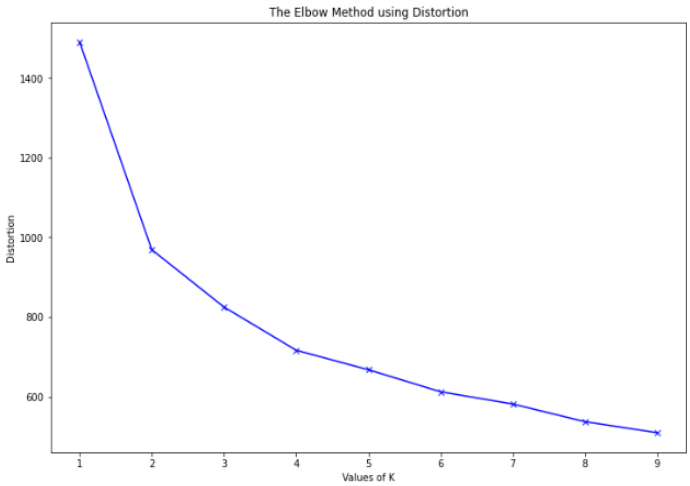
$$\sum_{i=1}^{n} \sum_{j=1}^{k} -\mathbf{1}(c_i = j)[\mathbf{x}_i - \mu_j^*] = 0$$

$$\leftrightarrow \mu_j^* = \frac{\sum_{i=1}^{n} \mathbf{1}(c_i = j)\mathbf{x}_i}{\sum_{i=1}^{n} \mathbf{1}(c_i = j)}$$

Hàm biến dạng luôn giảm sau mỗi vòng lặp và bị chặn dưới bởi 0 nên là một chuỗi hội tụ. Do vậy sau hữu hạn các bước thì thuật toán K-means sẽ dừng.

3. Phương pháp Elbow (Xác định K-number clustering)

Sử dụng phương pháp Elbow để xác định được số lượng cụm phù hợp thông qua đồ thị trực quan hóa bằng cách nhìn vào sự suy giảm của *hàm biến dạng* và lựa chọn ra *elbow point. Elbow point* là điểm mà ở đó tốc độ suy giảm của *hàm biến dạng* giảm đáng kể. Nếu thuật toán phân chia theo số lượng cụm tại vị trí này sẽ đạt được tính chất phân cụm một cách tốt nhất mà không gặp hiện tượng overfitting.



Điểm *Elbow point* là điểm mà ở đó tốc độ suy giảm của *hàm biến dạng* sẽ thay đổi nhiều nhất. Tức là sau vị trí này thì gia tăng thêm số lượng cụm cũng không giúp *hàm biến dạn* giảm đáng kể. Trong hình trên thì *Elbow point* k = 2.

II. Thuật toán phân cụm phân cấp (Thuật toán Hierachical)

Thuật toán Hierachical không yêu cầu khai báo trước số lượng cụm như thuật toán K-means nhưng phải xác định thước đo về sự khác nhau giữa các cụm, dựa trên sự khác biệt từng cặp giữa các quan sát trong hai cụm. Theo thuật toán này này, các cấp của biểu diễn phân cụm được biểu diễn trong đồ thị *dendrogram* .

1. Thuật toán Agglomerative

Thực hiện theo chiều *bottum-up*. Quá trình phân cụm bắt đầu ở dưới cùng tại các *leaf node*. Ban đầu mỗi quan sát sẽ được xem là một cụm tách biệt được thực hiện bởi một *leaf node*. Tại mỗi mức, thực hiện hợp một cặp cụm thành một cụm duy nhất nhằm tạo ra một cụm mới ở mức cao hơn tiếp theo. Cụm mới này tương ứng với *non-leaf node*. Như vậy khi hợp cụm thì số lượng cụm ít hơn. Một cặp được chọn để hợp nhất là những cụm trung gian không giao nhau.

2. Khoảng cách giữa hai cụm

Một số phương pháp xác định khoảng cách giữa hai cụm:

- Ward linkage: Phương pháp này đo lường khoảng cach giữa hai tâm cụm thông qua sự suy giảm phương sai. Mức độ suy giảm phương sai dữ liệu giảm nhiều hay ít phụ thuộc vào khoảng cách tâm (centroids) giữa hai cụm.
- Single linkage (nearest – neighbor): Phương pháp đo lường sự khác biệt giữa hai cụm bằng cách lấy ra cặp điểm gần nhất giữa hai cụm.
- Complete linkage: Phương pháp đo lường sự khác biệt giữa hai cụm bằng cách lấy ra hai cặm điểm xa nhau nhất hai cụm.
- Group average: Phương pháp này sẽ lấy trung bình toàn bộ khoảng cách giữa các cặp điểm được lấy từ hai cụm.

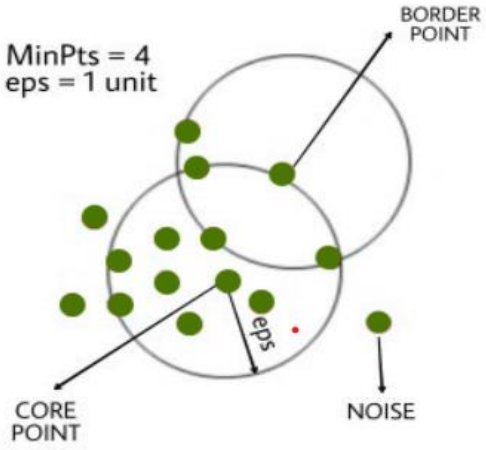
Cả bốn phương pháp *ward linkage, single linkage, complete linkage, group linkage* đều giúp tạo ra một thước đo về sự tương đồng.

3. Xác định điều kiện dừng của thuật toán phân cụm.

Quá trình phân cụm đều thu được đồ thị *dendrogram* dạng cây nhị phân. Mỗi một node trong cây nhị phân sẽ xác định một cụm dữ liệu. Nhưng làm thế nào để xác định khi nào ngừng tiếp tục phân chia hoặc hợp nhất đối với một node để tạo thành kết quả phân cụm. Đồ thị dendrogram là phương pháp xác định trước số lượng cụm cho Hierachical.

III. Thuật toán phân cụm dựa trên mật độ (Thuật toán DBSCAN)

Là một thuật toán phân cụm dựa trên mật độ giả định rằng các cụm là các vùng dày đặc trong không gian được phân tách bằng các cùng có mật độ điểm dữ liệu thấp hơn. Các điểm dữ liệu có mật độ dày đặc được kết hợp thành một cụm.



K-means và Hierarchical hoạt động tốt khi các cụm đơn giản để phát hiện. Nhưng chúng sẽ không tạo ra kết quả tốt khi các mô hình có cấu trúc phức tạp và chứa điểm dữ liệu ngoại lệ.

DBSCAN: yêu cầu 2 tham số ***eps, MinPts***

1. Tham số Eps và MinPts

Eps: Xác định vùng lân cận xung quanh một điểm dữ liệu, tức là nếu khoảng cách giữa hai điểm thấp hơn hoặc bằng "esp" thì chúng được coi là hàng xóm. Nếu giá trị eps được chọn quá nhỏ thì phần lớn dữ liệu sẽ coi là ngoại lệ. Nếu nó được chọn rất lớn thì các cụm sẽ hợp nhất và phần lớn các điểm dữ liệu sẽ nằm trong cùng một cụm.

MinPts: Số lân cận (điểm dữ liệu) tối thiểu trong bán kính eps. Tập dữ liệu càng lớn thì phải chọn giá trị MinPts càng lớn. Giá trị tối thiểu của MinPts phải được chọn là 3.

Hai tham số trên giúp xác định ba loại điểm:

- Điểm lõi (core): Đây là một điểm có ít nhất *minPts* điểm trong vùng lân cận *epsilon* của chính nó.
- Điểm biên (border): Đây là một điểm có ít nhất một *điểm lõi* nằm ở vùng lân cận *epsilon* nhưng mật độ không đủ minPts điểm.
- Điểm nhiễu (noise): Đây là điểm không phải là điểm lõi hay điểm biên.

Đối với một cặp điểm (P, Q) bất kỳ có ba khả năng:

- Cả P và Q đều có khả năng *kết nối mật độ* với nhau. Khi đó P, Q đều thuộc chung một cụm.
- P có khả năng *kết nối mật độ* được với Q nhưng Q không *kết nối mật độ* được với P. Khi đó P là *điểm lõi* của cụm còn Q là một *điểm biên.*
- P và Q đều không *kết nối mật độ* được với nhau. Trường hợp này P và Q sẽ rơi vào cụm khác nhau hoặc trong hai điểm là *điểm nhiễu.*

2. Thuật toán DBSCAN

Bước 1: Tìm tất cả các điểm lân cận trong eps và xác định các điểm cốt lõi hoặc

Bước 2: Đối với mỗi điểm cốt lõi nếu chưa được gán cho một cụm thì tạo một cụm mới

Bước 3: Tìm đệ quy tất cả các điểm được kết nối mật độ của nó và gán chung cho cùng một cụm làm điểm cốt lõi.

