

Apprentissage non-supervisé

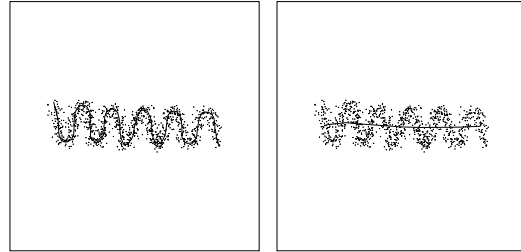
1

- Donnée "crue" – pas de classe: $\mathcal{X}_n = \{\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n\}$
- Variantes, synonymes, aspects
 - estimation de densité
 - extraction de traits
 - réduction de dimensionnalité
 - compression de donnée
 - clustering
 - visualisation

Apprentissage non-supervisé

2

- Deux critères en compétition
 - représentation **fidèle** – préservation d'information
 - représentation **concise** – compression



Apprentissage non-supervisé

3

- Méthodes de base
 - projection sur un **sous-espace linéaire** → ACP
 - projection sur un **ensemble fini de points** → k-moyennes

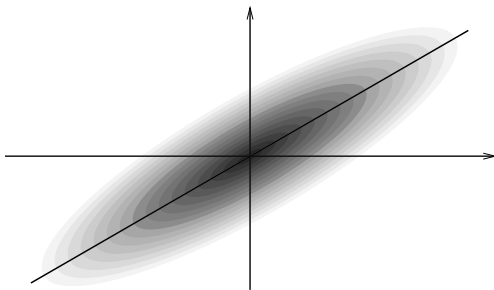
Apprentissage non-supervisé

4

- **Analyse en composantes principales (ACP)** (transformation de Karhunen-Loève)
 - Trouver le sous-espace linéaire qui **maximise la variance des projections**
 - Trouver le sous-espace linéaire qui **minimise la distance entre les points et leur projection**

Apprentissage non-supervisé

5



Apprentissage non-supervisé

6

- ACP
 - $\mathbf{X} = (X_1, \dots, X_d)$: **observation** aléatoire, $E[\mathbf{X}] = \mathbf{0}$, $\text{Var}[\mathbf{X}] < \infty$
 - $\mathbf{u} \in \mathbb{R}^d$: **vecteur d'unité** arbitraire
 - $s(t) = t\mathbf{u}$: **ligne droite** qui correspond à \mathbf{u}
 - $Y = t_s(\mathbf{X}) = \mathbf{X}'\mathbf{u}$: **l'indice de projection** de \mathbf{X} à s
 - $s(t_s(\mathbf{X})) = s(\mathbf{X}'\mathbf{u})$: **point de projection** de \mathbf{X} à s

• ACP

- $E[\mathbf{X}] = \mathbf{0} \implies E[Y] = E[\mathbf{X}'\mathbf{u}] = 0$
- variance de Y :

$$\begin{aligned}\text{Var}[Y] &= E[(\mathbf{X}'\mathbf{u})^2] = E[(\mathbf{u}'\mathbf{X})(\mathbf{X}'\mathbf{u})] \\ &= \mathbf{u}'E[\mathbf{X}\mathbf{X}']\mathbf{u} = \mathbf{u}'\mathbf{R}\mathbf{u} \\ &= \psi(\mathbf{u})\end{aligned}$$
- $\mathbf{R} = E[(\mathbf{X} - E[\mathbf{X}])(\mathbf{X} - E[\mathbf{X}])'] = E[\mathbf{X}\mathbf{X}']$: matrice de covariance
- $\mathbf{R}_{ij} = E[X_i X_j]$
- \mathbf{R} est symétrique $\implies \mathbf{R} = \mathbf{R}'$,
- $\forall \mathbf{v}, \mathbf{w} \in \mathbb{R}^d: \mathbf{v}'\mathbf{R}\mathbf{w} = \mathbf{w}'\mathbf{R}\mathbf{v}$

