

CẢI TIẾN DẠY HỌC TRONG THỜI ĐẠI DỮ LIỆU: KẾT NỐI ĐẠI SỐ TUYẾN TÍNH, XÁC SUẤT THỐNG KÊ VÀ PCA TRONG HỌC MÁY

Lê Bích Phương^{1,*}, Nguyễn Văn Chung¹, Nguyễn Thị Hằng¹, Phạm Tuấn Cường¹

¹ Trường Đại học Mở - Địa chất, Số 18 Phố Viên, Phường Đức Thắng,
Quận Bắc Từ Liêm, TP Hà Nội

Tác giả liên hệ: Email: Lebichphuong@humb.edu.vn; [Tel:0988782112](tel:0988782112)

Tóm tắt. Bài báo này nghiên cứu mối liên hệ giữa xác suất thống kê, đại số tuyến tính và phân tích thành phần chính (PCA) trong học máy. PCA là một phương pháp giảm chiều dữ liệu hiệu quả, giúp tối ưu hóa thuật toán học máy bằng cách sử dụng các công cụ của đại số tuyến tính như ma trận, giá trị riêng và vector riêng. Chúng tôi trình bày chi tiết quy trình thực hiện PCA, bao gồm chuẩn hóa dữ liệu, tính toán ma trận hiệp phương sai, xác định giá trị và vector riêng, lựa chọn thành phần chính và biến đổi dữ liệu. Bài báo cũng cung cấp ví dụ giả lập để minh họa ứng dụng thực tế của PCA trong phân tích dữ liệu. Ngoài ra, chúng tôi đề xuất phương pháp giảng dạy kết hợp lý thuyết và thực hành, giúp sinh viên hiểu rõ PCA thông qua lập trình và các tình huống thực tế, từ đó nâng cao tư duy thuật toán và khả năng ứng dụng trong khoa học dữ liệu.

Từ khóa: đại số tuyến tính, giá trị riêng, vector riêng, học máy, PCA, ma trận hiệp phương sai.

1. ĐẶT VẤN ĐỀ

Trong kỷ nguyên dữ liệu bùng nổ, việc xử lý dữ liệu có số chiều lớn là một thách thức của khoa học dữ liệu, trí tuệ nhân tạo và thống kê. Số lượng đặc trưng quá lớn gây khó khăn cho trực quan hóa, phân tích và mô hình hóa, làm tăng chi phí tính toán và nguy cơ quá khớp. Để khắc phục, các phương pháp giảm chiều dữ liệu, đặc biệt là Phân tích thành phần chính (PCA), đã được áp dụng rộng rãi [2,3,5]. PCA sử dụng giá trị riêng và vector riêng của ma trận hiệp phương sai để tìm các hướng giữ nhiều thông tin nhất, giúp giảm số chiều mà vẫn bảo toàn phần lớn phương sai, tối ưu hóa phân tích dữ liệu và huấn luyện mô hình. Phương pháp này có nhiều ứng dụng trong nhận dạng ảnh, phân loại dữ liệu, nén ảnh, dự báo kinh tế và học sâu [4].

Bài báo này kết nối đại số tuyến tính, xác suất thống kê và PCA trong học máy, trình bày chi tiết quy trình thực hiện PCA và đề xuất phương pháp giảng dạy giúp sinh viên tiếp cận PCA trực quan và thực hành lập trình hiệu quả.

