**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI: Tìm hiều về thuật toán phân cụm Spectral Clustering**

**Giáo viên hướng dẫn: TS. BÙI HẢI PHONG**

**Sinh viên thực hiện:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Stt** | **Mã sv** | **Họ và tên** | **Lớp** |
| 1 | 1671020200 | Lê Đức Mạnh | CNTT-1601 |
| 2 | 1671020307 | Nguyễn Xuân Thuận | CNTT-1601 |
| 3 | 1671020291 | Dương Xuân Thành | CNTT-1601 |

**Hà Nội, năm 2024**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI: Tìm hiểu về thuật toán phân cụm Spectral Clustering**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Điểm | |
| Bằng Số | Bằng Chữ |
| 1 | 1671020200 | Lê Đức Mạnh |  |  |  |
| 2 | 1671020307 | Nguyễn Xuân Thuận | 06/11/2004 |  |  |
| 3 | 1671020291 | Dương Xuân Thành |  |  |  |

### 

### CÁN BỘ CHẤM THI 1 CÁN BỘ CHẤM THI 2

**Hà Nội, năm 2024**

**LỜI NÓI ĐẦU**

(Nếu có)

**MỤC LỤC**

**Tìm hiểu thuật toán phân cụm Spectral Clustering**

A network of colored circles and lines

Description automatically generated

Phân cụm đồ thị là một lĩnh vực trong phân tích cụm nhằm tìm kiếm các nhóm đỉnh có liên quan trong một đồ thị. Phân cụm đồ thị cho kết quả trong mỗi cụm các đỉnh có nhiều cạnh kết nối gần, trong khi giữa các cụm thì chỉ có vài cạnh kết nối. Thuật toán Spectral Clustering sử dụng thông tin từ các giá trị riêng (phổ) của các ma trận đặc biệt được xây dựng từ đồ thị hoặc tập dữ liệu, nên nó có tên là spectral. Thuật toán xây dựng một đồ thị tương tự, chiếu dữ liệu lên không gian chiều thấp hơn và phân cụm dữ liệu. Spectral Clustering có ứng dụng trong nhiều lĩnh vực bao gồm: phân đoạn hình ảnh, khai thác dữ liệu giáo dục, phân giải thực thể, tách giọng nói, phân cụm quang phổ của chuỗi protein, phân đoạn hình ảnh văn bản.

**1.Bài toán phân cụm**

**1.1 Học máy (Machine learning)**

    Học máy hay còn gọi với cái tên Tiếng Anh là Machine Learning . Có 2 định nghĩa về Machine Learning được cung cấp. Theo Arthur Samuel mô tả: “Lĩnh vực nghiên cứu mang lại cho máy tính khả năng học hỏi mà không cần được lập trình rõ ràng.” Đây là một định nghĩa cũ, không chính thức.Tom Mitchell đưa ra một định nghĩa hiện đại và rõ ràng hơn:  
*“Một chương trình máy tính được cho là học hỏi từ kinh nghiệm E đối với một số loại nhiệm vụ T và thước đo hiệu suất P, nếu hiệu suất của nó ở các nhiệm vụ trong T, được đo bằng P, cải thiện theo kinh nghiệm E.”*  
Ví dụ: chơi cờ caro.  
E = kinh nghiệm chơi nhiều ván cờ caro  
T = nhiệm vụ chơi cờ caro.  
P = xác suất chương trình sẽ thắng trong trò chơi tiếp theo.  
    Theo phân nhóm dựa theo phương thức học, Machine learning thường được chia thành 4 loại:

* Học có giám sát (Supervised learning): thuật toán dự đoán đầu ra (outcome) của một dữ liệu mới (new input) dựa trên các cặp (input, outcome) đã biết từ trước.
* Học không giám sát (Unsupervised learning): chỉ có dữ liệu vào X mà không biết label Y tương ứng. Chúng ta không biết được đầu ra hay nhãn mà chỉ có dữ liệu đầu vào. Thuật toán unsupervised learning sẽ dựa vào cấu trúc của dữ liệu để thực hiện một công việc nào đó.
* Học bán giám sát (Semi-Supervised Learning): chúng ta có một lượng lớn dữ liệu X nhưng chỉ một phần trong chúng được gán nhãn.
* Học củng cố (Reinforcement Learning): các bài toán giúp cho một hệ thống tự động xác định hành vi dựa trên hoàn cảnh để đạt được lợi ích cao nhất (maximizing the performance).A diagram of a machine learning