• ACP

- objectif: maximiser $\text{Var}[Y] = \psi(\mathbf{u}) = \mathbf{u}'\mathbf{R}\mathbf{u}$ par rapport à \mathbf{u}
- considérer une petite perturbation $\delta\mathbf{u}$ de \mathbf{u}
- telle que $\|\mathbf{u} + \delta\mathbf{u}\| = 1$:

$$\begin{aligned}\psi(\mathbf{u} + \delta\mathbf{u}) &= (\mathbf{u} + \delta\mathbf{u})'\mathbf{R}(\mathbf{u} + \delta\mathbf{u}) \\ &= \mathbf{u}'\mathbf{R}\mathbf{u} + 2(\delta\mathbf{u})'\mathbf{R}\mathbf{u} + (\delta\mathbf{u})'\mathbf{R}\delta\mathbf{u}\end{aligned}$$

• ACP

- ignorer le terme d'ordre 2:

$$\begin{aligned}\psi(\mathbf{u} + \delta\mathbf{u}) &= \mathbf{u}'\mathbf{R}\mathbf{u} + 2(\delta\mathbf{u})'\mathbf{R}\mathbf{u} \\ &= \psi(\mathbf{u}) + 2(\delta\mathbf{u})'\mathbf{R}\mathbf{u}\end{aligned}$$
- si $\psi(\mathbf{u})$ est stationnaire:

$$\psi(\mathbf{u} + \delta\mathbf{u}) = \psi(\mathbf{u})$$
- donc

$$(\delta\mathbf{u})'\mathbf{R}\mathbf{u} = 0$$

• ACP

- puisque $\|\mathbf{u} + \delta\mathbf{u}\|^2 = \|\mathbf{u}\|^2 + 2(\delta\mathbf{u})'\mathbf{u} + \|\delta\mathbf{u}\|^2 = 1$:

$$(\delta\mathbf{u})'\mathbf{u} = 0$$
- $\delta\mathbf{u}$ est orthogonal à \mathbf{u}
- l'équation à résoudre:

$$(\delta\mathbf{u})'\mathbf{R}\mathbf{u} - l(\delta\mathbf{u})'\mathbf{u} = 0$$
- également

$$\begin{aligned}(\delta\mathbf{u})'(\mathbf{R}\mathbf{u} - l\mathbf{u}) &= 0 \\ \mathbf{R}\mathbf{u} &= l\mathbf{u}\end{aligned}$$

• ACP

- les solutions l_1, \dots, l_d : valeurs propres
- les solutions $\mathbf{u}_1, \dots, \mathbf{u}_d$: vecteurs propres
- $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_d]$
- simplification: les valeurs propres sont toutes différentes: $l_i \neq l_j$ si $i \neq j$
- trier les valeurs propres: $l_1 > \dots > l_d$
- les vecteurs propres forment une base orthonormale:

$$\begin{aligned}0 &= (\mathbf{u}_i\mathbf{R}\mathbf{u}_j - \mathbf{u}_j\mathbf{R}\mathbf{u}_i) = (\mathbf{u}_i\mathbf{R}\mathbf{u}_j - \mathbf{u}_j\mathbf{R}\mathbf{u}_i) = (\mathbf{u}_i, \mathbf{u}_j - \mathbf{u}_j, l_i\mathbf{u}_i) \\ &= (l_j - l_i)(\mathbf{u}_i\mathbf{u}_j)\end{aligned}$$

• ACP

- le résultat:

$$\begin{aligned}\max_{\|\mathbf{u}\|=1} \psi(\mathbf{u}) &= l_1 \\ \arg \max_{\|\mathbf{u}\|=1} \psi(\mathbf{u}) &= \mathbf{u}_1\end{aligned}$$
- les lignes de composantes principales: $s_i(t) = t\mathbf{u}_i, i = 1, \dots, d$
- les composantes principales: $t_i = \mathbf{u}_i\mathbf{x}, i = 1, \dots, d$
- l'analyse en composantes principales: $\mathbf{t} = \mathbf{U}'\mathbf{x}$
- reconstruction: $\mathbf{x} = (\mathbf{U}')^{-1}\mathbf{t} = \mathbf{U}\mathbf{t} = \sum_{i=1}^d t_i\mathbf{u}_i$

