

Réduction

Primary Component Analysis

Imagine qu'on ait une liste de pays avec différentes valeurs pour le PIB, les services sociaux, l'espérance de vie, etc. On ne peut pas tout représenter sur un unique graphe. On utilise PCA pour obtenir 1, 2, ou 3 axes qui nous permettent d'obtenir une visualisation qui sépare les pays selon des critères fusionnés (par exemple, les pays "très bonne espérance de vie et très bons services sociaux" et les autres). Chaque axe suit un ordre d'importance (on fait une première séparation avec le premier axe, puis on divise en sous-groupe avec les autres axes), et est décorrélié des autres.

🔗 Intuitivement: la matrice de covariance

- "Quand le PIB augmente, est-ce que l'espérance de vie augmente aussi ?" → Covariance positive
- "Quand un augmente, l'autre baisse ?" → Covariance négative
- "Pas de lien ?" → Covariance proche de 0

❓ Pourquoi la matrice de covariance est utile ?

Pour cela, on veut arriver à savoir quelles variables varient ensemble, pour les rattacher à un même axe.

Au début, on veut donc prendre l'axe qui nous fait perdre le moins d'information. Quand on projette un point **centré** ($x = x_{\text{original}} - \text{moyenne des } x$) x sur un vecteur \vec{u} (tel que c'est un vecteur unitaire $\|u\| = u^T u = 1$) on obtient un nouveau vecteur $x' = (\vec{x}^T u) \vec{u}$.

On peut mesurer l'information préservée avec $(x^T u)^2$.

Au début, on veut donc prendre le vecteur qui maximise l'information préservée.

$$\begin{aligned} \max_u \text{information} &= \max_u \frac{1}{n} \sum_i (x_i^T u)^2 = \max_u \frac{1}{n} \sum_i (x_i^T u) (x_i^T u) \\ &= \max_u \frac{1}{n} u^T \left(\sum_i x_i x_i^T \right) u = \max_u u^T C u \end{aligned}$$

où $C = \frac{1}{n} \sum_i x_i x_i^T$ est la matrice de covariance.

Cela revient à résoudre le problème (car $u^T u - 1 = 0$):

$$\max_u u^T C u - \lambda (u^T u - 1)$$

puis en différenciant :

$$2Cu - 2\lambda u = 0 \Leftrightarrow Cu = -\lambda u$$

donc le vecteur u qui préserve le plus d'information est un vecteur propre de la matrice de covariance !

Et on voit que la quantité d'information préservée ($u^T Cu$) est égale à λ ! donc on va prendre le vecteur avec la plus grande valeur propre. Une fois qu'on a trouvé le vecteur propre (par exemple $u = [0.5, 0.6, 0.4]$), ça nous donne les poids de chaque colonne pour le graphe.

🔍 Quelles étapes ?

1. Centrer les données

On centre chaque colonne, on soustrait la moyenne de chaque variable à toutes les valeurs de cette variable. On obtient une matrice centrée de taille $n \times d$.

2. Calcul de la **matrice de covariance**

$$\text{Cov}(X) = \frac{1}{n} X_{\text{centered}}^T X_{\text{centered}}$$

Résultat : une matrice $d \times d$ (une case pour chaque paire de variables).

3. On calcule les vecteurs propres de la matrice.

Chaque vecteur propre pointe vers une **direction importante** dans l'espace des variables (ex. une direction combinant PIB + espérance de vie).

Les valeurs propres disent **quelle proportion de la variance totale** est expliquée par chaque axe.

🔄 Dans le sens inverse

On peut trouver le point projeté dans une dimension plus faible $y = W^T (x - \mu_x)$ mais aussi repartir du point projeté pour essayer de retrouver le point d'origine $x = \mu_x + Wy$ (avec perte de données **parce que W n'est pas une matrice carrée** en général, sauf si on choisit **tous** les eigenvectors).

Linear Discriminant Analysis

Maximise explicitement la variance inter-classes et minimise la variance intra-classes pour améliorer la séparation des classes.

② within-class scatter matrix

$$S_W = \sum_{c=1}^C \sum_{i \in c} (x_i - \mu_c) (x_i - \mu_c)^T$$

avec $E_W(w_1) = w_1 S_W w_1$

② between-class scatter matrix

$$S_B = \sum_{c=1}^C N_c (\mu_c - \text{moyenne des } x) (\mu_c - \text{moyenne des } x)^T$$

avec $E_B(w_1) = w_1 S_B w_1$

On veut minimiser E_W et maximiser E_B

Auto-encoder

Il prend un input, le map dans un plus petit espace, et essaye de le retrouver (donc pas comme PCA maximise pas forcément la variance).