2. NỘI DUNG

2.1 Kiến thức cơ bản [1] Định nghĩa 1. Cho ma trận $A = (a_{ij})_{m \times n}$ và $B = (b_{jk})_{n \times p}$ trong đó số cột của A bằng số hàng của B . Khi đó, tích của A và B , ký hiệu là

$$A.B = C = (c_{ik})_{m \times p} \text{ với } c_{ik} = \sum_{j=1}^n a_{ij} b_{jk} = (a_{i1} \ a_{i2} \ \dots \ a_{in}) \begin{pmatrix} b_{1k} \\ b_{2k} \\ \vdots \\ b_{nk} \end{pmatrix} \quad (1)$$

Định nghĩa 2. Cho ma trận vuông $A = (a_{ij})_{m \times n}$. Định thức của A là một số, ký hiệu là $\det(A)$ hoặc $|A|$ được xác định như sau

$$\det(A) = \sum (-1)^{N(j_1, j_2, \dots, j_n)} a_{1j_1} a_{2j_2} \dots a_{nj_n} \quad (2)$$

Tổng được lấy theo mọi hoán vị của tập $\{1, 2, \dots, n\}$.

Định nghĩa 3. Ma trận đơn vị là ma trận vuông mà các phần tử trên đường chéo chính bằng 1, các phần tử còn lại bằng 0. Ma trận đơn vị thường được ký hiệu là I .

Cho A là ma trận vuông. Ma trận B được gọi là ma trận nghịch đảo của A khi và chỉ khi $AB = BA = I$. Trong đó I là ma trận đơn vị cùng cấp với ma trận A . Khi đó, ma trận A gọi là ma trận khả nghịch.

Định nghĩa 4. Cho $A \in M_n$, nếu tồn tại $\lambda \in \mathbb{R}$, $x \in \mathbb{R}^n, x \neq \theta$, sao cho $Ax = \lambda x$ thì λ được gọi là giá trị riêng của A , và x được gọi là véc tơ riêng của A ứng với giá trị riêng λ . Đa thức $P_A(\lambda) = \det(A - \lambda I)$ được gọi là đa thức đặc trưng của A .

Cách tìm giá trị riêng: λ là giá trị riêng của A khi và chỉ khi λ là nghiệm của phương trình đặc trưng $P_A(\lambda) = 0$.

Cách tìm véc tơ riêng: Ứng với mỗi giá trị riêng λ , giải hệ phương trình tuyến tính thuần nhất $(A - \lambda I)x = 0$. Mỗi nghiệm của hệ là một véc tơ riêng. Tập hợp mọi nghiệm của hệ gọi là không gian riêng ứng với giá trị riêng λ .

Định nghĩa 5. Ma trận hiệp phương sai (covariance matrix) là một ma trận dùng để mô tả mối quan hệ giữa các biến ngẫu nhiên trong một tập hợp dữ liệu. Ma trận này cho biết mức độ tương quan giữa các đặc trưng. Công thức tính hiệp phương sai giữa 2 biến ngẫu nhiên X và Y là:

$$\text{cov}(X, Y) = E[(X - EX)(Y - EY)] \quad (3)$$

Ma trận hiệp phương sai cho một tập hợp các biến ngẫu nhiên, có thể biểu diễn dưới dạng:

$$\Sigma = \begin{pmatrix} \text{cov}(X_1, X_1) & \text{cov}(X_1, X_2) & \dots & \text{cov}(X_1, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(X_n, X_1) & \text{cov}(X_n, X_2) & \dots & \text{cov}(X_n, X_n) \end{pmatrix} \quad (4)$$

Trong các bài toán thống kê học máy, ma trận hiệp phương sai được tính bởi công thức:

$$C = \frac{1}{n-1} X'^T X' \text{ hoặc } C = \frac{1}{n} X'^T X' \quad (5)$$

trong đó X' là ma trận dữ liệu chuẩn hóa, n là số lượng mẫu dữ liệu.

Chuẩn hóa dữ liệu: Chuẩn hóa dữ liệu là quá trình biến đổi các đặc trưng (features) về cùng một đơn vị đo lường hoặc phạm vi giá trị, thường là thông qua việc đưa dữ liệu về một thang chuẩn như $[0, 1]$ hoặc có trung bình bằng 0 và độ lệch chuẩn bằng 1.

