4.8

Phùng Thị Điệp

Ngày 26 tháng 12 năm 2021

1 Phân tích thành phần chính (PCA)

Ý tưởng chính của phân tích thành phần chính (PCA) là làm giảm kích thước của một tập dữ liệu bao gồm nhiều biến. PCA là một cơ chế giảm tính năng (hoặc trích xuất tính năng), giúp chúng ta xử lý dữ liệu chiều cao với nhiều các tính năng thông minh hơn để giải thích.

1.1 Động lực: Trục chính của một Ellipsoid

Xem xét một phân phối chuẩn d-chiều với vectơ trung bình 0 và ma trận hiệp phương sai $\sum$. Hàm mật độ xác suất tương ứng (xem (2.33)) là

$$\boldsymbol{f(x) = \frac{1}{\sqrt{(2\pi)^n \left|\sum\right|}} e^{\frac{-1}{2}\boldsymbol{x}^T \sum^{-1} \boldsymbol{x}}, \ x \in R^d.}$$

Nếu chúng ta vẽ nhiều mẫu iid từ hàm mật độ xác suất này, các điểm gần như sẽ có dạng ellipsoid, như minh họa trong Hình 3.1, và tương ứng với các đường bao của f: tập hợp các điểm x sao cho $\boldsymbol{x^T \sum^{-1} x = c, \ c \geq 0}$. Đặc biệt, xem xét Ellipsoid

$$\boldsymbol{x^T \sum^{-1} x = 1, \ x \in R^d} \tag{4.42}$$

Giả sử $\boldsymbol{\sum = BB^T}$, ví dụ B là ma trận Cholesky (thấp hơn). Sau đó, như được giải thích trong Ví dụ A.5, ellipsoid (4.42) cũng có thể được xem như là phép biến đổi tuyến tính của hình cầu đơn vị d-chiều qua ma trận B. Hơn nữa, các trục chính của ellipsoid có thể được tìm thấy thông qua một phép phân rã giá trị đơn lẻ (SVD) của B (hoặc $\sum$); xem Phần A.6.5 và Ví dụ A.8. Đặc biệt, giả sử rằng SVD của B là

$$\boldsymbol{B = UDV^T}$$

(lưu ý là SVD của $\sum$ sau đó được $\boldsymbol{UD^2U^T}$).
Các cột của ma trận UD tương ứng với các trục chính của ellipsoid và độ lớn tương đối của các trục được cho bởi các phần tử của ma trận đường chéo D. Nếu một số độ lớn này nhỏ so với các độ lớn khác, thì kích thước sẽ giảm của không gian có thể đạt được bằng cách chiếu mỗi điểm $\boldsymbol{x} \in \boldsymbol{R^d}$ lên không gian con được kéo dài bởi các cột chính (giả sử $\boldsymbol{k} \ll \boldsymbol{d}$) của U - gọi là các thành phần chính. Giả sử không mất tính tổng quát rằng k thành phần chính đầu tiên được cho bởi k cột đầu tiên của U và đặt $\boldsymbol{U_k}$ là ma trận $\boldsymbol{d} \times \boldsymbol{k}$ tương ứng.
Với cơ sở tiêu chuẩn $\boldsymbol{\{e_i\}}$, vectơ $\boldsymbol{x = x_1 e_1 + \cdots + x_d e_d}$ được biểu diễn bằng vectơ d-chiều $\boldsymbol{[x_1, ..., x_d]^T}$. Đối với cơ sở trực chuẩn $\boldsymbol{\{u_i\}}$ được tạo thành bởi các cột của ma trận U, biểu diễn của x là $\boldsymbol{U^T x}$. Tương tự, hình chiếu của bất kỳ điểm x lên không gian con bao trùm bởi k vectơ chính đầu tiên được biểu diễn bằng vectơ k-chiều $\boldsymbol{U^T x}$, đối với cơ sở trực chuẩn được tạo thành bởi các cột của $\boldsymbol{U_k}$. Vì vậy, ý tưởng là nếu một điểm x nằm gần với hình chiếu $\boldsymbol{U_k U_k^T x}$ của nó, chúng ta có thể biểu diễn nó qua k số thay vì d, sử dụng các đặc trưng kết hợp được cho bởi k thành phần chính. Xem Phần A.4 để xem xét các phép chiếu và các cơ sở chính tắc. Ví dụ 4.10 (thành phần chính), xét ma trận

