10. Analiza componentelor principale

Normalizarea datelor
Baze ortogonale de reprezentare a datelor
Maximizarea varianţei datelor

Obiectivele centrale ale algoritmului de *Analiză a Componentelor Principale* (eng. Principal Components Analysis - PCA) sunt analiza datelor în scopul identificării de structuri în date şi de reduce a numărului de dimensiuni prin care sunt reprezentate datele. Formal, datele sunt reprezentate într-un spaţiu n-dimensional: $x \in \mathbb{R}^n$. PCA identifică un subspaţiu k-dimensional al datelor (unde $k < n$) prin calculul vectorilor proprii ai lui x.

Considerăm un set de antrenare $\{x^{(i)}; i = 1, ..., m\}$ ce conţine atribute (precum viteza maximă, comportamentul în viraje, etc) pentru m tipuri de automobile. Fie $x \in \mathbb{R}^n$ pentru fiecare i ($n \ll m$). Fără a ne fi cunoscut, două tipuri de atribute (x_i, repsectiv x_j) reprezintă viteza maximă a automobilului în kilometrii pe oră şi, respectiv, în mile pe oră. Aceste două atribute sunt liniar dependente. Astfel, datele sunt, de fapt, reprezentate într-un spaţiu $n-1$ dimensional. Metoda PCA este utilizată în identificarea şi înlăturarea acestor redundanţe.

Un alt exemplu este acela al unei baze de date compusă din părerile unor piloţi de elicoptere controlate prin radio, unde $x_1^{(i)}$ este o mărime a gradului de îndemânare al pilotului i, iar $x_2^{(i)}$ reprezintă gradul de bucurie pe care îl are în pilotare. Deoarece elicopterele controlate prin radio sunt dificil de manevrat, doar studenţii dedicaţi, care se bucură această activitate, ajung să fie buni piloţi. Astfel, atributele x_1 şi x_2 sunt puternic corelate. După cum este redat în figura 10.1, se poate spune că datele sunt reprezentate de-a lungul unei axe diagonale (direcţia u_1) ce descrie modul de comportare al unui pilot. În cele ce urmează vom calcula direcţia u_1 prin metoda PCA.

Anterior dezvoltării algoritmului PCA, vom normaliza datele utilizând media şi varianţa lor, astfel:

1. Fie $\mu = \frac{1}{m} \sum_{i=1}^{m} x^{(i)}$;
2. Fiecare $x^{(i)}$ va fi înlocuit cu $x^{(i)} - \mu$;
3. Fie $\sigma_j^2 = \frac{1}{m} \sum_i \left(x_j^{(i)} \right)^2$;
4. Fiecare $x_j^{(i)}$ va fi înlocuit cu $x_j^{(i)} / \sigma_j$.

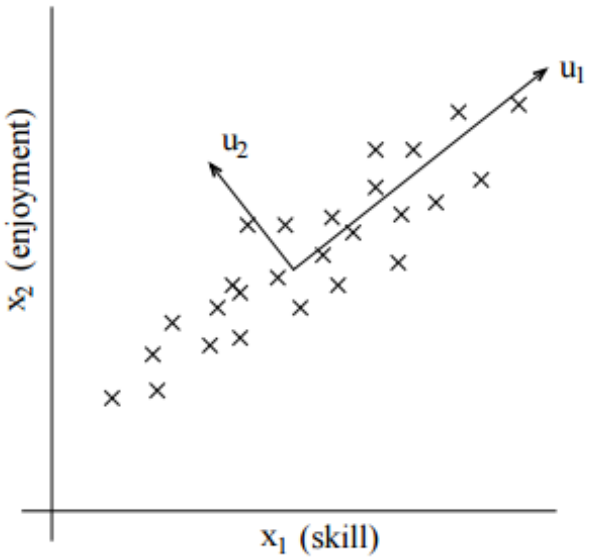

Fig. 10.1 Distribuţia caracteristicilor ce descriu comportamentul unui pilot de elicopter.

