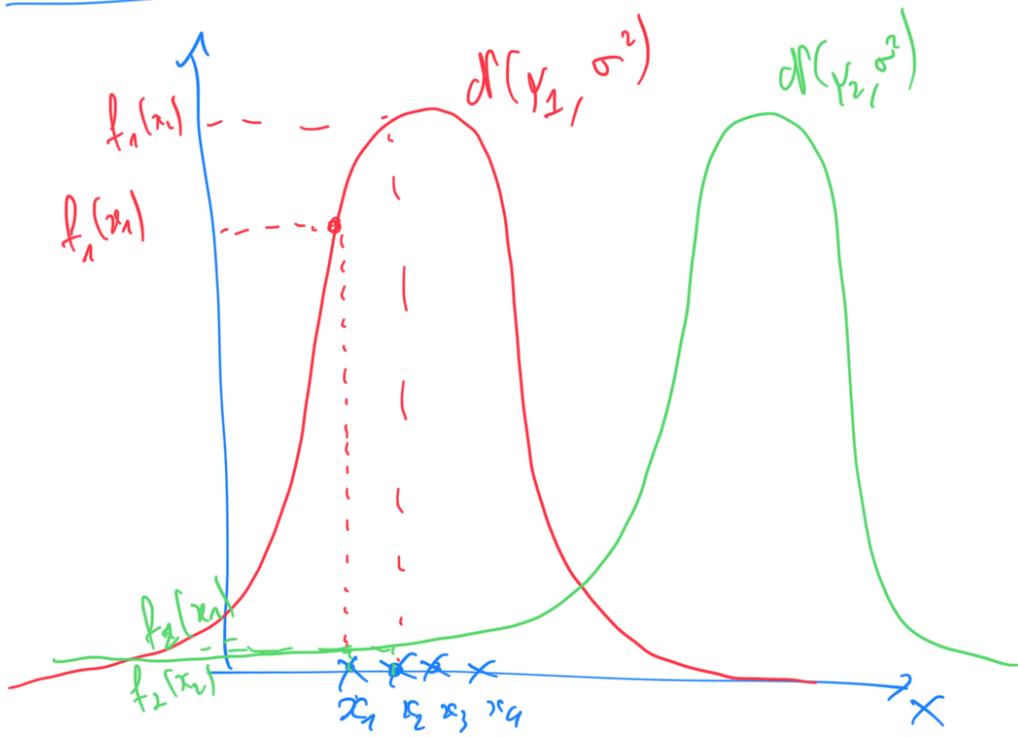


Vraisemblance



$$f_1(x_1) \times f_1(x_2) \times \dots \times f_1(x_4) > f_2(x_1) \times f_2(x_2) \times \dots \times f_2(x_4)$$

Vraisemblance de la
 $d(y_1, \sigma^2)$ pour l'éch
 (x_1, \dots, x_4)