• ACP

• soit $\mathbf{X}' = \sum_{i=1}^{d'} t_i \mathbf{u}_i$

• $S_{d'}$ maximise la variance de \mathbf{X}' :

$$E[\mathbf{X}'^2] = \sum_{i=1}^{d'} \psi(\mathbf{u}_i) = \sum_{i=1}^{d'} l_{i3}$$

• $S_{d'}$ minimise la variance de $\mathbf{X} - \mathbf{X}'$:

$$E[(\mathbf{X} - \mathbf{X}')^2] = \sum_{i=d'+1}^d \psi(\mathbf{u}_i) = \sum_{i=d'+1}^d l_{ij}$$

• ACP

• estimation: $\mathcal{X}_n = \{\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n\}$

• matrice de covariance d'échantillon:

$$\hat{\mathbf{R}}_n = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i'$$

• les solutions $\hat{\mathbf{u}}_1, \dots, \hat{\mathbf{u}}_d$: vecteurs propres

• algorithme naïf: trouver les vecteurs propres – $T = O(nd^3)$

• techniques sophistiquées: $T = O(nd^2)$

• algorithmes itératifs: $T = O(nds)$

• ACP

• algorithmes itératifs

```

ACPIITERATIVE( $\mathcal{X}_n$ )
1  $\mathbf{s}^{(0)}(t) \leftarrow t\mathbf{u}^{(0)}$  une ligne arbitraire
2 faire
3   Projection
4   Espérance
5 jusqu'à changement < seuil
    
```

• ACP

• algorithme de Roweis-Tipping-Bishop

• fixer les indices de projection et minimiser

$$\begin{aligned} \Delta_n(\mathbf{s} | \mathbf{t}^{(j)}) &= \sum_{i=1}^n \|\mathbf{x}_i - t_i^{(j)} \mathbf{u}\|^2 \\ &= \sum_{i=1}^n \|\mathbf{x}_i\|^2 + \|\mathbf{u}\|^2 \sum_{i=1}^n (t_i^{(j)})^2 - 2\mathbf{u}' \sum_{i=1}^n t_i^{(j)} \mathbf{x}_i \end{aligned}$$

• le résultat de la minimisation:

$$\mathbf{u}^{(j+1)} = \arg \min_{\|\mathbf{u}\|=1} \Delta(\mathbf{s} | \mathbf{t}^{(j)}) = \frac{\sum_{i=1}^n t_i^{(j)} \mathbf{x}_i}{\left\| \sum_{i=1}^n t_i^{(j)} \mathbf{x}_i \right\|}$$

• ACP

• algorithme de Roweis-Tipping-Bishop

```

ROWEISTIPPINGBISHOP( $\mathcal{X}_n$ )
1  $\mathbf{s}^{(0)}(t) \leftarrow t\mathbf{u}^{(0)}$  une ligne arbitraire
2  $j \leftarrow 0$ 
3 faire
4    $\mathbf{t}^{(j)} \leftarrow [t_1^{(j)}, \dots, t_n^{(j)}]'$  ←  $[\mathbf{x}_1' \mathbf{u}^{(j)}, \dots, \mathbf{x}_n' \mathbf{u}^{(j)}]'$ 
5    $\mathbf{u}^{(j+1)} \leftarrow \frac{\sum_{i=1}^n t_i^{(j)} \mathbf{x}_i}{\left\| \sum_{i=1}^n t_i^{(j)} \mathbf{x}_i \right\|}$ , and  $\mathbf{s}^{(j+1)}(t) \leftarrow t\mathbf{u}^{(j+1)}$ 
6    $j \leftarrow j + 1$ 
7 jusqu'à  $\left(1 - \frac{\Delta_n(\mathbf{s}^{(j+1)})}{\Delta_n(\mathbf{s}^{(j)})}\right) < \text{seuil}$ 
    
```