  Description automatically generated

**1.2 Phân cụm (Clustering)**

    Phân cụm (Clustering) thuộc loại học không giám sát (Unsupervised learning) là một dữ liệu là bài toán gom nhóm các đối tượng dữ liệu vào thánh từng cụm (cluster) sao cho các đối tượng trong cùng một cụm có sự tương đồng theo một tiêu chí nào đó. Ví dụ: phân nhóm khách hàng dựa trên hành vi mua hàng. Điều này cũng giống như việc ta đưa cho một đứa trẻ rất nhiều mảnh ghép với các hình thù và màu sắc khác nhau, ví dụ tam giác, vuông, tròn với màu xanh và đỏ, sau đó yêu cầu trẻ phân chúng thành từng nhóm. Mặc dù không cho trẻ biết mảnh nào tương ứng với hình nào hoặc màu nào, nhiều khả năng chúng vẫn có thể phân loại các mảnh ghép theo màu hoặc hình dạng. Đặc điểm của phân cụm: -Số cụm dữ liệu không được biết trước -Có nhiều các tiếp cận, mối cách lại có vài kỹ thuật -Các kỹ thuật khác nhau thường mang lại kết quả khác nhau.

**2.Thuật toán Spectral Clustering**

**2.1 Mã giả**

A white background with black text

Description automatically generated

1. **Tính ma trận kề**

Cho đồ thị G = (V, E) với V là tập gồm các đỉnh và E là tập gồm các cạnh. Mỗi cạnh thuộc E sẽ gồm 2 đỉnh trong tập V - một cặp đỉnh (𝑣\_𝑖, 𝑣\_𝑗), với trọng số cạnh là 𝑤\_𝑖𝑗.

**2. Tính ma trận Laplacian**

Ma trận laplacian: **L = D-A** với D là ma trận bậc. Ma trận bậc D được tính từ ma trận kề A, có số chiều giống với ma trận A. Mỗi phần tử trên đường chéo chính của ma trận bậc D là tổng của của các phần tử trên một hàng của ma trận A tương ứng. Các phần tử

khác ngoài đường chéo chính đều bằng 0.

A screenshot of a math book

Description automatically generated

**3. Tính k vector riêng đầu tiên của ma trận Laplacian**  
Cho một ma trận A, ta có 𝜆 là một giá trị riêng và 𝜈 là vector riêng của A nếu:  
**A𝝂 = 𝝀𝝂**  
Cho một đồ thị G có n nút, ma trận kề của nó sẽ có n giá trị riêng {𝜇\_1, 𝜇\_2, 𝜇\_3…, 𝜇\_𝑛} với 𝜇\_1 ≥ 𝜇\_2 ≥ 𝜇\_3 ≥…≥ 𝜇\_𝑛 và n vector riêng {𝑥\_1, 𝑥\_2, 𝑥\_3,…, 𝑥\_𝑛}

Ở đây ta sẽ đi tính véc tơ riêng và giá trị riêng cho ma trận Laplacian, sau đó lấy ra k vector đầu tiên. Có thể nói việc tính ma trận Laplacian và tính k giá trị riêng và vector riêng của ma trận này là trái tim của thuật toán Spectral Clustering. Các giá trị riêng cho biết các thuộc tính toàn cục không rõ ràng của đồ thị từ cấu trúc cạnh. Xét đồ thị Laplacian của G, 𝐿\_𝐺 có tập giá trị riêng {𝜆\_1, 𝜆\_2, 𝜆\_3,…, 𝜆\_𝑛} và tập vector {𝑥\_1,𝑥\_2,𝑥\_3,…,𝑥\_𝑛}:

A table with numbers and equations

Description automatically generated with medium confidence

* Nếu 0 là giá trị riêng của L (𝜆\_1=𝜆\_2=…=𝜆\_𝑘=0) với k vector riêng thì đồ thị G có k kết nối thành phần
* Nếu đồ thị đã được kết nối, 𝜆\_2 > 0 và 𝜆\_2 là kết nối đại số của G, 𝜆\_2 càng lớn thì G càng có nhiều kết nối.