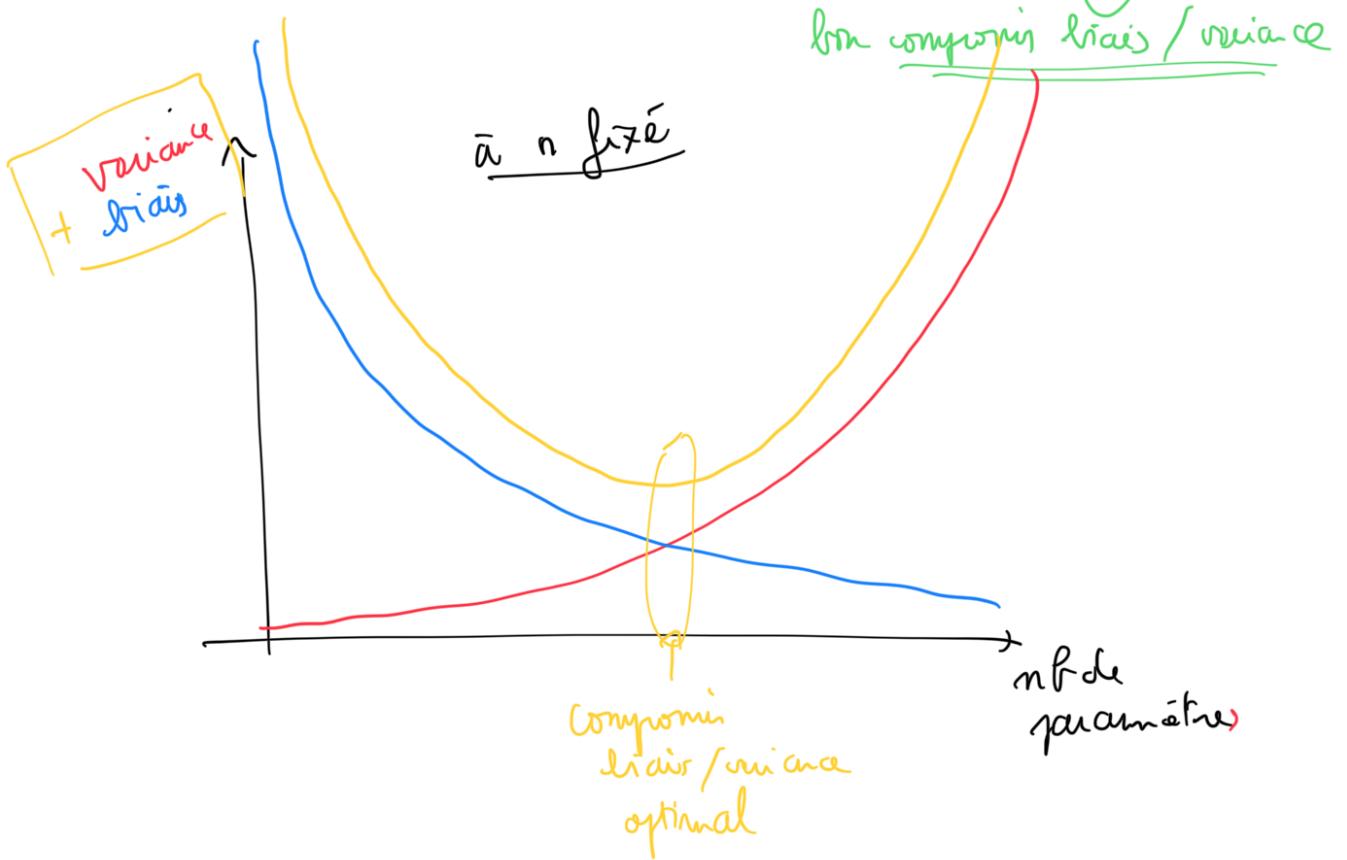
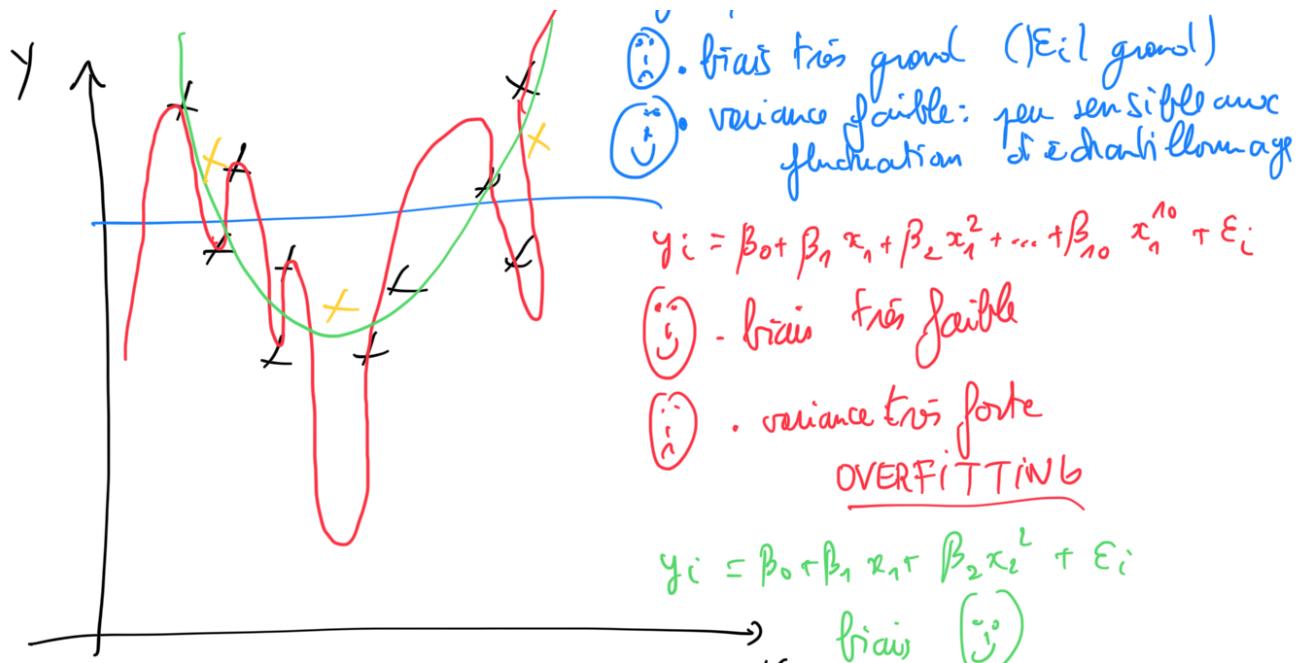
$d(y_2, \sigma^2)$