Bước 4: Một điểm A và B được gọi là liên thông mật độ nếu tồn tại một điểm C có đủ số lượng các điểm lân cận của nó và cả hai điểm A, B đều nằm trong khoảng cách eps.

3. Xác định tham số

- MinPts: có thể được tính theo số chiều D trong tập dữ liệu đó là minPts >= D + 1. Một giá trị minPts = 1 không có ý nghĩa, vì khi đó mọi điểm bản thân nó đều là một cụm. Với minPts $\leq$ 2 kết quả sẽ giống như phân cụm phân cấp (hierarchical clustering) với *single linkage* với biểu đồ *dendrogram* được tách ở độ cao y = epsilon. Do đó minPts phải chọn ít nhất là 3. Tuy nhiên, các giá trị lớn hơn thường tốt hơn cho các tập dữ liệu có nhiễu và kết quả phân cụm thường hợp lý hơn.
- Epsilon: Giá trị $\in$ được chọn bằng cách vẽ một biểu đồ k - distance (tương tự phương pháp Elbow trong K-measn). Đây là biểu đồ thể hiện giá trị khoảng cách trong thuật toán k-means với k = minPts – 1 điểm láng giềng gần nhất. Ứng với mỗi điểm chúng ta chỉ lựa chọn ra khoảng cách lớn nhất. Ứng với mỗi điểm chúng ta chỉ lựa chọn ra khoảng cách lớn nhất trong k khoảng cách. Nếu $\in$ quá nhỏ thì phần lớn dữ liệu sẽ coi là nhiễu; giá trị $\in$ quá cao, các cụm sẽ hợp nhất và phần phần lớn các điểm sẽ nằm trong một cụm.

4. Thuật toán KNN (K-Nearest Neighbors)

KNN là thuật toán đi tìm đầu ra của một điểm dữ liệu mới bằng cách *chỉ* dựa trên thông tin của K điểm dữ liệu trong training set gần nó nhất (K-lân cận), *không quan tâm đến việc có một vài điểm dữ liệu trong những điểm gần nhất này là nhiễu*.

Các bước trong KNN:

Bước 1: Ta có D là tập các điểm dữ liệu đã được gắn nhãn và A là dữ liệu chưa được phân loại.

Bước 2: Đo khoảng cách (Euclidian, Manhattan, Minkowski hoặc Trọng số) từ dữ liệu mới A đến tất cả các dữ liệu khác đã được phân loại trong D.

Bước 3: Chọn K (K là tham số mà bạn định nghĩa) khoảng cách nhỏ nhất.

Bước 4: Kiểm tra danh sách các lớp có khoảng cách ngắn nhất và đếm số lượng của mỗi lớp xuất hiện.

Bước 5: Lấy đúng lớp (lớp xuất hiện nhiều lần nhất).

Bước 6: Lớp của dữ liệu mới là lớp mà bạn đã nhận được ở bước 5.

IV. Thuật toán phân cụm dựa trên lưới (Thuật toán Mean-Shift)

Phân cụm Mean-Shift là một thuật toán phân cụm không tham số, dựa trên mật độ, có thể được sử dụng để xác định các cụm trong một tập dữ liệu. Nó đặc biệt hữu ích cho các tập dữ liệu mà các cụm có hình dạng tùy ý và không được phân tách rõ ràng bởi các ranh giới tuyến tính.

Ý tưởng cơ bản đằng sau phân cụm Mean-Shift là dịch chuyển mỗi điểm dữ liệu về phía chế độ (tức là mật độ cao nhất) của phân phối các điểm trong một bán kính nhất định. Thuật toán lặp đi lặp lại thực hiện các bước dịch chuyển này cho đến khi các điểm hội tụ về một cực trị địa phương của hàm mật độ. Những cực trị địa phương này đại diện cho các cụm trong dữ liệu.

Quá trình của thuật toán phân cụm Mean-Shift có thể được tóm tắt như sau:

Bước 1: Khởi tạo các điểm dữ liệu làm trung tâm cụm.

Bước 2: Lặp lại các bước sau cho đến khi hội tụ hoặc đạt số lần lặp tối đa: Đối với mỗi điểm dữ liệu, tính trung bình của tất cả các điểm trong một bán kính nhất định (tức là "hạt nhân") có tâm ở điểm dữ liệu.

Bước 3: Dịch chuyển điểm dữ liệu về trung bình.

Bước 4: Xác định các trung tâm cụm là những điểm không di chuyển sau khi hội tụ.

Bước 5: Trả về các trung tâm cụm cuối cùng và việc gán các điểm dữ liệu vào các cụm.

V. Đánh giá các phương pháp phân cụm (Davis - Bouldin)

Chỉ số Davies - Bouldin là đo độ tương đồng trung bình của mỗi cụm với cụm tương tự nhất của nó. Độ tương đồng được tính bằng tỷ lệ khoảng cách trong cụm với khoảng cách giữa các cụm. Do vậy các cụm càng xa nhau và ít phân tán sẽ cho kết quả tốt nhất. Chỉ số nhỏ nhất là 0 và các giá trị càng thấp hơn cho biết phân cụm tốt hơn.

$$DB = \frac{1}{n}\sum_{i=1}^{n} max_{j \neq i}\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right)$$

Trong đó: n là số cụm và σ_i là khoảng cách trung bình của tất cả các điểm trong cụm i từ tâm của cụm c_i.

CHƯƠNG 3: CẤU TRÚC MẠNG CNN VÀ CÁC ĐẠI LƯỢNG ĐO ĐỘ CHÍNH XÁC CỦA MÔ HÌNH

I. Cấu trúc mạng CNN

Mạng CNN là một tập hợp các lớp *Convolution* chồng lên nhau và sử dụng các hàm *nonlinear activation* như *ReLU* để kích hoạt các trọng số trong các node. Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo.

Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Do đó mà mỗi neuron ở lớp kế tiếp sinh ra từ kết quả của filter áp đặt lên một vùng ảnh một bộ neuron trước đó.

Mỗi một lớp được sử dụng nhiều filter khác nhau, thông thường như pooling/sub-sampling layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu).

Trong quá trình huấn luyện mạng (training) CNN tự động học các giá trị của các lớp filter dựa vào cách thức mà chúng ta thực hiện. Trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra các thông số tối ưu cho các filter tương ứng theo thứ tự raw pixel > edges > shapes > facial > high – level features. Layer cuối cùng dùng để phân lớp ảnh.

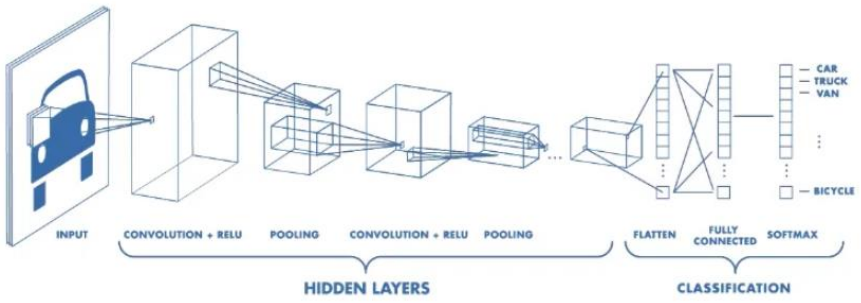
Trong mô hình CNN có 2 khía cạnh cần lưu tâm: *tính bất biến (Location Invariance)* và *tính kết hợp (Compositionality).*

Với cùng một đối tượng, nếu đối tượng này được chiếu theo các *(translation, rotation, scaling)* khác nhau thì độ chính xác của thuật toán sẽ ảnh hưởng đáng kể.

Pooling layer sẽ cho tính bất biến với phép dịch chuyển *(translation),* phép quay *(rotation),* phép co giãn *(scaling).*

Tính kết hợp (Compositionality) cho các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn qua convolution từ các filter.

Do đó CNN cho mô hình với độ chính xác rất cao.