Paşii (1) şi (2) centrează datele pe media zero. Aceşti paşi pot fi omişi atunci când se cunoaşte faptul că datele au media zero (spre exemplu, pentru seriile de timp ce descriu semnale acustice). Paşii (3) şi (4) scalează fiecare coordonată a datelor la varianţa unitate. Această scalare asigură că diferitele atribute au aceeaşi scală şi sunt astfel tratate la fel. Spre exemplu, dacă x_1 este viteza maximă a unui autoturism, măsurată în km/h (ce are valori de ordinul a sute de km/h), iar x_2 repezintă numărul de scaune (ce în mod normal sunt în număr de 2 sau 4), atunci scalarea prin varianţă face atributele mai compatibile între ele. Paşii (3) şi (4) pot fi omişi atunci când ştim că atributele sunt reprezentate la o scală compatibilă. Un exemplu în acest sens este atunci când considerăm fiecare pixel dintr-o imagine gri ca fiind o caracteristică $x_j^{(i)}$, fiecare valoare variind în intervalul $\{0, 1, ..., 255\}$, ce corespunde intensităţii pixelului j din imaginea i.

În următoarea fază se calculează axa principală de variaţie u pe care sunt distribuite datele. O modalitate de descriere a acestei probleme este de a găsi vectorul unitate u, în aşa fel încât varianţa datelor proiectate să fie maximă atunci când sunt proiectate de-a lungul direcţiei lui u. Intuitiv, datele vor avea pentru început un anumit grad de varianţă. Dorim să găsim o direcţie a lui u în aşa fel încât să se păstreze cât mai multă varianţă atunci când datele sunt proiectate pe direcţia/sub-spaţiul lui u.

Consideraţi setul de date normalizat reprezentat în figura 10.2.

O posibilă direcţie pentru u este ilustrată în figura 10.3. Cercurile reprezintă proiecţia datelor originale pe direcţia lui u.

Se poate observa din figura 10.3 că datele proiectate păstrează încă un grad ridicat de varianţă, cu puncte îndepărtate de valoarea zero. În contrast cu acest caz, varianţa datelor proiectate de-a lungul direcţiei ilustrate în figura 10.4 este mult mai mică, cu puncte mult mai apropiate de origine.

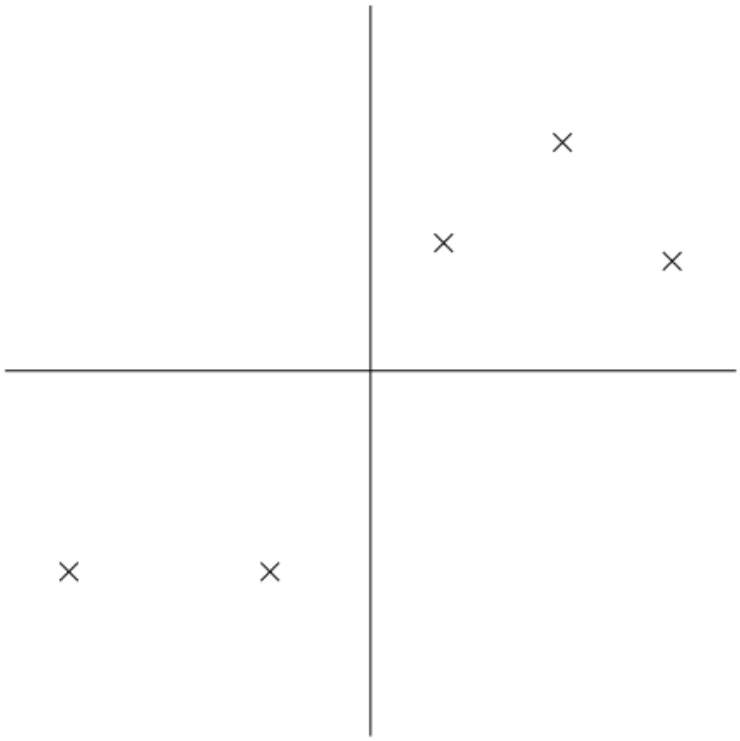

Fig. 10.2 Set de date normalizat.

