

# E-mail categorization using KNN, Gaussian Naive Bayes, and Logistic Regression

Nguyễn Duy Vũ - 22000132

Vũ Đức Duy - 22000078

Đoàn Phạm Ngọc Linh - 22000102



**HUS**  
VNU UNIVERSITY OF SCIENCE



# Nội dung

- 1 Giới thiệu đề tài
- 2 Phân cụm dữ liệu
- 3 Các phương pháp sử dụng
- 4 Dữ liệu và thực nghiệm
- 5 Tài liệu tham khảo

# Nội dung

- 1 Giới thiệu đề tài
- 2 Phân cụm dữ liệu
- 3 Các phương pháp sử dụng
- 4 Dữ liệu và thực nghiệm
- 5 Tài liệu tham khảo

# Tổng quan bài toán phân loại email

- **Phân loại email** là một bài toán quan trọng trong **học máy** và **xử lý ngôn ngữ tự nhiên**
- **Mục tiêu:** Xây dựng hệ thống tự động phân loại email vào các nhóm chủ đề. ví dụ: *công việc, spam, quảng cáo, cá nhân,...*
- **Dựa trên:** Các đặc trưng như: **nội dung, tiêu đề, người gửi, metadata...**
- **Lợi ích:** Hỗ trợ quản lý hộp thư hiệu quả, tiết kiệm thời gian, phát hiện email quan trọng, lọc thư rác tự động.

# Ứng dụng thực tế

**Phân loại email** được ứng dụng rộng rãi trong các hệ thống quản lý thông tin và truyền thông:

- **Lọc thư rác:** Phát hiện và chuyển email rác vào thư mục riêng.
- **Tự động gắn nhãn:** Phân loại email thành các nhóm: công việc, cá nhân, quảng cáo,...
- **Trợ lý ảo:** Ưu tiên thư quan trọng, hỗ trợ phản hồi tự động.
- **Chăm sóc khách hàng:** Phân tích nội dung email để hiểu nhu cầu người dùng.

# Phân chia công việc

- **Đoàn Phạm Ngọc Linh:** Tiền xử lý dữ liệu, chuẩn hóa dữ liệu, giảm chiều dữ liệu bằng PCA. Xây dựng, huấn luyện và đánh giá mô hình K-Nearest Neighbors (KNN). Thực hiện đánh giá kết quả thuật toán phân cụm DBSCAN thông qua các chỉ số Silhouette Score, Davies-Bouldin Index (DBI), Adjusted Rand Index (ARI)
- **Nguyễn Duy Vũ:** Xây dựng, huấn luyện và tối ưu hóa mô hình Logistic Regression, đánh giá hiệu quả mô hình bằng các chỉ số Accuracy, Precision, Recall, F1-Score. Xây dựng và triển khai thuật toán phân cụm DBSCAN trên dữ liệu đã giảm chiều.
- **Vũ Đức Duy:** Xây dựng, huấn luyện và đánh giá mô hình Bernoulli Naive Bayes. Thực hiện trực quan hóa dữ liệu, bao gồm vẽ biểu đồ phương sai tích lũy (PCA), biểu đồ phân tán để phân tích và đánh giá dữ liệu trước và sau khi giảm chiều. Triển khai thuật toán phân cụm K-Means và đánh giá hiệu quả phân cụm.

# Nội dung

- 1 Giới thiệu đề tài
- 2 Phân cụm dữ liệu**
- 3 Các phương pháp sử dụng
- 4 Dữ liệu và thực nghiệm
- 5 Tài liệu tham khảo

# K-Means Clustering

**K-Means** là thuật toán phân cụm phổ biến thuộc nhóm học không giám sát.

- Nhóm dữ liệu thành  $K$  cụm dựa trên khoảng cách Euclidean.
- Ứng dụng: phân tích thị trường, nhận diện ảnh, khai phá dữ liệu văn bản, v.v.
- Phù hợp khi dữ liệu có cấu trúc hình cầu, các cụm đồng đều về kích thước.



