

Báo cáo bài tập thực hành 1

CLUSTERING

Trịnh Mẫn Hoàng - 14520320

GV: TS. Lê Đình Duy

Môn học: Máy học trong thị giác máy tính

Mục Lục

Mục Lục Ảnh	2
I. Giới thiệu bài toán clustering	3
a. Input	3
b. Output [1]	3
II. Giới thiệu thuật toán	3
a. Kmeans clustering [2]	3
i. Ý tưởng:	3
ii. Các bước thực hiện	3
b. Spectral clustering [1]	3
i. Ý tưởng	3
ii. Các bước thực hiện:	3
c. DBSCAN [3]	4
i. Ý tưởng	4
ii. Các bước thực hiện	4
d. Agglomerative [4]	4
i. Ý tưởng	4
ii. Các bước thực hiện	4
III. Thực nghiệm [5]	5
Link Github: https://github.com/HoangTrinh/Machine-Leaning-in-Computer-Vision.git	5
a. Dùng dữ liệu point tự tạo.	5
i. Dữ liệu:	5
ii. Kết quả:	5
b. Dùng bộ dữ liệu chữ số viết tay	6
i. Dữ liệu	6
ii. Kết quả	6
iii. Đánh giá và so sánh	6
c. Dùng bộ dữ liệu face	6
i. Dữ liệu	6
ii. Các bước thực hiện	6
iii. Kết quả	7
iv. Đánh giá và so sánh	7
d. Dùng bộ dữ liệu và phương pháp rút trích đặc trưng tự chọn	8
i. Dữ liệu	8
ii. Các bước thực hiện:	8
iii. Kết quả	8
iv. Đánh giá và so sánh	8
Tài Liệu Tham Khảo	10

Mục Lục Ảnh

Figure 1 - Hand digits clustering result	6
Figure 2 - LBP face result	7
Figure 3 - HOG face result	8

I. Giới thiệu bài toán clustering

a. Input

- Tập dữ liệu không dán nhãn

b. Output [1]

- Tập dữ liệu đã được phân chia thành các cụm (cluster)
- Dữ liệu trong cùng một cluster có tính chất giống nhau

II. Giới thiệu thuật toán

a. Kmeans clustering [2]

i. Ý tưởng:

- Tìm cách phân nhóm các đối tượng (objects) đã cho vào K cụm (K là số các cụm được xác định trước, K nguyên dương) sao cho tổng bình phương khoảng cách giữa các đối tượng đến tâm nhóm (centroid) là nhỏ nhất.

ii. Các bước thực hiện

- Bước 1:
Chọn ngẫu nhiên K tâm (centroid) cho K cụm (cluster). Mỗi cụm được đại diện bằng các tâm của cụm.
- Bước 2:
Tính khoảng cách giữa các đối tượng (objects) đến K tâm (thường dùng khoảng cách Euclidean)
- Bước 3:
Nhóm các đối tượng vào nhóm gần nhất
- Bước 4:
Xác định lại tâm mới cho các nhóm (dùng means hoặc median)
- Bước 5:
Thực hiện lại bước 2 cho đến khi không có sự thay đổi nhóm nào của các đối tượng (hội tụ)

b. Spectral clustering [1]

i. Ý tưởng

- Tìm cách đưa các đối tượng về dạng đồ thị tương đồng, sau đó dùng k dimensions để phân hoạch đồ thị

ii. Các bước thực hiện:

- Bước 1:
Chọn hàm tính độ tương đồng kernel(x,y)
- Bước 2:
Xây dựng đồ thị với mỗi node là một đối tượng và các cạnh chứa giá trị tương đồng $w_{ij} = \text{kernel}(X_i, X_j)$

- Bước 3:
Tính toán đồ thị Laplacian: $L = D - W$
Trong đó : D là ma trận đường chéo của ma trận G (Ma trận kề của W), với $d_{ij} = \deg[i]$
- Bước 4:
Tìm eigenvector e tương ứng với eigenvalue nhỏ thứ hai cho một trong hai phương trình sau:
 $Lf = \lambda Df$ normalized
 $Lf = \lambda f$ unnormalized
- Bước 5:
Kết luận các nhóm được phân hoạch dựa trên eigenvalue

c. DBSCAN [3]

