BÀI THU HOẠCH
XỬ LÝ SỐ LIỆU THỐNG KÊ

Nhóm 4:
19110311 - Nguyễn Ngô Trung Hậu
19110257 - Trần Bửu Ân
19110281 - Phùng Thị Điệp
19110315 - Trịnh Ngọc Hiến
19110327 - Nguyễn Thị Cẩm Hương

Chương 4: Unsupervised Learning
(Học tập không giám sát)

Nhóm 4

2021

4.1 - 4.2 - 4.3

Trịnh Ngọc Hiến

2021

1 Giới thiệu.

Ngược lại với học có giám sát, trong đó biến y "đầu ra" (phản hồi) được giải thích bằng một vectơ "đầu vào" (giải thích) x, trong học tập không giám sát, không có biến phản hồi và mục tiêu tổng thể là trích xuất thông tin hữu ích và các mẫu từ dữ liệu, ví dụ: cho dạng $\boldsymbol{\tau := \{x_1, ..., x_n\}}$ hoặc dưới dạng ma trận $\boldsymbol{X^T = [x_1, ..., x_n]}$. Về bản chất, mục tiêu của học không giám sát là học về phân phối xác suất cơ bản của dữ liệu.

Chúng ta bắt đầu trong **Section 4.2** bằng cách thiết lập một khuôn khổ cho việc học tập không giám sát đó là tương tự như khung được sử dụng cho học có giám sát trong Section 2.3. Đó là, chúng tôi lập công thức học tập không có giám sát trong điều kiện giảm thiểu rủi ro và tổn thất; nhưng bây giờ liên quan đến rủi ro chéo entropy, thay vì rủi ro sai số bình phương. Theo cách tự nhiên, điều này dẫn đến học các khái niệm như khả năng xảy ra, thông tin Fisher và tiêu chí thông tin Akaike. **Section 4.3** giới thiệu thuật toán Kỳ vọng – Tối đa hóa (EM) như một thuật toán hữu ích phương pháp để tối đa hóa các hàm hợp lý khi không thể dễ dàng tìm thấy giải pháp của chúng trong dạng đóng.

Nếu dữ liệu tạo thành một mẫu iid từ một số phân phối không xác định, thì "phân phối thực nghiệm" của dữ liệu cung cấp thông tin có giá trị về phân phối chưa biết. Trong **Section 4.4** chúng tôi chính thức khái niệm hóa về phân phối thực nghiệm (tổng quát của hàm phân phối chuẩn tích luỹ theo kinh nghiệm) và giải thích cách chúng tôi có thể đưa ra ước tính về xác suất cơ bản của hàm mật độ của dữ liệu bằng cách sử dụng công cụ ước tính mật độ hạt nhân.

Hầu hết các kỹ thuật học tập không giám sát tập trung vào việc xác định các đặc điểm nhất định của phân phối cơ bản, chẳng hạn như các điểm tối đa hóa cục bộ của nó. Một ý tưởng liên quan là phân vùng dữ liệu thành các cụm điểm theo nghĩa nào đó "tương tự" với nhau. Trong **Section 4.5**, chúng tôi hình thành vấn đề phân cụm theo mô hình hỗn hợp. Đặc biệt, dữ liệu được giả định đến từ hỗn hợp các phân phối (thường là Gaussian) và mục tiêu là khôi phục các tham số của sự phân bố hỗn hợp từ dữ liệu. Công cụ chính cho tham số ước lượng trong các mô hình hỗn hợp là thuật toán EM.

Section 4.6 nói về một cách tiếp cận heuristic hơn để phân cụm, nơi dữ liệu được nhóm theo một số "trung tâm cụm" nhất định, có vị trí được tìm thấy bằng cách giải quyết một vấn đề tối ưu hóa. **Section 4.7** mô tả cách các cụm có thể được xây dựng theo cách phân cấp.

Cuối cùng, trong **Section 4.8**, chúng ta nói về kỹ thuật học tập không giám sát được gọi là Phân tích Thành phần Chính (PCA), đây là một công cụ quan trọng để giảm bớt các chiều của dữ liệu.

Chúng ta sẽ gặp lại các kỹ thuật học tập không có giám sát khác nhau trong các chương tiếp theo về học tập có giám sát. Ví dụ: giảm thiểu tổn thất đào tạo qua entropy sẽ nằm trong hồi quy logistic (**Section 5.7**) và phân loại (**Chương 7**), và PCA có thể được sử dụng trong hồi quy logistic (**Section 5.7**) và phân loại (**Chương 7**), và PCA có thể được sử dụng cho lựa chọn biến và giảm kích thước, để làm cho các mô hình dễ đào tạo hơn và tăng sức mạnh dự đoán của họ; xem ví dụ: **Section 6.8** và **7.4**.

2 Rủi ro và mất mát trong học không giám sát (Risk and Loss in Unsupervised Learning).

Trong học tập không giám sát, dữ liệu đào tạo $\boldsymbol{\mathcal{T} := \{X_1, ..., X_n\}}$ chỉ bao gồm (những gì là thường được giả định) các bản sao độc lập của một vectơ đặc trưng X; không có phản hồi dữ liệu. Giả sử mục tiêu của chúng ta là tìm hiểu hàm mật độ xác suất $\boldsymbol{f}$ chưa biết của X dựa trên một kết quả $\boldsymbol{\tau := \{x_1, ..., x_n\}}$ của dữ liệu huấn luyện T. Chúng ta có thể thuận tiện tuân theo cùng một dòng lý luận như đối với học có giám sát, được thảo luận trong **Section 2.3–2.5**. Bảng 4.1 - Tóm tắt định nghĩa cho trường hợp học tập không có giám sát. So sánh điều này với Bảng 2.1 cho trường hợp được giám sát.

Tương tự như học có giám sát, chúng ta muốn tìm một hàm $\boldsymbol{g}$, thứ mà bây giờ là một xác suất mật độ (liên tục hoặc rời rạc), gần đúng nhất với hàm mật độ xác suất $\boldsymbol{f}$ về mặt giảm thiểu rủi ro

$$\boldsymbol{\ell(g) := \mathbb{E}Loss(f(X), g(X)),} \tag{4.1}$$

trong đó Loss là một hàm mất mát. Trong (2.27), chúng tôi đã gặp phải rủi ro Kullback – Leibler

$$\boldsymbol{\ell(g) := \mathbb{E} \ln \frac{f(X)}{g(X)} = \mathbb{E} \ln f(X) - \mathbb{E} \ln g(X)} \tag{4.2}$$

Nếu $\boldsymbol{\mathcal{G}}$ là một lớp hàm chứa $\boldsymbol{f}$, thì việc giảm thiểu rủi ro Kullback – Leibler qua $\boldsymbol{\mathcal{G}}$ sẽ mang lại bộ thu nhỏ (đúng) $\boldsymbol{f}$. Tất nhiên, vấn đề là việc giảm thiểu (4.2) phụ thuộc vào $\boldsymbol{f}$, thường không được biết đến. Tuy nhiên, vì thuật ngữ $\boldsymbol{\mathbb{E} \ln f(X)}$ không phụ thuộc vào $\boldsymbol{g}$, nó không đóng vai trò gì trong việc giảm thiểu rủi ro Kullback-Leibler. Bằng cách loại bỏ số hạng này, chúng ta thu được rủi ro chéo entropy (đối với X rời rạc thay thế tích phân bằng một tổng):

$$\boldsymbol{\ell(g) := -\mathbb{E} \ln g(X) = -\int f(x) \ln g(x) dx} \tag{4.3}$$

$\boldsymbol{x}$	Fixed feature vector.
$\boldsymbol{X}$	Random feature vector.
$f(\boldsymbol{x})$	Pdf of $\boldsymbol{X}$ evaluated at the point $\boldsymbol{x}$.
τ or τ_n	Fixed training data $\{\boldsymbol{x}_i, i = 1, \ldots, n\}$.
$\mathcal{T}$ or $\mathcal{T}_n$	Random training data $\{\boldsymbol{X}_i, i = 1, \ldots, n\}$.
g	Approximation of the pdf f.
$\text{Loss}(f(\boldsymbol{x}), g(\boldsymbol{x}))$	Loss incurred when approximating $f(\boldsymbol{x})$ with $g(\boldsymbol{x})$.
$\ell(g)$	Risk for approximation function g; that is, $\mathbb{E}\,\text{Loss}(f(\boldsymbol{X}), g(\boldsymbol{X}))$.
$g^{\mathcal{G}}$	Optimal approximation function in function class $\mathcal{G}$; that is, $\text{argmin}_{g\in\mathcal{G}}\,\ell(g)$.
$\ell_\tau(g)$	Training loss for approximation function (guess) g; that is, the sample average estimate of $\ell(g)$ based on a fixed training sample τ.
$\ell_{\mathcal{T}}(g)$	The same as $\ell_\tau(g)$, but now for a random training sample $\mathcal{T}$.
$g_\tau^{\mathcal{G}}$ or g_τ	The *learner*: $\text{argmin}_{g\in\mathcal{G}}\,\ell_\tau(g)$. That is, the optimal approximation function based on a fixed training set τ and function class $\mathcal{G}$. We suppress the superscript $\mathcal{G}$ if the function class is implicit.
$g_{\mathcal{T}}^{\mathcal{G}}$ or $g_{\mathcal{T}}$	The learner for a random training set $\mathcal{T}$.

Hình 1: Bảng 4.1: Tóm tắt các định nghĩa cho việc học không giám sát.

Do đó, việc giảm thiểu rủi ro chéo entropy (4.3) trên tất cả $\boldsymbol{g} \in \boldsymbol{\mathcal{G}}$,một lần nữa cho bộ giảm thiểu $\boldsymbol{f}$, với điều kiện là $\boldsymbol{f} \in \boldsymbol{\mathcal{G}}$. Thật không may, việc giải quyết (4.3) nói chung cũng không khả thi, vì nó vẫn phụ thuộc vào $\boldsymbol{f}$. Thay vào đó, chúng tôi tìm cách giảm thiểu tổn thất do tạo entropy chéo:

$$\boldsymbol{\ell_r(g) := \frac{1}{n}\sum_{i=1}^{n} Loss(f(x_i), g(x_i)) = -\frac{1}{n}\sum_{i=1}^{n} \ln g(x_i)} \tag{4.4}$$

trên lớp của các hàm $\boldsymbol{\mathcal{G}}$, trong đó $\boldsymbol{\tau := \{x_1, ..., x_n\}}$ là một mẫu iid từ $\boldsymbol{f}$. Việc tối ưu hóa này có thể thực hiện được mà không cần biết $\boldsymbol{f}$ và tương đương với việc giải quyết vấn đề tối đa hóa

$$\boldsymbol{\max_{g\in\mathcal{G}} \sum_{i=1}^{n} \ln g x_i}$$

Bước quan trọng trong việc thiết lập quy trình học là chọn một lớp chức năng $\boldsymbol{\mathcal{G}}$ phù hợp để tối ưu hóa. Cách tiếp cận tiêu chuẩn là tham số hóa $\boldsymbol{g}$ với một tham số $\boldsymbol{\theta}$ và gọi $\boldsymbol{\mathcal{G}}$ là lớp của các hàm $\boldsymbol{\{g(\cdot|\theta), \theta \in \Theta\}}$ đối với một tập tham số p-chiều $\boldsymbol{\Theta}$. Đối với phần còn lại của Section 4.2, chúng tôi sẽ sử dụng lớp hàm này, cũng như nguy cơ entropy chéo.

Hàm $\boldsymbol{\theta \mapsto g(x|\theta)}$ được gọi là *hàm hợp lý xảy ra.* Nó cho khả năng xuất hiện của vectơ đặc trưng quan sát $\boldsymbol{x}$ dưới $\boldsymbol{g(\cdot|\theta)}$, như một hàm của tham số $\boldsymbol{\theta}$.

Lôgarit tự nhiên của hàm hợp lý được gọi là hàm log - hàm hợp lý xảy ra và gradient của nó đối với $\boldsymbol{\theta}$ được gọi là hàm điểm, ký hiệu là $\boldsymbol{S(x|\theta)}$; đó là,

$$\boldsymbol{S}(\boldsymbol{x}\,|\,\boldsymbol{\theta}) := \frac{\partial \ln g(\boldsymbol{x}\,|\,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{\frac{\partial g(\boldsymbol{x}\,|\,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}}{g(\boldsymbol{x}\,|\,\boldsymbol{\theta})}. \tag{4.6}$$

Điểm ngẫu nhiên $\boldsymbol{S(X|\theta)}$, với $\boldsymbol{X \sim g(\cdot|\theta)}$, được quan tâm đặc biệt. Trong nhiều trường hợp, kỳ vọng của nó *bằng vectơ không*; cụ thể là

$$\begin{aligned} \mathbb{E}_{\boldsymbol{\theta}}\boldsymbol{S}(\boldsymbol{X}\,|\,\boldsymbol{\theta}) &= \int \frac{\frac{\partial g(\boldsymbol{x}\,|\,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}}{g(\boldsymbol{x}\,|\,\boldsymbol{\theta})}\, g(\boldsymbol{x}\,|\,\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{x} \\ &= \int \frac{\partial g(\boldsymbol{x}\,|\,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\,\mathrm{d}\boldsymbol{x} = \frac{\partial \int g(\boldsymbol{x}\,|\,\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{x}}{\partial \boldsymbol{\theta}} = \frac{\partial 1}{\partial \boldsymbol{\theta}} = \mathbf{0}, \end{aligned} \tag{4.7}$$

với điều kiện là sự xen kẽ của sự khác biệt và tích hợp là hợp lý. Điều này đúng với một số lượng lớn các phân phối, bao gồm cả phân phối chuẩn, hàm mũ và nhị thức. Các trường hợp ngoại lệ đáng chú ý là các bản phân phối có hỗ trợ phụ thuộc vào tham số phân phối; ví dụ phân phối $\boldsymbol{\mathcal{U}(0,\theta)}$.

Lưu ý:
Điều quan trọng là phải xem liệu các kỳ vọng được coi là $X \sim g(\cdot|\theta)$ hay $X \sim f$. Chúng tôi sử dụng các ký hiệu kỳ vọng $\mathbb{E}_\theta$ và $\mathbb{E}$ để phân biệt hai trường hợp.