# Nguyên lý toán học K-Means

Ký hiệu:

- $\mathbf{X}$ : Ma trận dữ liệu  $N$  điểm, mỗi điểm  $d$  chiều.
- $K$ : Số cụm;  $\mathbf{m}_k$ : trung tâm cụm  $k$ .
- Nhãn  $\mathbf{y}_i$  là vector one-hot xác định cụm của điểm  $i$ .

$$L(\mathbf{Y}, \mathbf{M}) = \sum_{i=1}^N \sum_{j=1}^K y_{ij} \|\mathbf{x}_i - \mathbf{m}_j\|_2^2$$

Tối ưu lặp lại hai bước:

- 1 Gán mỗi điểm vào cụm gần nhất.
- 2 Cập nhật trung tâm cụm bằng trung bình cộng các điểm trong cụm.

# Đánh giá kết quả K-Means (PCA)

Chỉ số	Giá trị
Silhouette Score	0.6924
Davies-Bouldin Index	2.6078
Adjusted Rand Index	0.0400

**Bảng:** Các chỉ số đánh giá phân cụm K-Means

## Nhận xét:

- Silhouette Score cao ( $\sim 0.69$ ): các cụm khá tách biệt nhau.
- DBI còn cao ( $\sim 2.6$ ): các cụm vẫn còn chồng lấn, chưa tối ưu.
- ARI rất thấp ( $\sim 0.04$ ): phân cụm không khớp nhiều với nhãn thật.

# Phân tích tỷ lệ nhãn trong cụm

Cụm	Tỷ lệ nhãn 0	Tỷ lệ nhãn 1
0	72.03%	27.97%
1	31.30%	68.70%

**Bảng:** Tỷ lệ nhãn gốc trong các cụm K-Means

- **Cụm 0:** Chủ yếu là nhãn 0 (non-spam), nhưng có gần 28% nhãn 1.
- **Cụm 1:** Phần lớn là nhãn 1 (spam), đạt 68.7%.

# Kết luận về phân cụm

- K-Means tách biệt được một phần email spam và non-spam trong không gian PCA.
- Tuy nhiên, cụm 1 rất nhỏ, dẫn đến ARI thấp. Cụm 0 còn chứa nhiều dữ liệu ”lẫn”.
- Kết quả cho thấy K-Means chưa lý tưởng để tái hiện cấu trúc phân lớp của dữ liệu gốc, đặc biệt khi dữ liệu không chia cụm rõ ràng hoặc không có hình cầu.
- Các chỉ số đánh giá giúp hiểu sâu hơn về đặc tính dữ liệu cũng như hạn chế của thuật toán.

# Nguyên lý toán học DBSCAN

Ký hiệu và tham số:

- $\varepsilon$ : bán kính lân cận.
- $\text{MinPts}$ : số điểm tối thiểu để một điểm trở thành điểm lõi.

Phân loại điểm:

- **Điểm lõi**: có ít nhất  $\text{MinPts}$  điểm trong  $\varepsilon$ .
- **Điểm biên**: không phải lõi, nhưng nằm trong vùng  $\varepsilon$  của điểm lõi.
- **Điểm nhiễu**: không thuộc cụm nào.

Quy trình:

- 1 Tìm điểm lõi và bắt đầu cụm mới từ điểm đó.
- 2 Kết nối các điểm density-reachable để mở rộng cụm.
- 3 Lặp lại cho đến khi mọi điểm được xét.

# Đánh giá kết quả DBSCAN (PCA)

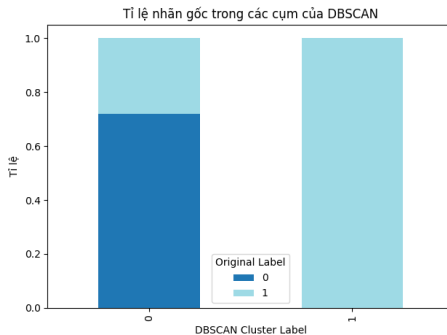
Chỉ số	Giá trị
Silhouette Score	0.5995
Davies-Bouldin Index	0.3347
Adjusted Rand Index	0.0311

**Bảng:** Các chỉ số đánh giá phân cụm DBSCAN

## Nhận xét:

- Silhouette Score khá tốt ( $\sim 0.6$ ): cụm rõ ràng, hợp lý.
- DBI rất thấp ( $\sim 0.33$ ): cụm cách biệt và tập trung tốt.
- ARI thấp ( $\sim 0.03$ ): không khớp nhãn thật, nhưng phù hợp tính chất không giám sát.