- i. Ý tưởng
 - Gom nhóm các điểm chứa lẫn nhau và chứa đủ nhiều hơn một ngưỡng, nếu thấp hơn ngưỡng, xem như là một noise.
- ii. Các bước thực hiện
 - Bước 1:
Tìm số lượng neighbors cho mỗi điểm trong một phạm vi ϵ , chọn các điểm có số lượng neighbors $\geq \min PTs$ (định trước) làm core point
 - Bước 2:
Tìm tất cả các điểm có liên kết với core points trong đồ thị biểu diễn neighbors, bỏ qua các điểm non-core
 - Bước 3:
Đưa các điểm non-core vào cluster gần nhất trong phạm vi ϵ , nếu không có, xem như là noise.

d. Agglomerative [4]

- i. Ý tưởng
 - Đi từ bottom-up, gom nhóm các clusters gần nhau nhất cho đến khi chỉ còn 1 cluster
- ii. Các bước thực hiện
 - Bước 1:
Xem mỗi đối tượng là 1 cluster
 - Bước 2:
Tính khoảng cách từng đôi một giữa các cluster
 - Bước 3:
Tạo một ma trận khoảng cách từ bước 2
 - Bước 4:
Tìm cặp cluster có khoảng cách gần nhất
 - Bước 5:

Xóa cặp trên ra khỏi ma trận và nhóm lại

- Bước 6:
Tính khoảng cách từ cluster mới tới các cluster đã biết, cập nhật ma trận
- Bước 7:
Lặp lại bước 4 cho đến khi ma trận chỉ còn một phần tử

III. Thực nghiệm [5]

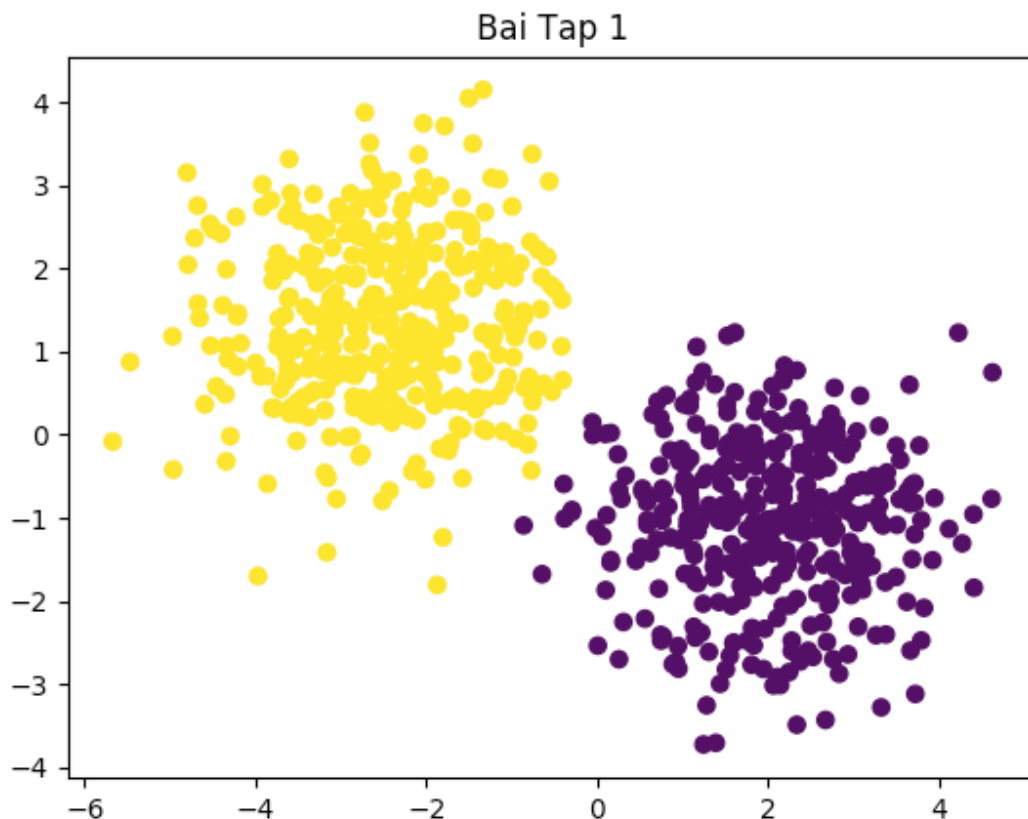
Link Github: <https://github.com/HoangTrinh/Machine-Leaning-in-Computer-Vision.git>

a. Dùng dữ liệu point tự tạo.