Từ bây giờ, chúng tôi chỉ đơn giản giả định rằng sự xen kẽ của sự khác biệt và tích hợp là được phép; xem, ví dụ, [76] để biết các điều kiện đủ. Ma trận hiệp phương sai của điểm ngẫu nhiên $\boldsymbol{S(X|\theta)}$ được gọi là *ma trận thông tin Fisher*, chúng ta ký hiệu là $\boldsymbol{F}$ hoặc $\boldsymbol{F(\theta)}$ để thể hiện sự phụ thuộc của nó vào $\boldsymbol{\theta}$. Vì điểm số dự kiến là $\mathbf{0}$, chúng tôi đã

$$\boldsymbol{F(\theta)} = \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{S(X|\theta)S(X|\theta)^T}] \tag{4.8}$$

Ma trận liên quan là ma trận Hessian kỳ vọng của - $\boldsymbol{\ln g(X|\theta)}$:

Lưu ý rằng kỳ vọng ở đây là đối với $\boldsymbol{X \sim f}$. Nó chỉ ra rằng nếu $\boldsymbol{f = g(\cdot|\theta)}$, thì hai ma trận giống nhau; đó là,

$$\boldsymbol{F(\theta)} = \boldsymbol{H(\theta)}, \tag{4.10}$$

$$\mathbf{H}(\boldsymbol{\theta}) := \mathbb{E}\left[-\frac{\partial \boldsymbol{S}(\boldsymbol{X};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right] = -\mathbb{E}\begin{bmatrix} \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial^2 \theta_1} & \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial \theta_1 \partial \theta_2} & \cdots & \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial \theta_1 \partial \theta_p} \\ \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial^2 \theta_2} & \cdots & \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial \theta_2 \partial \theta_p} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial \theta_p \partial \theta_1} & \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial \theta_p \partial \theta_2} & \cdots & \frac{\partial^2 \ln g(\boldsymbol{X}|\boldsymbol{\theta})}{\partial^2 \theta_p} \end{bmatrix}. \tag{4.9}$$

với điều kiện là chúng ta có thể hoán đổi thứ tự của sự khác biệt và tích hợp (kỳ vọng). Kết quả này được gọi là *ma trận bình đẳng thông tin.*

Các ma trận $\boldsymbol{F}(\boldsymbol{\theta})$ và $\boldsymbol{H}(\boldsymbol{\theta})$ đóng vai trò quan trọng trong việc tính gần đúng nguy cơ chéo entropy với n lớn. Để thiết lập bối cảnh, hãy đặt $\boldsymbol{g}^{\boldsymbol{\mathcal{G}}} = \boldsymbol{g}(\cdot|\boldsymbol{\theta}^*)$ là bộ giảm thiểu rủi ro chéo entropy

$$\boldsymbol{r}(\boldsymbol{\theta}) := -\mathbb{E} \ln \boldsymbol{g}(\boldsymbol{X}|\boldsymbol{\theta})$$

Chúng tôi giả định rằng $\boldsymbol{r}$, như một hàm của $\boldsymbol{\theta}$, được xử lý tốt; đặc biệt, trong vùng lân cận của $\boldsymbol{\theta}^*$, nó là lồi nghiêm ngặt và có thể phân biệt hai lần liên tục (điều này đúng, ví dụ, nếu $\boldsymbol{g}$ là mật độ Gauss). Theo đó $\boldsymbol{\theta}^*$ là một gốc của $\mathbb{E}\boldsymbol{S}(\boldsymbol{X}|\boldsymbol{\theta})$, bởi vì

$$\mathbf{0} = \frac{\partial \boldsymbol{r}(\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}} = -\frac{\partial \mathbb{E} \ln \boldsymbol{g}(\boldsymbol{X}|\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}} = -\mathbb{E}\frac{\partial \ln \boldsymbol{g}(\boldsymbol{X}|\boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}} = -\mathbb{E}\boldsymbol{S}(\boldsymbol{X}|\boldsymbol{\theta}^*),$$

một lần nữa với điều kiện là thứ tự của sự khác biệt và tích hợp (kỳ vọng) có thể được hoán đổi. Theo cách tương tự, $\boldsymbol{H}(\boldsymbol{\theta})$ khi đó là ma trận Hessian của r. Gọi $\boldsymbol{g}(\cdot|\widehat{\boldsymbol{\theta}_n})$ là giá trị tối thiểu của tổn thất đào tạo

$$\boldsymbol{r}_{\boldsymbol{\mathcal{T}}_n} := -\frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} \ln \boldsymbol{g}(\boldsymbol{X}_i|\boldsymbol{\theta})$$

trong đó $\boldsymbol{\mathcal{T}}_n = \{\boldsymbol{X}_1, \ldots \boldsymbol{X}_n\}$ là một tập hợp huấn luyện ngẫu nhiên. Gọi $\boldsymbol{r}^*$ là rủi ro chéo entropy nhỏ nhất có thể, được tính trên tất cả các hàm; rõ ràng, $\boldsymbol{r}^* = -\mathbb{E} \ln \boldsymbol{f}(\boldsymbol{X})$, trong đó $\boldsymbol{X} \sim \boldsymbol{f}$. Tương tự như trường hợp học có giám sát, chúng ta có thể phân rã rủi ro tổng quát hóa, $\boldsymbol{\ell}(\boldsymbol{g}(\cdot|\widehat{\boldsymbol{\theta}_n})) = \boldsymbol{r}(\widehat{\boldsymbol{\theta}_n})$, thành

$$r(\widehat{\boldsymbol{\theta}_n}) = r^* + \underbrace{r(\boldsymbol{\theta}^*) - r^*}_{\text{approx. error}} + \underbrace{r(\widehat{\boldsymbol{\theta}_n}) - r(\boldsymbol{\theta}^*)}_{\text{statistical error}} = r(\boldsymbol{\theta}^*) - \mathbb{E} \ln \frac{g(\boldsymbol{X}|\boldsymbol{\theta}^*)}{g(\boldsymbol{X}|\widehat{\boldsymbol{\theta}_n})}.$$

Định lý sau đây chỉ rõ hành vi tiệm cận của các thành phần của rủi ro tổng quát hóa. Trong giả thuyết, chúng tôi giả định rằng $\widehat{\boldsymbol{\theta}_n} \longrightarrow \mathbb{P}\boldsymbol{\theta}^*$ khi n $\longrightarrow \boldsymbol{\infty}$

Định lý 4.1: Xấp xỉ Rủi ro chéo Entropy
Tiệm cận với (n $\rightarrow \boldsymbol{\infty}$) là

$$\mathbb{E}r(\widehat{\theta_n}) - r(\theta^*) \simeq tr(F(\theta^*)H^{-1}(\theta^*))/(2n), \tag{4.11}$$

trong đó

$$r(\theta^*) \simeq \mathbb{E}r_{\mathcal{T}_n}(\widehat{\theta_n}) + tr(F(\theta^*)H^{-1}(\theta^*))/(2n). \tag{4.12}$$

Chứng minh: Khai triển Taylor của $r(\widehat{\theta_n})$ xung quanh θ^* đưa ra sai số thống kê

$$r(\widehat{\theta}_n) - r(\theta^*) = (\widehat{\theta}_n - \theta^*)^\top \underbrace{\frac{\partial r(\theta^*)}{\partial \theta}}_{=\mathbf{0}} + \frac{1}{2}(\widehat{\theta}_n - \theta^*)^\top \mathbf{H}(\bar{\theta}_n)(\widehat{\theta}_n - \theta^*), \tag{4.13}$$

trong đó $\bar{\theta}_n$ nằm trên đoạn thẳng giữa θ^* và $\bar{\theta}_n$. Với n lớn, chúng ta có thể thay $H(\bar{\theta}_n)$ bằng $H(\theta^*)$ vì theo giả thiết, $\widehat{\theta_n}$ hội tụ thành θ^*. Ma trận $H(\theta^*)$ xác định dương vì $r(\theta)$ là hàm lồi nghiêm ngặt tại θ^* theo giả thiết, và do đó khả nghịch. Điều quan trọng là nhận ra rằng $\widehat{\theta_n}$ trên thực tế là một ước lượng M của θ^*. Đặc biệt, trong ký hiệu của Định lý C.19, ta có $\Psi = S$, $A = H(\theta^*)$, và $B = F(\theta^*)$. Do đó, theo cùng một định lý,

$$\sqrt{n}\,(\widehat{\theta}_n - \theta^*) \xrightarrow{\mathrm{d}} \mathcal{N}\left(\mathbf{0}, \mathbf{H}^{-1}(\theta^*)\,\mathbf{F}(\theta^*)\,\mathbf{H}^{-\top}(\theta^*)\right). \tag{4.14}$$

Kết hợp (4.13) với (4.14), theo Định lý C.2, về mặt tiệm cận, sai số ước lượng dự kiến được đưa ra bởi (4.11).

Tiếp theo, chúng ta xét một khai triển Taylor của $r_{\mathcal{T}_n(\theta^*)}$ xung quanh $\widehat{\theta_n}$:

$$r_{\mathcal{T}_n}(\theta^*) = r_{\mathcal{T}_n}(\widehat{\theta}_n) + (\theta^* - \widehat{\theta}_n)^\top \underbrace{\frac{\partial r_{\mathcal{T}_n}(\widehat{\theta}_n)}{\partial \theta}}_{=\mathbf{0}} + \frac{1}{2}(\theta^* - \widehat{\theta}_n)^\top \mathbf{H}_{\mathcal{T}_n}(\bar{\theta}_n)(\theta^* - \widehat{\theta}_n), \tag{4.15}$$

trong đó, $H_{\mathcal{T}_n}(\bar{\theta_n}) := -\frac{1}{n}\sum_{i=1}^{n} \frac{\partial S(X_i|\bar{\theta_n})}{\partial \theta}$ là Hessian của $r_{\mathcal{T}_n}(\theta)$ tại một số $\bar{\theta_n}$ giữa $\widehat{\theta_n}$ và θ^*. Với kỳ vọng ở cả hai mặt của (4.15), chúng tôi nhận được

$$r(\theta^*) = \mathbb{E}r_{\mathcal{T}_n}(\widehat{\theta_n}) + \frac{1}{2}\mathbb{E}(\theta^* - \widehat{\theta_n})^T H_{\mathcal{T}_n}(\bar{\theta_n})(\theta^* - \widehat{\theta_n}).$$

Thay $H_{\mathcal{T}_n}(\bar{\theta_n})$ bằng $H(\theta^*)$ cho n lớn và sử dụng (4.14), chúng ta được

$$n\mathbb{E}(\theta^* - \widehat{\theta_n})^T H_{\mathcal{T}_n}(\bar{\theta_n})(\theta^* - \widehat{\theta_n}) \to tr(F(\theta^*)H^{-1}(\theta^*)), \quad n \to \infty.$$

Do đó, tiệm cận là $n \to \infty$, ta có (4.12).
Định lý 4.1 có một số hệ quả thú vị:

1. Tương tự như Section 2.5.1, mất mát trong đào tạo $\ell_{\mathcal{T}_n}(g_{\mathcal{T}_n}) = r_{\mathcal{T}_n}(\widehat{\theta_n})$ có xu hướng đánh giá thấp rủi ro $\ell(g^{\mathcal{G}}) = r(\theta^*)$, bởi vì tập huấn luyện $\mathcal{T}_n$ được sử dụng để huấn luyện $g \in \mathcal{G}$ (nghĩa là ước lượng θ^*) và ước tính rủi ro. Quan hệ (4.12) cho chúng ta biết rằng trung bình tổn thất đào tạo đánh giá thấp rủi ro thực sự bằng $tr(F(\theta^*)H^{-1}(\theta^*))/(2n)$.

2. Thêm các phương trình (4.11) và (4.12), sẽ mang lại giá trị xấp xỉ tiệm cận sau cho rủi ro tổng quát hóa dự kiến:

$$\mathbb{E}\, r(\widehat{\theta}_n) \simeq \mathbb{E}\, r_{\mathcal{T}_n}(\widehat{\theta}_n) + \frac{1}{n}\mathrm{tr}\left(\mathbf{F}(\theta^*)\,\mathbf{H}^{-1}(\theta^*)\right) \tag{4.16}$$

Số hạng đầu tiên ở phía bên phải của (4.16) có thể được ước lượng (không có độ lệch) thông qua tổn thất huấn luyện $r_{\mathcal{T}_n}(\widehat{\theta_n})$. Đối với thuật ngữ thứ hai, chúng ta đã đề cập rằng khi mô hình thực $f \in \mathcal{G}$, thì $F(\theta^*) = H(\theta^*)$. Do đó, khi $\mathcal{G}$ được coi là một tập đầy đủ được tham số hóa bởi vectơ p- chiều θ, chúng ta có thể tính gần đúng số hạng thứ hai là $tr(F(\theta^*)H^{-1}(\theta^*))/n \approx tr(I_p)/n = p/n$. Điều này gợi ý sự xấp xỉ theo kinh nghiệm sau đối với rủi ro tổng quát hóa (dự kiến):

$$\mathbb{E}\, r(\widehat{\theta}_n) \approx r_{\mathcal{T}_n}(\widehat{\theta}_n) + \frac{p}{n}. \tag{4.17}$$

3. Nhân cả hai vế của (4.16) với 2n và thay $tr(F(\theta^*)H^{-1}(\theta^*)) \approx p$, ta được giá trị gần đúng:

$$2n\, r(\widehat{\theta}_n) \approx -2\sum_{i=1}^{n} \ln g(X_i \,|\, \widehat{\theta}_n) + 2p. \tag{4.18}$$

Vế phải của (4.18) được gọi là tiêu chí thông tin Akaike (AIC). Cũng giống như (4.17), phép gần đúng AIC có thể được sử dụng để so sánh sự khác biệt về rủi ro tổng quát hóa của hai hoặc nhiều người học. Chúng tôi muốn người học có rủi ro tổng quát hóa (ước tính) nhỏ nhất.

Giả sử rằng, đối với một tập huấn luyện $\mathcal{T}$, tổn thất huấn luyện $r_{\mathcal{T}}(\boldsymbol{\theta})$ có một điểm cực tiểu duy nhất $\widehat{\boldsymbol{\theta}}$ nằm trong phần bên trong của $\boldsymbol{\Theta}$. Nếu $r_{\mathcal{T}}(\boldsymbol{\theta})$ là một hàm phân biệt đối với $\boldsymbol{\theta}$, thì chúng ta có thể tìm tham số tối ưu $\widehat{\boldsymbol{\theta}}$ bằng cách giải

$$\frac{\partial r_{\mathcal{T}}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \underbrace{\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{S}(\boldsymbol{X}_i \,|\, \boldsymbol{\theta})}_{\boldsymbol{S}_{\mathcal{T}}(\boldsymbol{\theta})} = \mathbf{0}.$$

Nói cách khác, ước lượng hợp lý lớn nhất $\widehat{\boldsymbol{\theta}}$ cho $\boldsymbol{\theta}$ thu được bằng cách giải quyết gốc của hàm điểm trung bình, nghĩa là, bằng cách giải

$$\boldsymbol{S}_{\mathcal{T}}(\boldsymbol{\theta}) = \mathbf{0}. \tag{4.19}$$

Thông thường ta không thể tìm thấy $\widehat{\boldsymbol{\theta}}$ ở dạng rõ ràng. Trong trường hợp đó, người ta cần giải phương trình (4.19) bằng số. Tồn tại nhiều kỹ thuật tiêu chuẩn để tìm gốc, ví dụ: thông qua phương pháp của Newton (xem Phần B.3.1), theo đó, bắt đầu từ phỏng đoán ban đầu $\boldsymbol{\theta_0}$, các lần lặp tiếp theo được thu thập thông qua quá trình lặp

$$\boldsymbol{\theta_{t+1}} = \boldsymbol{\theta_t} + \boldsymbol{H}_{\mathcal{T}}^{-1}(\boldsymbol{\theta_t})\boldsymbol{S}_{\mathcal{T}}(\boldsymbol{\theta_t}),$$

trong đó,

$$\boldsymbol{H}_{\mathcal{T}}(\boldsymbol{\theta}) := \frac{-\partial \boldsymbol{S}_{\mathcal{T}}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{1}{n}\sum_{i=1}^{n} -\frac{\partial \boldsymbol{S}(\boldsymbol{X}_i|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$$

là ma trận Hessian trung bình của $\{-\ln g(\boldsymbol{X}_i|\boldsymbol{\theta})_{i=1}^{n}\}$. Dưới $f = g(\cdot|\boldsymbol{\theta})$, kỳ vọng của $\boldsymbol{H}_{\mathcal{T}}(\boldsymbol{\theta})$ bằng ma trận thông tin $\boldsymbol{F}(\boldsymbol{\theta})$, không phụ thuộc vào dữ liệu. Điều này gợi ý một sơ đồ lặp thay thế, được gọi là *phương pháp tính điểm của Fisher*:

$$\boldsymbol{\theta_{t+1}} = \boldsymbol{\theta_t} + \boldsymbol{F}^{-1}(\boldsymbol{\theta_t})\boldsymbol{S}_{\mathcal{T}}(\boldsymbol{\theta_t}), \tag{4.20}$$

điều này không chỉ dễ thực hiện hơn (nếu ma trận thông tin có thể được đánh giá dễ dàng) mà còn ổn định hơn về mặt số học.