# Phân tích tỷ lệ nhãn trong cụm (DBSCAN)



- DBSCAN tạo ra các cụm với tỷ lệ nhãn khác nhau.
- Có khả năng phát hiện các nhóm điểm ít rõ ràng hơn K-Means, đồng thời loại bỏ được nhiễu.

# Kết luận về phân cụm DBSCAN

- DBSCAN phân cụm hiệu quả trong không gian PCA mà không cần chỉ định trước số cụm.
- Đánh giá Silhouette và DBI đều tốt, cho thấy cấu trúc cụm rõ ràng và rời rạc.
- ARI thấp phản ánh sự khác biệt với nhãn thật, nhưng phù hợp vì DBSCAN phát hiện cấu trúc ẩn không tuyến tính.
- DBSCAN là lựa chọn tốt trong dữ liệu có hình dạng cụm phức tạp hoặc chứa nhiễu.



# Nội dung

- 1 Giới thiệu đề tài
- 2 Phân cụm dữ liệu
- 3 Các phương pháp sử dụng**
- 4 Dữ liệu và thực nghiệm
- 5 Tài liệu tham khảo

# Các phương pháp được lựa chọn

Trong nghiên cứu này, nhóm sử dụng ba thuật toán phổ biến trong học máy để phân loại email:

- **K-Nearest Neighbors (KNN)**
- **Bernoulli Naive Bayes (BNB)**
- **Logistic Regression (LR)**

## Lý do lựa chọn:

- Phổ biến, dễ triển khai.
- Phù hợp với bài toán phân loại văn bản.
- Mỗi thuật toán đại diện cho một hướng tiếp cận khác nhau.

# So sánh ba phương pháp

- **KNN** — thuật toán phi tham số, dựa trên khoảng cách giữa các điểm.
- **BNB** — phương pháp phân loại dựa trên lý thuyết Bayes, thiết kế cho dữ liệu có **đặc trưng nhị phân** (0 hoặc 1).
- **Logistic Regression** — mô hình tuyến tính, học trọng số cho từng đặc trưng.

## Tiêu chí đánh giá:

- Độ chính xác (Accuracy)
- Độ nhạy (Recall)
- Độ chính xác theo lớp dương (Precision )
- Ma trận nhầm lẫn (Confusion Matrix)

# KNN – K-Nearest Neighbors

**Ý tưởng:** Phân loại dựa trên số lượng các điểm gần nhất trong không gian đặc trưng.

**Công thức khoảng cách Euclidean:**

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

**Thường áp dụng cho:** vector hóa bằng TF-IDF, sử dụng khoảng cách Cosine hoặc Euclidean để so sánh.

# KNN – Ưu và nhược điểm

## Ưu điểm:

- Không cần huấn luyện mô hình.
- Đơn giản, trực quan.
- Hoạt động tốt với các dữ liệu không có quá nhiều nhiễu.

## Nhược điểm:

- Hiệu suất kém nếu dữ liệu lớn.
- Kết quả phụ thuộc vào  $k$  và khoảng cách.
- Chi phí tính toán cao khi dữ liệu lớn.

# BNB – Bernoulli Naive Bayes

**Công thức Bayes:**

$$p(k \mid \mathbf{x}) \propto p(\mathbf{x} \mid k) \cdot p(k)$$

Với giả thiết độc lập:

$$p(\mathbf{x} \mid k) = \prod_{i=1}^d p(x_i \mid k)$$

**Tính xác suất điều kiện:**

$$p(x_i \mid k) = (p_{ki})^{x_i} \cdot (1 - p_{ki})^{1-x_i}$$

Trong đó:

$$p_{ki} = \frac{\text{Số mẫu lớp } k \text{ có } x_i = 1 + \alpha}{\text{Tổng số mẫu lớp } k + 2\alpha}$$

( $\alpha = 1$  cho làm mịn Laplace)