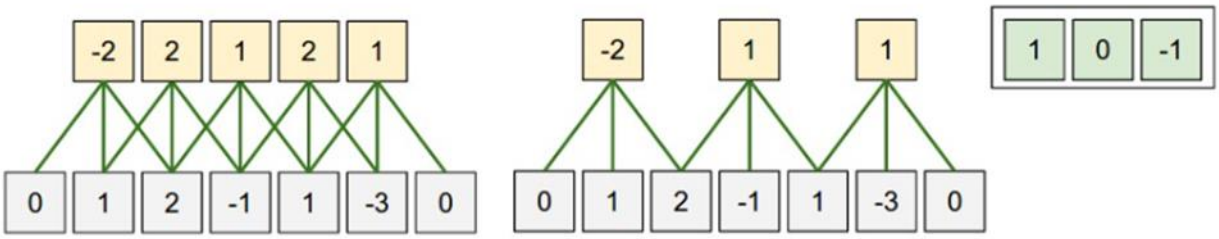
1. Convolution Layer

Convolution là lớp đầu tiên trích xuất các đặc tính từ hình ảnh. Tham số lớp này bao gồm các Filter có thể học được. Các Filter đều nhỏ thường có kích cỡ hai chiều 3x3 hoặc 5x5,... và có độ sâu bằng với độ sâu của đầu vào. Bằng cách trượt dần Filter theo chiều ngang

và dọc trên ảnh, chúng ta thu được một *Feature Map* chứa đặc trưng được trích xuất từ trên hình ảnh đầu vào.

Quá trình trượt các filter thường có các giá trị được quy định bao gồm: padding và stride.

- Zero – padding: cho phép chúng ta kiểm soát kích thước không gian của matrix đầu ra chẳng hạn như bảo toàn kích thước không gian của matrix đầu vào sao cho chiều rộng, chiều cao đầu vào và đầu ra là như nhau.
- Stride: Chỉ định bước tiến của Filter dẫn tới ảnh hưởng matrix đầu ra nhỏ hơn theo không gian.
- Công thức tính khối đầu ra: *(W – F + 2P) / S + 1* trong đó W là kích thước matrix đầu vào, F là kích thước matrix Filter, P là padding, S là stride.



Với mỗi kernel khác nhau thì mô hình sẽ học được những đặc trưng khác nhau của ảnh, nên trong mỗi *convolution layer* dùng nhiều kernel để học nhiều thuộc tính của ảnh. Vì mỗi kernel cho ra output là 1 matrix nên *k* kernel sẽ cho ra *k* matrix. Sau đó kết hợp *k* output matrix thành 1 tensor 3 chiều có chiều sâu *k*. Output của *convolution layer* sẽ qua hàm *acitvation function* trước khi trở thành input của *convolution layer* tiếp theo.

2. Activation Function

Một số hàm kích hoạt phi tuyến tính (*nonlinear activation*) như *ReLU* để kích hoạt trọng số của các node. Mỗi một lớp thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho lớp tiếp theo.

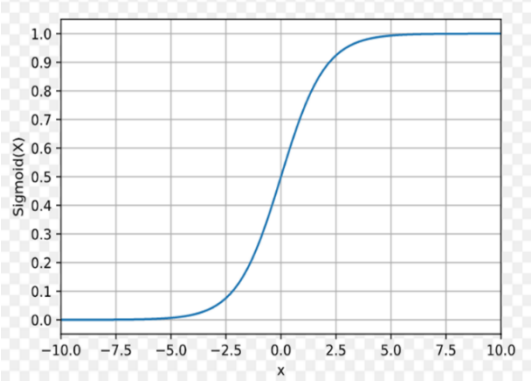
Một số hàm kích hoạt thường dùng: *Sigmoid*, *ReLU*.

a. Sigmoid

Công thức:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

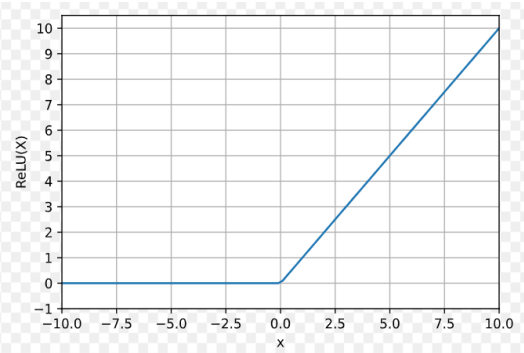
Hàm Sigmoid nhận đầu vào là một số thực và chuyển thành một giá trị trong khoảng (0,1). Đầu vào là số thực âm rất nhỏ sẽ cho đầu ra tiệm cận với 0, ngược lại, nếu đầu vào là một số thực dương rất lớn thì đầu ra tiệm cận với 1. Hàm Sigmoid ngày nay rất ít được sử dụng bởi vì gradient của hàm số này sẽ rất gần với 0 khi đầu vào có giá trị tuyệt đối lớn, dẫn tới trọng số tương ứng với đơn vị đang xét sẽ gần như không được cập nhật (vanishing gradient). Ngoài ra hàm Sigmoid không có trung tâm là 0 nên gây khó khăn cho việc hội tụ.



b. ReLU

Công thức:

$$f(x) = max(0, x)$$



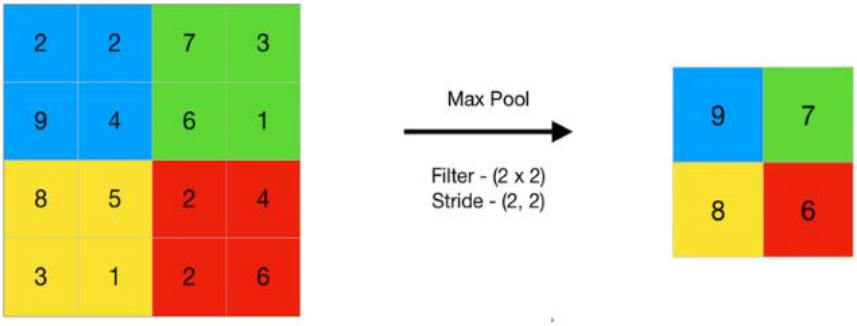
Hàm ReLU được dùng nhiều trong khi huấn luyện các mạng neuron. ReLU đơn giản lọc các giá trị < 0.

Ưu điểm: Tốc độ hội tụ nhanh hơn hẳn. ReLU có tốc độ hội tụ nhanh gấp nhiều lần do không bị bão hòa ở 2 đầu như Sigmoid. Tính toán nhanh hơn do không phải hàm exp.

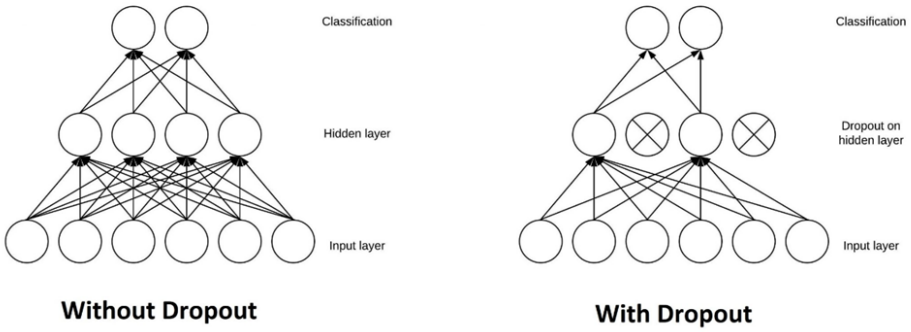
Nhược điểm: Khi learning rate lớn, các trọng số có thể thay đổi theo cách làm tất cả neuron dừng việc cập nhật. Do đó chọn learning rate nhỏ. Ngoài ra nhược điểm nữa là, các node có giá trị nhỏ hơn 0, qua ReLU activation sẽ thành 0. Nếu các node bị chuyển thành 0 thì sẽ không có ý nghĩa với bước linear activation ở lớp tiếp theo và hệ số tương ứng từ node đấy cũng không được cập nhật với gradient decent.

3. Pooling Layer

Pooling layer thường được dùng giữa các convolutional layer để giảm kích thước dữ liệu nhưng vẫn giữ được các thuộc tính quan trọng. Kích thước dữ liệu giảm giúp giảm việc tính toán trong model. Trong quá trình này, quy tắc về *stride* và *padding* áp dụng như phép tính convolution trên ảnh. Một số loại pooling: *Max Pooling*, *Average Pooling*, *Sum Pooling*.