VD: 4.1 (Khả năng tối đa cho Phân phối Gamma) Chúng tôi muốn tính gần đúng mật độ của phân phối Gamma $(\boldsymbol{\alpha^*}, \boldsymbol{\lambda^*})$ cho một số tham số đúng nhưng chưa biết $\boldsymbol{\alpha^*}$ và $\boldsymbol{\lambda^*}$, trên cơ sở tập huấn luyện $\boldsymbol{\tau} := \{\boldsymbol{x_1}, ..., \boldsymbol{x_n}\}$ trong số các mẫu iid từ bản phân phối này. Chọn hàm gần đúng của chúng ta $\boldsymbol{g}(\cdot|\boldsymbol{\alpha}, \boldsymbol{\lambda})$ trong cùng một loại mật độ gamma,

$$g(\cdot|\alpha,\lambda) = \frac{\lambda^{\alpha} x^{\alpha-1} e^{-\lambda x}}{\Gamma(\alpha)}, \quad x \geqslant 0, \tag{4.21}$$

với $\alpha > 0$ và $\lambda > 0$, chúng ta tìm cách giải (4.19). Lấy lôgarit trong (4.21), hàm hợp lý được cho bởi

$$l(x|\alpha,\lambda) := \alpha \ln \lambda - \ln \Gamma(\alpha) + (\alpha - 1) \ln x - \lambda x.$$

Theo sau đó,

$$S(\alpha,\lambda) = \begin{bmatrix} \frac{\partial}{\partial\alpha} l(x\,|\,\alpha,\lambda) \\ \frac{\partial}{\partial\lambda} l(x\,|\,\alpha,\lambda) \end{bmatrix} = \begin{bmatrix} \ln\lambda - \psi(\alpha) + \ln x \\ \frac{\alpha}{\lambda} - x \end{bmatrix},$$

trong đó, Ψ là đạo hàm của $\ln\Gamma$: cái gọi là *hàm digamma*. Kể từ đây

$$\mathbf{H}(\alpha,\lambda) = -\mathbb{E}\begin{bmatrix} \frac{\partial^2}{\partial\alpha^2} l(X\,|\,\alpha,\lambda) & \frac{\partial^2}{\partial\alpha\partial\lambda} l(X\,|\,\alpha,\lambda) \\ \frac{\partial^2}{\partial\alpha\partial\lambda} l(X\,|\,\alpha,\lambda) & \frac{\partial^2}{\partial\lambda^2} l(X\,|\,\alpha,\lambda) \end{bmatrix} = -\mathbb{E}\begin{bmatrix} -\psi'(\alpha) & \frac{1}{\lambda} \\ \frac{1}{\lambda} & -\frac{\alpha}{\lambda^2} \end{bmatrix} = \begin{bmatrix} \psi'(\alpha) & -\frac{1}{\lambda} \\ -\frac{1}{\lambda} & \frac{\alpha}{\lambda^2} \end{bmatrix}.$$

Phương pháp tính điểm của Fisher (4,20) hiện có thể được sử dụng để giải (4,19), với

$$S_{\tau}(\alpha,\lambda) = \begin{bmatrix} \ln\lambda - \psi(\alpha) + n^{-1}\sum_{i=1}^{n} \ln x_i \\ \frac{\alpha}{\lambda} - n^{-1}\sum_{i=1}^{n} x_i \end{bmatrix}$$

và $F(\alpha,\lambda) = H(\alpha,\lambda)$.

3 Thuật toán tối đa hóa kỳ vọng (EM) (Expectation-Maximization (EM) Algorithm).

Thuật toán Kỳ vọng – Tối đa hóa (EM) là một thuật toán chung để tối đa hóa các hàm hợp lý phức tạp (log-), thông qua việc giới thiệu các biến phụ trợ.

Lưu ý:
Để đơn giản hóa ký hiệu trong phần này, chúng tôi sử dụng hệ thống ký hiệu Bayes, trong đó cùng một ký hiệu được sử dụng cho các mật độ xác suất (có điều kiện) khác nhau.

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmax}}\, g(\tau\,|\,\boldsymbol{\theta}), \tag{4.22}$$

Như trong phần trước, các quan sát độc lập đã cho $\boldsymbol{\tau} = \{\boldsymbol{x_1}, ..., \boldsymbol{x_n}\}$ từ một số hàm mật độ xác suất f chưa biết, mục tiêu là tìm giá trị gần đúng nhất cho $\boldsymbol{f}$ trong một lớp hàm $\boldsymbol{\mathcal{G}} = \{\boldsymbol{g(\cdot|\theta), \theta \in \Theta}\}$ bằng cách giải bài toán hợp lý xảy ra tối đa:

trong đó $\boldsymbol{g(\tau|\theta) := g(x_1|\theta)...g(x_n|\theta)}$. Yếu tố quan trọng của thuật toán EM là sự gia tăng dữ liệu $\boldsymbol{\tau}$ với một vectơ phù hợp của *các biến tiềm ẩn, z*, sao cho

$$\boldsymbol{g(\tau|\theta) = \int g(\tau, z|\theta)dz.}$$

Hàm $\boldsymbol{\theta \mapsto g(\tau, z|\theta)}$ thường được gọi là hàm hợp lý dữ liệu đầy đủ. Việc lựa chọn các biến tiềm ẩn được hướng dẫn bởi mong muốn làm cho việc tối đa hóa $\boldsymbol{g(\tau, z|\theta)}$ dễ dàng hơn nhiều so với $\boldsymbol{g(\tau, \theta)}$.

Giả sử $\boldsymbol{p}$ biểu thị mật độ tùy ý của các biến tiềm ẩn $\boldsymbol{z}$. Sau đó, chúng ta có thể viết:

$$\begin{aligned}
\ln g(\tau\,|\,\boldsymbol{\theta}) &= \int p(z) \ln g(\tau\,|\,\boldsymbol{\theta})\, \mathrm{d}z \\
&= \int p(z) \ln\left(\frac{g(\tau, z\,|\,\boldsymbol{\theta})/p(z)}{g(z\,|\,\tau, \boldsymbol{\theta})/p(z)}\right) \mathrm{d}z \\
&= \int p(z) \ln\left(\frac{g(\tau, z\,|\,\boldsymbol{\theta})}{p(z)}\right) \mathrm{d}z - \int p(z) \ln\left(\frac{g(z\,|\,\tau, \boldsymbol{\theta})}{p(z)}\right) \mathrm{d}z \\
&= \int p(z) \ln\left(\frac{g(\tau, z\,|\,\boldsymbol{\theta})}{p(z)}\right) \mathrm{d}z + \mathcal{D}(p, g(\cdot\,|\,\tau, \boldsymbol{\theta})),
\end{aligned} \tag{4.23}$$

trong đó $\boldsymbol{\mathcal{D}(p, g(\cdot|\tau, \theta))}$ là phân kỳ Kullback-Leibler từ mật độ $\boldsymbol{p}$ đến $\boldsymbol{g(\cdot|\tau, \theta)}$. Vì $\boldsymbol{\mathcal{D} \geqslant 0}$, nó theo sau rằng

$$\boldsymbol{\ln g(\tau|\theta) \geqslant \int p(z) \ln(\frac{g(\tau, z|\theta)}{p(z)})dz =: \mathcal{L}(p, \theta)}$$

cho tất cả $\boldsymbol{\theta}$ và bất kỳ mật độ $\boldsymbol{p}$ nào của các biến tiềm ẩn. Nói cách khác, $\boldsymbol{\mathcal{L}(p, \theta)}$ là giới hạn dưới của log hợp lý liên quan đến khả năng dữ liệu đầy đủ. Sau đó, thuật toán EM nhằm mục đích tăng giới hạn dưới này nhiều nhất có thể bằng cách bắt đầu với phỏng đoán ban đầu $\boldsymbol{\theta^{(0)}}$ và sau đó, đối với $\boldsymbol{t = 1, 2, ...,}$ giải quyết theo hai bước sau:

1. $p^{(t)} = \operatorname{argmax}_p \mathcal{L}(p, \boldsymbol{\theta}^{(t-1)})$,

2. $\boldsymbol{\theta}^{(t)} = \operatorname{argmax}_{\boldsymbol{\theta}\in\Theta} \mathcal{L}(p^{(t)}, \boldsymbol{\theta})$.

$$p^{(t)} = \operatorname*{argmin}_p \mathcal{D}(p, g(\cdot \mid \tau, \boldsymbol{\theta}^{(t-1)})) = g(\cdot \mid \tau, \boldsymbol{\theta}^{(t-1)}).$$

Vấn đề tối ưu hóa đầu tiên có thể được giải quyết một cách rõ ràng. Cụ thể, bởi (4.23), chúng tôi có

Tức là, mật độ tối ưu là mật độ có điều kiện của các biến tiềm ẩn cho dữ liệu $\boldsymbol{\tau}$ và tham số $\boldsymbol{\theta^{(t-1)}}$. Bài toán tối ưu hóa thứ hai có thể được đơn giản hóa bằng cách viết $\boldsymbol{\mathcal{L}(p^{(t)}, \theta) = \mathcal{Q}^{(t)}(\theta) - \mathbb{E}_{p^{(t)}} \ln p^{(t)}(Z)}$, trong đó

$$\boldsymbol{\mathcal{Q}^{(t)}(\theta) := \mathbb{E}_{p^{(t)}} \ln g(\tau, Z|\theta)}$$

là dữ liệu đầy đủ của log hợp lý dự kiến theo $\boldsymbol{Z \sim p^{(t)}}$. Do đó, việc tối đa hóa $\boldsymbol{\mathcal{L}(p^{(t)}, \theta)}$ đối với $\boldsymbol{\theta}$ tương đương với việc tìm

$$\boldsymbol{\theta^{(t)} = arg \max_{\theta\in\Theta} Q^{(t)}(\theta)}$$

Điều này dẫn đến thuật toán EM chung sau đây.

Thuật toán 4.3.1:Thuật toán EM chung
input: Dữ liệu $\boldsymbol{\tau}$, phỏng đoán ban đầu $\boldsymbol{\theta^{(0)}}$.
output: Tính gần đúng của ước lượng hợp lý xảy ra tối đa.

input: Data τ, initial guess $\boldsymbol{\theta}^{(0)}$.
output: Approximation of the maximum likelihood estimate.

$t \leftarrow 1$
while a stopping criterion is not met **do**
  Expectation Step: Find $p^{(t)}(z) := g(z \mid \tau, \boldsymbol{\theta}^{(t-1)})$ and compute the expectation

$$Q^{(t)}(\boldsymbol{\theta}) := \mathbb{E}_{p^{(t)}} \ln g(\tau, \boldsymbol{Z} \mid \boldsymbol{\theta}). \tag{4.24}$$

  Maximization Step: Let $\boldsymbol{\theta}^{(t)} \leftarrow \operatorname{argmax}_{\boldsymbol{\theta} \in \Theta} Q^{(t)}(\boldsymbol{\theta})$.
  $t \leftarrow t + 1$
return $\boldsymbol{\theta}^{(t)}$

Điều kiện dừng khả thi là khi

$$\left| \frac{\ln g(\tau \mid \boldsymbol{\theta}^{(t)}) - \ln g(\tau \mid \boldsymbol{\theta}^{(t-1)})}{\ln g(\tau \mid \boldsymbol{\theta}^{(t)})} \right| \leqslant \varepsilon$$

đối với một số dung sai nhỏ $\boldsymbol{\epsilon > 0}$.

Nhận xét 4.1 (Các thuộc tính của thuật toán EM) Mục (4.23) có thể được sử dụng để cho thấy rằng khả năng $\boldsymbol{g(\tau|\theta^{(t)})}$ không giảm với mỗi lần lặp lại thuật toán. Tính chất này là một trong những điểm mạnh của thuật toán. Ví dụ, nó có thể được sử dụng để gỡ lỗi máy tính triển khai thuật toán EM: nếu khả năng được quan sát thấy giảm ở bất kỳ lần lặp nào, thì người ta đã phát hiện ra lỗi trong chương trình.

Sự hội tụ của dãy $\boldsymbol{\{\theta^{(t)}\}}$ đến cực đại toàn cục (nếu nó tồn tại) phụ thuộc nhiều vào giá trị ban đầu $\boldsymbol{\theta^{(0)}}$ và trong nhiều trường hợp, lựa chọn thích hợp của $\boldsymbol{\theta^{(0)}}$ có thể không rõ ràng. Thông thường, các học viên chạy thuật toán từ các điểm bắt đầu ngẫu nhiên khác nhau trên $\boldsymbol{\Theta}$, để xác định theo kinh nghiệm rằng đạt được mức tối ưu phù hợp.

VD 4.2 (Dữ liệu được kiểm duyệt): Giả sử vòng đời (tính bằng năm) của một loại máy nhất định được mô hình hóa thông qua phân phối $\boldsymbol{\mathcal{N}(\mu, \sigma^2)}$. Để ước tính $\boldsymbol{\mu}$ và $\boldsymbol{\sigma^2}$, tuổi thọ của n máy (độc lập) được ghi lại lên đến c năm. Ký hiệu các vòng đời được kiểm duyệt này bằng $\boldsymbol{x_1, ..., x_n}$. Do đó, $\boldsymbol{\{x_i\}}$ là các biến ngẫu nhiên iid $\boldsymbol{\{X_i\}}$, được phân phối dưới dạng $\boldsymbol{min\{Y, c\}}$, trong đó $\boldsymbol{Y \sim \mathcal{N}(\mu, \sigma^2)}$.

Theo định luật luật xác suất toàn phần (xem (C.9)), hàm mật độ xác suất biên của mỗi $\boldsymbol{X}$ có thể được viết như sau:

$$g(x|\mu,\sigma^2) = \underbrace{\Phi((c-\mu)/\sigma)}_{\mathbb{P}[Y<c]} \frac{\varphi_{\sigma^2}(x-\mu)}{\Phi((c-\mu)/\sigma)} \mathbb{1}\{x<c\} + \underbrace{\overline{\Phi}((c-\mu)/\sigma)}_{\mathbb{P}[Y\geqslant c]} \mathbb{1}\{x=c\},$$

trong đó $\boldsymbol{\varphi_{\sigma^2}(\cdot)}$ là hàm mật độ xác suất của phân phối $\boldsymbol{N(0,\sigma^2)}$, $\boldsymbol{\Phi}$ là hàm phân phối chuẩn tích luỹ của phân phối chuẩn và $\boldsymbol{\overline{\Phi} := 1 - \Phi}$. Theo đó khả năng xảy ra của dữ liệu $\boldsymbol{\tau = \{x_1, ..., x_n\}}$ dưới dạng một hàm của tham số $\boldsymbol{\theta := [\mu, \sigma^2]^T}$ là:

$$g(\tau|\theta) = \prod_{i:x_i<c} \frac{\exp\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)}{\sqrt{2\pi\sigma^2}} \times \prod_{i:x_i=c} \overline{\Phi}((c-\mu)/\sigma).$$

Gọi $\boldsymbol{n_c}$ là tổng số $\boldsymbol{x_i}$ sao cho $\boldsymbol{x_i = c}$. Sử dụng $\boldsymbol{n_c}$ biến tiềm ẩn $\boldsymbol{z = [z_1, ..., z_{n_c}]^T}$, chúng ta có thể viết hàm mật độ xác suất chung:

$$\boldsymbol{g(\tau, z|\theta) = \frac{1}{(2\pi\sigma^2)^{n/2}} exp(-\frac{\sum_{i:x_i<c}(x_i-\mu)^2)}{2\sigma^2} - \frac{\sum_{i=1}^{n_c}(z_i-\mu)^2}{2\sigma^2}) \mathbb{1}\{\min_i z_i \geqslant c\}}$$

sao cho $\boldsymbol{\int g(\tau, z|\theta)dz = g(\tau|\theta)}$. Do đó, chúng tôi có thể áp dụng thuật toán EM để tối đa hóa khả năng xảy ra, như sau.

Đối với bước "Kỳ vọng", chúng tôi có một $\boldsymbol{\theta}$ cố định:

$$\boldsymbol{g(z|\tau,\theta) = \prod_{i=1}^{n_c} g(z_i|\tau,\theta),}$$

trong đó, $\boldsymbol{g(z|\tau,\theta) = \mathbb{1}\{z \geqslant c\}\varphi_{\sigma^2}(z-\mu)/\bar{\Phi}((c-\mu)/\sigma)}$ chỉ đơn giản là hàm mật độ xác suất của phân phối $\boldsymbol{N(\mu,\sigma^2)}$, được cắt bớt thành $\boldsymbol{[c,\infty)}$.