# BNB – Bernoulli Naive Bayes

## **Ưu điểm:**

- Hiệu quả với dữ liệu nhị phân
- Tốc độ huấn luyện nhanh

## **Hạn chế:**

- Giả thiết độc lập không thực tế
- Không xử lý tốt tần suất xuất hiện

# LR – Logistic Regression

**Hàm sigmoid:**

$$h_{\theta}(x) = \frac{1}{1 + \exp(-\theta^T x)}$$

**Xác suất phân lớp:**

$$P(y = 1|x; \theta) = h_{\theta}(x), \quad P(y = 0|x; \theta) = 1 - h_{\theta}(x)$$

**Hàm log-likelihood cần tối ưu:**

$$\ell(\theta) = \sum_{n=1}^N [y_n \log h_{\theta}(x_n) + (1 - y_n) \log(1 - h_{\theta}(x_n))]$$

**Tối ưu bằng:** Gradient Descent hoặc Newton-Raphson.



# Logistic Regression – Ưu và nhược điểm

## Ưu điểm:

- Xử lý tốt dữ liệu thưa.
- Dễ huấn luyện, dễ diễn giải.
- Mở rộng được cho bài toán đa lớp.

## Nhược điểm:

- Không phù hợp dữ liệu phi tuyến.
- Dễ bị ảnh hưởng bởi ngoại lệ và đa cộng tuyến.

# Nội dung

- 1 Giới thiệu đề tài
- 2 Phân cụm dữ liệu
- 3 Các phương pháp sử dụng
- 4 Dữ liệu và thực nghiệm**
- 5 Tài liệu tham khảo

# Tổng quan bộ dữ liệu

**Bộ dữ liệu:** 5172 email, 3002 cột.

**Thông tin chính:**

- Tần suất từ, độ dài email, metadata.
- Dữ liệu dạng số thực, số nguyên và chuỗi.
- Nhãn Label dùng cho huấn luyện.

**Cân bằng lớp:** Spam vs. Non-spam ~ ngang nhau.

# Các đặc trưng đáng chú ý

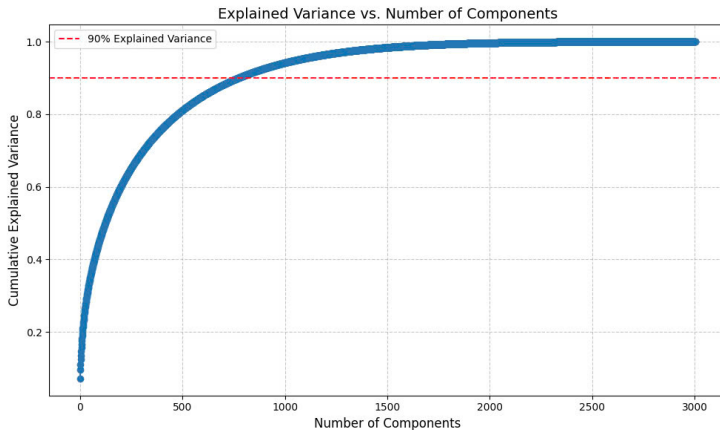
- **Most Common Word 1 - 9:** Tần suất từ phổ biến ("the", "to", "you",...)
- **Thông số tần suất:** VD: "you" trung bình 55.5, std 87.6
- **Email Name, ID:** Bỏ khi huấn luyện

# Tiền xử lý dữ liệu

## Các bước xử lý:

- **Loại bỏ cột không cần thiết:**
  - Email No. — chỉ mang tính tra cứu.
- **Chuẩn hóa dữ liệu:**
  - Sử dụng StandardScaler đưa kỳ vọng về 0, phương sai = 1.
- **Giảm chiều dữ liệu:**
  - Áp dụng PCA để giảm từ 3000 xuống 50 thành phần chính.
- **Chia dữ liệu:**
  - Tập huấn luyện (80%) — kiểm tra (20%), dùng phân tầng.