A diagram of a number

Description automatically generated

Dễ thấy rằng L\_G có λ\_1 = λ\_2 = λ\_3 = 0 và λ\_4 > 0 nên G có 3 đường kết nối. L\_G1 và L\_G2 có λ\_1 = 0 nên 2 đồ thị G1 và G2 có một đường kết nối.  
Lại có λ\_2 (L\_G1 ) < λ\_2 (L\_G2) nên G2 có nhiều đường kết nối hơn.

**4. Sử dụng K-means để phân cụm**  
Sắp xếp thành ma trận U với k véc tơ riêng đầu tiên của ma trận L và các đỉnh của đồ thị G (ký hiệu v trong hình là véc tơ riêng, không phải đỉnh). Sau đó áp dụng phân cụm đối với ma trận U sử dụng thuật toán K-mean để phân các đỉnh của G (hàng của U).

A diagram of a number of objects

Description automatically generated with medium confidence

**Thuật toán K-mean**  
Input: K (số lượng các cụm), Training set {𝑥^((1)), 𝑥^((2)),…,𝑥^((𝑚)) } 𝑥^((𝑖))∈ℝ^𝑛  
Output:  Các center 𝜇\_𝐾 và label cho từng điểm dữ liệu 𝑥^((𝑖)).

1. Chọn K điểm bất kỳ làm các center ban đầu.
2. Phân mỗi điểm dữ liệu vào cụm có center gần nó nhất.
3. Nếu việc gán dữ liệu vào từng cụm ở bước 2 không thay đổi so với vòng lặp trước nó thì ta dừng thuật toán.
4. Cập nhật center cho từng cụm bằng cách lấy trung bình cộng của tất các các điểm dữ liệu đã được gán vào cluster đó sau bước 2.
5. Quay lại bước 2.

**2.2 Ưu và nhược điểm của thuật toán**

     Spectral Clustering giúp chúng ta khắc phục hai vấn đề chính trong phân cụm: một là hình dạng của cụm và vấn đề khác là xác định tâm của cụm. Thuật toán K-mean thường giả định rằng các cụm là hình cầu hoặc tròn, dùng nhiều lần lặp để xác định tâm cụm và phân các điểm. Trong spectral clustering, các cụm không tuân theo một hình dạng hoặc khuôn mẫu cố định. Các điểm ở xa nhau nhưng được kết nối thuộc cùng một cụm và các điểm ít xa nhau hơn có thể thuộc các cụm khác nhau nếu chúng không được kết nối. Điều này có nghĩa là thuật toán có thể hiệu quả đối với dữ liệu có hình dạng và kích thước khác nhau.  
     Khi so sánh với các thuật toán khác, nó nhanh về mặt tính toán đối với các tập dữ liệu thưa thớt vài nghìn điểm dữ liệu mặc dù có thể tốn kém để tính toán cho các tập dữ liệu lớn vì các giá trị riêng và vector cần được tính toán và sau đó mới thực hiện phân cụm.

**3. Cài đặt thuật toán**

* Code python cài đặt thuật toán từng bước theo mã giả ở trên, mình đã trình bày chi tiết trên Google Colab: [https://drive.google.com/file/d/1tua2gx4J7k8jhVCkm8-7gjFVx-j0wqrc/view?usp=sharing](https://drive.google.com/file/d/1tua2gx4J7k8jhVCkm8-7gjFVx-j0wqrc/view?usp=sharing" \t "_blank)
* Khi đã nắm chắc, các bạn có thể sử dụng hàm có sẵn của thư viện Sklearn cho thuật toán: [https://scikit-learn.org/stable/modules/generated/sklearn.cluster.SpectralClustering.html](https://scikit-learn.org/stable/modules/generated/sklearn.cluster.SpectralClustering.html" \t "_blank)