Ý nghĩa của chuẩn hóa dữ liệu: Chuẩn hóa giúp đảm bảo tất cả các đặc trưng đóng vai trò tương đương trong quá trình phân tích và mô hình hóa. Nếu không chuẩn hóa, các đặc trưng

có giá trị lớn có thể lấn át các đặc trưng có giá trị nhỏ, gây sai lệch trong kết quả phân tích. **Cách thực hiện chuẩn hóa dữ liệu:** Chuẩn hóa có thể được thực hiện bằng nhiều phương pháp khác nhau, phổ biến nhất là:

Min-Max Scaling: Chuyển dữ liệu về khoảng $[0, 1]$ bằng công thức:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (6)$$

Z-score Standardization: Đưa dữ liệu về phân phối chuẩn có trung bình 0 và độ lệch chuẩn 1 bằng công thức:

$$x' = \frac{x - \mu}{\sigma} \quad (7)$$

trong đó x là giá trị ban đầu của đặc trưng, μ là trung bình của đặc trưng, σ là độ lệch chuẩn của đặc trưng.

2.2 Phương pháp Phân tích thành phần chính (PCA) [2-7]

PCA là một kỹ thuật giảm chiều (dimensionality reduction) phổ biến trong học máy và thống kê. Mục tiêu của PCA là giảm số lượng biến số trong một tập dữ liệu, đồng thời vẫn giữ được càng nhiều thông tin quan trọng càng tốt. Điều này cực kỳ hữu ích khi dữ liệu có quá nhiều chiều, gây khó khăn trong việc trực quan hóa, phân tích hoặc xử lý dữ liệu. Quá trình PCA dựa trên đại số tuyến tính, đặc biệt là việc sử dụng ma trận hiệp phương sai và phân tích giá trị riêng. Các bước thực hiện PCA như sau:

Tính ma trận hiệp phương sai: Ma trận hiệp phương sai là nền tảng của PCA, giúp phát hiện các chiều (hướng) mà dữ liệu biến thiên nhiều nhất — từ đó hỗ trợ việc giảm chiều dữ liệu mà vẫn giữ lại thông tin quan trọng.

Tính giá trị riêng và vector riêng: Đây là bước quan trọng nhất trong PCA. Các giá trị riêng và vector riêng của ma trận hiệp phương sai được tính toán. Các vector riêng này là những "hướng" tối ưu mà dữ liệu có thể chiếu lên để giảm chiều. Giá trị riêng biểu thị mức độ quan trọng của từng "hướng".

Chọn số lượng thành phần chính: Sau khi tính toán, chúng ta chọn số lượng thành phần chính (principal components) mà chúng ta muốn giữ lại. Thường thì, những thành phần chính này có giá trị riêng lớn, tức là chúng chiếm phần lớn thông tin của dữ liệu. Mục tiêu: Lựa chọn số lượng thành phần chính cần giữ lại để giảm chiều dữ liệu mà vẫn bảo toàn phần lớn thông tin. Cách thực hiện:

- Tính tổng các giá trị riêng
- Tính tỉ lệ phần trăm thông tin được giữ lại bởi mỗi thành phần chính:

$$TỈ LỆ TÍCH LŨY = \frac{\text{GIÁ TRỊ RIÊNG THÀNH PHẦN CHÍNH}}{\text{TỔNG GIÁ TRỊ RIÊNG}} \quad (8)$$

- Quyết định số lượng thành phần chính dựa trên tỉ lệ tích lũy.

Biến đổi dữ liệu: Tạo ra ma trận thành phần chính P chứa các vector riêng tương ứng với các giá trị riêng lớn nhất. Sau đó biến đổi dữ liệu bởi công thức:

$$Z = X' \cdot P \quad (9)$$

Ta được dữ liệu Z , có số chiều giảm, nhưng bảo toàn được phần lớn thông tin.

Lưu ý: PCA phù hợp với dữ liệu có quan hệ tuyến tính giữa các đặc trưng.