Đối với bước "Tối đa hóa", chúng tôi tính toán kỳ vọng của khả năng xảy ra hoàn toàn liên quan đến một $\boldsymbol{g(z|\tau,\theta)}$ cố định và sử dụng thực tế rằng $\boldsymbol{Z_1, ..., Z_{n_c}}$ là iid:

$$\boldsymbol{\mathbb{E}\ln g(\tau, Z|\theta) = -\frac{\sum_{i:x_i<c}(x_i-\mu)^2}{2\mu^2} - \frac{n_c\mathbb{E}(Z-\mu)^2}{2\sigma^2} - \frac{n}{2}\ln\sigma^2 - \frac{n}{2}\ln(2\pi),}$$

trong đó Z có phân phối $\boldsymbol{\mathcal{N}(\mu,\sigma^2)}$, được cắt ngắn thành $\boldsymbol{[c,\infty)}$. Để tối đa hóa biểu thức cuối cùng đối với $\boldsymbol{\mu}$, chúng ta đặt đạo hàm đối với $\boldsymbol{\mu}$ thành 0 và thu được:

$$\boldsymbol{\mu = \frac{n_c\mathbb{E}Z + \sum_{i:x_i<c}}{n}.}$$

Tương tự, đặt đạo hàm đối với $\boldsymbol{\sigma^2}$ thành 0 sẽ thành:

$$\boldsymbol{\sigma^2 = \frac{n_c \mathbb{E}(Z-\mu)^2 + \sum_{i:x_i<c}(x_i-\mu)^2}{n}.}$$

Tóm lại, EM lặp lại cho $t = 1, 2, \ldots$ được thể hiện như sau.

Bước E: Với ước lượng hiện tại $\boldsymbol{\theta_t := [\mu_t, \sigma_t^2]^T}$, tính các kỳ vọng $\boldsymbol{v_t :=} \mathbb{E}\boldsymbol{Z}$ và $\boldsymbol{\zeta_t^2 := \mathbb{E}(Z-\mu_t)^2}$, trong đó $\boldsymbol{Z \sim \mathcal{N}(\mu, \sigma_t^2)}$, với điều kiện $\boldsymbol{Z \geqslant c}$; đó là,

$$\boldsymbol{v_t := \mu_t + \sigma^2 \frac{\varphi_{\sigma^2}(c-\mu_t)}{\bar{\Phi}((c-\mu_t)/\sigma_t)}}$$

$$\boldsymbol{\zeta_t^2 := \sigma_t^2 (1 + (c-\mu_t) \frac{\varphi_{\sigma_t^2}(c-\mu_t)}{\bar{\Phi}((c-\mu_t)/\sigma_t)}).}$$

Bước M: Cập nhật ước tính thành $\boldsymbol{\theta_{t+1} := [\mu_{t+1}, \sigma_{t+1}^2]^T}$ thông qua công thức:

$$\boldsymbol{\mu_{t+1} = \frac{n_c v_t + \sum_{i:x_i<c} x_i}{n}}$$

$$\boldsymbol{\sigma_{t+1}^2 = \frac{n_c \zeta_t^2 + \sum_{i:x_i<c}(x_i - \mu_{t+1})^2}{n}}$$

4.4 - 4.7

Nguyễn Ngô Trung Hậu

Ngày 26 tháng 12 năm 2021

1 Empirical Distribution and Density Estimation (Hàm phân phối thực nghiệm và ước lượng mật độ)

Trong mục 1.5.2.3 chúng ta thấy hàm phân phối tính lũy thực nghiệm $\widehat{F_n}$, thu được từ một tập huấn luyện độc lập và phân phối đồng nhất $\tau = \{x_1, ..., x_n\}$ từ một phân phối không xác định trên R, đưa ra ước tính về hàm phân phối tích lũy F chưa biết của phân phối lấy mẫu này. Hàm $\widehat{F_n}$ là một hàm phân phối tích lũy luôn đúng vì nó liên tục phải, tăng dần và nằm trong khoảng từ 0 đến 1. Phân phối xác suất rời rạc tương ứng được gọi là phân phối thực nghiệm của dữ liệu. Một biến ngẫu nhiên X được phân phối theo phân phối thực nghiệm này nhận các giá trịx_1,. . . , x_n với xác suất bằng nhau $\frac{1}{n}$.Khái niệm phân phối thực nghiệm tổng quát một cách tự nhiên theo các kích thước lớn hơn: một vectơ ngẫu nhiên X được phân phối theo phân phối thực nghiệm của x_1 ,. . . , x_n có hàm mật độ xác suất rời rạc P $[X = x_i] = \frac{1}{n}$ với i = 1 ,. . . , n.

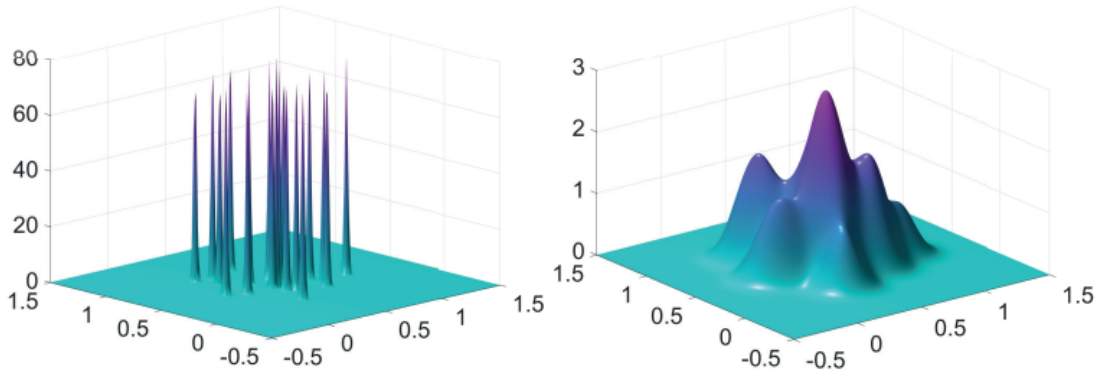
Theo một cách nào đó, phân phối thực nghiệm là câu trả lời tự nhiên cho câu hỏi học tập không giám sát: phân phối xác suất cơ bản của dữ liệu là gì?Tuy nhiên, theo định nghĩa, phân phối thực nghiệm là một phân phối rời rạc, trong khi phân phối lấy mẫu thực sự có thể liên tục.Đối với dữ liệu liên tục cũng hợp lý khi xem xét ước tính của hàm mật độ xác suất dữ liệu. Một cách tiếp cận phổ biến là ước tính mật độ thông qua ước tính mật độ hạt nhân (KDE)

Định nghĩa : Gaussian KDE

Cho $x_1, ..., x_n \in R^d$ là kết quả của một mẫu độc lập và phân phối đồng nhất từ một hàm mật độ xác suất liên tục f. Ước tính mật độ hạt nhân Gaussian của f là hỗn hợp của các hàm một độ xác suất chuẩn, có dạng:

$$g_{\tau_n}(x|\sigma) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{(2\pi)^{d/2}\sigma^d}e^{-\frac{\|x-x_i\|^2}{2\sigma^2}}\right), \text{ với } x \in R^d$$

trong đó $\sigma > 0$ được gọi là độ lệch chuẩn
Chúng ta thấy rằng g_{τ_n} trong định nghĩa Gaussian KDE là giá trị trung bình của một tập hợp hàm mật độ xác suất chuẩn n, trong đó mỗi phân phối chuẩn được căn giữa tại điểm dữ liệu x_i và có ma trận hiệp phương sai $\sigma^2 I_d$. Một câu hỏi chính là làm thế nào để chọn độ lệch chuẩn σ sao cho gần đúng nhất với hàm mật độ xác suất f chưa biết. Việc chọn σ rất nhỏ sẽ dẫn đến một ước tính "nhạy cảm", trong khi một σ lớn sẽ tạo ra một ước tính quá mượt mà có thể không xác định được các đỉnh quan trọng có trong hàm mật độ xác suất không xác định.Hình sau minh họa hiện tượng này. Trong trường hợp này, dữ liệu bao gồm 20 điểm được vẽ thống nhất từ hình vuông đơn vị. Do đó, hàm mật độ xác suất thực sự là 1 trên $[0, 1]^2$ và 0 ở những nơi khác.



Hình 1: Hai KDE Gaussian hai chiều, với σ= 0,01 (trái) và σ = 0,1 (phải)

Chúng ta viết lại định nghĩa Gaussian KDE dưới dạng

$$g_{\tau_n}(x|\sigma) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sigma^d}\phi\left(\frac{x-x_i}{\sigma}\right), \quad (1)$$

trong đó

$$\phi(z) = \frac{1}{(2\pi)^{d/2}} e^{-\frac{\|z\|^2}{2}}, z \in R^d \ (2)$$

là hàm mật độ xác suất của phân phối chuẩn chuẩn d-chiều. Bằng cách chọn một mật độ xác suất khác ϕ trong (1), thỏa mãn $\phi(x) = \phi(-x)$ với mọi x, chúng ta có thể thu được nhiều ước tính mật độ hạt nhân. Ví dụ: một hàm mật độ xác suất đơn giản ϕ là hàm mật độ xác suất thống nhất trên $[-1, 1]^d$

$$\phi(z) = \begin{cases} 2^{-}d, & \text{if } z \in [-1,1]^d. \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Hình 2 cho thấy đồ thị của KDE tương ứng, sử dụng dữ liệu tương tự như trong Hình 1 và với độ lệch chuẩn $\sigma = 0{,}1$. Chúng tôi quan sát hành vi tương tự về mặt chất lượng đối với các KDE Gaussian và đồng nhất. Theo quy luật, sự lựa chọn của hàm ϕ ít quan trọng hơn sự lựa chọn của độ lệch chuẩn trong việc xác định chất lượng của ước lượng.

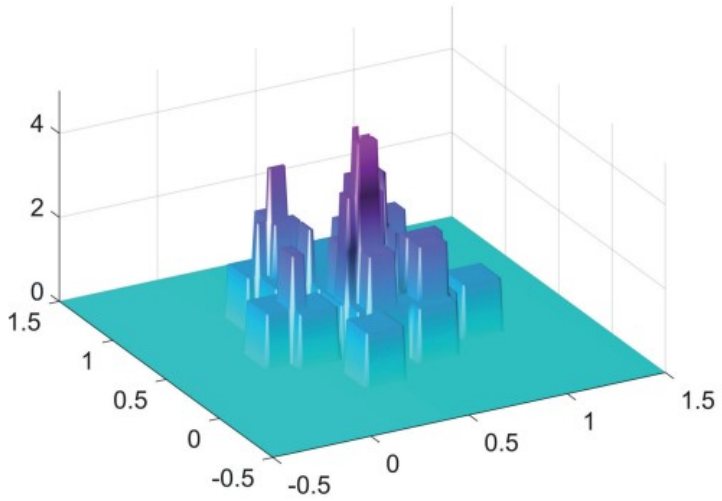
Vấn đề quan trọng của việc lựa chọn băng thông đã được nghiên cứu rộng rãi đối với dữ liệu một chiều. Để giải thích các ý tưởng, chúng tôi sử dụng thiết lập thông thường của chúng tôi và đặt $\tau = \{x_1, ..., x_n\}$ là dữ liệu quan sát (một chiều) từ hàm mật độ xác suất f chưa xác định. Đầu tiên, chúng tôi định nghĩa hàm mất mát là

$$\text{Loss}(f(x), g(x)) = \frac{(f(x) - g(x))^2}{f(x)} \ (3)$$

Do đó, rủi ro cần giảm thiểu là $\ell(g) := \mathbb{E}_f \text{Loss}(f(x), g(x)) = \int (f(x) - g(x))^2 dx$

Chúng tôi bỏ qua việc chọn một lớp hàm xấp xỉ bằng cách chọn người học được chỉ định bởi định nghĩa Gaussian KDE cho một σ cố định. Mục tiêu bây giờ là tìm một σ giảm thiểu rủi ro tổng quát hóa $\ell(g_\tau(\cdot|\sigma))$ hoặc rủi ro tổng quát hóa dự kiến $\mathbb{E}\ell(g_\tau(\cdot|\sigma))$. Rủi ro tổng quát hóa là trong trường hợp này

$$\int (f(x) - g_\tau(x|\sigma))^2 dx = \int f^2(x) dx - 2\int f(x) g_\tau(x|\sigma) dx + \int {g_\tau}^2(x|\sigma) dx$$



Hình 2: một KDE đồng nhất hai chiều, với σ= 0,1

Việc giảm thiểu biểu thức này đối với σ tương đương với việc giảm thiểu hai số hạng cuối cùng, có thể được viết là

$$-2\mathbb{E}_f\left(g_\tau\left(X|\sigma\right)\right) + \int\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sigma}\phi\left(\frac{x-x_i}{\sigma}\right)\right]^2 dx$$

Biểu thức này có thể được ước lượng bằng cách sử dụng một mẫu thử nghiệm $\{x'_1, ..., x'_{n'}\}$ từ f, đưa ra bài toán tối thiểu hóa sau:

$$min_\sigma - \frac{2}{n'}\sum_{i=1}^{n'} g_\tau\left(x'_i|\sigma\right) + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\int\frac{1}{\sigma^2}\phi\left(\frac{x-x_i}{\sigma}\right)\phi\left(\frac{x-x_j}{\sigma}\right)dx,$$

Trong đó $\int\frac{1}{\sigma^2}\phi\left(\frac{x-x_i}{\sigma}\right)\phi\left(\frac{x-x_j}{\sigma}\right)dx = \frac{1}{\sqrt{2}\sigma}\phi\left(\frac{x_i-x_j}{\sqrt{2}\sigma}\right)$ trong trường hợp của hạt nhân Gaussian (2) với d = 1. Để ước lượng σ theo cách này rõ ràng cần phải có một mẫu thử nghiệm, hoặc ít nhất là một ứng dụng xác nhận chéo. Một cách tiếp cận khác là giảm thiểu rủi ro tổng quát hóa dự kiến, (nghĩa là, tính trung bình trên tất cả các tập huấn luyện):

$$\mathbb{E}\int\left(f\left(x\right) - g_\tau\left(x|\sigma\right)\right)^2 dx$$

Đây được gọi là lỗi bình phương tích hợp trung bình (MISE). Nó có thể được phân tách thành thành phần thiên vị bình phương tích hợp và thành phần phương sai tích hợp:

$$\int \left(f\left(x\right) - \mathbb{E}g_{\tau}\left(x|\sigma\right)\right)^{2} dx + \int Var\left(g_{\tau}\left(x|\sigma\right)\right) dx$$

Một phân tích điển hình hiện đang được tiến hành bằng cách điều tra MISE hoạt động như thế nào đối với n lớn, dưới nhiều giả thiết khác nhau về f. Đối với $\sigma \to 0$ và $n\sigma \to \infty$, xấp xỉ tiệm cận với MISE của công cụ ước lượng mật độ hạt nhân Gaussian (đối với d = 1) được cho bởi

$$\frac{1}{4}\|f''\|^{2} + \frac{1}{2n\sqrt{\pi\sigma^{2}}}$$

trong đó $\|f''\|^{2} := \int \left(f''\left(x\right)\right)^{2} dx$ The asymptotically optimal value of σ is the minimizer

$$\sigma* := \left(\frac{1}{2n\sqrt{\pi}\|f''\|^{2}}\right)^{\frac{1}{5}}$$

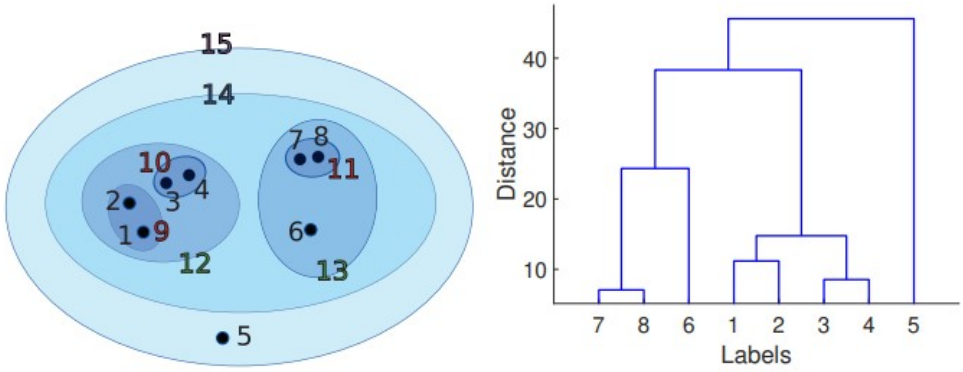
Để tính $\sigma*$ tối ưu trong,, người ta cần ước lượng hàm $\|f''\|^{2}$. Quy tắc ngón tay cái của Gaussian là giả định rằng f là mật độ của phân phối $\mathbb{N}(\overline{x}, s^{2})$ trong đó x và s^{2} lần lượt là trung bình mẫu và phương sai của dữ liệu. Trong trường hợp này$\|f''\|^{2} = s^{-5}\pi^{\frac{-1}{2}}\frac{3}{8}$ và quy tắc ngón tay cái của Gaussian trở thành:

$$\sigma_{rot} = \left(\frac{4s^{5}}{3n}\right)^{1/5} \approx 1.06sn^{-1/5}$$

We recommend, however, the fast and reliable theta KDE of [14], which chooses the bandwidth in an optimal way via a fixed-point procedure. Figures 4.1 and 4.2 illustrate a common problem with traditional KDEs: for distributions on a bounded domain, such as the uniform distribution on [0, 1]2 , the KDE assigns positive probability mass outside this domain. An additional advantage of the theta KDE is that it largely avoids this boundary effect. We illustrate the theta KDE with the following example.