likelihood

$$\begin{aligned} L(x_1, \dots, x_n; \gamma, \sigma^2) &= \prod_{i=1}^n f(x_i; \gamma, \sigma^2) \\ &= \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\left(\frac{x_i - \gamma}{\sigma}\right)^2\right) \end{aligned}$$

Choix de modèle,

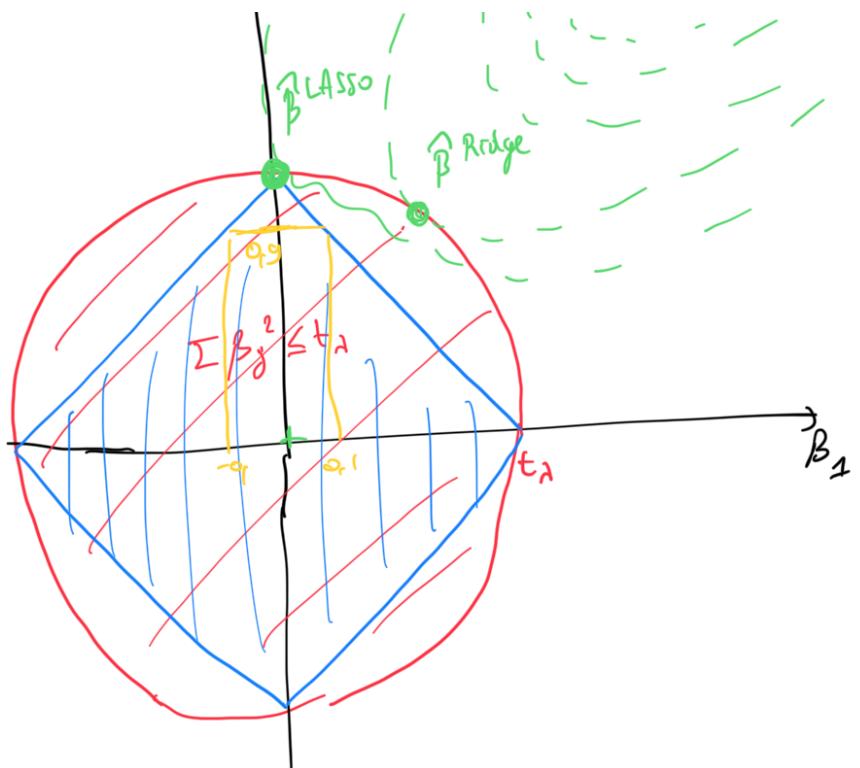
$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$



Regressions ridge et LASSO

$$\hat{\beta}_{\text{Ridge}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^n (y_i - (\beta_0 + \sum_j \beta_j x_{ij}))^2 \right\} \text{ tq } \sum \beta_j^2 \leq t_2$$

