Apprentissage non-supervisé

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

Apprentissage non-supervisé

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

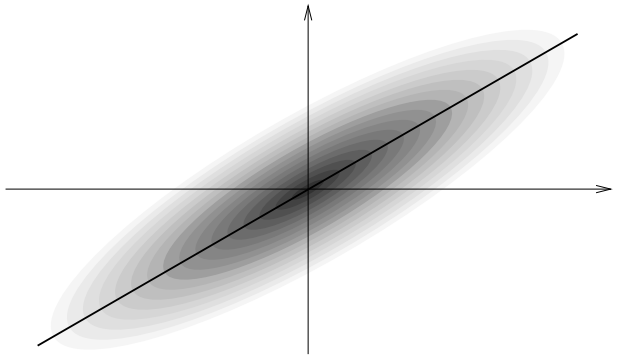
Apprentissage non-supervisé

- Méthodes de base
  - projection sur un sous-espace linéaire $\longrightarrow$ ACP
  - projection sur un ensemble fini de points $\longrightarrow$ k-moyennes

Apprentissage non-supervisé

- 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



Apprentissage non-supervisé

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

Apprentissage non-supervisé

- ACP
  - $E[\mathbf{X}] = \mathbf{0} \Longrightarrow E[Y] = E[\mathbf{X}^t\mathbf{u}] = 0$
  - variance de Y:

$$\begin{aligned} \mathrm{Var}[Y] &= E[(\mathbf{X}^t\mathbf{u})^2] = E[(\mathbf{u}^t\mathbf{X})(\mathbf{X}^t\mathbf{u})] \\ &= \mathbf{u}^t E[\mathbf{X}\mathbf{X}^t]\mathbf{u} = \mathbf{u}^t\mathbf{R}\mathbf{u} \\ &= \psi(\mathbf{u}) \end{aligned}$$

  - $\mathbf{R} = E\left[(\mathbf{X} - E[\mathbf{X}])(\mathbf{X} - E[\mathbf{X}])^t\right] = E\left[\mathbf{X}\mathbf{X}^t\right]$: matrice de covariance
  - $\mathbf{R}_{ij} = E\left[X_i X_j\right]$
  - R est symétrique $\Longrightarrow \mathbf{R} = \mathbf{R}^t$,
  - $\forall \mathbf{v}, \mathbf{w} \in \mathbb{R}^d$: $\mathbf{v}^t\mathbf{R}\mathbf{w} = \mathbf{w}^t\mathbf{R}\mathbf{v}$

Apprentissage non-supervisé

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

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

- ACP
  - ignorer le terme d'ordre 2:

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

  - si $\psi(\mathbf{u})$ est stationnaire:

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

  - donc

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

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

$$(\delta\mathbf{u})^t\mathbf{u}=0$$

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

$$(\delta\mathbf{u})^t\mathbf{R}\mathbf{u}-l(\delta\mathbf{u})^t\mathbf{u}=0$$

  - également

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

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

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

- 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: $\mathbf{s}_i(t)=t\mathbf{u}_i, i=1,\ldots,d$
  - les composantes principales: $t_i=\mathbf{u}_i\mathbf{x}, i=1,\ldots,d$
  - l'analyse en composantes principales: $\mathbf{t}=\mathbf{U}^t\mathbf{x}$
  - reconstruction: $\mathbf{x}=(\mathbf{U}^t)^{-1}\mathbf{t}=\mathbf{U}\mathbf{t}=\sum_{i=1}^{d}t_i\mathbf{u}_i$

- ACP
  - soit $\mathbf{X}' = \sum_{i=1}^{d'} t_i \mathbf{u}_i$
  - $S_{d'}$ maximise la variance de $\mathbf{X}'$:

$$E\left[\mathbf{X}'^2\right] = \sum_{i=1}^{d'} \psi(\mathbf{u}_i) = \sum_{i=1}^{d'} l_i,$$

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

$$E\left[(\mathbf{X} - \mathbf{X}')^2\right] = \sum_{i=d'+1}^{d} \psi(\mathbf{u}_j) = \sum_{i=d'+1}^{d} l_j,$$

- ACP
  - estimation: $X_n = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$
  - matrice de covariance d'échantillon:

$$\widehat{\mathbf{R}}_n = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_n \mathbf{x}_n^t$$