În analiza componentelor principale, dorim să selectăm automat direcţia lui u ce corespunde figurii 10.3. Formal, pentru un vector unitate u şi un punct x, lungimea proiecţiei lui x pe u este dată de $x^T u$. Spre exemplu, dacă $x^{(i)}$ este un punct din baza de date originală, atunci proiecţia sa pe u este distanţa $x^T u$ de la origine. Astfel, pentru a maximiza varianţa proiecţiilor, dorim să alegem vectorul unitate u în aşa fel încât să maximizăm:

$$\begin{aligned}\frac{1}{m}\sum_{i=1}^{m}\left(x^{(i)^T}u\right)^2 &= \frac{1}{m}\sum_{i=1}^{m}u^T x^{(i)}x^{(i)^T}u \\ &= u^T\left(\frac{1}{m}\frac{1}{m}\sum_{i=1}^{m}x^{(i)}x^{(i)^T}\right)u.\end{aligned}$$

Se poate observa că maximizarea expresiei de mai sus utilizând constrângerea $\|u\|_2 = 1$ rezultă în vectorii proprii principali ai expresiei:

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m}x^{(i)}x^{(i)^T}. \tag{10.1}$$

Expresia 10.1 reprezintă matricea de covarianţă a datelor, luând în considerare că datele au media zero (sunt centrate pe valoarea zero).

Astfel, pentru a găsi un subspaţiu 1-dimensional ce poate aproxima datele 2-dimensionale, ar trebui ales u ca fiind vectorii proprii ai matricei Σ. Generalizat, dacă dorim să proiectăm datele într-un subspaţiu k-dimensional ($k < n$), ar trebui sa alegem $u_1, u_2, ..., u_k$ ca şi primii

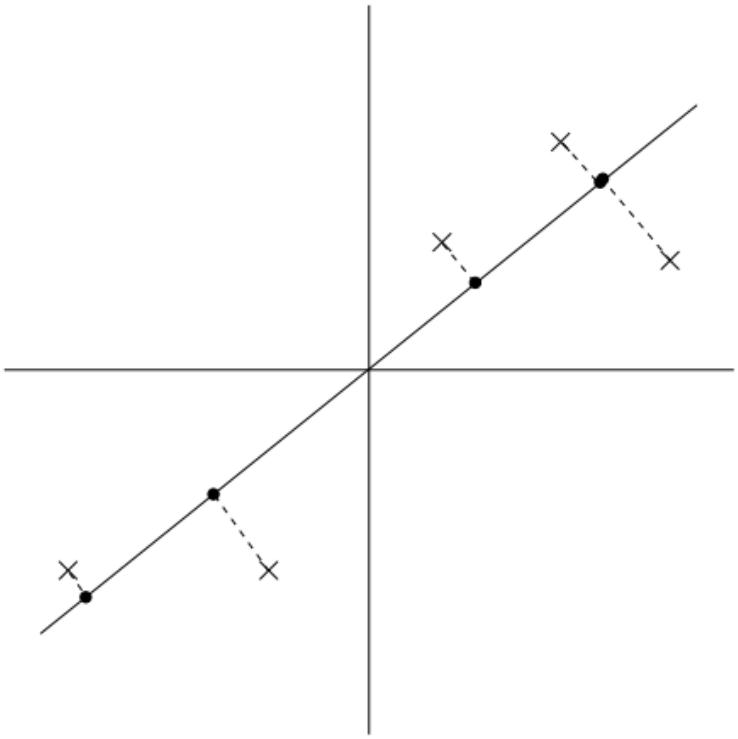

Fig. 10.3 Posibila proiecţie a setului de date din figura 10.2.

k vectori proprii ai matricei Σ. Noile valori u_i reprezintă noua bază ortogonală a datelor. Deoarece Σ este simetrică, vectorii u_i pot fi aleşi ortogonali unul faţă de celălalt.