$\beta_2 \uparrow$ $\hat{\beta}_{\text{OLS}}$



$$\hat{\beta}^\text{LASSO} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^n (y_i - (\dots))^2 \right\} \text{ tq } \sum |\beta_j| \leq t_\lambda$$

LARS

• x_k le + corrélé avec γ

$$\hat{\beta}_k^\text{OLS} = 3.8$$

$$\Rightarrow \gamma = 0.02 x_k + \varepsilon$$

qui est la corrélation

si x_k

$$\gamma = 0.02 x_k + \varepsilon$$



$$\gamma = 0.03 x_k + \varepsilon$$

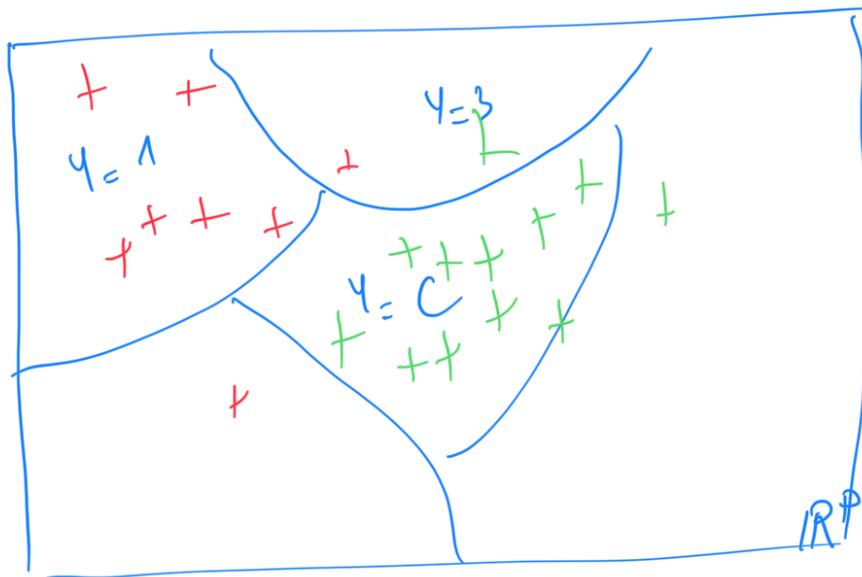
x_j

$$\gamma = 0.03 x_k + 0.01 x_j + \varepsilon$$

\Rightarrow

Classification

output $Y \in \{1, -1\}$
 input $x \in \mathbb{R}^p$

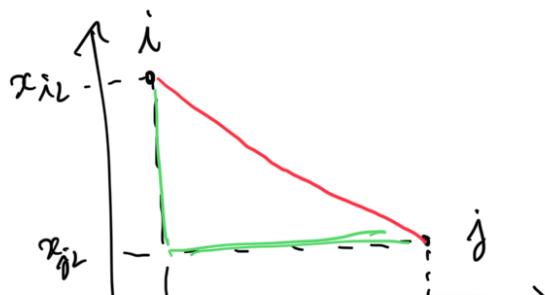


Distance euclidienne

$x_i \in \mathbb{R}^p : (x_{i1}, \dots, x_{ip})$

$x_j \quad (x_{j1}, \dots, x_{jp})$

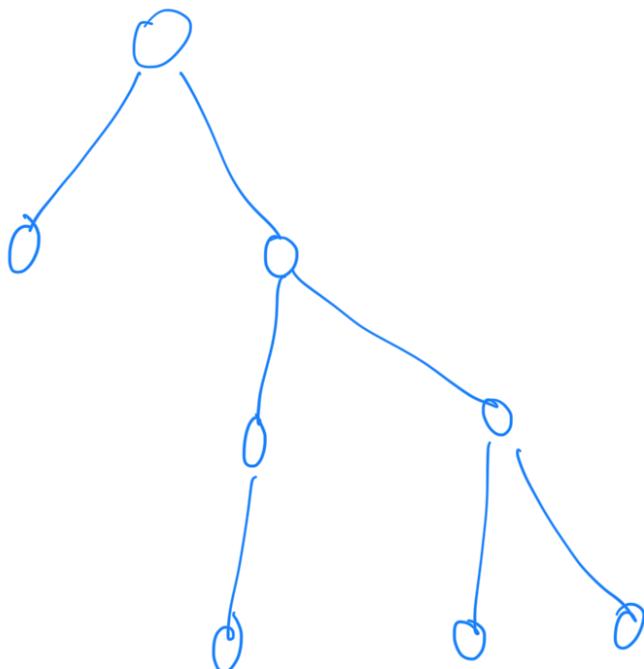
$$d(x_i, x_j) = \sqrt{\sum_{l=1}^p (x_{il} - x_{jl})^2}$$