• Quantification vectorielle

• collection des points de code (centres): $C = \{\mathbf{v}_1, \dots, \mathbf{v}_k\} \subset \mathbb{R}^d$

• quantificateur vectoriel de k points: $q: \mathbb{R}^d \rightarrow C$

• partition: $\mathcal{V} = \{V_1, \dots, V_k\}$

$$V_i = q^{-1}(\mathbf{v}_i) = \{\mathbf{x} : q(\mathbf{x}) = \mathbf{v}_i\}$$

• Quantification vectorielle

- mesure de **distorsion**: $\Delta(\mathbf{x}, \hat{\mathbf{x}})$

- le plus souvent

$$\Delta(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2$$

- objectif: **minimiser l'espérance**

$$\Delta(q) = E[\Delta(\mathbf{X}, q(\mathbf{X}))]$$

par rapport à \mathcal{C}

- q^* est **globalement optimal** si $\Delta(q^*) \leq \Delta(q)$

- q^* est **très difficile à trouver!!!**

• Quantification vectorielle

- optimalité **locale**

• Condition du **plus proche voisin**

- étant donné \mathcal{C} , $\mathcal{V} = \{V_1, \dots, V_k\}$ est **optimal** si

$$V_\ell = \{\mathbf{x} : \Delta(\mathbf{x}, \mathbf{v}_\ell) \leq \Delta(\mathbf{x}, \mathbf{v}_m), m = 1, \dots, k\}$$

- V_ℓ est la **région de Voronoi** de \mathbf{v}_ℓ

• Condition de **centroïde**

- étant donné \mathcal{V} , $\mathcal{C} = \{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ est **optimal** si

$$\mathbf{v}_\ell = \arg \min_{\mathbf{v}} E[\Delta(\mathbf{X}, \mathbf{v}) | \mathbf{X} \in V_\ell]$$

- distorsion quadratique ($\Delta(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2$):

$$\mathbf{v}_\ell = E[\mathbf{X} | \mathbf{X} \in V_\ell]$$

• Quantification vectorielle

- algorithme de **Max-Lloyd** (k -moyennes)

- fixer \mathcal{C} et optimiser \mathcal{V}

- fixer \mathcal{V} et optimiser \mathcal{C}

- jusqu'à **changement** < seuil

• Quantification vectorielle

- algorithme de **Max-Lloyd** (k -moyennes)

MAXLLOYD(\mathbf{X})

1 $\mathcal{C}^{(0)} \leftarrow \{\mathbf{v}_1^{(0)}, \dots, \mathbf{v}_k^{(0)}\}, j \leftarrow 0$

2 **faire**

3 **pour** $\ell \leftarrow 1$ à k **faire**

4 $V_\ell^{(j)} \leftarrow \{\mathbf{x} : \Delta(\mathbf{x}, \mathbf{v}_\ell^{(j)}) \leq \Delta(\mathbf{x}, \mathbf{v}_m^{(j)}), m = 1, \dots, k\}$

5 **pour** $\ell \leftarrow 1$ à k **faire**

6 $\mathbf{v}_\ell^{(j+1)} \leftarrow \arg \min_{\mathbf{v}} E[\Delta(\mathbf{X}, \mathbf{v}) | \mathbf{X} \in V_\ell^{(j)}] \leftarrow E[\mathbf{X} | \mathbf{X} \in V_\ell^{(j)}]$

7 $j \leftarrow j + 1$

8 **jusqu'à** $\left(1 - \frac{\Delta(q^{(j+1)})}{\Delta(q^{(j)})}\right) < \text{seuil}$

• Quantification vectorielle

- algorithme de **Max-Lloyd** (k -moyennes) pour $\mathcal{X}_n = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$