2.3 Ví dụ giả lập

Đây là một ví dụ giả lập và tính toán cụ thể về PCA để chúng ta có thể thấy rõ cách thức PCA hoạt động. Chúng ta sẽ giả lập một bộ dữ liệu đơn giản với 2 đặc trưng (là x_1 và x_2) với 5 mẫu, rồi thực hiện PCA để giảm chiều dữ liệu và phân tích các thành phần chính.

Bước 1: Tạo dữ liệu giả lập

Giả sử chúng ta có một bảng dữ liệu mẫu như sau:

Mẫu	x_1	x_2
1	2	3
2	3	4
3	4	5
4	5	6
5	6	7

Bước 2: Tính ma trận hiệp phương sai

Sử dụng đại số tuyến tính để tính toán các giá riêng và vector riêng từ ma trận hiệp phương sai, và giảm chiều dữ liệu

Tạo ma trận dữ liệu:

$X = \begin{bmatrix} 2 & 3 \\ 3 & 4 \\ 4 & 5 \\ 5 & 6 \\ 6 & 7 \end{bmatrix}$

Chuẩn hoá dữ liệu:

Chuẩn hoá dữ liệu có nghĩa là trừ giá trị trung bình của mỗi đặc trưng từ dữ liệu.

Tính trung bình và chuẩn hoá:

$$\text{Trung bình của } x_1 : \bar{x}_1 = \frac{2+3+4+5+6}{5} = 4 \quad (10)$$

$$\text{Trung bình của } x_2 : \bar{x}_2 = \frac{3+4+5+6+7}{5} = 5 \quad (11)$$

Sau đó, chuẩn hoá dữ liệu:

$$X_{\text{scaled}} = X - \begin{bmatrix} 4 & 5 \end{bmatrix}$$

$$X_{\text{scaled}} = \begin{bmatrix} -2 & -2 \\ -1 & -1 \end{bmatrix}$$

[0, 0],
[1, 1],
[2, 2])

Tính ma trận hiệp phương sai:

Ma trận hiệp phương sai đo lường mối quan hệ giữa các đặc trưng. Đối với dữ liệu chuẩn hoá, ta tính ma trận hiệp phương sai như sau:

$$\text{Cov}(X_scaled) = \left(\frac{1}{n}\right) * (X_scaled)^T * X_scaled \quad \text{với } (n=5)$$

Dễ dàng tính toán ma trận hiệp phương sai:

$$\text{Cov}(X_scaled) = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$$

Tính giá trị riêng và vector riêng:

Sử dụng đại số tuyến tính, chúng ta tính giá trị riêng và vector riêng của ma trận hiệp phương sai.

Giá trị riêng (λ) và vector riêng của ma trận hiệp phương sai sẽ cho ta các thành phần chính.

Giá sử ta tính được các giá trị riêng và vector riêng sau:

Giá trị riêng 1: $\lambda_1 = 4$

Giá trị riêng 2: $\lambda_2 = 0$

Vector riêng tương ứng:

$$\text{Vector riêng 1: } \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

$$\text{Vector riêng 2: } \left[-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

Chọn thành phần chính:

Trong Phân tích thành phần chính (PCA), mục tiêu chính là tìm hướng trong không gian dữ liệu mà dữ liệu có độ biến thiên lớn nhất. Điều này giúp giảm chiều dữ liệu trong khi vẫn giữ được nhiều thông tin nhất có thể. Chọn thành phần chính có giá trị riêng lớn nhất vì nó là hướng chứa nhiều thông tin nhất. Vì vậy ta chọn vector riêng 1.

$$\text{Do đó vector riêng là: } \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

Bước 3: Chiếu dữ liệu lên không gian các thành phần chính

Để giảm chiều dữ liệu, ta chiếu dữ liệu gốc lên vector riêng 1.

$$X_pca = X_scaled * \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

$$X_{pca} = [-2\sqrt{2}, -\sqrt{2}, 0, \sqrt{2}, 2\sqrt{2}]$$

Kết quả là chúng ta đã giảm chiều dữ liệu từ 2 chiều xuống 1 chiều, với các giá trị là các điểm chiếu trên thành phần chính

Bước 4: Kết luận

Dữ liệu gốc có 2 đặc trưng, và sau khi áp dụng PCA, chúng ta đã giảm chiều dữ liệu xuống còn 1 đặc trưng (thành phần chính đầu tiên).