Toujours normaliser les données avant d'utiliser la distance euclidienne.

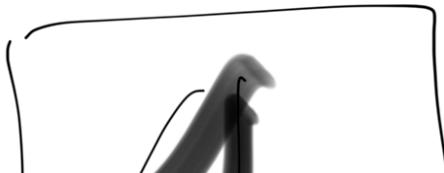


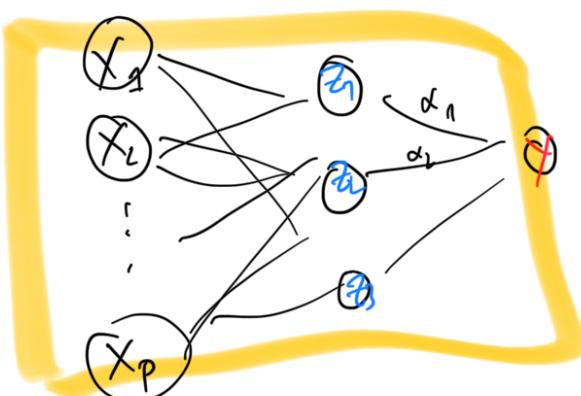
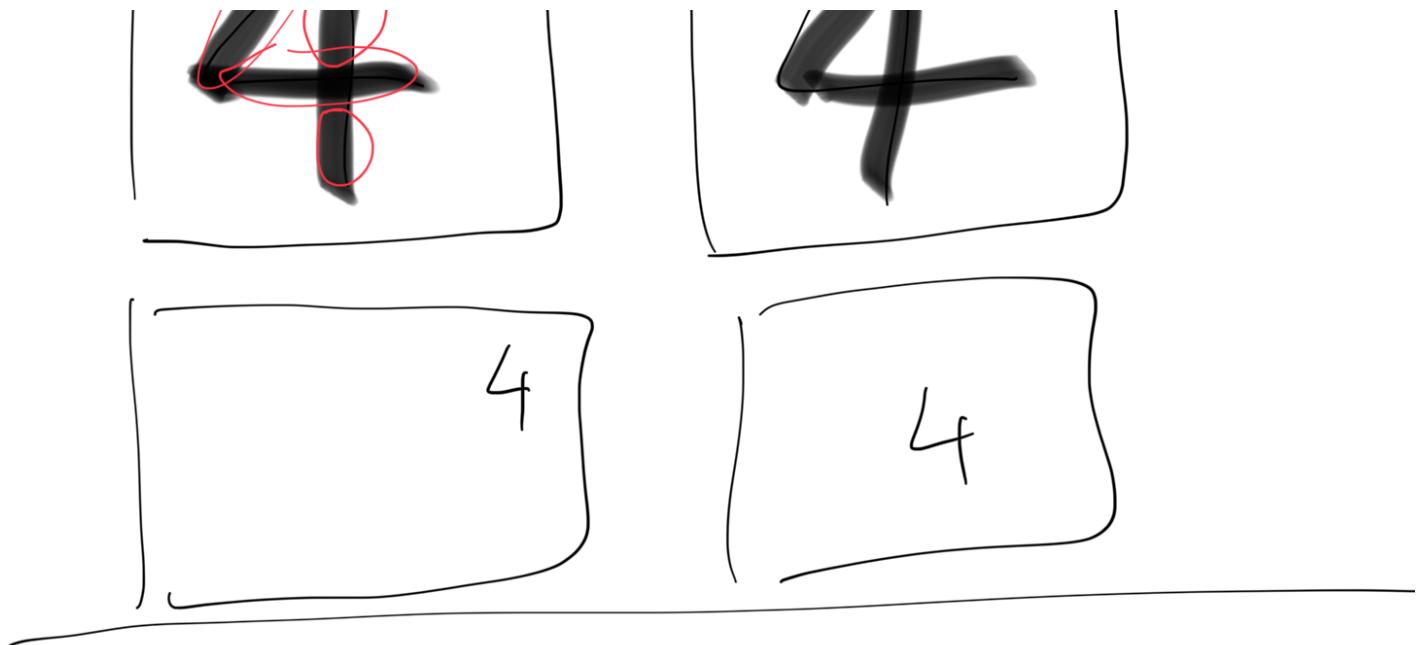
Méthode CART