2 Hierarchical Clustering (Phân cụm phân cấp)

Thuật toán phân cụm K-means cho thấy cần phải cấu hình trước số lượng cụm cần phân chia. Ngược lại, phương pháp phân cụm phân cấp (Hierachical Clustering) không yêu cầu khai báo trước số lượng cụm. Thay vào đó, thuật toán chỉ yêu cầu xác định trước thước đo về sự khác biệt giữa các cụm (không giao nhau), dựa trên sự khác biệt từng cặp giữa các quan sát trong hai cụm. Theo phương pháp này, chúng tạo ra những biểu diễn phân cấp trong đó các cụm ở mỗi cấp của hệ thống phân cấp được tạo bằng cách hợp nhất các cụm ở cấp độ thấp hơn bên dưới. Ở cấp thấp nhất, mỗi cụm chứa một quan sát. Ở cấp cao nhất, chỉ có một cụm chứa tất cả dữ liệu



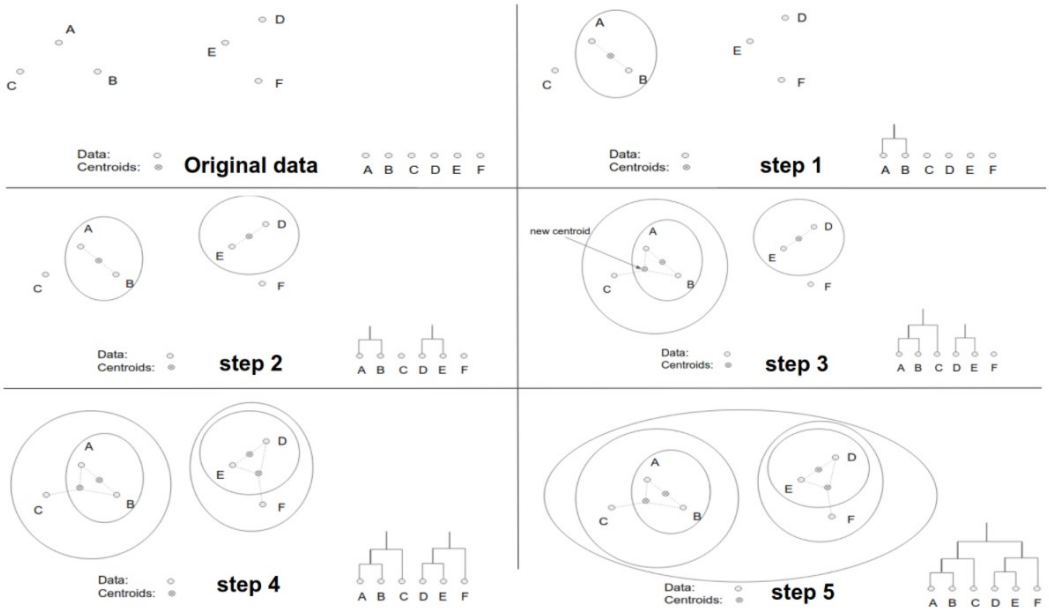
Hình 3: Bên trái: một hệ thống phân cấp cụm gồm 15 cụm. Phải: biểu đồ dendrogram tương ứng

Thuật toán phân cụm phân cấp được xây dựng trên bộ dữ liệu có kích thước N thì sẽ trải qua tổng cộng N bước phân chia. Có hai chiến lược phân chia chính phụ thuộc vào chiều di chuyển trên biểu đồ dendrogram mà chúng ta sẽ tìm hiểu bên dưới:Chiến lược hợp nhất và chiến lược phân chia

2.1 Chiến lược hợp nhất(agglomerative)

Chiến lược này sẽ đi theo chiều bottum-up (từ dưới lên trên). Quá trình phân cụm bắt đầu ở dưới cùng tại các node lá (còn gọi là leaf node hoặc termial

node). Ban dầu mỗi quan sát sẽ được xem là một cụm tách biệt được thể hiện bởi một node lá. Ở mỗi level chúng ta sẽ tìm cách hợp một cặp cụm thành một cụm duy nhất nhằm tạo ra một cụm mới ở level cao hơn tiếp theo. Cụm mới này tương ứng với các node quyết định (non-leaf node). Như vậy sau khi hợp cụm thì số lượng cụm ít hơn. Một cặp được chọn để hợp nhất sẽ là những cụm trung gian không giao nhau.
Chiến lược hợp nhất sẽ bắt đầu biểu diễn mỗi quan sát là một cụm đơn lẻ. Giả định chúng ta có N quan sát, thuật toán cần thực hiện N-1 bước để hợp nhất hai nhóm có khoảng cách gần nhất lại với nhau và đồng thời giảm số lượng cụm trước khi chúng đạt được tới node gốc gồm toàn bộ các quan sát.Ta có ví dụ minh họa về chiến lược hợp nhất



Hình 4: Hình minh hoạ các bước được thực hiện trên thuật toán phân cụm phân cấp sử dụng chiến lược hợp nhất đối với 6 điểm dữ liệu (A,B,C,D,E,F) . Chấm tròn thể hiện cho các điểm dữ liệu, chấm tròn có dấu x ở giữa là tâm của các cụm. Các đường elipse bao ngoài thể hiện cho các điểm được phân về cùng một cụm. Ở bên phải dưới cùng của mỗi hình là đồ thị dendrogram thể hiện sự gộp nhóm.

Bộ dữ liệu ở hình 2 bao gồm 6 điểm nên sẽ trải qua 5 bước dữ liệu để nhóm dữ liệu. Thứ tự nhóm sẽ như sau:
Step 1: Dựa trên khoảng cách gần nhất giữa các điểm chúng ta sẽ nhóm 2 điểm A và B thành 1 cụm. Khi đó điểm đại diện cho một cụm (A,B) sẽ là trung bình cộng giữa hai điểm A và B, được thể hiện bằng dấu $\oplus$ giữa A và B trên hình.
Step 2: Lựa chọn ngẫu nhiên một điểm chưa được gộp cụm, chẳng hạn điểm D. Đo khoảng cách tới các điểm còn lại và với tâm cụm (A,B) ta sẽ thu được khoảng cách d(D,E) là nhỏ nhất. Như vậy ta sẽ thu được một cụm(D,E).
Step 3: Xuất phát từ điểm C , ta đo khoảng cách tới các tâm cụm (A,B) và (D,E) và tới điểm F. Khoảng cách gần nhất là d(C,(A,B)) nên ta nhóm C vào cụm (A,B) để thu được cụm mới
Step 4: Xuất phát từ F ta đo khoảng cách tới các tâm cụm (A,B,C) và (D,E). Điểm F gần cụm (D,E) hơn nên sẽ được gộp vào thành cụm (D,E,F). .
Step 5: Gộp cả 2 cụm (A,B,C) và Chung qui lại xuất phát từ node lá, thuật toán gộp dần thành các cụm theo chiều từ dưới lên trên. Sau đó sẽ thực hiện truy hồi việc gộp cụm (cụm ở đây có thể gồm một điểm hoặc nhiều điểm). Khoảng cách giữa hai cụm được đo lường thông qua một thước đo sẽ được làm rõ hơn ở bên dưới, trong ví dụ này chính là khoảng cách trong không gian euclidean giữa tâm của mỗi cụm. Trong đó tâm cụm được xác định bằng trung bình cộng của các quan sát bên trong cụm.(D,E,F) ta thu được cụm cuối cùng là node gốc bao trùm toàn bộ dữ liệu.
Chung qui lại xuất phát từ node lá, thuật toán gộp dần thành các cụm theo chiều từ dưới lên trên. Sau đó sẽ thực hiện truy hồi việc gộp cụm (cụm ở đây có thể gồm một điểm hoặc nhiều điểm). Khoảng cách giữa hai cụm được đo lường thông qua một thước đo sẽ được làm rõ hơn ở bên dưới, trong ví dụ này chính là khoảng cách trong không gian euclidean giữa tâm của mỗi cụm. Trong đó tâm cụm được xác định bằng trung bình cộng của các quan sát bên trong cụm.

2.2 Khoảng cách giữa hai cụm

Trước hết ta cùng ôn lại khoảng cách eulidean, chính là độ dài đoạn thẳng nối trực hiếp hai điểm trong không gian euclidean

$$\text{dist}(x, x') = \|x - x'\| = \sqrt{\sum_{i=1}^{n} (x_i - {x'}_i)^2}$$

Giả định tại một level cụ thể trong biểu đồ dendrogram chúng ta có hai cụm trung gian không trùng nhau là $S_1 = \{x_i, i = I\}$ và $S_2 = \{x_j, j = J\}$. Khoảng cách giữa hai cụm chính là sự khác biệt giữa chúng. Có những phương pháp giúp xác định khoảng cách giữa hai cụm như sau:
Single linkage : Phương pháp này đo lường sự khác biệt giữa hai cụm bằng cách lấy ra cặp điểm gần nhất giữa hai cụm. Độ đo sự khác biệt được tính theo công thức:

$$d_{min}(I, J) := \min_{i \in I, j \in J} \text{dist}(x_i, x_j)$$

Complete linkage : Phương pháp này đo lường sự khác biệt giữa hai cụm bằng cách lấy ra hai cặp điểm xa nhau nhất giữa hai cụm:

$$d_{max}(I, J) := \max_{i \in I, j \in J} \text{dist}(x_i, x_j)$$

Group average. : Khoảng cách trung bình giữa các cụm. Lưu ý rằng điều này phụ thuộc vào kích thước cụm:

$$d_{avg}(I, J) := \frac{1}{|I||J|} \sum_{i \in I} \sum_{j \in J} \text{dist}(x_i, x_j)$$

Ward' linkage. : Khoảng cách trung bình giữa các cụm. Lưu ý rằng điều này phụ thuộc vào kích thước cụm:

$$d_{Ward}(I, J) := \sum_{k \in I \cup J} \|x_k - \overline{x}_{I \cup J}\|^2 - \left(\sum_{i \in I} \|x_i - \overline{x}_I\|^2 + \sum_{j \in J} \|x_J - \overline{x}_I\|^2 \right)$$

Cả bốn phương pháp ward linkage, sinlge linkage, complete linkage, group average đều giúp tạo ra một thước đo về sự không tương đồng hay chính là khoảng cách giữa hai cụm. Khi giữa các cụm có sự tách biệt thể hiện qua phân phối dữ liệu và đường biên phân chia rõ rệt thì kết quả trả $d(S_1, S_2)$ về đều thu được lớn và trái lại. Tuy nhiên phương pháp single linkage và complete linkage thường bị ảnh hưởng bởi những điểm dữ liệu outliers. Chẳng hạn hai cụm rất cách xa nhau nhưng do hai điểm outliers của chúng lại rất gần nhau có thể trả về một khoảng cách theo single linkage rất bé. Một tình huống khác, khi hai cụm rất gần nhau nhưng do hai điểm outliers của chúng rất xa nên khoảng cách được đo theo complete linkage lại rất lớn. trong khi đó ward linkage và group average ít bị ảnh hưởng bởi outliers hơn. Tuy nhiên ward linkage lại chỉ có thể hoạt động khi các điểm dữ liệu tồn tại trong không gian

euclidean.
Chú ý: Trong triển khai phần mềm, hàm Ward' linkage thường được thay đổi tỷ lệ bằng cách nhân nó với hệ số 2. Bằng cách này, khoảng cách giữa các cụm một điểm $\{x_i\}$ và $\{x_j\}$ là khoảng cách Euclid bình phương $\|x_i - x_j\|^2$.
Sau khi chọn một khoảng cách trên X và một tiêu chí liên kết, một thuật toán phân cụm tích hợp chung sẽ tiến hành theo cách "greedy" sau đây.
Thuật toán: Greedy Agglomerative Clustering.
Đầu vào: Hàm khoảng cách dist, hàm liên kết d, số lượng cụm K.
Đầu ra: Bộ nhãn cho cây
1. Khởi tạo bộ nhận dạng cụm: I =$\{1, ..., n\}$
2. Khởi tạo các bộ nhãn tương ứng:$L_i = \{i\}, i \in I$.
3. Khởi tạo ma trận khoảng cách D = $[d_{ij}]$ với $d_{ij} = d(\{i\}, \{j\})$.
4. for k = n + 1 to 2n -K do
5. Tìm i và $j > i$ trong I sao cho d_{ij}là nhỏ nhất.
6. Tạo bộ nhãn mới $L_k := L_i \cup L_j$
7. Thêm số nhận dạng mới k vào I và xóa các số nhận dạng cũ i và j khỏi I
8.Cập nhật ma trận khoảng cách D đối với các định danh i, j và k.
9. return $L_i, i = 1, ..., 2n - K$

2.3 Chiến lược phân chia (divisive)

Chiến lược này sẽ thực hiện theo chiều top-down. Tức là phân chia bắt đầu từ node gốc của đồ thị. Node gốc bao gồm toàn bộ các quan sát, tại mỗi level chúng ta phân chia một cách đệ qui các cụm đang tồn tại tại level đó thành hai cụm mới. Phép phân chia được tiến hành sao cho tạo thành hai cụm mới mà sự tách biệt giữa chúng là lớn nhất. Sự tách biệt này sẽ được đo lường thông qua một thước đo khoảng cách mà ta sẽ tìm hiểu kĩ hơn bên dưới.
Đầu tiên thuật toán sẽ chọn ra một điểm từ toàn bộ tập dữ liệu S sao cho điểm này thoả mãn điều kiện trung bình khoảng cách từ điểm đó tới toàn bộ những điểm còn lại là nhỏ nhất. Chúng ta đưa điểm này vào tậpS_1,tập còn lại gồm $N - 1$ điểm là tập S_2 . Tiếp theo ta sẽ thực hiện các lượt phân chia sao cho mỗi một lượt lựa chọn ra một điểm x_i từ tập S_2 đưa sang S_1 .Điểm này cần thoả mãn hai điều kiện:
- Trung bình khoảng cách từ điểm đó tới toàn bộ các điểm còn lại trong S_1 phải là nhỏ nhất. Điều đó có nghĩa là $x_i$là điểm tách biệt nhất so với phần còn lại của S_1

$$x_i = \underset{x_i}{\operatorname{argmax}} \frac{1}{|S_1| - 1} \sum_{j=1, j \neq i}^{|S_1|} d\left(x_i, x_j\right)$$

Khoảng cách tối thiểu từ x_i tới các điểm trong S_2 phải lớn hơn khoảng cách tối thiểu tới các điểm trong S_1 .Điều này nhằm mục đích khiến cho điểm x_i phải gần với cụm S_2 hơn cụm S_1.

$$\mathrm{d}(x_i, S_1) \geq d\left(x_i, S_2\right)$$

Trong đó:

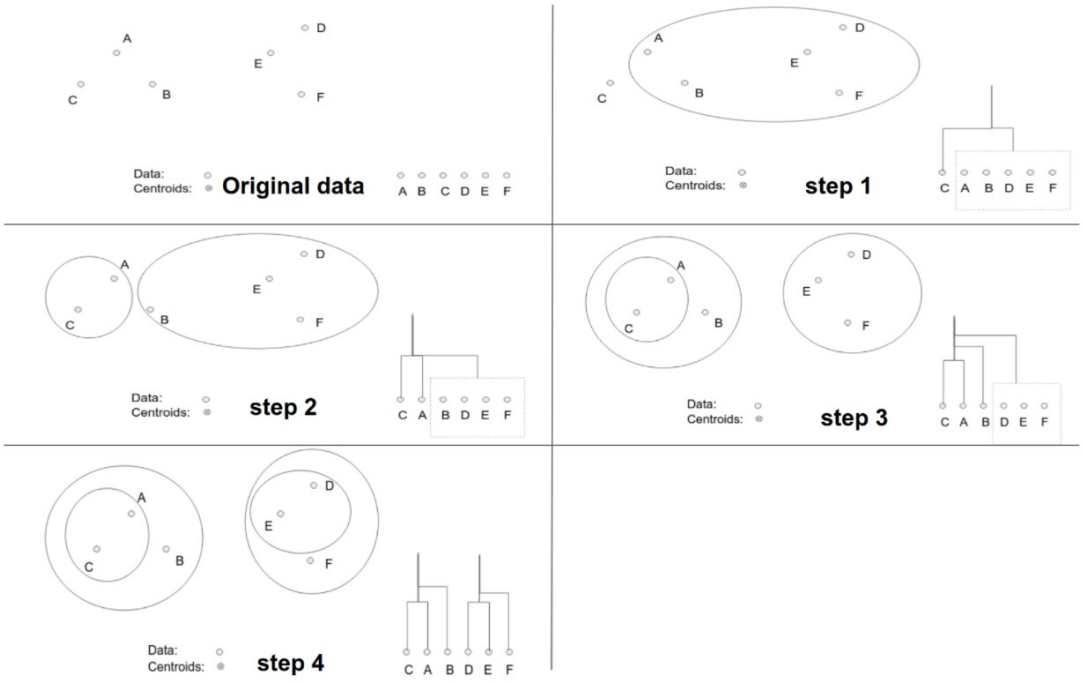
$$(x_i, S_k) = \min_{x_i, x_j \in S_k} d\left(x_i, x_j\right)$$

Qúa trình chuyển cụm sẽ kết thúc khi không còn điểm nào thoả mãn hai điều kiện trên. Khi đó chúng ta lại thực hiện đệ qui lại quá trình trên trên từng tập S_1 và S_2.
Chúng ta cùng xem ví dụ bên dưới để hiểu rõ hơn vấn đề này:

Bước 1: chúng ta sẽ lựa chọn ra điểm là điểm đầu tiên thuộc cụm mới dựa trên khoảng cách so với các điểm còn lại là xa nhất.Khi đó ta thu được tập $S_1 = \{C\}$ và $S_1 = \{A, B, D, E, F\}$.
Bước 2:lựa chọn trong số các điểm thuộc S_2 ra điểm mà có khoảng cách xa nhất so với những điểm còn lại sao cho điểm này gần với C hơn so với các điểm thuộc tập S_2 ,đó chính là điểm A.Di chuyển điểm này sang S_1.
Bước 3: lại tiếp tục thực hiện như vậy và lựa chọn được điểm để đưa sang S_1.
Bước 4: dừng quá trình chuyển cụm cho các điểm thuộc S_2 vì thuật toán đã đạt sự hội tụ về hai cụm. Khi đó ta lại tiếp tục tiến hành đệ qui thuật toán trên từng cụm con.

2.4 Thuật toán Lance-Williams

Ban đầu, ma trận khoảng cách D chứa khoảng cách (liên kết) giữa các cụm một điểm chứa một trong các điểm dữ liệu$x_1, ..., x_n$, và do đó với các số nhận dạng 1,. . . , n.ìm khoảng cách ngắn nhất tương ứng với tra cứu bảng trong D. Khi các cụm gần nhất được tìm thấy, chúng được hợp nhất thành một cụm mới và số nhận dạng mới k (số nguyên dương nhỏ nhất chưa được sử dụng làm định danh) được gán cho cụm này. Các số nhận dạng cũ i và j bị



Hình 5: Hình minh hoạ phương pháp phân chia trong thuật toán phân cụm phân cấp

xóa khỏi tập nhận dạng cụm I. Sau đó, ma trận D được cập nhật bằng cách thêm cột và hàng thứ k chứa khoảng cách giữa k và$m \in I$. Bước cập nhật này có thể khá tốn kém về mặt tính toán. nếu kích thước cụm lớn và khoảng cách liên kết giữa các cụm phụ thuộc vào tất cả các điểm trong cụm. May mắn thay, đối với nhiều hàm liên kết, ma trận D có thể được cập nhật một cách hiệu quả.
Giả sử rằng tại một số giai đoạn trong thuật toán, các cụm I và J, với số nhận dạng i và j, được hợp nhất thành một cụm $K = I \cup J$ với số nhận dạng k. Cho M, với số nhận dạng m, là một cụm đã được gán trước đó. Quy tắc cập nhật về khoảng cách liên kết d_{km} giữa K và M được gọi là cập nhật Lance – Williams nếu nó có thể được viết dưới dạng

$$d_{km} = \alpha d_{im} + \beta d_{jm} + \gamma d_{ij} + \delta |d_{im} - d_{jm}|,$$

trong đó $\alpha, ..., \delta$ chỉ phụ thuộc vào các đặc điểm đơn giản của các cụm liên quan, chẳng hạn như số lượng phần tử trong các cụm.

Linkage	α	β	γ	δ
Single	$1/2$	$1/2$	0	$-1/2$
Complete	$1/2$	$1/2$	0	$1/2$
Group avg.	$\frac{n_i}{n_i + n_j}$	$\frac{n_j}{n_i + n_j}$	0	0
Ward	$\frac{n_i + n_m}{n_i + n_j + n_m}$	$\frac{n_j + n_m}{n_i + n_j + n_m}$	$\frac{-n_m}{n_i + n_j + n_m}$	0

Hình 6: Các hằng số cho quy tắc cập nhật Lance-Williams cho các hàm liên kết khác nhau với n_i, n_j, n_m biểu thị số phần tử trong các cụm tương ứng

4.5

Trần Bửu Ân

Ngày 25 tháng 12 năm 2021

Phân cụm liên quan đến việc nhóm các vectơ đặc trưng không được gắn nhãn thành các cụm, sao cho các mẫu trong một cụm giống với nhau hơn các mẫu thuộc các cụm khác nhau. Thông thường, người ta cho rằng số lượng cụm được biết trước, nhưng nếu không thì không có thông tin trước nào được đưa ra về dữ liệu. Các ứng dụng của phân cụm có thể được tìm thấy trong các lĩnh vực truyền thông, nén và lưu trữ dữ liệu, tìm kiếm cơ sở dữ liệu, đối sánh mẫu và nhận dạng đối tượng.
Một cách tiếp cận phổ biến để phân tích phân cụm là giả định rằng dữ liệu đến từ một tổ hợp các phân phối (thường là Gaussian), và do đó mục tiêu là ước tính các tham số của mô hình hỗn hợp bằng cách tối đa hóa hàm khả năng cho dữ liệu. Tối ưu hóa trực tiếp hàm khả năng trong trường hợp này không phải là một nhiệm vụ đơn giản, do những ràng buộc cần thiết đốivới các tham số (sẽ nói thêm về điều này sau) và tính chất phức tạp của hàm khả năng, nói chung có rất nhiều địa phương cực đại và điểm yên ngựa.
Một phương pháp phổ biến để ước lượng các tham số của mô hình hỗn hợp là thuật toán EM, đã được thảo luận trong một cài đặt tổng quát hơn trong Phần 4.3. Trong phần này, chúng tôi giải thích những điều cơ bản về mô hình hỗn hợp(trong active 128)và giải thích hoạt động của phương pháp EM trong bối cảnh này.
Ngoài ra, chúng ta chỉ ra cách các phương pháp tối ưu hóa trực tiếp có thể được sử dụng để tối đa hóa khả năng xảy ra.

4.5.1 Mô hình hỗn hợp.

Cho $\tau = \{X_1, X_2, ..., X_n\}$ là các vectơ ngẫu nhiên nhận các giá trị trong một số tập $\chi \subseteq R^d$, mỗi X_i được phân phối theo mật độ hỗn hợp

$$g(x|\theta) = w_1\phi_1(x) + ... + w_n\phi_n(x) \tag{4.31}$$

trong đó $\phi_1, ..., \phi_K$ là mật độ xác suất (rời rạc hoặc liên tục) trên χ và trọng số dương $w_1, ..., w_K$ tổng lên đến 1. Bản pdf hỗn hợp này có thể được diễn giải theo cách sau. Gọi Z là một biến ngẫu nhiên rời rạc nhận các giá trị $1, 2, ..., K$ với các xác suất $w_1, ..., w_K$ và cho X là vectơ ngẫu nhiên có pdf có điều kiện, cho trước $Z = z$, là ϕ_z .
Theo quy tắc tích số (C.17), joint pdf của Z và X được đưa ra bởi $\phi_{Z,X}(z, x) = \phi_Z(z)\phi_{Z|X}(x|z) = w_z\phi_z(x)$ và pdf cận biên của X được tìm thấy bằng cách tính tổng joint pdf với các giá trị của z,cho (4.31). Do đó, một vectơ ngẫu nhiên

$X \sim g$ có thể được mô phỏng theo hai bước:
1. Đầu tiên, vẽ Z theo các xác suất

$$P[Z = z] = w_z, z = 1, ..., K.$$

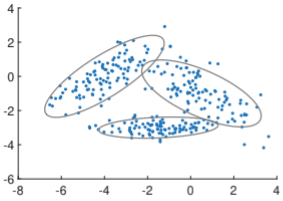
2. Sau đó vẽ X theo pdf ϕ_z
Vì τ chỉ chứa các biến $\{X_i\}$, nên $\{Z_i\}$ được xem như là các biến tiềm ẩn. Chúng ta có thể gọi Z_i là nhãn ẩn của cụm mà X_i thuộc về
Thông thường, mỗi ϕ_k trong (4.31) được giả định là đã biết đến một số vector tham số η_k. Thông thường 1 trong phân tích phân cụm là làm việc với các hỗn hợp Gaussian; nghĩa là, mỗi mật độ ϕ_k là Gaussian với một số vectơ kỳ vọng chưa biết μ_k và ma trận hiệp phương sai Σ_k. Chúng ta tập hợp tất cả các tham số chưa biết, bao gồm cả trọng số w_k, vào một vectơ tham số θ Như thường lệ, $\tau = x_1, ..., x_n$ biểu thị kết quả của τ . Vì các thành phần của τ là iid, joint pdf của chúng được đưa ra bởi

$$g(\tau|\theta) = \prod_{i=1}^{n} g(x_i|\theta) = \prod_{i=1}^{n} \sum_{k=1}^{K} w_k \phi_k(x_i|\mu_k, |\Sigma_k) \tag{4.32}$$

Theo lý luận tương tự như đối với (4.5), chúng ta có thể ước tính θ từ một kết quả τ bằng cách tối ưu hóa hàm log-khả năng

$$l(\theta|\tau) = \sum_{i=1}^{n} \ln g(x_i|\theta) = \sum_{i=1}^{n} \ln(\sum_{k=1}^{K} w_k \phi_k(x_i|\mu_k, \Sigma_k)) \tag{4.33}$$

Tuy nhiên, việc tìm giá trị cực đại của $L(\theta|\tau)$ nói chung là không dễ dàng, vì hàm thường là đa chiều
Ví dụ 4.4 (Phân cụm thông qua các mô hình hỗn hợp) Dữ liệu được mô tả trong Hình 4.4 bao gồm 300 điểm dữ liệu được tạo độc lập từ ba phân phối chuẩn hai biến , có các tham số được đưa ra trong cùng một hình. Đối với mỗi trong số ba giải thưởng này , chính xác 100 điểm đã được tạo ra. Tốt nhất, chúng tôi muốn phân cụm dữ liệu thành ba cụm tương ứng với ba trường hợp.
Để phân cụm dữ liệu thành ba nhóm, một mô hình khả thi cho dữ liệu là giả định rằng các điểm được lấy từ hỗn hợp (chưa biết) của ba phân biệt Gaussian 2 chiều. Đây là một cách tiếp cận hợp lý, mặc dù trên thực tế, dữ liệu không được mô phỏng theo cách này. Đó là hướng dẫn để hiểu sự khác biệt giữa hai mô hình.
Trong mô hình hỗn hợp, mỗi nhãn cụm Z nhận giá trị $1, 2, 3$ với xác suất bằng nhau và do đó, vẽ các nhãn một cách độc lập, tổng số điểm trong mỗi cụm sẽ được $B(300, 1/3)$ (phân phối nhị thức). Tuy nhiên, trong mô phỏng thực tế, số điểm trong mỗi cụm chính xác là 100. Tuy nhiên, mô hình hỗn hợp sẽ là một mô hình chính xác (mặc dù không chính xác) cho những dữ liệu này.
Hình 4.5 hiển thị mật độ hỗn hợp Gaussian "mục tiêu" cho dữ liệu trong Hình 4.4; nghĩa là hỗn hợp có khối lượng bằng nhau và với các thông số chính xác như quy định trong Hình 4.4



cluster	mean vector	covariance matrix
1	$\begin{bmatrix}-4\\0\end{bmatrix}$	$\begin{bmatrix}2 & 1.4\\1.4 & 1.5\end{bmatrix}$
2	$\begin{bmatrix}0.5\\-1\end{bmatrix}$	$\begin{bmatrix}2 & -0.95\\-0.95 & 1\end{bmatrix}$
3	$\begin{bmatrix}-1.5\\-3\end{bmatrix}$	$\begin{bmatrix}2 & 0.1\\0.1 & 0.1\end{bmatrix}$

Figure 4.4: Cluster the 300 data points (left) into three clusters, without making any assumptions about the probability distribution of the data. In fact, the data were generated from three bivariate normal distributions, whose parameters are listed on the right.

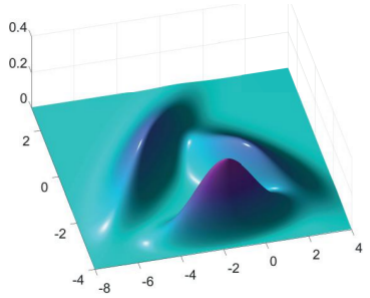


Figure 4.5: The target mixture density for the data in Figure 4.4.

Trong phần tiếp theo, chúng ta sẽ thực hiện phân cụm bằng thuật toán EM

4.5.2 Thuật Toán EM Cho Các Mô Hình Hỗn Hợp.

Như chúng ta đã thấy trong Phần 4.3, thay vì tối đa hóa hàm khả năng log trong (4.33) trực tiếp từ dữ liệu $\tau = \{x_1, ..., x_n\}$, thuật toán EM trước tiên tăng cường dữ liệu bằng vectơ của các biến tiềm ẩn - trong trường hợp này là các nhãn cụm ẩn $z = \{z_1, ..., z_n\}$.

Ý tưởng là τ là chỉ phần quan sát được của dữ liệu ngẫu nhiên hoàn chỉnh (τ, Z) , được tạo ra thông qua quy trình hai bước được mô tả ở trên. Nghĩa là, đối với mỗi điểm dữ liệu X, trước tiên hãy vẽ nhãn cụm $Z \in \{1, ..., K\}$ theo xác suất $\{w_1, ..., w_K\}$ và sau đó, cho $Z = z$, vẽ X từ ϕ_z. Hàm joint pdf của τ và Z là

$$g(\tau, z|\theta) = \prod_{i=1}^{n} w_{zi}\phi_{zi}(x_i)$$

có dạng đơn giản hơn nhiều so với (4.32).

Tiếp theo là complete-data log-likelihood function

$$\widetilde{l}(\theta|\tau, z) = \sum_{i=1}^{n} \ln(w_{zi}\phi_{zi}(x_i)) \tag{4.34}$$

thường dễ tối đa hóa hơn khả năng log ban đầu (4.33), đối với bất kỳ (τ, z) đã cho. Tuy nhiên, tất nhiên các biến tiềm ẩn z không được quan sát và $\widetilde{l}(\theta|\tau, z)$ không thể được đánh giá.

Trong bước E của thuật toán EM, the complete-data log-likelihood được thay thế bằng kỳ vọng $E_p\widetilde{l}(\theta|\tau, z)$ trong đó chỉ số con p trong kỳ vọng chỉ ra rằng Z được phân phối theo pdf có điều kiện của Z cho trước $T = \tau$; nghĩa là, với pdf

$$p(z) = g(z|\tau, \theta) \propto g(\tau, z|\theta) \tag{4.35}$$

Chú ý rằng $p(z)$ có dạng $p_1(z_1)....p_n(z_n)$ sao cho $T = \tau$ cho trước, các thành phần của Z là độc lập với nhau. Thuật toán EM cho các mô hình hỗn hợp hiện có thể được xây dựng như sau.