PCA giúp loại bỏ các thành phần ít quan trọng (có giá trị riêng thấp) và chỉ giữ lại những thành phần có giá trị riêng lớn nhất.

Trong ví dụ trên, ta chỉ cần một thành phần chính để biểu diễn toàn bộ thông tin trong dữ liệu, giúp giảm độ phức tạp và cải thiện khả năng xử lý, trực quan hóa. Đây là quy trình tính toán PCA đơn giản bằng tay với một bộ dữ liệu nhỏ. Khi áp dụng PCA vào dữ liệu lớn hoặc phức tạp hơn, ta thường sử dụng các thư viện như sklearn để tính toán nhanh chóng và chính xác.

2.4 Ứng dụng của PCA trong học máy

Bài toán 1: Giảm chiều dữ liệu và phân tích bộ dữ liệu Iris

Đề bài:

Bộ dữ liệu Iris bao gồm 150 mẫu hoa, được mô tả bằng 4 đặc trưng: chiều dài và chiều rộng của đài hoa, chiều dài và chiều rộng của cánh hoa. Ba loại hoa trong bộ dữ liệu này là *Setosa*, *Versicolor*, và *Virginica*. Làm thế nào để giảm số chiều của dữ liệu từ 4 chiều xuống 2 chiều, đồng thời giữ lại được thông tin quan trọng để phân loại các loài hoa?

Mục tiêu: Giảm số chiều của dữ liệu từ 4 chiều xuống 2 chiều, giúp trực quan hóa và phân tích dễ dàng hơn.

Duy trì càng nhiều thông tin biến thiên càng tốt trong quá trình giảm chiều.

Phân loại và nhận diện các loài hoa dựa trên các đặc trưng đã giảm chiều.

Cách giải:

Áp dụng phương pháp PCA để giảm từ 4 chiều xuống 2 chiều.

Tính toán ma trận hiệp phương sai của bộ dữ liệu và tìm các vector riêng và giá trị riêng.

Chọn 2 thành phần chính có giá trị riêng lớn nhất để tạo ra không gian mới với 2 chiều.



Iris setosa



Iris versicolor



Iris virginica

Bước 1. Cài đặt thư viện cần thiết

```
# Cài đặt các thư viện cần thiết
import numpy as np
import pandas as pd
from sklearn.decomposition import PCA
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt
```

Bước 2. Tải dữ liệu Iris

```
python

# Tải bộ dữ liệu Iris
data = load_iris()
X = data.data # Dữ liệu đầu vào
y = data.target # Nhãn Lớp
```

Bước 3. Chuẩn hóa dữ liệu

PCA yêu cầu dữ liệu phải được chuẩn hóa vì các đặc trưng có thể có đơn vị và phạm vi khác nhau

```
python

# Chuẩn hóa dữ liệu
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Bước 4. Áp dụng PCA để giảm chiều

```
# Khởi tạo PCA với 2 thành phần chính
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

# In ra tỷ lệ phương sai của các thành phần chính
print(f"Explained variance ratio: {pca.explained_variance_ratio_}")

# Chúng ta chỉ giữ 2 thành phần chính, vì vậy sẽ có một không gian 2 chiều mới
```

Bước 5. Trực quan hóa dữ liệu sau PCA

```
# Trực quan hóa dữ liệu sau khi giảm chiều
plt.figure(figsize=(8, 6))
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y, cmap='viridis')
plt.title("Iris Dataset after PCA")
plt.xlabel("Principal Component 1")
plt.ylabel("Principal Component 2")
plt.colorbar()
plt.show()
```

Bước 6. Chia dữ liệu thành tập Huấn luyện và tập Kiểm tra