# Trực quan hóa: Phân tích PCA

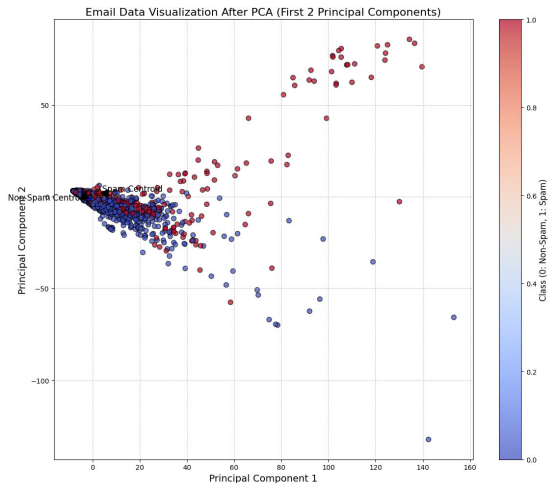


Hình: Phương sai tích lũy theo số lượng thành phần chính

# Phân tích

- **Phân bố dữ liệu:** Biểu đồ cho thấy cách các email được phân bố trong không gian hai chiều của hai thành phần chính đầu tiên. Điểm màu xanh đại diện cho email không phải spam, điểm màu đỏ đại diện cho email spam.
- **Mức độ phân tách:** Mức độ phân tách giữa các điểm màu xanh và đỏ cho thấy khả năng phân biệt giữa email spam và không spam của hai thành phần chính đầu tiên. Nếu các điểm phân tách rõ ràng, điều này chứng tỏ PCA đã giữ lại được thông tin phân biệt quan trọng.

# Phân bố dữ liệu sau PCA



Hình: Phân tán dữ liệu theo 2 thành phần chính đầu tiên



# Nhận xét

## **Nhận xét:**

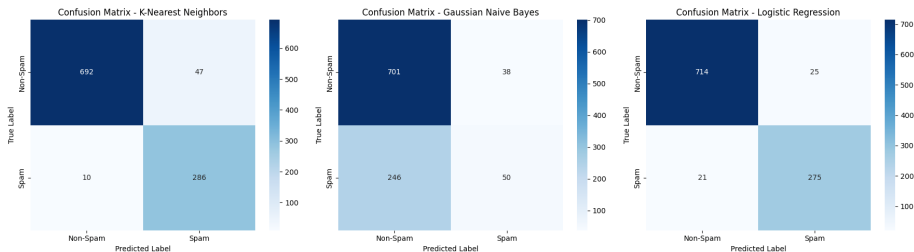
- Các điểm xanh (non-spam) và đỏ (spam) tách biệt rõ ràng.
- Các dấu sao thể hiện trọng tâm 2 nhóm, đường nối là ranh giới phân loại.
- Phân tách này hỗ trợ mô hình học tốt ngay cả với phân tích đơn giản.

# Kết quả trên dữ liệu gốc

Mô hình	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.9556	0.9167	0.9291	0.9228
K-Nearest Neighbors	0.9449	0.8589	0.9662	0.9094
Bernoulli Naive Bayes	0.8802	0.7688	0.8311	0.7987

**Tỷ lệ chính xác cao nhất:** Logistic Regression

# Ma trận nhầm lẫn trên dữ liệu gốc



# Phân tích kết quả (Dữ liệu gốc)

## **Logistic Regression:**

- Accuracy cao nhất: 95.56%, F1 đạt 0.92.
- Phân biệt tốt giữa spam và non-spam.

## **KNN:**

- Hiệu suất tốt, recall cao, nhưng dễ bị ảnh hưởng bởi nhiễu và cấu trúc dữ liệu.

## **Bernoulli Naive Bayes:**

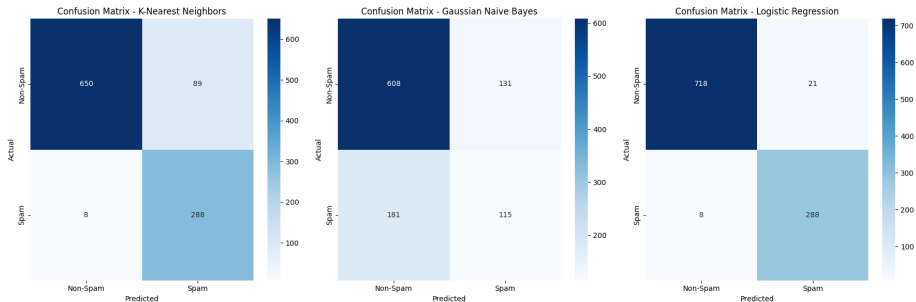
- Đơn giản, tốc độ nhanh, cho accuracy 0.88 và F1 0.80.
- Hiệu quả thấp hơn so với LR và KNN, do giả định độc lập không thực tế với dữ liệu văn bản.