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{14} & \mathbf{8} & \mathbf{3} \\ \mathbf{8} & \mathbf{5} & \mathbf{2} \\ \mathbf{3} & \mathbf{2} & \mathbf{1} \end{bmatrix},$$

có thể được viết thành $\boldsymbol{\Sigma = BB^T}$, với

$$\boldsymbol{B} = \begin{bmatrix} \mathbf{1} & \mathbf{2} & \mathbf{3} \\ \mathbf{0} & \mathbf{1} & \mathbf{2} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix},$$

Hình 1 mô tả ellipsoid $\boldsymbol{x^T \Sigma x = 1}$, có thể thu được bằng cách biến đổi tuyến tính các điểm trên hình cầu đơn vị nhờ ma trận B. Các trục chính và kích thước của ellipsoid được tìm thấy thông qua sự phân rã giá trị đơn lẻ $\boldsymbol{B = UDV^T}$, trong đó U và D là

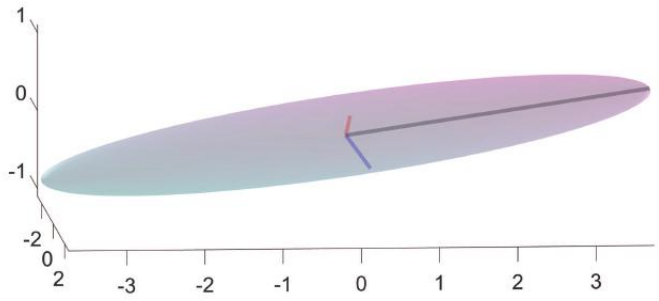
$$\boldsymbol{U} = \begin{bmatrix} \mathbf{0.8460} & \mathbf{0.4828} & \mathbf{0.2261} \\ \mathbf{0.4973} & \mathbf{-0.5618} & \mathbf{-0.6611} \\ \mathbf{0.1922} & \mathbf{-0.6718} & \mathbf{0.7154} \end{bmatrix}$$

và

$$\boldsymbol{D} = \begin{bmatrix} \mathbf{4.4027} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0.7187} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0.3160} \end{bmatrix}$$

Các cột của U cho biết hướng của các trục chính của ellıpsoid, và các phần tử khác nhau của D chỉ ra độ lớn tương đối của các trục chính. Chúng ta thấy rằng thành phần chính đầu tiên được cho bởi cột đầu tiên của U và thành phần chính thứ hai được cho bởi cột thứ hai của U.
Hình chiếu của điểm $\boldsymbol{x = [1.052, 0.6648, 0.2271]^T}$ lên không gian 1 chiều

được bao bởi thành phần chính thứ nhất $\boldsymbol{u_1} = [\mathbf{0.8460, 0.4972, 0.1922}]^{\boldsymbol{T}}$ là $\boldsymbol{z} = \boldsymbol{u_1 u_1^T x} = [\mathbf{1.0696, 0.6287, 0.2429}]^{\boldsymbol{T}}$. Đối với vectơ cơ sở $\boldsymbol{u_1}, \boldsymbol{z}$ được biểu diễn bằng số $\boldsymbol{u_1^T z} = \mathbf{1.2643}$. Tức là, $\boldsymbol{z} = \mathbf{1.2643}\boldsymbol{u_1}$.



Hình 1: Một ellipsoid "ván lướt sóng" trong đó một trục chính lớn hơn đáng kể so với hai trục còn lại.