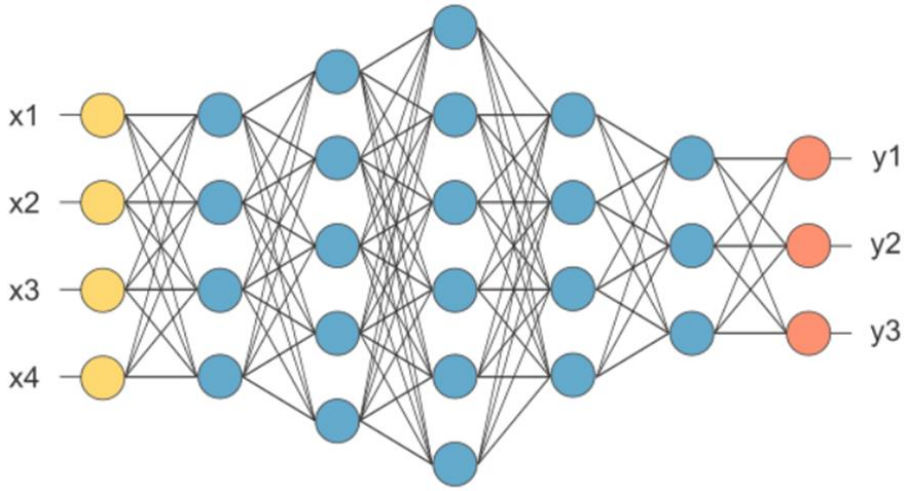
4. Dropout



Trong mạng neural network, kỹ thuật dropout là việc chúng ta sẽ bỏ qua một vài unit trong suốt quá trình training trong mô hình, những unit bị bỏ qua được lựa chọn ngẫu nhiên (tức là unit sẽ không tham gia và đóng góp vào quá trình huấn luyện). Tại mỗi giai đoạn huấn luyện, mỗi node có xác suất bị bỏ qua là 1 – p và xác suất được chọn là p.
Việc sử dụng kỹ thuật dropout, nhằm chống *over-fitting*. Khi chúng ta sử dụng *fully connected layer*, các neural sẽ phụ thuộc "mạnh" lẫn nhau trong suốt quá trình huấn luyện, điều này làm giảm sức mạnh cho mỗi neural và dẫn đến overfiting trong tập training.

5. Fully connected layer

Sau khi ảnh được truyền qua nhiều convolutional layer và pooling layer thì model đã học được tương đối các đặc điểm của ảnh thì tensor của output của layer cuối cùng sẽ được là phẳng thành vector và đưa vào một lớp được kết như một mạng neurol. Với FC layer được kết hợp với các tính năng lại với nhau để tạo ra một mô hình. Cuối cùng sử dụng softmax hoặc sigmoid để phân loại đầu ra.



6. Cách chọn tham số cho CNN

Để lựa chọn tham số cho mô hình CNN tốt nhất cần lưu tâm đến số lượng các convolution layer, kích thước filter, kích thước Pooling và việc thực hiện train test.

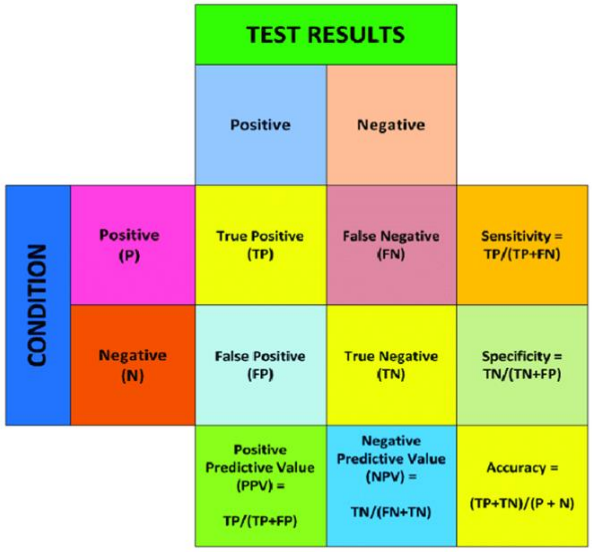
- Số các convolution layer: càng nhiều các convolution layer thì performance càng được cải thiện. Để mô hình tốt thì nên chọn ít nhất 3 layer.
- Kích thước Filter: thường là kích thước 5x5 hoặc 3x3
- Kích thước Pooling: thường là 2x2 hoặc 4x4 cho ảnh đầu vào lớn.
- Thực hiện việc train test nhiều lần để chọn ra tham số tốt nhất.

II. Đại lượng đo độ chính xác của một mô hình

Một số độ đo dùng để đánh giá hiệu năng của một mô hình: *Accuracy, Area Under the Curve (AUC).*

Đối với bài toán phân loại hai lớp, giả thiết hai phân loại dữ liệu sẽ là *positive* (+) và *negative* (-). Ma trận *Confusion matrix* như sau:

Trong đó, các chỉ số TP, FP, TN, FN lần lượt có ý nghĩa là:



- TP (True Positive): Tổng số trường hợp dự báo đúng Positive.
- TN (True Negative): Tổng số trường hợp dự báo đúng Negative.
- FP (False Positive): Tổng số trường hợp các quan sát (thực tế) thuộc nhãn Negative bị dự báo sai thành Positive.
- FN (False Negative): Tổng số trường hợp các quan sát (thực tế) thuộc nhãn Positive bị dự báo sai thành Negative.

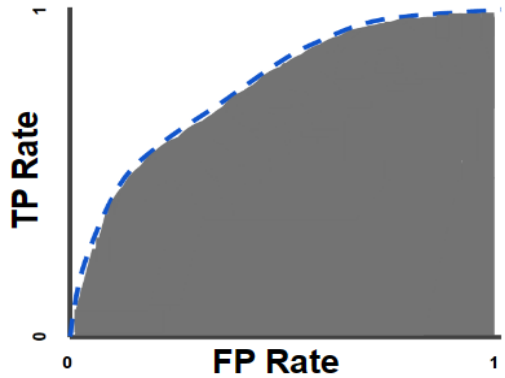
1. Accuracy

Đại lượng Độ chính xác - Accuracy: đo tỷ lệ số mẫu được mô hình dự đoán đúng trên tổng số mẫu dữ liệu được dùng để kiểm tra mô hình.

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

2. Area Under the Curve (AUC)

Một mô hình có hiệu quả hay không dựa trên ROC curve. Một mô hình hiệu quả khi có FPR thấp và TPR cao.
AUC là diện tích bên dưới bị giới hạn bởi ROC curve (một thước đo đánh giá mô hình khác).



- TPR: True Positive Rate

$$TPR = \frac{TP}{TP + FN}$$

- FPR: False Positive Rate

$$FPR = \frac{FP}{FP + TN}$$

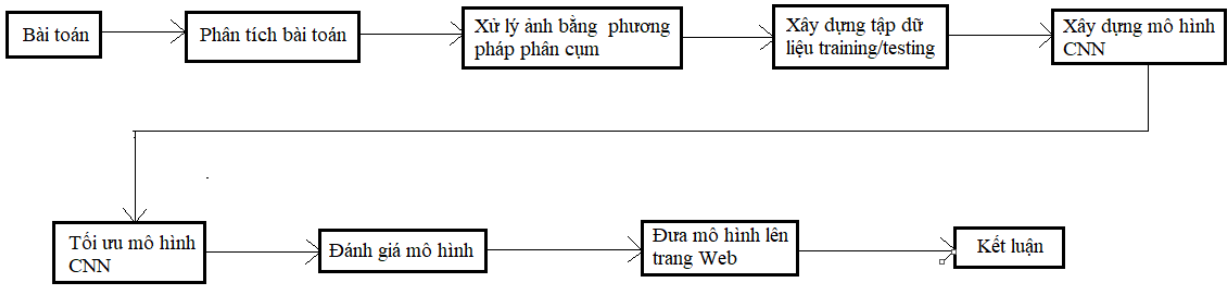
- AUC nằm trong khoảng giá trị từ 0 đến 1. Một mô hình có dự đoán sai 100% có AUC là 0,0; dự đoán chính xác 100% có AUC là 1.

CHƯƠNG 4: THỰC NGHIỆM VÀ ĐÁNH GIÁ

I. Mô tả dữ liệu sử dụng

Dưới đây là 30 ảnh minh họa cho bộ dữ liệu.