Algorithm 4.5.1: EM Algorithm for Mixture Models

input: Data τ, initial guess $\boldsymbol{\theta}^{(0)}$.
output: Approximation of the maximum likelihood estimate.

$t \leftarrow 1$
while a stopping criterion is not met **do**
  Expectation Step: Find $p^{(t)}(\mathbf{z}) := g(\mathbf{z}\,|\,\tau, \boldsymbol{\theta}^{(t-1)})$ and $Q^{(t)}(\boldsymbol{\theta}) := \mathbb{E}_{p^{(t)}}\widetilde{l}(\boldsymbol{\theta}\,|\,\tau, \mathbf{Z})$.
  Maximization Step: Let $\boldsymbol{\theta}^{(t)} \leftarrow \operatorname{argmax}_{\boldsymbol{\theta}} Q^{(t)}(\boldsymbol{\theta})$.
  $t \leftarrow t + 1$
return $\boldsymbol{\theta}^{(t)}$

Điều kiện kết thúc có thể xảy ra là dừng khi

$$|l(\theta^{(t)}|\tau) - l(\theta^{(t-1)}|\tau)|/|l(\theta^t|\tau)| < \epsilon$$

đối với một số dung sai nhỏ $\epsilon > 0$

Như đã được đề cập trong Phần 4.3, chuỗi các giá trị khả năng xảy ra log không giảm theo mỗi lần lặp. Theo những điều kiện liên tục nhất định , chuỗi $\{\theta^{(t)}\}$ được đảm bảo hội tụ tới một cực đại cục bộ của khả năng l. Sự hội tụ thành công cụ tối đa hóa toàn cầu (nếu nó tồn tại) phụ thuộc vào lựa chọn thích hợp cho giá trị bắt đầu. Thông thường, thuật toán được chạy từ các điểm bắt đầu ngẫu nhiên khác nhau .

Đối với trường hợp hỗn hợp Gauss, mỗi $\phi_k = \phi(\cdot|\mu_k, \Sigma_k)$, k=1,..,K là mật độ của phân bố Gaussian d-chiều. Gọi $\theta^{(t-1)}$ là dự đoán hiện tại cho vectơ tham số tối ưu , bao gồm trọng số $\{w_k^{(t-1)}\}$, vectơ trung bình $\{\mu_k^{(t-1)}\}$ và ma trận hiệp phương sai $\{\Sigma_k^{(t-1)}\}$.

Đầu tiên chúng ta xác định $p^{(t)}$ - pdf của Z với điều kiện $T = \tau$ – đối với dự đoán $\theta^{(t-1)}$ đã cho. Như đã đề cập trước đây, các thành phần của Z cho trước $T = \tau$ là độc lập, vì vậy nó đủ để chỉ định pdf rời rạc, $p_i^{(t)}$ mà tôi nói, của mỗi Z_i cho điểm quan sát $X_i = x_i$. Cái sau có thể được tìm thấy từ công thức của Bayes:

$$p_i^t(k) \propto w_k^{(t-1)}\phi_k(x_i|\mu_k^{(t-1)}, \Sigma_k^{(t-1)}), k = 1, .., K \tag{4.36}$$

Tiếp theo, theo quan điểm của (4.34) , hàm $Q^{(t)}(\theta)$ có thể được viết dưới dạng

$$Q^{(t)}(\theta) = E_{p^{(t)}} \sum_{i=1}^{n} (\ln w_{Z_i} + (\ln \phi_{Z_i}(x_i|\mu_{Z_i}, \Sigma_{Z_i}))$$

$$= \sum_{i=1}^{n} (E_{p^{(t)}}(\ln w_{Z_i} + \ln(\phi_{Z_i}(x_i|\mu_{Z_i}, \Sigma_{Z_i})))$$

trong đó $\{Z_i\}$ là độc lập và Z_i được phân phối theo $p_i^{(t)}$ trong (4.36). Điều này làm giảm tốc độ của E-step.

Trong bước M, chúng ta cực đại $Q^{(t)}$ đối với tham số θ; nghĩa là đối với $\{w_k\}$, $\{\mu_k\}$ và $\{\Sigma_k\}$.Đặc biệt, chúng ta tối đa hóa

$$\sum_{i=1}^{n} \sum_{k=1}^{K} p_i^k(t)(\ln w_k + \ln \phi_k(x_i|\mu_k, \Sigma_k))$$

với điều kiện $\sum_k w_k = 1$. Sử dụng nhân Lagrange và thực tế là $\sum_{k=1}^{K} p_i^{(t)}(k) = 1$ đưa ra nghiệm cho $\{w_k\}$:

$$w_k = \frac{1}{n} \sum_{i=1}^{n} p_i^{(t)}(k), k = 1, 2, .., K \tag{4.37}$$

Các giải pháp cho μ_k và Σ_k bây giờ tuân theo từ việc tối đa hóa $\sum_{i=1}^{n} p_i^{(t)}(k) \ln \phi_k(x_i|\mu_k, \Sigma_k)$

$$\mu_k = \frac{\sum_{i=1}^{n} p_i^{(t)}(k) x_i}{\sum_{i=1}^{n} p_i^{(t)}(k)}, k = 1, .., K \tag{4.38}$$

and

$$\Sigma_k = \frac{\sum_{i=1}^{n} p_i^{(t)}(k)(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{n} p_i^{(t)}(k)}, k = 1, .., K \tag{4.39}$$

những công thức rất giống với các công thức nổi tiếng cho MLE của các tham số của phân phối Gauss. Sau khi gán các tham số nghiệm cho $\theta^{(t)}$ và tăng bộ đếm lặp t lên 1, các bước (4.36), (4.37), (4.38) và (4.39) được lặp lại cho đến khi đạt được sự kết hợp. Sự hội tụ của thuật toán EM rất nhạy cảm với việc lựa chọn các tham số ban đầu. Do đó, chúng tôi đề nghị thử nhiều điều kiện bắt đầu khác nhau. Để thảo luận thêm về các khía cạnh lý thuyết và thực tiễn của thuật toán EM, chúng tôi đề cập đến [85]. .

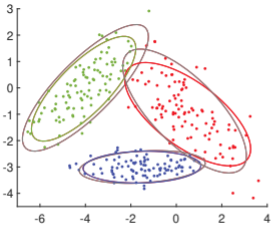
Ví dụ 4.5 (Phân cụm qua EM) Chúng tôi quay lại dữ liệu trong Ví dụ 4.4, được mô tả trong Hình 4.4 và áp dụng mô hình mà dữ liệu đến từ hỗn hợp của ba phân phối Gaussian hai biến.
Mã Python bên dưới thực hiện thủ tục EM được mô tả trong Thuật toán 4.5.1.

Các vectơ trung bình ban đầu $\{\mu_k\}$ của phân bố Gaussian hai biến được chọn (từ việc kiểm tra bằng mắt) để nằm gần đúng ở giữa mỗi cụm, trong trường hợp này là $[-2, -3]^T, [-4, 1]^T, [0, -1]^T$.
Các ma trận hiệp phương sai tương ứng ban đầu được chọn làm ma trận nhận dạng, điều này thích hợp với sự trải rộng dữ liệu quan sát được trong Hình 4.4. Cuối cùng, các trọng lượng ban đầu là 1/3, 1/3, 1/3. Để đơn giản, thuật toán dừng sau 100 lần lặp, trong trường hợp này là quá đủ để đảm bảo sự hội tụ.

Mã và dữ liệu có sẵn từ trang web của cuốn sách trong thư mục GitHub Chương 4

```
import numpy as np
from scipy.stats import multivariate_normal

Xmat = np.genfromtxt('clusterdata.csv', delimiter=',')
K = 3
n, D = Xmat.shape

W = np.array([[1/3,1/3,1/3]])
M  = np.array([[-2.0,-4,0],[-3,1,-1]], dtype=np.float32)
# Note that if above *all* entries were written as integers, M would
# be defined to be of integer type, which will give the wrong answer

C = np.zeros((3,2,2))

C[:,0,0] = 1
C[:,1,1] = 1

p = np.zeros((3,300))

for i in range(0,100):

#E-step
    for k in range(0,K):
        mvn = multivariate_normal( M[:,k].T, C[k,:,:] )
        p[k,:] = W[0,k]*mvn.pdf(Xmat)

# M-Step
    p = (p/sum(p,0))   #normalize
    W = np.mean(p,1).reshape(1,3)

    for k in range(0,K):
        M[:,k] = (Xmat.T @ p[k,:].T)/sum(p[k,:])
        xm = Xmat.T - M[:,k].reshape(2,1)
        C[k,:,:] = xm @ (xm*p[k,:]).T/sum(p[k,:])
```

Các thông số ước lượng của sự phân bố hỗn hợp được cho ở bên phải của Hình 4.6. Sau khi gắn nhãn lại cho các cụm, chúng ta có thể quan sát thấy sự trùng khớp chặt chẽ với các thông số trong Hình 4.4.Các hình elip ở phía bên trái của Hình 4.6 cho thấy sự phù hợp chặt chẽ giữa các hình elip xác suất 95% của các phân phối Gaussian ban đầu (màu xám) và các phân bố ước tính.
Một cách tự nhiên để phân cụm từng điểm x_i là gán nó vào cụm k mà xác suất có điều kiện $p_i(k)$ là cực đại (với các mối quan hệ được giải quyết tùy ý). Điều này cho phép nhóm các điểm thành các cụm màu đỏ, xanh lục và xanh lam trong hình.



weight	mean vector	covariance matrix
0.33	$\begin{bmatrix}-1.51\\-3.01\end{bmatrix}$	$\begin{bmatrix}1.75 & 0.03\\0.03 & 0.095\end{bmatrix}$
0.32	$\begin{bmatrix}-4.08\\-0.033\end{bmatrix}$	$\begin{bmatrix}1.37 & 0.92\\0.92 & 1.03\end{bmatrix}$
0.35	$\begin{bmatrix}0.36\\-0.88\end{bmatrix}$	$\begin{bmatrix}1.93 & -1.20\\-1.20 & 1.44\end{bmatrix}$

Figure 4.6: The results of the EM clustering algorithm applied to the data depicted in Figure 4.4.

Để thay thế cho thuật toán EM, tất nhiên người ta có thể sử dụng các thuật toán tối ưu hóa đa văn bản liên tục để tối ưu hóa trực tiếp hàm khả năng đăng nhập $l(\theta|\tau) = \ln g(\tau|\theta)$ trong (4.33) trên tập Θ tất cả những gì có thể θ. Điều này được thực hiện chẳng hạn trong [15], chứng tỏ kết quả vượt trội so với EM khi có ít điểm dữ liệu .
Điều tra kỹ hơn về hàm khả năng cho thấy rằng có một vấn đề tiềm ẩn với bất kỳ cách tiếp cận khả năng tối đa nào để phân cụm nếu Θ được chọn càng lớn càng tốt – tức là, bất kỳ phân phối hỗn hợp nào cũng có thể xảy ra.

Để chứng minh vấn đề này, hãy xem xét Hình 4.7, mô tả hàm mật độ xác suất, $g(\cdot|\theta)$ của hỗn hợp hai phân bố Gaussian, trong đó $\theta = [w, \mu_1, \sigma_1^2, \mu_2, \sigma_2^2]^T$ là véc tơ tham số của phân bố hỗn hợp.
Hàm log-khả năng xảy ra được cho bởi

$$l(\theta|\tau) = \sum_{i=1}^{4} \ln g(x_i|\theta)$$

trong đó $x_1, ..., x_4$ là dữ liệu (được biểu thị bằng các dấu chấm trong hình)

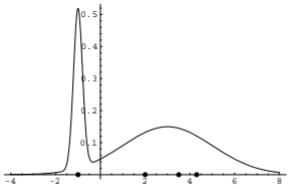


Figure 4.7: Mixture of two Gaussian distributions.

Rõ ràng là bằng cách cố định hằng số trộn w ở $0,25$ (giả sử) và căn giữa cụm đầu tiên ở x_1, người ta có thể thu được giá trị khả năng lớn tùy ý bằng cách lấy phương sai của cụm đầu tiên nhỏ tùy ý.
Tương tự, đối với dữ liệu có chiều cao hơn, bằng cách chọn các cụm "điểm" hoặc "dòng" hoặc các cụm "suy giảm" nói chung, người ta có thể làm cho giá trị của khả năng là vô hạn.

Đây là biểu hiện của vấn đề trang bị quá mức quen thuộc đối với mất đào tạo mà chúng ta đã gặp trong Chương 2.
Do đó, cực đại không bị giới hạn của hàm khả năng log-khả năng là một bài toán khó, bất kể việc lựa chọn thuật toán tối ưu hóa là gì!

Hai giải pháp khả thi cho vấn đề "overfitting" này là:
1. Hạn chế tập tham số Θ theo cách không cho phép các cụm suy biến (đôi khi được gọi là cụm giả).
2. Chạy thuật toán đã cho và nếu giải pháp bị suy biến, hãy loại bỏ nó và chạy thuật toán mới. Tiếp tục khởi động lại thuật toán cho đến khi thu được giải pháp không suy biến .
Cách tiếp cận đầu tiên thường được áp dụng cho các thuật toán tối ưu hóa đa chiều và cách tiếp cận thứ hai được sử dụng cho thuật toán EM.

4.6

Nguyễn Thị Cẩm Hương

Ngày 24 tháng 12 năm 2021

1 Phân cụm thông qua lượng tử hóa vectơ

Trong phần trước, chúng tôi đã giới thiệu phân cụm thông qua các mô hình hỗn hợp, như một hình thức ước tính mật độ tham số (trái ngược với ước tính mật độ phi tham số trong Phần 4.4).
Các cụm được mô hình hóa một cách tự nhiên thông qua các biến tiềm ẩn và thuật toán EM cung cấp một cách thuận tiện để gán các thành viên cụm. Trong phần này, chúng tôi xem xét một cách tiếp cận heuristic (khám phá sâu) hơn để phân cụm bằng cách bỏ qua các thuộc tính phân phối của dữ liệu. Các thuật toán kết quả có xu hướng mở rộng tốt hơn với số lượng mẫu n và số chiều d.
Trong phần này, ta sẽ xem xét một cách tiếp cận sâu hơn để phân cụm bằng cách bỏ qua các thuộc tính phân phối của dữ liệu. Các thuật toán kết quả có xu hướng mở rộng tốt hơn với số lượng mẫu n và số chiều d.
Cho tập hợp $\tau := \{x_1, ..., x_n\}$ các điểm dữ liệu trong một số d chiều không gian X, chia tập dữ liệu này thành K nhóm sao cho một số hàm mất được tối thiểu hóa.
Đầu tiên, ta phân chia toàn bộ không gian X, sử dụng một số hàm khoảng cách dist $(\cdot, \cdot)$ trên không gian này. Một lựa chọn tiêu chuẩn là Euclidean (hoặc L_2) khoảng cách:

$$dist(x, x') = \|x - x'\| = \sqrt{\sum_{i=1}^{d}(x_i - x'_i)^2}$$

Các thước đo khoảng cách thường được sử dụng khác trên $\mathbb{R}^d$ bao gồm khoảng cách Manhattan

$$\sum_{i=1}^{d} |x_i - x'_i|$$

và khoảng cách tối đa

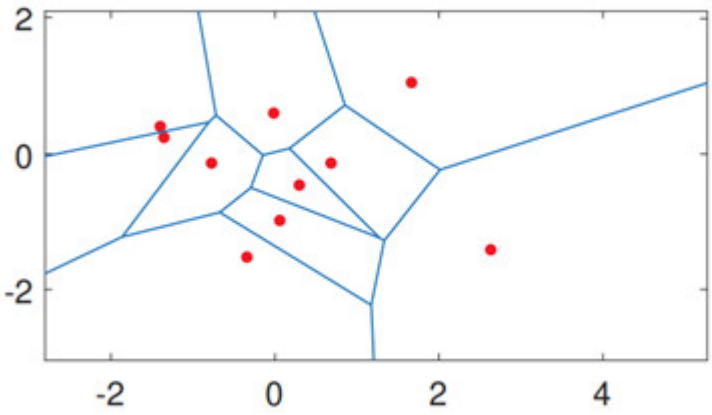
$$\max_{i=1,..,d} |x_i - x'_i|$$

Trên tập hợp các chuỗi có độ dài d, một thước đo khoảng cách thường được sử dụng là khoảng cách Hamming

$$\sum_{i=1}^{d} 1x_i \neq x'_i$$

Nghĩa là số lượng ký tự không khớp. Ví dụ, khoảng cách Hamming giữa 010101

và 011010 là 4.
Chúng ta có thể phân vùng không gian X thành các vùng như sau:
Đầu tiên, chúng ta chọn K điểm $c_1, ..., c_K$ được gọi là tâm cụm hoặc vectơ nguồn . Với mỗi k = 1, ..., K , sao cho:
$R_k = \{x \in X : dist(x, c_k) \leq dist(x, c_i)$ cho tất cả i ≠ j } là tập hợp các điểm trong X nằm gần c_k hơn bất kỳ tâm nào khác. Các vùng hoặc ô R_k chia không gian X thành cái được gọi là Voronoi diagram (biểu đồ Voronoi) hoặc Voronoi tessellatio.
Hình 4.8 cho thấy sự phân chia Voronoi của máy bay thành mười vùng, sử dụng Euclidean khoảng cách. Lưu ý rằng ở đây ranh giới giữa các ô Voronoi là đường thẳng. Đặc biệt, nếu ô R_i vàR_j có chung một đường viền thì một điểm trên đường viền này phải thỏa mãn $x - c_i = x - c_j$; nghĩa là nó phải nằm trên đường thẳng đi qua điểm $(cj + ci)/2$ (nghĩa là trung điểm của đoạn thẳng giữa c_i và c_j) và vuông góc với $c_j - c_i$