1.2 PCA và Phân tích Giá trị Số ít (SVD)

Trong cài đặt trên, chúng ta không xem xét bất kỳ tập dữ liệu nào được rút ra từ hàm phân phối xác suất đa biến f. Toàn bộ phân tích dựa trên đại số tuyến tính. Trong phân tích thành phần chính (PCA), chúng ta bắt đầu với dữ liệu $\boldsymbol{x_1, ..., x_n}$, trong đó mỗi x là d-chiều. PCA không yêu cầu giả định về cách thu thập dữ liệu, nhưng để tạo liên kết với phần trước, chúng ta có thể nghĩ về dữ liệu khi iid lấy từ một hàm phân phối xác suất thông thường đa biến. Hãy để chúng ta thu thập dữ liệu trong ma trận X theo cách thông thường; đó là,

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x_{11}} & \boldsymbol{x_{12}} & \boldsymbol{\dots} & \boldsymbol{x_{1d}} \\ \boldsymbol{x_{21}} & \boldsymbol{x_{22}} & \boldsymbol{\dots} & \boldsymbol{x_{2d}} \\ \boldsymbol{\vdots} & \boldsymbol{\vdots} & \boldsymbol{\vdots} & \boldsymbol{\vdots} \\ \boldsymbol{x_{n1}} & \boldsymbol{x_{n2}} & \boldsymbol{\dots} & \boldsymbol{x_{nd}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x_1^T} \\ \boldsymbol{x_2^T} \\ \boldsymbol{\vdots} \\ \boldsymbol{x_n^T} \end{bmatrix}$$

Ma trận X sẽ là đầu vào của PCA. Theo cài đặt này, dữ liệu bao gồm các điểm trong không gian d-chiều và mục tiêu của chúng ta là trình bày dữ liệu bằng cách sử dụng n vectơ đặc trưng của kích thước k < d. Theo phần trước, chúng ta giả định rằng phân phối cơ bản của dữ liệu có vectơ kỳ vọng 0. Trong thực tế, điều này có nghĩa là trước khi áp dụng PCA, dữ liệu cần được tập trung bằng cách trừ đi giá trị trung bình của cột trong mỗi cột: $\boldsymbol{x'_{ij}} = \boldsymbol{x_{ij}} - \boldsymbol{\bar{x}_j}$, trong đó $\boldsymbol{\bar{x}_j} = \dfrac{\mathbf{1}}{\boldsymbol{n}} \displaystyle\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{x_{ij}}$.

Từ bây giờ, chúng ta giả định rằng dữ liệu đến từ phân phối d-chiều tổng quát với vectơ trung bình 0 và một số ma trận hiệp phương sai $\boldsymbol{\Sigma}$. Theo định nghĩa, ma trận hiệp phương sai $\boldsymbol{\Sigma}$ bằng với kỳ vọng của ma trận ngẫu nhiên $\boldsymbol{XX^T}$ và có thể được ước tính từ dữ liệu $\boldsymbol{x_1, ..., x_n}$ qua giá trị trung bình mẫu

$$\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} x_i x_i^T = \frac{1}{n} X^T X.$$