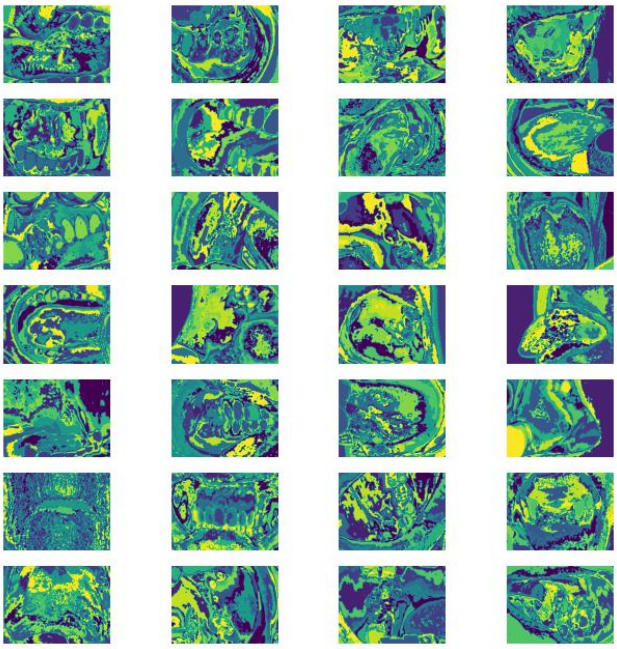
II. Xử lý bài toán



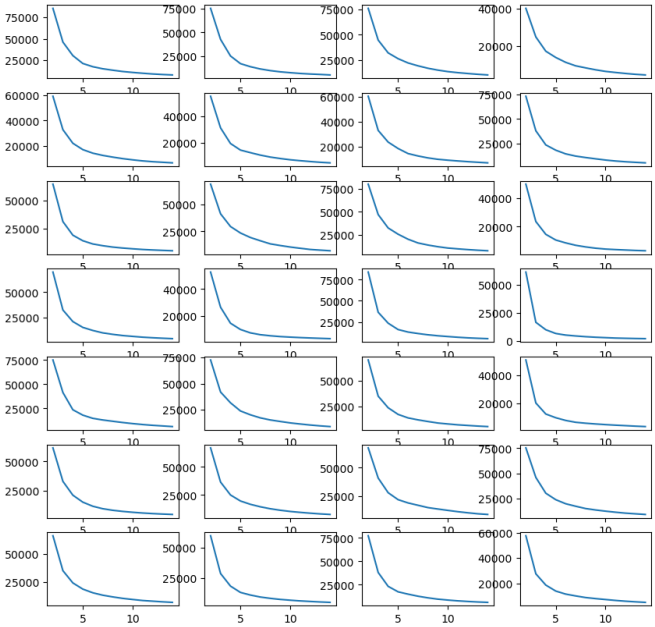
III. Đánh giá kết quả phân cụm

1. Đánh giá phân cụm trên các ảnh với thuật toán K-Means

Đối với thuật toán K-Means, chúng ta có một tham số duy nhất, đó là n_clustering (số lượng cụm). Để chọn ra số lượng cụm tối ưu cho mỗi ảnh, chúng em sử dụng các phương pháp khác nhau như Enboil, Shilhouse, Visualize.



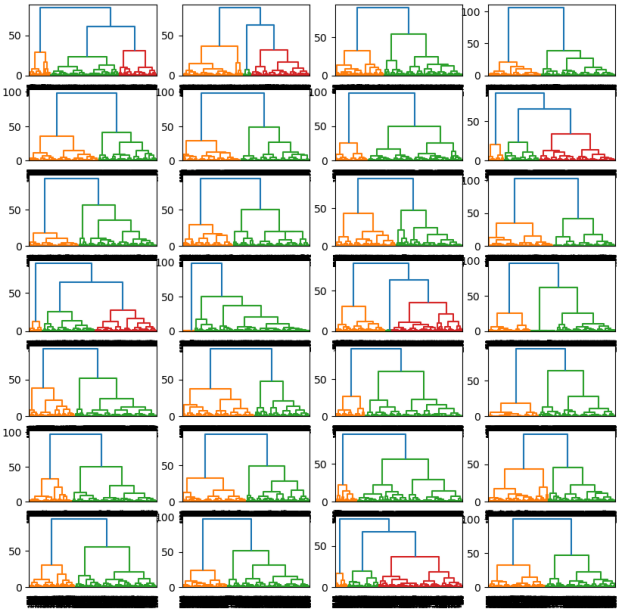
Như đã nói ở trên, để chọn ra số cụm phù hợp cho bộ dữ liệu này, chúng ta sẽ sử dụng phương pháp Elbow.



Từ kết quả trên, chúng ta có thể thấy nhìn chung, các ảnh đều phân đoạn ra rõ ràng khi số cụm có thể là 5 hoặc 6.

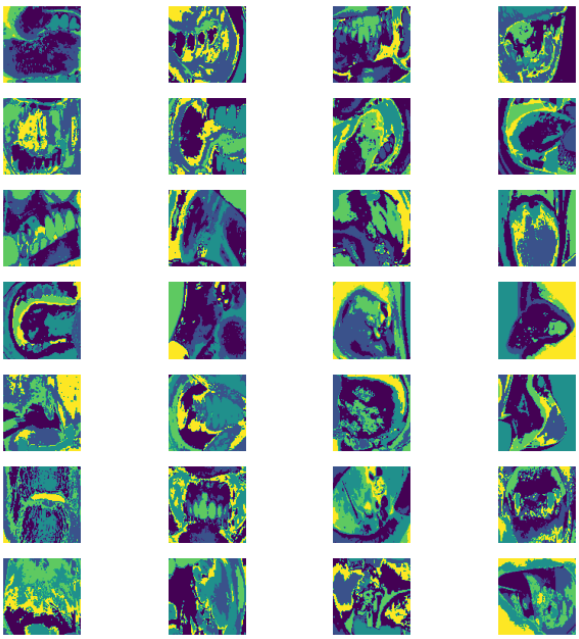
2. Đánh giá phân cụm trên các ảnh với Hierarchical

Đối với phân cụm phân cấp, chúng ta cần chọn ra số lượng cụm so cho mức độ khác biệt của nó đủ phù hợp. Chúng ta sẽ sử dụng đồ thi Dendrogram cho việc này.



Từ các kết quả trên, ta có thể đánh giá rằng số lượng cụm tối ưu của phương pháp này là khoảng 3, 4, 5 cụm thì các kết quả cho các ảnh khá tốt. Tuy nhiên, với việc phải giảm kích thước ảnh để phân cụm làm một phần nào đó lượng thông tin đã bị mất mát đi. Thêm vào đó, thời gian chạy lâu cùng với yêu cầu xử lý lượng lớn điểm ảnh đồng thời khiên cho việc dùng phương pháp này để phân đoạn ảnh là chưa thực sự tối ưu.

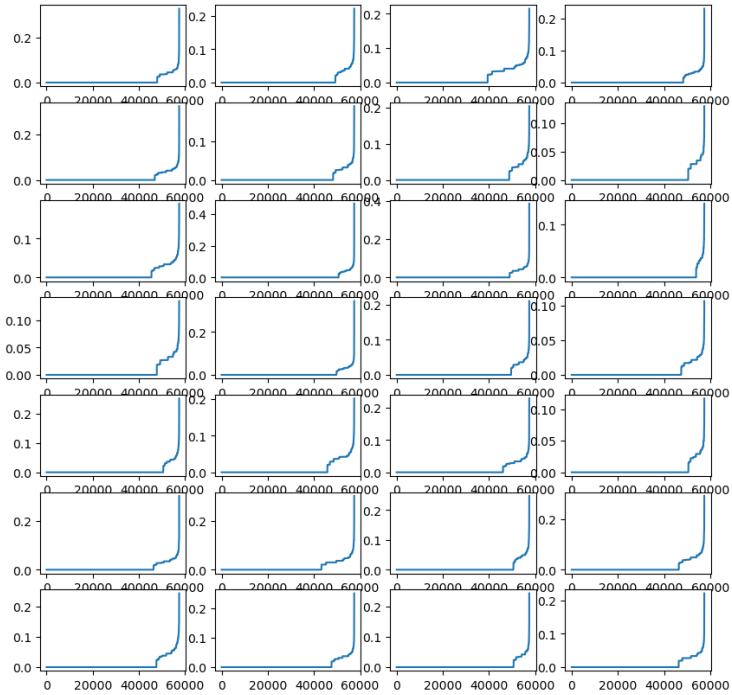
Dưới đây là chúng em đã giảm kích thước ảnh về 100 * 100, sau đó áp dụng phân cụm phân cấp để phân đoạn ảnh với số cụm là 5.