Exercice MNIST

- extraire les 5000 premiers images comme base d'apprentissage
- les 1000 suivantes comme base de test
- comparer
 - le NN (en réglant k)
 - un arbre
 - une forêt aléatoire





$$Y = g \left(x_0 + \sum_j \alpha_j z_j \right)$$

\uparrow
 $g \left(\alpha_0^{2j} + \sum_e \alpha_e^{2j} x_e \right)$

$$Y = g \left(x_0 + \sum_j \alpha_j g \left(\alpha_0^{2j} + \sum_e \alpha_e^{2j} x_e \right) \right)$$

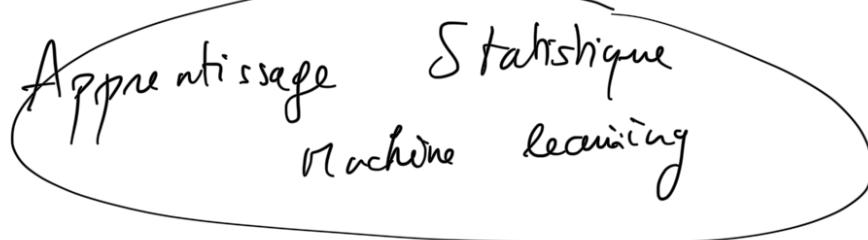
paramètre α ordinaire
hyper paramètre

Quiz J3

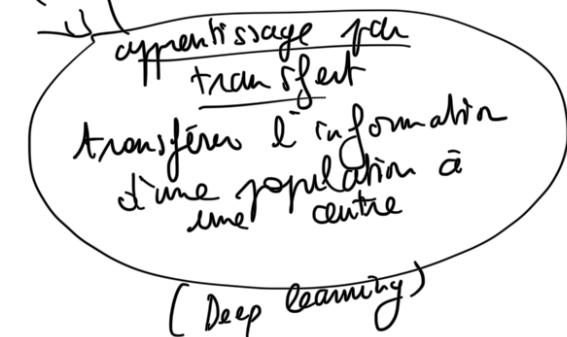
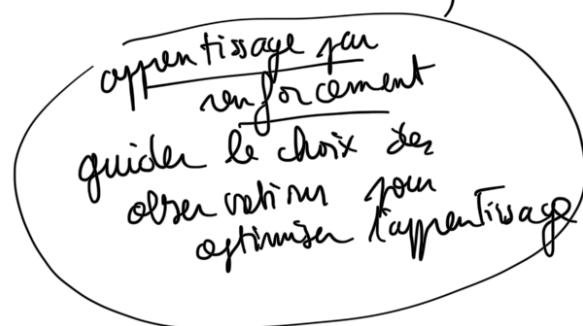
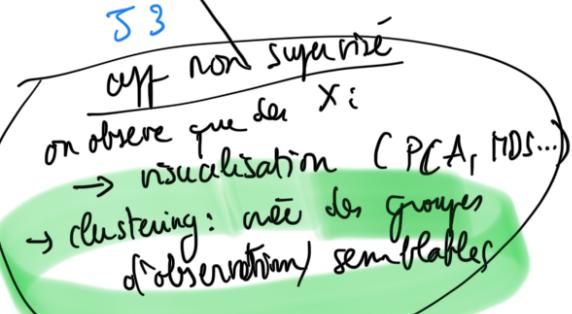
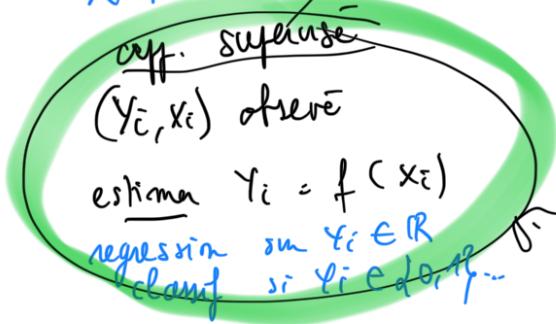
- Que je travaille avec des données quantitatives, des images, des textes, des données catégorielles, finalement je peu toujours me ramener dans un espace quantitatif et utiliser des techniques classiques d'apprentissage :
⚠ au nb de variables
vrai faux
- Lorsque je dispose de beaucoup d'observations : (n grand)
 - je risque de faire de l'overfitting
 - je suis sûr d'avoir de bonnes estimations
 - je me prends qu'une partie des données si elle ne tiennent pas toutes en mémoire sur ma machine NON, on adaptera les moyens informatiques
- Lorsque mon nombre de variables est grand devant le nombre d'observations : (p grand : $p \gg n$)
 - je risque de faire de l'overfitting
 - je peu utiliser une méthode de sélection de variables [oui si p est de l'ordre de 20-30] non si $p \geq 100$
 - je peu utiliser une estimation généralisée de mon modèle (LASSO, Ridge)
 - je peu résumer mes variables à l'aide de métavariables (PCR, PLS, ...)
 - je risque d'observer de la corrélation entre mes variables