i. Dữ liệu:

- Tạo random 750 điểm dữ liệu dùng hàm `mat_blobs` của `sklearn.datasets`, chia làm 2 Gaussians

ii. Kết quả:



b. Dùng bộ dữ liệu chữ số viết tay

i. Dữ liệu

- Dùng bộ dữ liệu lấy được từ hàm `load_digits` của `sklearn.datasets`, bao gồm 10 nhóm khác nhau

ii. Kết quả

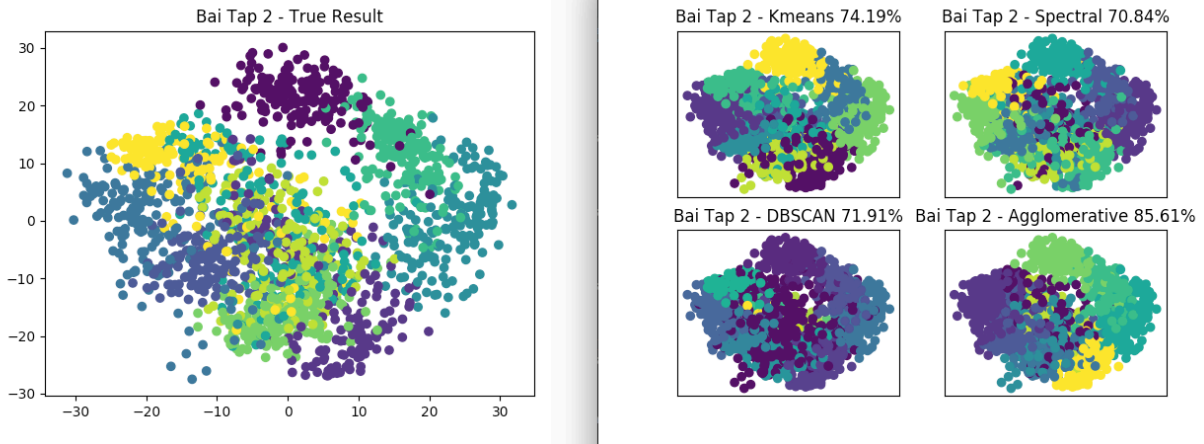


Figure 1 - Hand digits clustering result

iii. Đánh giá và so sánh

- Đánh giá bằng cách sử dụng hàm `metric_adjusted_mutual_info_score()`
- Các thuật toán cho kết quả gần tương đồng với kết quả thực
- Kmeans không thể xử lý các trường hợp outlier (vị trí (-10,-20) và xung quanh vị trí (0,0) nơi có một vài điểm thuộc cluster khác)
- Kmeans, Spectral, DBSCAN mang lại kết quả có độ chính xác gần như nhau
- Agglomerative là thuật giải mang lại kết quả gần như tốt nhất trong cả 4 thuật giải đã cài đặt trong trường hợp này

c. Dùng bộ dữ liệu face

i. Dữ liệu

- Dùng bộ dữ liệu lấy được từ hàm `fetch_lfw_people` của `sklearn.datasets`, mỗi nhóm sẽ có trên 40 ảnh, giảm size ảnh xuống còn 40% so với ảnh gốc
- Dùng Local Binary Pattern [6] để xử lý
- Vì không biết rõ số lượng nhóm thực của ảnh nên ta sẽ nhóm một cách tượng trưng là 7 nhóm