```
# Chia dữ liệu thành tập huấn luyện và kiểm tra (80% huấn luyện, 20% kiểm tra)
X_train, X_test, y_train, y_test = train_test_split(X_pca, y, test_size=0.2, random_state=
```

Bước 7. Huấn luyện mô hình phân loại

Ở đây ta sử dụng mô hình SVM (Support Vector Machine) để huấn luyện mô hình phân loại

```
python

# Khởi tạo mô hình SVM
svm = SVC(kernel='linear')

# Huấn luyện mô hình
svm.fit(X_train, y_train)
```

Bước 8. Dự đoán và đánh giá mô hình

```
python

# Dự đoán trên tập kiểm tra
y_pred = svm.predict(X_test)

# Tính độ chính xác của mô hình
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy of the model: {accuracy * 100:.2f}%")
```

Kết quả: Sau khi giảm chiều dữ liệu bằng PCA, ta thấy việc trực quan hóa dữ liệu trong không gian 2D dễ dàng hơn, mặc dù dữ liệu gốc có 4 chiều.

Bài toán 2: Phát hiện biến số quan trọng trong dự báo điểm rớt môn

Đề bài:

Tiến hành khảo sát 400 sinh viên, với các đặc trưng: số giờ học/tuần, điểm kiểm tra giữa kỳ, điểm chuyên cần, số giờ ngủ, số giờ dùng mạng xã hội, mức độ stress. Hãy sử dụng PCA để tìm ra những “yếu tố” ảnh hưởng lớn nhất đến “kết quả trượt môn”.

Cách giải:

Bước 1: Chuẩn hóa dữ liệu

Cần **chuẩn hóa (standardize)** các biến để có trung bình = 0 và độ lệch chuẩn = 1.

Các biến là: X1: Số giờ học/tuần; X2: Điểm kiểm tra giữa kỳ; X3: Điểm chuyên cần; X4: Số giờ ngủ; X5: Số giờ dùng mạng xã hội; X6: Mức độ stress

Bước 2: Thực hiện PCA

Sử dụng Python, sau khi chuẩn hóa, ta thực hiện PCA và thu được:

Các thành phần chính (PC1, PC2, ..., PC6)

Ma trận trọng số: Cho biết mỗi thành phần chính là tổ hợp tuyến tính của các biến gốc với các hệ số tương ứng. PCA cho kết quả như sau với PC1:

Biến	Trọng số trong PC1
Số giờ học/tuần	+0,589631
Điểm kiểm tra giữa kỳ	+0,471139
Điểm chuyên cần	+0,350821
Mức độ stress	+0,294467
Số giờ dùng mạng xã hội	-0,318639
Mức độ stress Số giờ ngủ	-0,345031

Bước 3: Phân tích PC1

- Vì PC1 có phương sai lớn nhất nên nó giải thích nhiều nhất sự khác biệt trong dữ liệu.
- Các trọng số cho thấy: Nếu “Tăng số giờ học, điểm kiểm tra, điểm chuyên cần” thì giá trị PC1 tăng. Nếu “Tăng thời gian mạng xã hội, số giờ ngủ” thì giá trị PC1 giảm.

Giá trị cao của PC1 ứng với sinh viên học hành nghiêm túc, ít ngủ, ít dùng mạng xã hội thì khả năng trượt môn giảm.

Bước 4: Kết luận

Trong bài toán này, sáu biến đầu vào bao gồm: số giờ học mỗi tuần, điểm kiểm tra giữa kỳ, mức độ chuyên cần, thời lượng ngủ, thời gian sử dụng mạng xã hội và mức độ căng thẳng. Các hệ số cho thấy điểm giữa kỳ, chuyên cần và số giờ học có hệ số dương cao, trong khi thời gian sử dụng mạng xã hội và giờ ngủ có hệ số âm lớn. Sinh viên có kết quả trượt môn thường là nhóm sinh viên có thời gian học thấp, điểm giữa kỳ thấp, chuyên cần thấp; mức độ sử dụng mạng xã hội và giờ ngủ cao. Kết quả này nhấn mạnh vai trò của việc duy trì kết quả học tập tích cực, thói quen học tập ổn định và quản lý thời gian dùng mạng xã hội trong việc cải thiện thành tích học tập, giảm nguy cơ thi trượt.