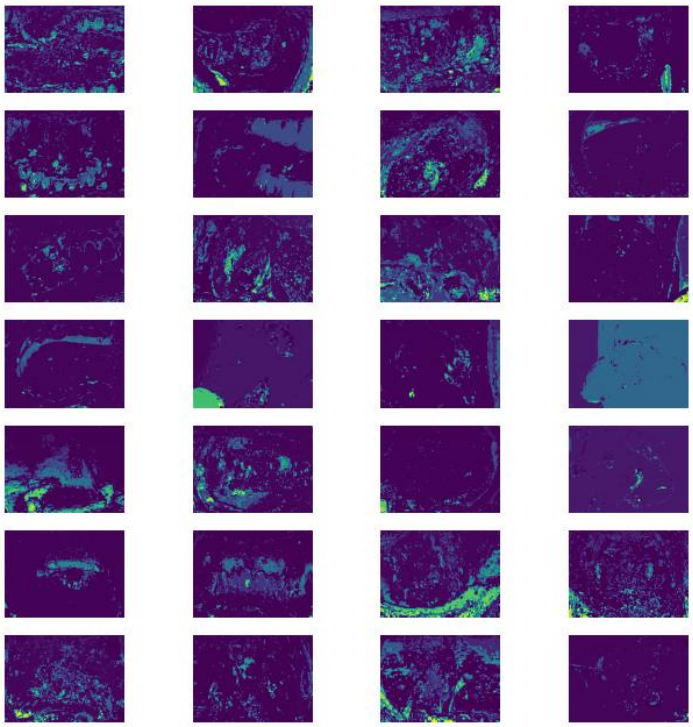
3. Đánh giá phân cụm trên các ảnh với thuật toán DBSCAN

Đối với thuật toán DBSCAN, chúng ta có hai tham số, eps (bán kính) và min_samples

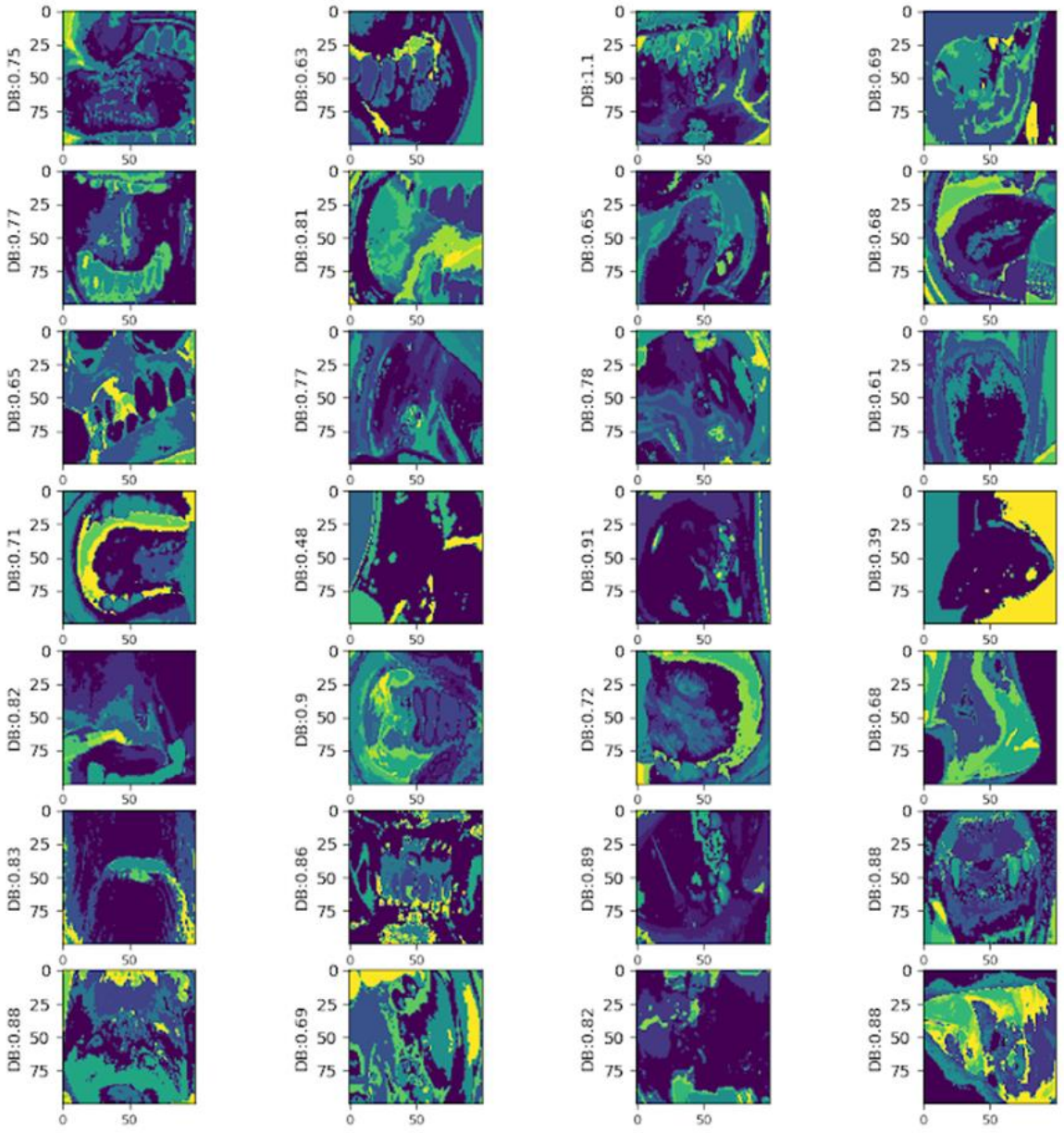
Để tìm được eps tối ưu, chúng em sử dụng thuật toán KNN (hàng xóm gần nhất) việc này giúp ta chọn được eps chung cho toàn bộ dữ liệu ảnh.



Nhận xét, ta thấy được khoảng eps hợp lý là khoảng 0.05. Vì vậy ta sẽ thử DBSCAN với eps = 0.05 và với min_sample = 5 để phân đoạn ảnh.



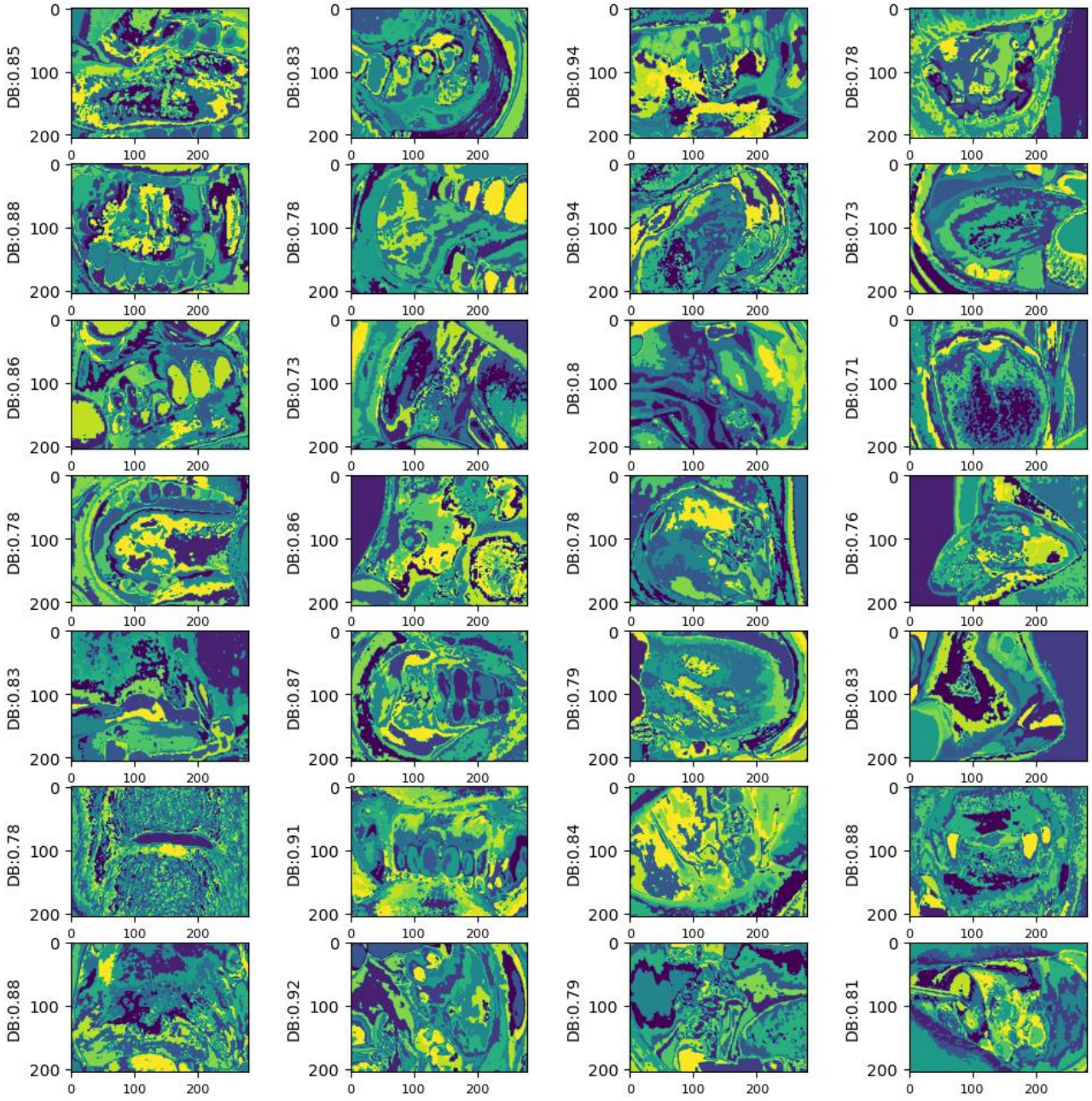
4. Đánh giá phân cụm trên các ảnh với thuật toán Mean-Shift