# Kết quả trên dữ liệu đã chuẩn hóa và giảm chiều

Mô hình	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.9507	0.8840	0.9527	0.9171
K-Nearest Neighbors	0.9507	0.9181	0.9088	0.9134
Bernoulli Naive Bayes	0.8841	0.7821	0.8243	0.8026

**LR vẫn là mô hình tốt nhất sau chuẩn hóa và PCA**

# Ma trận nhầm lẫn sau chuẩn hóa



# Phân tích kết quả (Chuẩn hóa và PCA)

## **Logistic Regression:**

- Accuracy giữ ở mức cao (0.95), F1 score đạt 0.92.
- Rõ lợi ích từ chuẩn hóa và giảm chiều.

## **KNN:**

- Accuracy và F1 score cao, nhưng có thể giảm so với dữ liệu gốc do PCA có thể làm thay đổi cấu trúc khoảng cách.

## **Bernoulli Naive Bayes:**

- Hiệu suất giữ nguyên hoặc cải thiện nhẹ sau chuẩn hóa và PCA.
- Vẫn thấp hơn LR và KNN, nhưng là lựa chọn phù hợp khi tốc độ và đơn giản là ưu tiên.

# Kết quả phân cụm K-Means

## Chỉ số đánh giá:

- Silhouette Score: 0.6924 (cụm khá tách biệt)
- Davies-Bouldin Index: 2.6078 (còn chồng lấn)
- Adjusted Rand Index: 0.0400 (không khớp nhãn gốc)

## Tỷ lệ nhãn trong từng cụm:

- Cụm 0: 72.03% nhãn 0, 27.97% nhãn 1
- Cụm 1: 31.30% nhãn 0, 68.70% nhãn 1

**Kết luận:** K-Means tách được một số nhóm, nhưng chưa phản ánh đúng cấu trúc phân lớp gốc.



# Kết quả phân cụm DBSCAN

## Chỉ số đánh giá:

- Silhouette Score: 0.5995 (tách cụm tương đối)
- Davies-Bouldin Index: 0.3347 (cụm cách biệt tốt)
- Adjusted Rand Index: 0.0311 (không khớp nhãn gốc)

## Nhận xét:

- DBSCAN xác định cụm với mật độ tốt, loại bỏ nhiễu, nhưng vẫn không trùng với nhãn phân loại.
- Điều này bình thường vì DBSCAN là thuật toán không giám sát.

# Tổng kết thực nghiệm

- **Logistic Regression** luôn vượt trội trong cả hai giai đoạn.
- **KNN** phù hợp hơn với dữ liệu gốc, hiệu suất giảm khi đặc trưng biến đổi qua PCA.
- **Bernoulli Naive Bayes** cải thiện nhẹ sau chuẩn hóa và giảm chiều, nhưng vẫn thấp hơn LR và KNN.
- **K-Means, DBSCAN** hỗ trợ khám phá cấu trúc tự nhiên, nhưng không phản ánh đúng nhãn phân loại.

**Kết luận:** Việc chọn mô hình cần cân nhắc đặc tính dữ liệu và bước tiền xử lý phù hợp.

# Nội dung

- 1 Giới thiệu đề tài
- 2 Phân cụm dữ liệu
- 3 Các phương pháp sử dụng
- 4 Dữ liệu và thực nghiệm
- 5 Tài liệu tham khảo**

# Tài liệu tham khảo

- ❶ Machine Learning cơ bản - Vũ Hữu Tiệp.
- ❷ Mining of Massive Datasets - Jure Leskovec, Anand Rajaraman, Jeff.
- ❸ Bài giảng Hệ gợi ý - Socit - Đại học Bách khoa Hà Nội.
- ❹ Machine Learning TextBook - Andreas Lindholm, Niklas Wahlstrom, Fredrik Lindsten, Thomas B.Schon.
- ❺ Bhuyan, R., Borah, S. (2023). *A Survey of Some Density Based Clustering Techniques*.
- ❻ Wang, D., Lu, X., Rinaldo, A. (2017). *DBSCAN: Optimal Rates For Density Based Clustering*.
- ❼ Chakraborty, S., Nagwani, N. K., Dey, L. (2014). *Performance Comparison of Incremental K-means and Incremental DBSCAN Algorithms*.