  - les solutions $\widehat{\mathbf{u}}_1, \ldots, \widehat{\mathbf{u}}_d$: 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

ACPITERATIVE(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

$$\begin{aligned}\Delta_n\left(\mathbf{s}\left|\mathbf{t}^{(j)}\right.\right) &= = \sum_{i=1}^{n} \left\|\mathbf{x}_i - t_i^{(j)}\mathbf{u}\right\|^2 \\ &= \sum_{i=1}^{n} \|\mathbf{x}_i\|^2 + \|\mathbf{u}\|^2 \sum_{i=1}^{n} \left(t_i^{(j)}\right)^2 - 2\mathbf{u}^t \sum_{i=1}^{n} t_i^{(j)} \mathbf{x}_i\end{aligned}$$

  - le résultat de la minimisation:

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

- ACP
  - algorithme de Roweis-Tipping-Bishop

ROWEISTIPPINGBISHOP(X_n)

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

- Quantification vectorielle
  - collection des points de code (centres): $C = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\} \subset \mathbb{R}^d$
  - quantificateur vectoriel de k points: $q : \mathbb{R}^d \to C$
  - partition: $V = \{V_1, \ldots, V_k\}$

$$V_\ell = q^{-1}(\mathbf{v}_\ell) = \{\mathbf{x} : q(\mathbf{x}) = \mathbf{v}_\ell\}$$

- 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 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 C, $V = \{V_1, \ldots, V_k\}$ est optimal si

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

  - V_ℓ est la région de Voronoi de $\mathbf{v}_\ell$

- Condition de centroïde
  - étant donné V, $C = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ 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 C et optimiser V
  - fixer V et optimiser C
  - jusqu'à $changement < seuil$

- Quantification vectorielle
  - algorithme de Max-Lloyd (k-moyennes)

MAXLLOYD($\mathbf{X}$)

1 $C^{(0)} \leftarrow \left\{\mathbf{v}_1^{(0)}, \ldots, \mathbf{v}_k^{(0)}\right\}$, $j \leftarrow 0$
2 **faire**
3 **pour** $\ell \leftarrow 1$ **à** k **faire**
4 $V_\ell^{(j)} \leftarrow \left\{\mathbf{x} : \Delta\left(\mathbf{x}, \mathbf{v}_\ell^{(j)}\right) \leq \Delta\left(\mathbf{x}, \mathbf{v}_m^{(j)}\right), m = 1, \ldots, k\right\}$
5 **pour** $\ell \leftarrow 1$ **à** k **faire**
6 $\mathbf{v}_\ell^{(j+1)} \leftarrow \arg\min_{\mathbf{v}} E\left[\Delta(\mathbf{X}, \mathbf{v}) \middle| \mathbf{X} \in V_\ell^{(j)}\right] \leftarrow E\left[\mathbf{X} \middle| \mathbf{X} \in V_\ell^{(j)}\right]$
7 $j \leftarrow j + 1$
8 **jusqu'à** $\left(1 - \frac{\Delta\left(q^{(j+1)}\right)}{\Delta\left(q^{(j)}\right)}\right) < seuil$

- Quantification vectorielle
  - algorithme de Max-Lloyd (k-moyennes) pour $X_n = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$
  - $\widehat{V}_\ell = V_\ell \cap X_n$, $n_\ell = |\widehat{V}_\ell|$
  - distorsion empirique:
$$\Delta_n(q) = \frac{1}{n} \sum_{i=1}^{n} \Delta(\mathbf{x}_i, q(\mathbf{x}_i)) = \frac{1}{n} \sum_{\ell=1}^{k} \sum_{\mathbf{x} \in \widehat{V}_\ell} \|\mathbf{v}_\ell - \mathbf{x}\|^2$$

- Quantification vectorielle
  - algorithme de Max-Lloyd (k-moyennes) pour $X_n = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$

MAXLLOYD(X_n)

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

- Quantification vectorielle
  - algorithme de Max-Lloyd (k-moyennes), version en-ligne
  - $\mathbf{x}_i$ appartient à $V_{(\mathbf{x}_i)}$
  - $\mathbf{x}_i$ change de $V_{(\mathbf{x}_i)}$ à V_ℓ:

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

MAXLLOYDENLIGNE(X_n)

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

- Quantification vectorielle