Hình 4.8: Một điểm dừng Voronoi của máy bay thành mười ô, được xác định bởi (màu đỏ) các trung tâm. Khi các tâm (và do đó các ô R_k) được chọn, các điểm trong τ có thể được nhóm lại theo trung tâm gần nhất của chúng. Các điểm trên ranh giới phải được xử lý riêng biệt. Điều này là điểm tranh luận cho dữ liệu liên tục, vì nói chung không có điểm dữ liệu nào nằm chính xác trên ranh giới. Vấn đề còn lại chính là làm thế nào để chọn các trung tâm để phân cụm dữ liệu trong một số cách tối ưu. Về khung học tập (không giám sát), muốn ước tính một vectơ x qua một trong $c_1, ..., c_K$, sử dụng một hàm có giá trị vectơ hằng số

$$g(x|\mathbb{C}) := \sum_{k=1}^{K} c_k 1\{x \in R_k\}$$

trong đó C là ma trận $d \times$ K $[c_1, ..., c_K]$. Do đó, $g(x|C) = c_k$ khi x nằm trong vùng R_k (chúng ta bỏ qua các ràng buộc). Trong lớp hàm G này, được tham số hóa bởi C, mục đích của chúng tôi là giảm thiểu sự mất mát trong đào tạo.

Đặc biệt, đối với tổn thất sai số bình phương. $Loss(x, x') = x - x'2$

$$l_{\tau n}(g(.|C)) = \frac{1}{n}\sum_{i=1}^{n}\|x_i - g(x_i|\mathbb{C})\|^2 = \frac{1}{n}\sum_{k=1}^{K}\sum_{x\in R_k\cap\tau n}\|x - c_k\|^2$$

Do đó, sự mất mát trong đào tạo giảm thiểu khoảng cách bình phương trung bình giữa các trung tâm. Điều này cũng kết hợp cả các bước mã hóa và giải mã trong lượng tử hóa vectơ
[125]. Cụ thể, chúng tôi muốn "lượng tử hóa" hoặc "mã hóa" các vectơ trong τ theo cách mà mỗi vectơ được biểu diễn bởi một trong K vectơ nguồn $c_1, ..., c_K$, sao cho (4.40) đại diện được giảm thiểu. Hầu hết các phương pháp phân cụm và lượng tử hóa vectơ nổi tiếng đều cập nhật vectơ của các trung tâm, bắt đầu từ một số lựa chọn ban đầu và sử dụng thủ tục lặp lại (thường dựa trên gradient). Điều quan trọng là nhận ra rằng trong trường hợp này (4.40) được coi là một chức năng của các trung tâm, trong đó mỗi điểm x được gán cho tâm gần nhất, do đó xác định các cụm. Nó là tốt được biết rằng loại vấn đề này - tối ưu hóa đối với các trung tâm - rất đa dạng, tùy thuộc vào các cụm ban đầu, các thủ tục dựa trên gradient(dốc, điểm) có xu hướng hội tụ ở mức tối thiểu cục bộ hơn là mức tối thiểu toàn cầu.

1.1 K- Phương tiện

Một trong những phương pháp đơn giản nhất để phân cụm là phương pháp K-mean. Nó là một phương pháp lặp lại trong đó, bắt đầu từ phỏng đoán ban đầu cho các trung tâm, các trung tâm mới được hình thành bằng cách lấy trung tâm phương tiện mẫu của các điểm hiện tại trong mỗi cụm. Do đó, các trung tâm mới là trung tâm của các điểm trong mỗi ô. Mặc dù tồn tại nhiều loại K-means khác nhau thuật toán, chúng về cơ bản đều có dạng sau:

Algorithm 4.6.1: K-Means

input: Collection of points $\tau = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, number of clusters K, initial centers $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_K$.
output: Cluster centers and cells (regions).

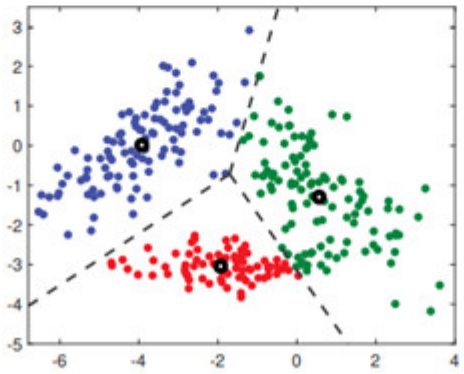
while a stopping criterion is not met **do**
  $\mathcal{R}_1, \ldots, \mathcal{R}_K \leftarrow \emptyset$ (empty sets).
  for $i = 1$ **to** n **do**
    $\boldsymbol{d} \leftarrow [\text{dist}(\boldsymbol{x}_i, \boldsymbol{c}_1), \ldots, \text{dist}(\boldsymbol{x}_i, \boldsymbol{c}_K)]$ // distances to centers
    $k \leftarrow \text{argmin}_j\, d_j$
    $\mathcal{R}_k \leftarrow \mathcal{R}_k \cup \{\boldsymbol{x}_i\}$ // assign $\boldsymbol{x}_i$ to cluster k
  for $k = 1$ **to** K **do**
    $\boldsymbol{c}_k \leftarrow \frac{\sum_{\boldsymbol{x}\in\mathcal{R}_k}\boldsymbol{x}}{|\mathcal{R}_k|}$ // compute the new center as a centroid of points
return $\{\boldsymbol{c}_k\}, \{\mathcal{R}_k\}$

Do đó, tại mỗi lần lặp, đối với một lựa chọn trung tâm nhất định, mỗi điểm trong τ được gán cho trung tâm gần nhất của nó. Sau khi tất cả các điểm đã được chỉ định, các trung tâm được tính lại thành trọng tâm của tất cả các điểm trong cụm hiện tại (Dòng 8). Một tiêu chí dừng điển hình là dừng lại khi các trung tâm không còn thay đổi nhiều. Vì thuật toán khá nhạy cảm với sự lựa chọn của các trung tâm ban đầu, nên thận trọng khi thử nhiều giá trị bắt đầu.
Ví dụ: đã chọn ngẫu nhiên từ hộp giới hạn của các điểm dữ liệu. Chúng ta có thể xem phương pháp K-mean như một phiên bản xác định (hoặc "cứng") của xác suất thuật toán EM ilistic (hoặc "mềm") như sau. Giả sử trong thuật toán EM, chúng ta có Gausian hỗn hợp với ma trận hiệp phương sai cố định $\sum_k \sigma^2 \mathbb{I}_d, k = 1, ..., K$, trong đó σ^2 phải là được coi là rất nhỏ. Xem xét lần lặp t của thuật toán EM. Dang có thu được vectơ kỳ vọng${\mu_k}^{t-}$) và trọng lượng ${w_k}^{t-1}$, k = 1, ..., K, mỗi điểm x_i là đã ký một nhãn cụm Z_i theo các xác suất $p_i^{(t)}(k)$, k = 1, ..., K cho trong (4.36). Trang 4 Chương 4. Học tập không giám sát 145
Nhưng đối với $\sigma^2 \to 0$ thì phân phối xác suất $p_i^{(t)}(k)$ trở nên thoái hóa, đặt tất cả khối lượng xác suất trên $argmin_k \|x_i - \mu_k\|^2$. Điều này tương ứng với quy tắc K-mean chỉ định x_i đến trung tâm cụm gần nhất của nó. Hơn nữa, ở bước M (4.38) mỗi trung tâm cụm $\mu_k^{(t)}$ Hiện tại là được cập nhật theo giá trị trung bình của x_i đã được gán cho cụm k. Do đó chúng tôi có được quy tắc cập nhật xác định tương tự như trong K-mean. Ví dụ 4.6 (K-means Clustering) Chúng tôi phân cụm dữ liệu từ Hình 4.8 qua K-means, bằng cách sử dụng triển khai Python bên dưới. Lưu ý rằng các điểm dữ liệu được lưu trữ dưới dạng 300 $\times$ 2 ma trận Xmat. Chúng ta lấy các tâm bắt đầu tương tự như trong ví dụ EM: $c_1 = [2, 3]^T, c_2 = [4, 1]^T$ và $c_3 = [0, 1]^T$. Cũng lưu ý rằng khoảng cách Euclid bình phương được sử dụng trong tính toán, vì chúng được tính toán nhanh hơn một chút so với khoảng cách Euclide (vì không có hình vuông tính toán gốc là bắt buộc) trong khi mang lại chính xác các đánh giá trung tâm cụm.

Kmeans.py

```
import numpy as np
Xmat = np.genfromtxt('clusterdata.csv', delimiter=',')
K = 3
n, D = Xmat.shape
c  = np.array([[-2.0,-4,0],[-3,1,-1]])  #initialize centers
cold = np.zeros(c.shape)
dist2 = np.zeros((K,n))
while np.abs(c - cold).sum() > 0.001:
   cold = c.copy()
   for i in range(0,K): #compute the squared distances
        dist2[i,:] = np.sum((Xmat - c[:,i].T)**2, 1)

   label = np.argmin(dist2,0) #assign the points to nearest centroid
   minvals = np.amin(dist2,0)
   for i in range(0,K): # recompute the centroids
      c[:,i] = np.mean(Xmat[np.where(label == i),:],1).reshape(1,2)

print('Loss = {:3.3f}'.format(minvals.mean()))
```

```
Loss = 2.288
```



Hình 4.9: Kết quả của thuật toán K-mean được áp dụng cho dữ liệu trong Hình 4.4 . Dày vòng tròn đen là tâm và các đường chấm xác định ranh giới tế bào

Chúng tôi tìm thấy các trung tâm cụm $c_1 = [1.9286, 3.0416]^T$,$c_2 = [-3.9237, 0.0131]^T$ và $c_3 = [0, 5611, 1, 2980]^T$, cho phép phân nhóm được mô tả trong Hình 4.9 . Tương ứng tổn thất (4,40) được tìm thấy là 2,288.

1.2 Phân cụm thông qua tối ưu hóa đa văn bản liên tục

Như đã đề cập, việc tối thiểu hóa chính xác hàm mất mát (4.40) là khó hoàn thành thông qua các phương pháp tìm kiếm cục bộ tiêu chuẩn, chẳng hạn như gradient descent, như hàm rất đa phương thức. Tuy nhiên, không có gì ngăn cản chúng tôi sử dụng tính năng tối ưu hóa toàn cầu các phương pháp như phương pháp CE hoặc SCO được thảo luận trong Phần 3.4.2 và 3.4.3 Ví dụ 4.7 (Phân cụm qua CE) Chúng tôi lấy cùng một tập dữ liệu như trong Ví dụ 4.6 và phân nhóm các điểm thông qua giảm thiểu tổn thất (4.40) bằng phương pháp CE. Con trăn mã dưới đây rất giống với mã trong Ví dụ 3.16, ngoại trừ việc bây giờ chúng ta đang xử lý 101 với một bài toán tối ưu hóa sáu chiều. Hàm mất mát được thực hiện trong function Scluster , về cơ bản sử dụng lại phép tính khoảng cách bình phương của K- mean mã trong Ví dụ 4.6. Chương trình CE thường quy về mức lỗ 2,287, tương ứng đi vào bộ thu nhỏ (toàn cục) $c_1 = [1.9286, 3.0416]^T$, $c_2 = [3.8681, 0.0456]^T$ và $c_3 = [0, 5880, 1, 3526]^T$, hơi khác so với các bộ giảm thiểu cục bộ cho K- mean thuật toán.

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

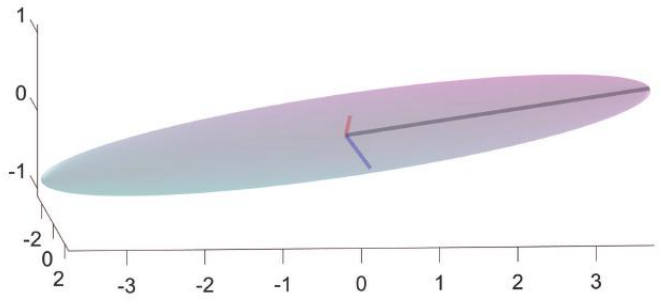
$$\boldsymbol{U} = \begin{bmatrix} \mathbf{0.8460} & \mathbf{0.4828} & \mathbf{0.2261} \\ \mathbf{0.4973} & \mathbf{-0.5618} & \mathbf{-0.6611} \\ \mathbf{0.1922} & \mathbf{-0.6718} & \mathbf{0.7154} \end{bmatrix}$$

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:

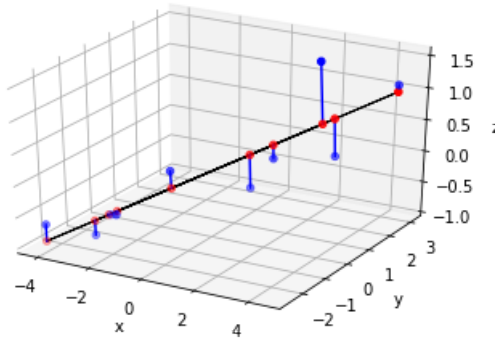
$$\boldsymbol{U} = \begin{bmatrix} \mathbf{-0.8277} & \mathbf{0.4613} & \mathbf{0.3195} \\ \mathbf{-0.5300} & \mathbf{-0.4556} & \mathbf{-0.7152} \\ \mathbf{-0.1843} & \mathbf{-0.7613} & \mathbf{0.6216} \end{bmatrix},$$

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()
```

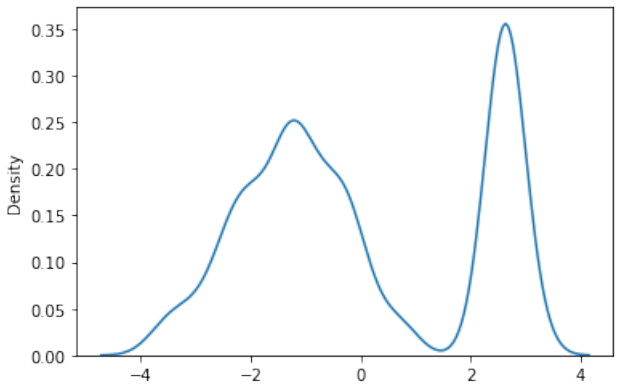
Tiếp theo là một ứng dụng của PCA cho bộ dữ liệu iris nổi tiếng của Fisher, đã được đề cập trong Phần 1.1 và Bài tập 1.5. Ví dụ 4.12 (PCA cho Tập dữ liệu Iris) Bộ dữ liệu iris chứa các phép đo về bốn đặc điểm của cây iris: chiều dài và chiều rộng của lá đài, chiều dài và chiều rộng của cánh hoa, với 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.