Lab 2 - PCA và bài toán phân cụm (clustering)

1 Lập trình PCA

1.1 Tải tập dữ liệu

- Ở phần này, ta sử dụng cơ sở dữ liệu về tròng mắt (iris) từ thư viện scikit learn. Cơ sơ dữ liệu này gồm 4 cột dữ liệu là Sepal Length, Sepal Width, Petal Length và Petal Width.
- Nhóm có thể tự download dataset về máy hoặc có thể chạy đoạn code sau trong notebook:

```
from sklearn.datasets import load_iris

iris = load_iris()
X = iris['data']
y = iris['target']
```

1.2 Yêu cầu

Trong phần này, nhóm cần hoàn thành các yêu cầu sau:

- Tải xuống và đọc được toàn bộ tập dữ liệu.
- In ra một số thông tin của dataset: Số dòng, tên các cột.

2 Lớp PCA (5 điểm)

2.1 Yêu cầu

Trong phần này, nhóm cần hoàn thành các yêu cầu sau:

- Tạo một class PCA với cấu trúc như sau và hoàn thành nội dung còn thiếu cho các hàm:

```
class MyPCA:
    def __init__(self, n_components):

    def fit(self, X):

    def transform(self, X):
```

Yêu cầu cho các hàm:

- Hàm init(self, n_components): Hàm khởi tạo cho PCA với n_components là số thành phần của PCA.
- Hàm fit(self, X): Hàm khớp mô hình PCA với dữ liệu X. Nhóm cần thực hiện cụ thể từng bước tính PCA trong hàm này và tính hai giá trị sau đây:
  - EVR (Explained variance ratio): Giá trị này thể hiện tỉ lệ phương sai (variance) của dữ liệu được cho đã được nắm giữ bởi các thành phần PCA là bao nhiêu.
  - CEVR (Cumulative explained variance ratio): Giá trị này thể hiện **tổng** tỉ lệ phương sai của dữ liệu được cho được nắm dữ bởi tất cả các thành phần trong PCA là bao nhiêu.

- Hàm transform(self, X): Hàm biến đổi dữ liệu X thành các thành phần PCA dựa vào giá trị trung bình (mean) và độ lệch chuẩn (standard deviation) được tính từ hàm fit(X).
- **Lưu ý:** Không được sử dụng thư viện scikit-learn ở đây. Tuy nhiên, nhóm có thể chạy thử PCA của scikit-learn để kiểm chứng xem cách lập trình của mình có đúng hay không.

3 Bài toán phân cụm (5 điểm)

3.1 Giới thiệu

Thực tế, các bộ cơ sở dữ liệu có rất nhiều cột/đặc trưng (features), nếu như ta chỉ chọn một số đặc trưng để huấn luyện mô hình như lab 1 thì có thể ta sẽ bỏ qua những đặc trưng tốt và việc chọn lọc đặc trưng sẽ rất khó khăn. Mặc khác, nếu ta chọn tất cả đặc trưng để huấn luyện thì sẽ khiến mô hình khó có thể khớp được với bộ cơ sở dữ liệu hoặc là mô hình máy học sẽ rất phức tạp.

Vì vậy, PCA là một phương pháp giảm số chiều hiệu quả giúp ta có thể tạo ra những đặc trưng "ảo" có thể biểu diễn được cho cả bộ cơ sở dữ liệu với số lương đặc trưng ít hơn rất nhiều.

Ở Lab này, nhóm cần xử lý một bộ cơ sở dữ liệu gồm rất nhiều đặc trưng và giải quyết bài toán phân cụm (clustering) cho bộ cơ sở dữ liệu này.

3.2 Cơ sở dữ liệu

Bộ cơ sở dữ liệu ABIDE II được giới thiệu ở NeuroHackademy 2020 bởi giáo sư Tal Yarkoni. Bộ cơ sở đã được thay đổi một ít để phù hợp với lab này, cụ thể sẽ là phân cụm bệnh nhân có bị ung thư (cancer) hay không (normal).