ii. Các bước thực hiện

- Bước 1:
Load ảnh từ bộ dữ liệu

- Bước 2:
Dùng hàm `local_binary_pattern` thuộc thư viện `skimage` để trích xuất đặc trưng LBP
Tính toán histogram cho ảnh LBP đã được rút trích bên trên -> feature
- Bước 3:
Lưu tập features được trích xuất dưới dạng file numpy: `data.npy`
Lưu kết quả đúng được thư viện hỗ trợ dưới dạng file numpy: `target.npy`
- Bước 4:
Mỗi khi chạy chương trình: Load hai file: `data.npy`, `target.npy`
- Bước 5:
Dùng các thuật giải để phân cụm dữ liệu trên, visualize kết quả, xem xét, đánh giá kết quả

iii. Kết quả

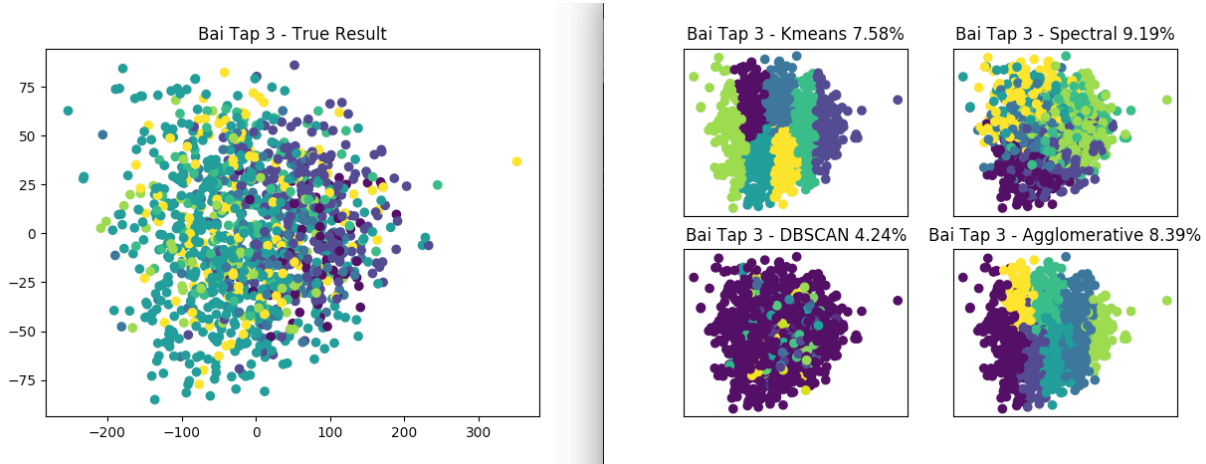


Figure 2 - LBP face result

iv. Đánh giá và so sánh

- Đánh giá bằng cách sử dụng hàm `metric_adjusted_mutual_info_score()`
- Hầu hết các thuật toán đều cho kết quả không tốt (<10% chính xác)
- Vì dữ liệu feature có xu hướng gom về phía phải đồ thị và tản ra về phía còn lại nên thuật toán phân cụm dựa vào neighbor như DBSCAN mang lại kết quả không tốt vì liên tục nhận được noise từ các cluster khác và nhập nhằng của core points.
- Spectral mang lại kết quả cao nhất trong 4 thuật toán nhờ khả năng tính toán trên ma trận tương đồng.