Đối với Mean Shift, thuật toán không yêu cầu trền tham số đầu vào. Tuy nhiên, khi không truyền tham số đầu vào, lượng tính toán cực lớn và đã xảy ra lỗi khi tính toán trên đa luồng. Chính vì vậy, chúng em đã tử truyền một tham số đầu: vào bandwidth=0.5. Ngoài ra, vì thuật toán có độ phức tạp khá cao nên chúng em đã giảm chiều từ dữ liệu gốc về 100 * 100 chiều. Lượng mất mát chi tiết khá nhiều.

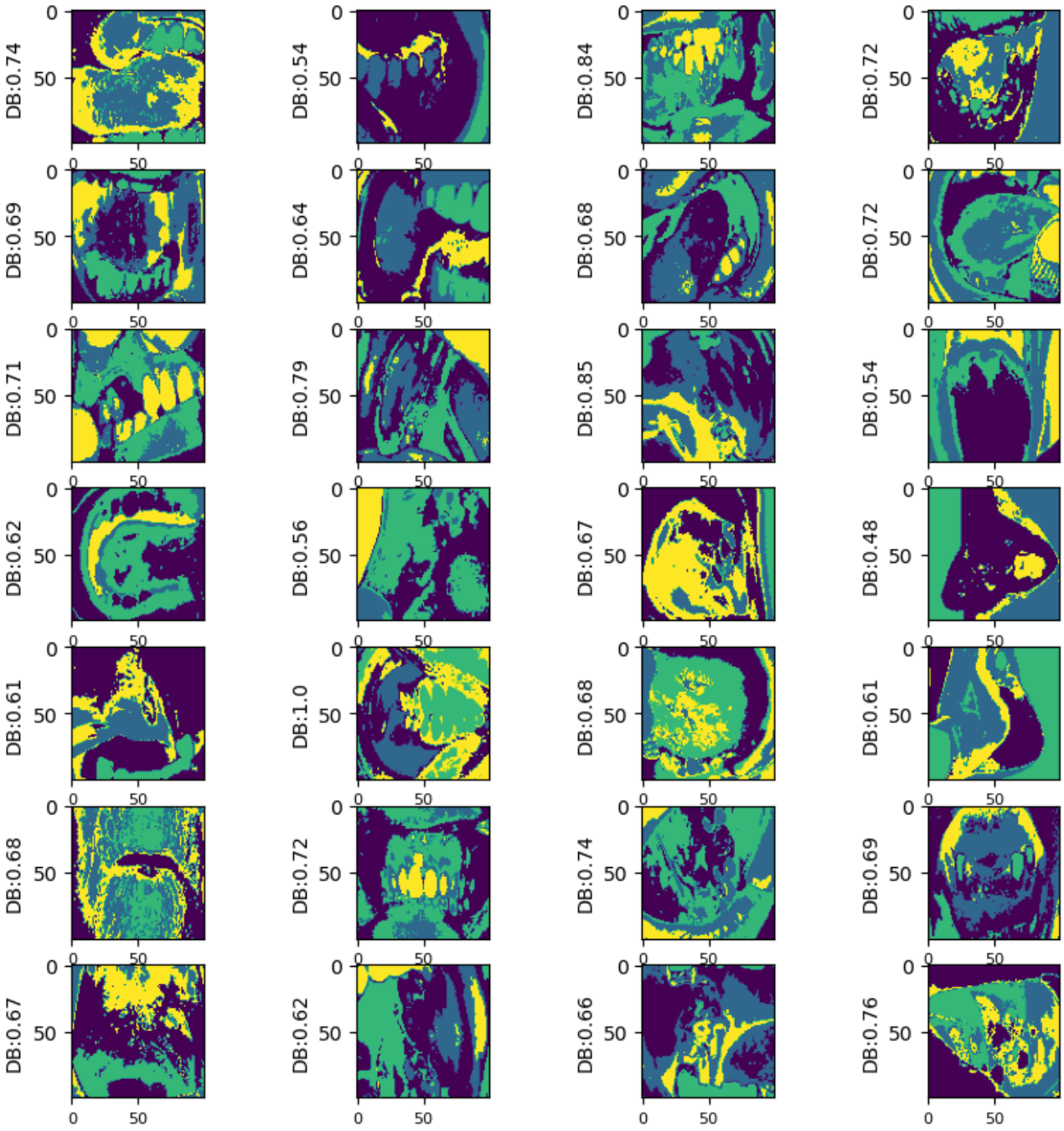
5. Đánh giá việc phân đoạn ảnh với các phương pháp K-Means, DBSCAN, Hierarchical, Mean-Shift bằng chỉ số Davies Bouldin

a. Đối với K-Means:



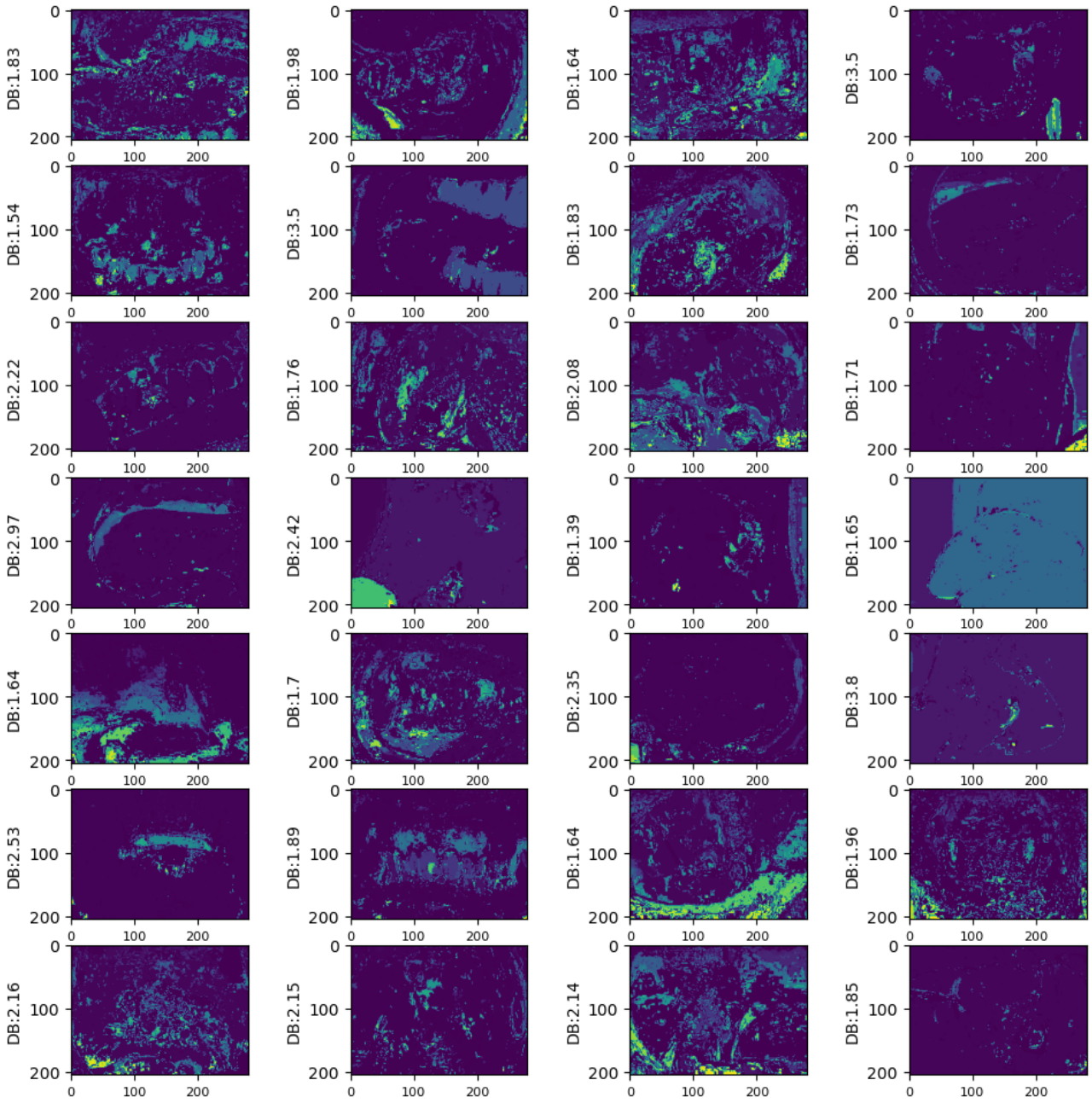
Từ kết quả trên, chúng ta có thể thấy kết quả của DB cho phương pháp K-Means khá ổn, các cụm cách xa nhau và ít phân tán chồng chéo. Các ảnh đều cho chỉ số DB nằm trong ngưỡng có thể chấp nhận được, từ 0.79 đến 0.94.