Khi $\hat{\Sigma}$ là một ma trận hiệp phương sai, chúng ta có thể tiến hành phân tích $\hat{\Sigma}$ tương tự như chúng ta đã làm cho Σ trong phần trước. Cụ thể, giả sử $\hat{\Sigma} = UD^2U^T$ là SVD của $\hat{\Sigma}$ và đặt U_k là ma trận có các cột là k thành phần chính; nghĩa là, k cột của U tương ứng với các phần tử đường chéo lớn nhất trong D^2. Lưu ý rằng chúng ta đã sử dụng D^2 thay vì D để tương đồng với phần trước. Phép biến đổi $z = U_k U_k^T x_i$ ánh xạ mỗi vectơ $x_i \in R^d$ (do đó, với d đặc điểm) thành một vectơ $z_i \in R^d$ nằm trong không gian con được kéo dài bởi các cột của U_k. Theo cơ sở này, điểm z_i có biểu diễn $z_i = U_k^T(U_k U_k^T x_i) = U_k^T x_i \in R^k$ (do đó có k đặc điểm). Ma trận hiệp phương sai tương ứng của $z_i, i = 1, ..., n$ là đường chéo. Các phần tử đường chéo d_{ll} của D có thể được hiểu là độ lệch chuẩn của dữ liệu theo hướng của các thành phần chính. Đại lượng $v = \Sigma_{l=1} d_{ll}^2$ (nghĩa là dấu vết của D^2) do đó là một thước đo cho lượng phương sai trong dữ liệu. Tỷ lệ d_{ll}^2/v cho biết mức độ phương sai trong dữ liệu được giải thích bằng thành phần chính thứ l.

Một cách khác để xem xét PCA là xem xét câu hỏi: Làm thế nào chúng ta có thể chiếu dữ liệu lên không gian con k-chiều một cách tốt nhất theo cách mà tổng bình phương khoảng cách giữa các điểm được chiếu và các điểm gốc là nhỏ nhất? Từ Phần A.4, chúng ta biết rằng bất kỳ phép chiếu trực giao nào lên không gian con k-chiều V_k có thể được biểu diễn bằng một ma trận $U_k U_k^T$, trong đó $U_k = [u_1, ..., u_k]$ và $\{u_l, \ell = 1, ..., k\}$ là các vectơ trực giao có độ dài 1 kéo dài V_k. Do đó, câu hỏi trên có thể được xây dựng dưới dạng phương trình giảm thiểu:

$$\min_{u_1,...,u_k} \sum_{i=1}^{n} \left\| x_i - U_k U_k^T x_i \right\|^2. \quad (4.43)$$

Bây giờ quan sát

$$\begin{aligned}
&\tfrac{1}{n}\sum_{i=1}^{n} \left\| x_i - U_k U_k^T x_i \right\|^2 = \tfrac{1}{n}\sum_{i=1}^{n} (x_i^T - x_i^T U_k U_k^T)(x_i - x_i U_k U_k^T) \\
&= \tfrac{1}{n}\sum_{i=1}^{n} \|x_i\|^2 - \tfrac{1}{n}\sum_{i=1}^{n} (x_i^T U_k U_k^T x_i) = c - \tfrac{1}{n}\sum_{i=1}^{n}\sum_{l=1}^{k} tr(x_i^T u_l u_l^T x_i) \\
&= c - \tfrac{1}{n}\sum_{l=1}^{k}\sum_{i=1}^{n} u_l^T x_i x_i^T u_l = c - \sum_{l=1}^{k} u_l^T \hat{\Sigma} u_l,
\end{aligned}$$

trong đó chúng ta đã sử dụng tính chất tuần hoàn của một vết (Định lý A.1) và thực tế là $U_k U_k^T$ có thể được viết dưới dạng $\sum_{l=1}^{k} u_l u_l^T$ theo đó bài toán tối thiểu hóa (4.43) tương đương với bài toán tối đa hóa $\max_{u_1,...,u_k} \sum_{l=1}^{k} u_l^T \hat{\Sigma} u_l$ (4.44).

Cực đại này có thể lớn nhất là $\sum_{l=1}^{k} \boldsymbol{d_{ll}}^2$ và đạt được chính xác khi $\boldsymbol{u_1, ..., u_k}$ là k thành phần chính đầu tiên của $\hat{\boldsymbol{\Sigma}}$.