**4. Bài toán**

**4.1. Ví dụ về bài toán**

4.1.1.Ảnh minh họa

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

4.1.2. Code

* Lưu ý: Trước khi chạy được thuật toán phải cài đặt môi trường bằng câu lệnh: ***pip install numpy scikit-learn matplotlib***

import numpy as np

import matplotlib.pyplot as plt

from sklearn.datasets import make\_blobs

from sklearn.cluster import SpectralClustering

# Tạo tập dữ liệu mẫu

X, y = make\_blobs(n\_samples=300, centers=4, cluster\_std=0.60, random\_state=0)

# Áp dụng Spectral Clustering

spectral\_model = SpectralClustering(n\_clusters=4, affinity='nearest\_neighbors', random\_state=0)

labels = spectral\_model.fit\_predict(X)

# Vẽ biểu đồ kết quả

plt.scatter(X[:, 0], X[:, 1], c=labels, cmap='viridis')

plt.title('Spectral Clustering')

plt.show()

* 1. **.Bài toán về phân cụm loài hoa Iris**



4.2.1. Ảnh minh họa:

A screenshot of a computer screen

Description automatically generated

* + 1. Code

import numpy as np

import matplotlib.pyplot as plt

from sklearn.datasets import load\_iris

from sklearn.decomposition import PCA

from sklearn.cluster import SpectralClustering

# Load dữ liệu Iris

iris = load\_iris()

X = iris.data

y = iris.target

# Áp dụng PCA để giảm chiều dữ liệu xuống còn 2 chiều để có thể vẽ biểu đồ

pca = PCA(n\_components=2)

X\_pca = pca.fit\_transform(X)

# Áp dụng Spectral Clustering

spectral\_model = SpectralClustering(n\_clusters=3, affinity='nearest\_neighbors', random\_state=0)

labels = spectral\_model.fit\_predict(X)

# Vẽ biểu đồ kết quả

plt.figure(figsize=(10, 5))

plt.subplot(1, 2, 1)

plt.scatter(X\_pca[:, 0], X\_pca[:, 1], c=y, cmap='viridis')

plt.title('Phân cụm thực tế')

plt.subplot(1, 2, 2)

plt.scatter(X\_pca[:, 0], X\_pca[:, 1], c=labels, cmap='viridis')

plt.title('Phân cụm bằng Spectral Clustering')

plt.show()

4.2.3.Giải thích, ứng dụng:

Trong đoạn mã trên:

* Chúng ta sử dụng tập dữ liệu Iris từ Scikit-learn.
* Sử dụng PCA để giảm chiều dữ liệu xuống còn 2 chiều để có thể vẽ biểu đồ.
* Áp dụng Spectral Clustering với 3 cụm và hiển thị kết quả phân cụm trên biểu đồ.
* Kết quả sẽ hiển thị hai biểu đồ: một là phân cụm thực tế của các loài hoa Iris và hai là kết quả phân cụm sử dụng Spectral Clustering.
* Lưu ý rằng, khi áp dụng Spectral Clustering vào bài toán này, chúng ta không cần biết trước số lượng cụm (loài hoa) mà thuật toán sẽ tự động xác định dựa trên đặc trưng của dữ liệu. Trong ví dụ này, tôi đã sử dụng 3 cụm vì chúng ta có biết từ tập dữ liệu Iris là có 3 loài hoa khác nhau. Tuy nhiên, trong thực tế, chúng ta có thể không biết trước số lượng cụm và Spectral Clustering có thể hữu ích trong việc xác định số lượng cụm một cách tự động.

Trong bài toán phân cụm loài hoa Iris, chúng ta có một tập dữ liệu gồm các đặc trưng của các loài hoa Iris, bao gồm chiều dài và chiều rộng của lá đài và cánh hoa. Mỗi mẫu dữ liệu trong tập dữ liệu này đã được gán nhãn với loại hoa tương ứng: Setosa, Versicolor và Virginica. Chúng ta muốn sử dụng thuật toán Spectral Clustering để phân cụm các mẫu dữ liệu này mà không cần biết trước nhãn của chúng.