b. Đối với Hierarchical



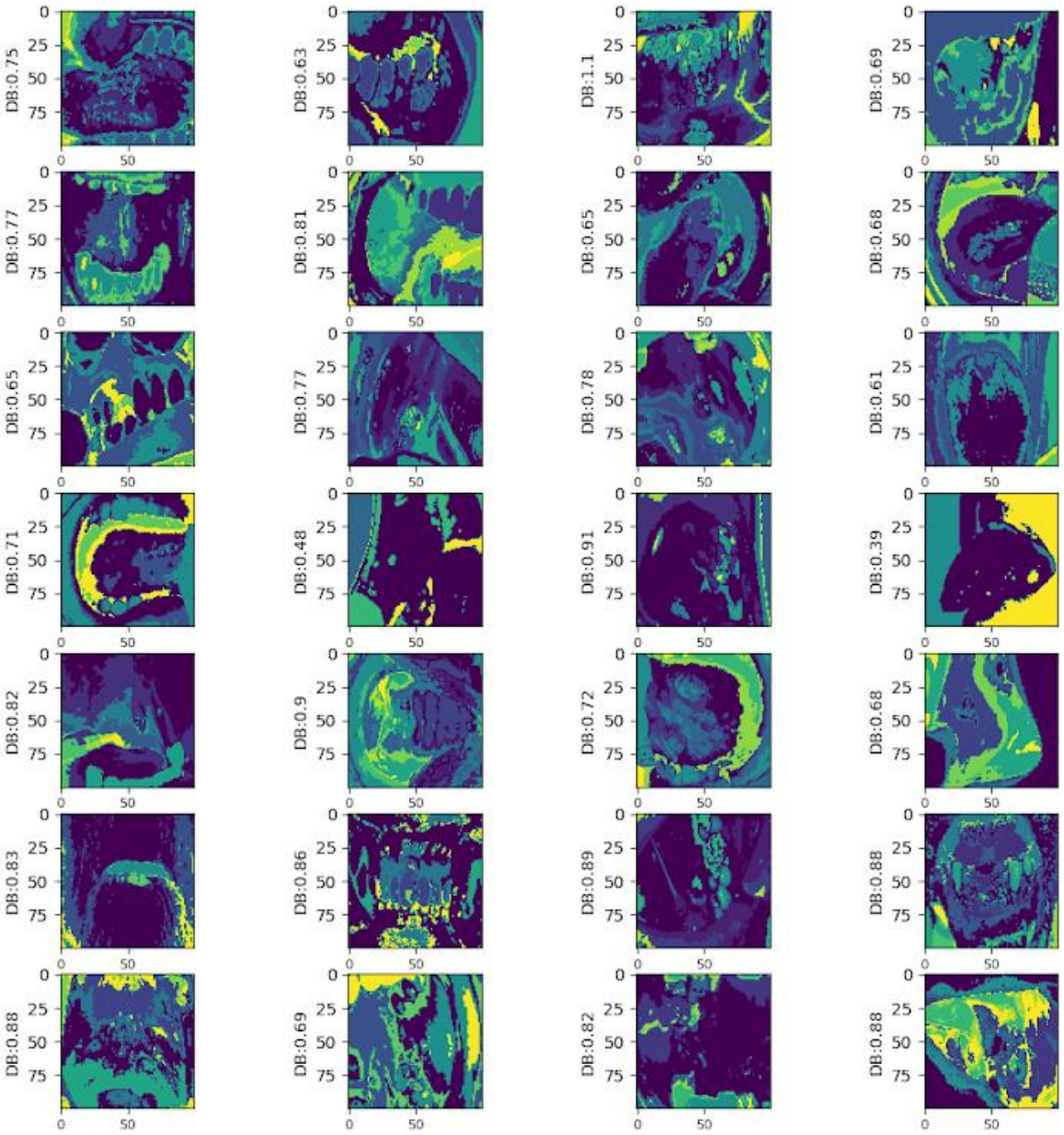
Từ kết quả trên, chúng ta có thể thấy rằng chỉ số DB thấp, ngoài ra có sự chênh lệch cao giữa các ảnh. Như vậy, một phần nào đó phương pháp này chưa thực sự hiệu quả với bộ dữ liệu.

c. Đối với DBSCAN



Từ kết quả trên, chúng ta có thể thấy rằng DB của DBSCAN lớn hơn so với thuật toán K-Means, tuy nhiên nó lại không đồng đều giữa các ảnh, sự chênh lệch đôi khi lên tới gấp đôi. Vì vậy, DBSCAN chưa thực sự phù hợp với bộ dữ liệu ảnh này.

d. Đối với Mean - Shift



Thời gian chạy cho mỗi ảnh là khoảng 1 tiếng, phương pháp này có độ phức tạp cao nhất, bù lại chỉ số DB khá ổn định giữa các ảnh. Tuy nhiên, vẫn không ổn định như K-Means.

6. So sánh các phương pháp phân cụm

Các phương pháp phân cụm	Thời gian chạy
K-Means	1 phút 38 giây
Agglomerative	22 phút 10 giây (ảnh đã giảm chiều tránh tràn bộ nhớ)
DBSCAN	1 phút 23 giây
Mean-Shift	Khoảng 60 phút/ảnh 29 phút (khi chạy đa luồng cho tất cả ảnh)

Thời gian chạy cao nhất là thuật toán Mean-Shift và nhanh nhất là K-Means. Đối với thuật toán Agglomerative và DBSCAN, thời gian chạy cũng khá lâu, Agglomerative có thời gian chạy gấp đôi DBSCAN.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. Kết luận

- Phân cụm K-Means cho thời gian nhanh nhất.
- David Bouldin của K-Means cho chất lượng phân cụm ổn định nhất.
- Mô hình CNN tốt nhất với độ chính xác là 71%.
- Giao diện đã đáp ứng được các nhu cầu cơ bản của người dùng.

2. Hướng phát triển

Trong thời gian tới, thứ nhất hiện tại độ chính xác còn chưa ổn định nên nhóm sẽ hướng tới cải thiện điều này đồng thời tăng kích thước của bộ dữ liệu. Tiếp đến, hiện tại trang web của dự án đang dừng lại ở việc trả ra ảnh kèm độ chính xác. Vì vậy nhóm chúng em dự định sẽ tích hợp mô hình vào một ứng dụng web tốt hơn, giao diện bắt mắt, thân thiện với người dùng để đáp ứng tốt hơn cho người dùng trong quá trình xác định bệnh tình. Việc xây dựng một ứng dụng web giúp cho việc truy cập và sử dụng mô hình trở nên dễ dàng hơn. Người dùng có thể tải lên hình ảnh của mình trên ứng dụng web và nhận kết quả chẩn đoán tự động một cách tiết kiệm thời gian.

TÀI LIỆU THAM KHẢO

[1] Jiawei Han, Micheline Kamber, Jian Pei (2012). *Data Mining Concepts and Techniques*, USA: Morgan Kaufmann.

[2] Jérémie du Boisberranger, Joris Van den Bossche, Loïc Estève (2007). *Scikit-learn Machine Learning in Python*. Truy cập ngày 25/5/2023, từ https://scikit-learn.org/stable/

[3] Pham Dinh Khanh, (2019). *Khoa học dữ liệu*. Truy cập ngày 25/5/2023, từ https://phamdinhkhanh.github.io/content

[4] Sandeep Jain, (2018). *Geeks for Geeks*. Truy cập ngày 25/5/2023, từ https://www.geeksforgeeks.org/

[5] Tiep Vu Huu, (2018). *Machine Learning cơ bản*. Truy cập từ ngày 25/5/2023, từ https://machinelearningcoban.com/