Một số thông tin về dataset:

- Số dòng: 1004, số cột: 1444. Trong đó có 463 bệnh nhân bị ung thư và 541 bệnh nhân không bị ung thư.
- Cột nhãn: "group". Cột này chỉ có 2 giá trị là: "Cancer" và "Normal".

3.3 Yêu cầu

Trong phần này, nhóm cần hoàn thành các yêu cầu sau:

Tải tập dữ liệu từ moodle. Lưu ý là tập dữ liệu **đã được thay đổi** so với dữ liệu gốc, vì thế không tải từ các nguồn khác.

Kiểm tra lại các thông tin về dataset và in ra.

3.4 Mô hình

Ở lab này, nhóm được áp dụng các kiến thức về phân cụm (Clustering) để huấn luyện mô hình dựa vào tập dữ liệu tập dữ liệu cho sẵn.

Nhóm có thể tự quyết định mô hình như thế nào là phù hợp tuy nhiên chỉ giới hạn cho các mô hình máy học (Machine Learning), không được sử dụng các mô hình học sâu (deep learning).

3.5 Yêu cầu

Trong phần này, nhóm cần hoàn thành các yêu cầu sau:

- Sử dụng lớp PCA đã được lập trình để giảm số đặc trưng của dataset xuống.
- Thực hiện các thuật toán phân cụm (Kmeans,...) để phân cụm các điểm dữ liệu. Lưu ý không được sử dụng cột "group" khi thực hiện giảm chiều bằng PCA và trong lúc phân cụm.
- So sánh các điểm giữ liệu với giá trị thât ở cột "group" và tiến hành đo độ hiểu quả của thuật toán phân cụm (Precision, accuracy,...)
- Số lượng thành phần của PCA cần được khảo sát và chọn ra giá trị "tốt nhất". Trong báo cáo hoặc file notebook cần có phần phân tích tại sao lại chọn số lượng thành phần của PCA đó và các thông số của thuật toán phân cụm đã chọn.

4 Các yêu cầu khác

- Ngôn ngữ sử dụng bắt buộc là Python, không được phép sử dụng ngôn ngữ khác. (nên sử dụng Jupiter Notebook).
- Giới hạn thư viện: nhóm chỉ được sử dụng các thư viện cho các tác vụ nằm ngoài việc huấn luyện mô hình (ví dụ: pandas, numpy,...) và không được sử dụng các thư việc cho tác vụ này (ví dụ: sklearn,...)
- Các nhóm cần kiểm tra mã nguồn trước khi nộp. Nếu mã nguồn không chạy được mà không phải do nguyên nhân khách quan (thiếu thư viện, lỗi do thư viện gây ra, sử dụng thư viện sai phiên bản,...) thì sẽ bị 0 điểm đồ án.
- Bài nộp phải gồm có 2 phần:
  + Report: Chứa các file báo cáo.
  + Source: Chứa các file mã nguồn.
- Trong các file nộp, nhóm cần ghi rõ thông tin về các thành viên gồm họ tên và MSSV. Riêng đối với mã nguồn, nhóm có thể ghi thông tin trên dưới dạng comment trong code của nhóm.
- Bài nộp sẽ được đặt trong thư mục có tên `MSSV01[_MSSV02[_MSSV03[...]]]` và được nén lại bằng định dạng ZIP với format `[group_number]`.zip . Ví dụ đặt tên nhóm có 1 nhóm là `MSSV01`, nhóm có 2 nhóm là `MSSV01_MSSV02`.
- Nghiêm cấm các hành vi gian lận, không trung thực trong học tập như sao chép bài làm giữa các nhóm với nhau, sao chép bài làm của các nhóm khóa trước hoặc các nhóm lớp khác trường khác, nhờ người làm hộ. Nếu phát hiện các hành vi trên thì cả nhóm sẽ bị 0 điểm và xử lý theo quy định của Khoa và Trường.