Pentru a reprezenta $x^{(i)}$ în această nouă bază, trebuie să calculăm vectorii corespunzători:

$$y^{(i)} = \begin{bmatrix} u_1^T x^{(i)} \\ u_2^T x^{(i)} \\ \vdots \\ u_k^T x^{(i)} \end{bmatrix}, \quad y^{(i)} \in \mathbb{R}^k. \tag{10.2}$$

Luând în considerare că $x^{(i)} \in \mathbb{R}^n$, vectorul $y^{(i)}$ oferă o aproximare/reprezentare a lui $x^{(i)}$ într-un spaţiu k-dimensional. Din această cauză metoda PCA este adesea întâlnită sub denumirea de algoritm de *reducere a dimensiunii*. Vectorii $u_1, u_2, ..., u_k$ sunt denumiţi primele k *componente principale* ale datelor.

Cu toate că în acestă lucrare de laborator vom reduce datele la subspaţiul 1-dimensional ($k = 1$), se poate demonstra că dintre toate bazele ortogonale $u_1, u_2, ..., u_k$, cea aleasă maximizează expresia $\sum_i \|y^{(i)}\|_2^2$. Astfel, alegerea unei baze ortogonale păstrează pe cât de mult posibil variabilitatea din datele originale.

Algoritmul PCA se utilizează într-o gamă largă de probleme, precum compresia datelor, vizualizarea distribuţiei caracteristicilor (eng. features) într-un spaţiu 1-, 2-, sau 3-dimensional, sau ca şi o modalitate de reducere a zgomotului.

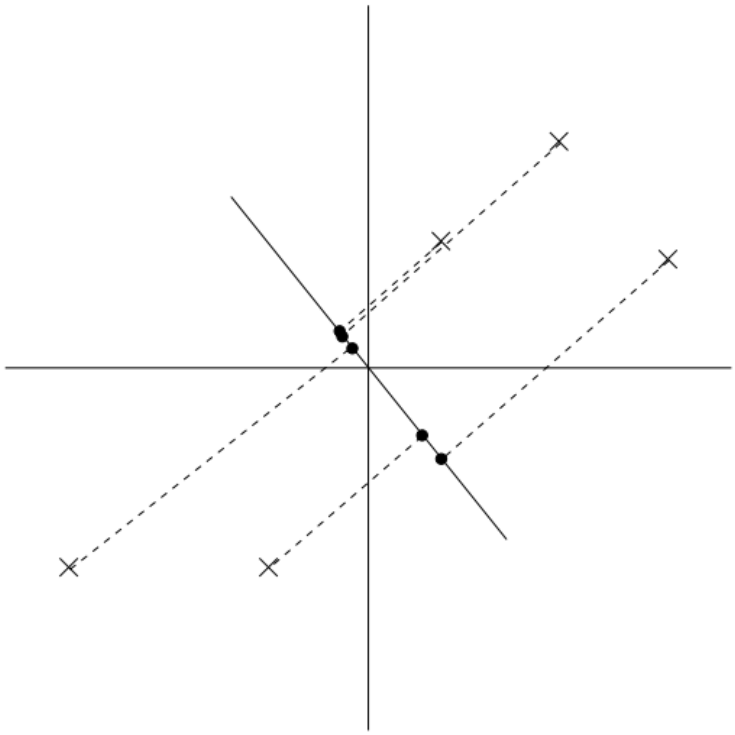

Fig. 10.4 Posibila proiecţie a setului de date din figura 10.2.

10.1 Cerinţe

Să se scrie un script Python ce calculează (i) vectorii şi valorile proprii ale unui set de date generat aleator dintr-o distribuţie Gaussiană normală şi (ii) transformă datele utilizând vectorii proprii obţinuţi.