3. KẾT LUẬN

PCA là một kỹ thuật quan trọng trong học máy, đặc biệt hữu ích trong việc giảm chiều dữ liệu, tối ưu hóa hiệu suất tính toán và trích xuất thông tin quan trọng từ tập dữ liệu lớn. Việc hiểu sâu về đại số tuyến tính và xác suất thống kê đóng vai trò nền tảng cho nhiều thuật toán hiện đại, bao gồm PCA và nhiều phương pháp học máy khác. Do đó, việc tích hợp giảng

dạy đại số tuyến tính, xác suất thống kê cùng với các ứng dụng thực tế như PCA không chỉ giúp sinh viên tiếp thu lý thuyết tốt hơn mà còn phát triển tư duy tính toán và khả năng giải quyết vấn đề trong lĩnh vực khoa học dữ liệu và trí tuệ nhân tạo.

TÀI LIỆU THAM KHẢO

- [1]. Nguyễn Văn Ngọc, Nguyễn Thị Lan Hương, Lê Bích Phương, Lê Thị Hương Giang, Hà Hữu Cao Trình. Giáo trình Đại số tuyến tính, Xuất bản lần thứ nhất. Nhà xuất bản Giao thông vận tải. (2020).
- [2]. Jolliffe, I. T. Principal Component Analysis, Second Edition. Springer-Verlag. (2002).
- [3]. Bishop, C. M.. Pattern Recognition and Machine Learning. Springer. (2006).
- [4]. Nhóm biên dịch DLBOOKVN. Đắm mình vào học sâu. <https://d2l.aivivn.com/> (2022). Truy cập ngày 4/2/2025.
- [5]. Jolliffe, I. T., & Cadima, J. Principal Component Analysis: A Review and Recent Developments. Philosophical Transactions of the Royal Society A, 374(2065) (2016), 20150202.
- [6]. Goodfellow, I., Bengio, Y., & Courville, A.. Deep Learning. MIT Press. (2016).

AN INTERDISCIPLINARY APPROACH TO TEACHING MACHINE LEARNING: INTEGRATING LINEAR ALGEBRA, STATISTICS, AND PCA

Lê Bích Phương^{1,*}, Nguyễn Văn Chung¹, Nguyễn Thị Hằng, Phạm Tuấn Cường

¹ Hanoi University of Mining and Geology, 18 Pho Vien Street, Duc Thang Ward,
Bac Tu Liem District, Hanoi, Vietnam

Corresponding author: Email: lebichphuong@humg.edu.vn; Tel: 0988782112

Abstract.

This paper explores the connection between statistics, linear algebra, and Principal Component Analysis (PCA) in the context of machine learning. PCA is an effective dimensionality reduction technique that optimizes machine learning algorithms by employing linear algebra tools such as matrices, eigenvalues, and eigenvectors. We provide a detailed description of the PCA implementation process, including data standardization, computation of the covariance matrix, identification of eigenvalues and eigenvectors, selection of principal components, and data transformation. The paper also includes a simulated example to demonstrate the practical application of PCA in data analysis. Furthermore, we propose a teaching method that combines theory with practice, enabling students to deeply understand PCA through programming and real-world scenarios. This approach enhances algorithmic thinking and the ability to apply PCA in data science.

Keywords: linear algebra, eigenvalue, eigenvector, machine learning, PCA, covariance matrix.