Khi thực hiện Spectral Clustering trên tập dữ liệu Iris, kết quả sẽ hiển thị hai biểu đồ như sau:

1. Biểu đồ phân cụm thực tế của các loài hoa Iris:

A screen shot of a graph

Description automatically generated

Trong biểu đồ này, chúng ta sử dụng thông tin nhãn đã biết trước để tạo ra biểu đồ phân cụm thực tế. Mỗi điểm dữ liệu trên biểu đồ được biểu diễn bởi một điểm trên không gian 2D hoặc 3D (tùy thuộc vào số chiều đặc trưng được sử dụng), với mỗi loại hoa có một màu sắc khác nhau. Điều này giúp chúng ta xác định được cách mà các loài hoa được phân chia trong không gian đặc trưng.

1. Biểu đồ kết quả phân cụm sử dụng Spectral Clustering:

A screen shot of a graph

Description automatically generated

Biểu đồ này hiển thị kết quả của Spectral Clustering sau khi áp dụng thuật toán lên tập dữ liệu Iris. Các điểm dữ liệu được phân thành các cụm khác nhau, mỗi cụm có thể được đánh dấu bằng một màu sắc hoặc ký hiệu riêng. Kết quả này cho thấy cách mà Spectral Clustering đã tự động phân loại các mẫu dữ liệu dựa trên cấu trúc không gian của chúng.

Kết luận: So sánh giữa hai biểu đồ này có thể giúp chúng ta đánh giá hiệu quả của Spectral Clustering trong việc phân cụm dữ liệu Iris so với phân cụm thực tế đã biết trước. Điều này có thể giúp chúng ta hiểu được độ chính xác và tính linh hoạt của thuật toán trong các bài toán phân cụm thực tế.

Dưới đây là một tổng hợp về cách ứng dụng Spectral Clustering vào bài toán này:

* Chuẩn bị dữ liệu: Thu thập và tiền xử lý dữ liệu loài hoa Iris, bao gồm các đặc trưng như chiều dài và chiều rộng của lá đài và cánh hoa.
* Xây dựng ma trận tương tự: Sử dụng các độ đo như Euclidean distance hoặc cosine similarity để tạo ra ma trận tương tự (similarity matrix) từ dữ liệu.
* Xây dựng ma trận Laplacian: Tính toán ma trận Laplacian từ ma trận tương tự, thường thông qua các phương pháp như Laplacian Eigenmaps.
* Phân cụm bằng Spectral Clustering: Áp dụng thuật toán Spectral Clustering vào ma trận Laplacian để phân chia dữ liệu thành các nhóm.
* Đánh giá kết quả: Sử dụng các chỉ số đánh giá như Adjusted Rand Index (ARI), Silhouette Score, hoặc Davies-Bouldin Index để đánh giá chất lượng của các cụm đã được tạo ra.

Hiển thị kết quả: Sử dụng biểu đồ để so sánh kết quả phân cụm của Spectral Clustering với phân phối thực tế của các loài hoa Iris.

Thông qua quy trình này, chúng ta có thể hiểu rõ hơn về cách Spectral Clustering được áp dụng để phân cụm dữ liệu và đưa ra kết quả phân cụm trực quan và thống kê, giúp chúng ta hiểu hơn về cấu trúc dữ liệu và tự động phân loại chúng thành các nhóm có ý nghĩa.

**5. Tài liệu tham khảo**

* [https://www.analyticsvidhya.com/blog/2021/05/what-why-and-how-of-spectral-clustering/](https://www.analyticsvidhya.com/blog/2021/05/what-why-and-how-of-spectral-clustering/" \t "_blank)
* [https://www.kdnuggets.com/2020/05/getting-started-spectral-clustering.html](https://www.kdnuggets.com/2020/05/getting-started-spectral-clustering.html" \t "_blank)
* [https://machinelearningcoban.com/2017/01/01/kmeans/](https://machinelearningcoban.com/2017/01/01/kmeans/" \t "_blank)
* [https://machinelearningcoban.com/2016/12/26/introduce/](https://machinelearningcoban.com/2016/12/26/introduce/" \t "_blank)