Ví dụ 4.11 (Phân tích giá trị đơn lẻ) Tập dữ liệu sau đây bao gồm các mẫu phụ thuộc từ phân bố Gaussian ba chiều với vectơ trung bình 0 và ma trận hiệp phương sai $\hat{\boldsymbol{\Sigma}}$ được cho trong ví dụ 4.10:

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{3.1209} & \mathbf{1.7438} & \mathbf{0.5479} \\ \mathbf{-2.6628} & \mathbf{-1.5310} & \mathbf{-0.2763} \\ \mathbf{3.7284} & \mathbf{3.0648} & \mathbf{1.8451} \\ \mathbf{0.4203} & \mathbf{0.3553} & \mathbf{0.4268} \\ \mathbf{-0.7155} & \mathbf{-0.6871} & \mathbf{-0.1414} \\ \mathbf{5.8728} & \mathbf{4.0180} & \mathbf{1.4541} \\ \mathbf{4.8163} & \mathbf{2.4799} & \mathbf{0.5637} \\ \mathbf{2.6948} & \mathbf{1.2384} & \mathbf{0.1533} \\ \mathbf{-1.1376} & \mathbf{-0.4677} & \mathbf{-0.2219} \\ \mathbf{-1.2452} & \mathbf{-0.9942} & \mathbf{-0.4449} \end{bmatrix}$$

Sau khi thay thế X bằng phiên bản ở giữa, SVD $\boldsymbol{UD^2U^T}$ của $\hat{\boldsymbol{\Sigma}} = \boldsymbol{X^T X/n}$ tạo ra ma trận thành phần chính U và ma trận đường chéo D:

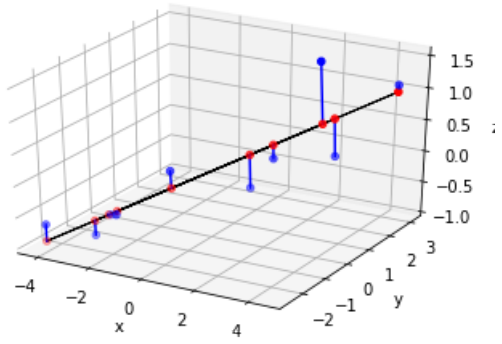
$$\boldsymbol{U} = \begin{bmatrix} \mathbf{-0.8277} & \mathbf{0.4613} & \mathbf{0.3195} \\ \mathbf{-0.5300} & \mathbf{-0.4556} & \mathbf{-0.7152} \\ \mathbf{-0.1843} & \mathbf{-0.7613} & \mathbf{0.6216} \end{bmatrix},$$

và

$$\boldsymbol{D} = \begin{bmatrix} \mathbf{3.3424} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0.4778} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0.1038} \end{bmatrix},$$

Chúng ta cũng nhận thấy rằng, ngoài dấu của cột đầu tiên, ma trận thành phần chính U tương tự như trong Ví dụ 4.10. Tương tự như vậy đối với ma trận D. Chúng ta thấy rằng **97.90%** tổng phương sai được giải thích bởi thành phần chính đầu tiên. Hình 2 cho thấy phép chiếu của dữ liệu được căn giữa lên không gian con được bao trùm bởi thành phần chính này.

Code python sau khi được sử dụng



Hình 2: Dữ liệu từ hàm mật độ phân phối xác suất "ván lướt sóng" được chiếu lên không gian con được kéo dài bởi thành phần chính lớn nhất.

[5]:
```
import numpy as np
X = np.genfromtxt('pcadat.csv', delimiter=',')
n = X.shape[0]
X = X - X.mean(axis=0)
G = X.T @ X
U, _ , _ = np.linalg.svd(G/n)
# Diem du kien
Y = X @ np.outer(U[:,0],U[:,0])
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.w_xaxis.set_pane_color((0, 0, 0, 0))
ax.plot(Y[:,0], Y[:,1], Y[:,2], c='k', linewidth=1)
ax.scatter(X[:,0], X[:,1], X[:,2], c='b')
ax.scatter(Y[:,0], Y[:,1], Y[:,2], c='r')
for i in range(n):
  ax.plot([X[i,0], Y[i,0]], [X[i,1],Y[i,1]], [X[i,2],Y[i,2]], 'b')
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
plt.show()
```