- $\hat{V}_\ell = V_\ell \cap \mathcal{X}_n, n_\ell = |\hat{V}_\ell|$

- **distorsion empirique**:

$$\Delta_n(q) = \frac{1}{n} \sum_{i=1}^n \Delta(\mathbf{x}_i, q(\mathbf{x}_i)) = \frac{1}{n} \sum_{\ell=1}^k \sum_{\mathbf{x} \in \hat{V}_\ell} \|\mathbf{v}_\ell - \mathbf{x}\|^2$$

• Quantification vectorielle

- algorithme de Max-Lloyd (k -moyennes) pour $\mathcal{X}_n = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$

```

MAXLLOYD( $\mathcal{X}_n$ )
1   $C^{(0)} \leftarrow \{\mathbf{v}_1^{(0)}, \dots, \mathbf{v}_k^{(0)}\}, j \leftarrow 0$ 
2  faire
3    pour  $\ell \leftarrow 1$  à  $k$  faire
4       $V_\ell^{(j)} \leftarrow \{\mathbf{x} : \Delta(\mathbf{x}, \mathbf{v}_\ell^{(j)}) \leq \Delta(\mathbf{x}, \mathbf{v}_m^{(j)}), m = 1, \dots, k\}$ 
5    pour  $\ell \leftarrow 1$  à  $k$  faire
6       $\mathbf{v}_\ell^{(j+1)} \leftarrow \arg \min_{\mathbf{v}} \sum_{\mathbf{x} \in V_\ell^{(j)}} \Delta(\mathbf{x}, \mathbf{v}) \leftarrow \frac{1}{n_\ell} \sum_{\mathbf{x} \in V_\ell^{(j)}} \mathbf{x}$ 
7     $j \leftarrow j + 1$ 
8  jusqu'à  $\left(1 - \frac{\Delta_n(q^{(j+1)})}{\Delta_n(q^{(j)})}\right) < \text{seuil}$ 

```

```

MAXLLOYDÉNIGNE( $\mathcal{X}_n$ )
1   $C^{(0)} \leftarrow \{\mathbf{v}_1^{(0)}, \dots, \mathbf{v}_k^{(0)}\}$ 
2   $j \leftarrow 0$ 
3  faire
4    pour  $i \leftarrow 1$  à  $n$  faire
5      si  $\exists v_\ell : \|\mathbf{x}_i - \mathbf{v}_\ell^{(j)}\| < \|\mathbf{x}_i - \mathbf{v}_{[x_i]}^{(j)}\|$ 
6         $\mathbf{v}_\ell^{(j+1)} \leftarrow \frac{\mathbf{v}_\ell^{(j)} n_\ell^{(j)} + \mathbf{x}_i}{n_\ell^{(j)} + 1}$ 
7         $\mathbf{v}_{[x_i]}^{(j+1)} \leftarrow \frac{\mathbf{v}_{[x_i]}^{(j)} n_{[x_i]}^{(j)} - \mathbf{x}_i}{n_{[x_i]}^{(j)} - 1}$ 
8         $V_{[x_i]}^{(j+1)} \leftarrow V_\ell^{(j)}$ 
9     $j \leftarrow j + 1$ 
10 jusqu'à il y a un changement

```

• Quantification vectorielle

- algorithme de Max-Lloyd (k -moyennes), version en-ligne

- \mathbf{x}_i appartient à $V_{[x_i]}$

- \mathbf{x}_i change de $V_{[x_i]}$ à V_j :

$$\mathbf{v}_\ell^{(j+1)} = \frac{\mathbf{v}_\ell^{(j)} n_\ell^{(j)} + \mathbf{x}_i}{n_\ell^{(j)} + 1}; \quad \mathbf{v}_{[x_i]}^{(j+1)} = \frac{\mathbf{v}_{[x_i]}^{(j)} n_{[x_i]}^{(j)} - \mathbf{x}_i}{n_{[x_i]}^{(j)} - 1}$$

• Quantification vectorielle