d. Dùng bộ dữ liệu và phương pháp rút trích đặc trưng tự chọn

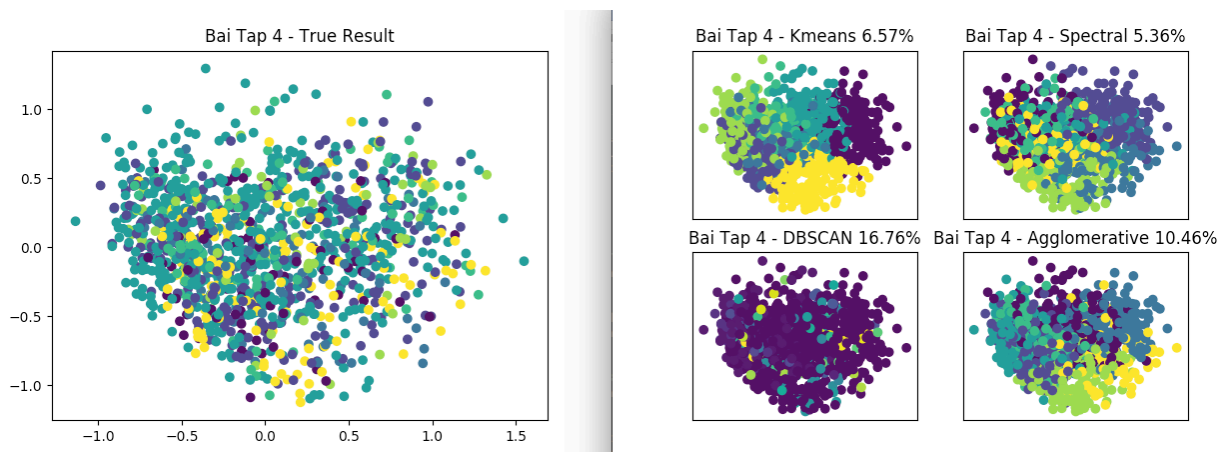
i. Dữ liệu

- Dùng bộ dữ liệu lấy được từ hàm `fetch_lfw_people` của `sklearn.datasets`, mỗi nhóm sẽ có trên 40 ảnh, giảm size ảnh xuống còn 40% so với ảnh gốc
- Dùng Histogram of Oriented Gradients [6] để xử lý

ii. Các bước thực hiện:

- Bước 1:
Load ảnh từ bộ dữ liệu
- Bước 2:
Dùng hàm `hog` thuộc thư viện `skimage` để trích xuất đặc trưng HOG
- Bước 3:
Lưu tập features được trích xuất dưới dạng file numpy: `data.npy`
- Bước 4:
Mỗi khi chạy chương trình: Load hai file: `data.npy`
- Bước 5:
Dùng các thuật giải để phân cụm dữ liệu trên, visualize kết quả, xem xét, đánh giá kết quả

iii. Kết quả



iv. Đánh giá và so sánh

- Đánh giá bằng cách sử dụng hàm `metric_adjusted_mutual_info_score()`
- Kết quả tuy không quá tốt nhưng vẫn có thuật toán cho độ chính xác >10% (DBSCAN, Agglomerative)

- Có thể thấy, với bộ feature phân bố đều, DBSCAN cho kết quả tốt hơn các thuật toán còn lại.
- Kmeans có khuynh hướng nhóm các tập điểm về dạng spherical nên không có hiệu quả tốt đối với dạng dữ liệu features phân bố đều.
- Agglomerative cải tiến hơn Kmeans nên có kết quả khả quan hơn, tuy nhiên, vì vẫn dựa vào khoảng cách nên tính chính xác cũng không quá cao.
- Vì feature phân bố đều nên việc tính ma trận tương đồng sẽ gây ra nhiều sự nhập nhằng khiến spectral không còn hiệu quả.

Tài Liệu Tham Khảo

- [1 M. B. O. B. Ulrike von Luxburg, “Consistency of Spectral Clustering,” [Trực tuyến]. Available:
] http://web.cse.ohio-state.edu/~belkin.8/papers/SC_AOS_07.pdf.
- [2 V. H. Tiệp, “machinelearningcoban.com,” [Trực tuyến]. Available:
] <https://machinelearningcoban.com/2017/01/01/kmeans/>.
- [3 “Wikipedia,” 25 9 2017. [Trực tuyến]. Available: <https://en.wikipedia.org/wiki/DBSCAN>.
]
- [4 “mprovedoutcomes,” [Trực tuyến]. Available:
] http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Agglomerative_Hierarchical_Clustering_Overview.htm.
- [5 “scikit-learn.org,” [Trực tuyến]. Available: <http://scikit-learn.org/stable/modules/clustering.html#clustering>.
]
- [6 “scikit-image.org,” [Trực tuyến]. Available: <http://scikit-image.org/docs/stable/api/skimage.feature.html>.
]
- [7 “Wikipedia,” [Trực tuyến]. Available:
] https://en.wikipedia.org/wiki/Histogram_of_oriented_gradients.