Tiếp theo là một ứng dụng của PCA cho bộ dữ liệu iris nổi tiếng của Fisher, đã được đề cập trong Phần 1.1 và Bài tập 1.5. Ví dụ 4.12 (PCA cho Tập dữ liệu Iris) Bộ dữ liệu iris chứa các phép đo về bốn đặc điểm của cây iris: chiều dài và chiều rộng của lá đài, chiều dài và chiều rộng của cánh hoa, với tổng số 150 mẫu vật. Tập dữ liệu đầy đủ cũng chứa tên loài, nhưng với mục đích của ví dụ này, chúng tôi bỏ qua nó.
Hình 1.9 cho thấy có mối tương quan đáng kể giữa các tính năng khác nhau. Có lẽ chúng ta có thể mô tả dữ liệu bằng cách sử dụng ít tính năng hơn bằng cách lấy một số tổ hợp tuyến tính nhất định của các đối tượng địa lý ban đầu

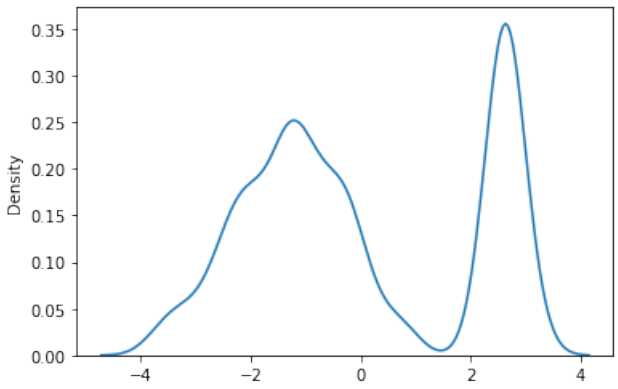
không? Để điều tra điều này, hãy để chúng tôi thực hiện PCA, trước tiên hãy căn giữa dữ liệu. Mã Python sau đây thực hiện PCA. Giả định rằng tệp CSV irisX.csv đã được tạo có chứa tập dữ liệu iris (không có thông tin loài).

[8]:
```
import seaborn as sns, numpy as np
import scipy.linalg
np.set_printoptions(precision=4)
X = np.genfromtxt('IrisX.csv',delimiter=',')
n = X.shape[0]
X = X - np.mean(X, axis=0)
[U,D2,UT]= np.linalg.svd((X.T @ X)/n)
print('U = \n', U)
print('\n diag(D^2) = ', D2)
z = U[:,0].T @ X.T
sns.kdeplot(z, bw=0.15)
```

```
U =
 [[-0.3614 -0.6566  0.582   0.3155]
 [ 0.0845 -0.7302 -0.5979 -0.3197]
 [-0.8567  0.1734 -0.0762 -0.4798]
 [-0.3583  0.0755 -0.5458  0.7537]]

 diag(D^2) =  [4.2001 0.2411 0.0777 0.0237]
```

Kết quả ở trên hiển thị ma trận thành phần chính (mà chúng ta gọi là U) cũng như đường chéo của ma trận $\boldsymbol{D^2}$. Chúng ta thấy rằng một tỷ lệ lớn của phương sai, $\mathbf{4.2001/(4.2001 + 0,2411 + 0,0777 + 0,0237) = 92.46\%}$ được giải thích bởi thành phần chính đầu tiên. Do đó, việc biến đổi mỗi điểm dữ liệu $\boldsymbol{x} \in \boldsymbol{R^4}$ thành $\boldsymbol{{u_1}^T x} \in \boldsymbol{R}$. Hình 3 cho thấy ước tính mật độ hạt nhân của dữ liệu đã biến đổi. Điều thú vị là chúng ta thấy có hai chế độ, chỉ ra ít nhất hai cụm trong dữ liệu.



Hình 3: Ước tính mật độ nhân của dữ liệu iris kết hợp PCA.