- Si je veux prédire une variable continue, je fais :
 - de l'apprentissage non supervisé
 - de l'apprentissage supervisé
 - de la classification
 - de la régression (sauf régression logistique)
mais qui est une méthode de classif
- Dans une analyse de classification, si je suis intéressé uniquement en la qualité de la prédiction, je priviliege :
 - kNN
 - Random Forest
 - arbre de décision
 - SVM si bien réglés
 - réseau de neurones
 - régression logistique
- En régression, si je suis essentiellement intéressé à détecter les variables importantes (parmi $p=100$ variable), je utilise :
 - une sélection forward (faut de l'overfitting)
 - une régression PCR
 - une régression Lasso
 - une régression PLS
 - une régression Ridge
 - une sélection stepwise
- Si en classification je veux détecter les variables influentes, je priviliege :

- Random Forest
 - kNN
 - Reg logistique LASSO
 - Reg logistique Ridge
 - Neural Network
-



2 premiers journées

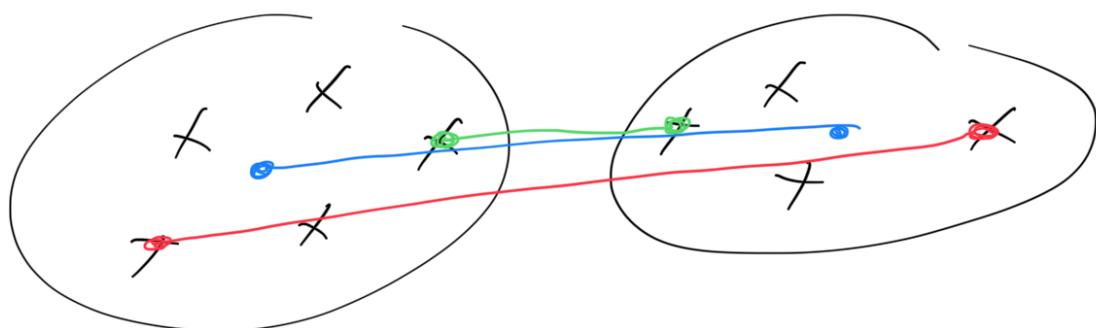


Exercice

faire un clustering sur les données MNIST avec kmeans.

- choisir K
 - représenter les moyennes des clusters
 - représenter les individus (images) dans le plan de l'ACP -
-

distance entre clusters

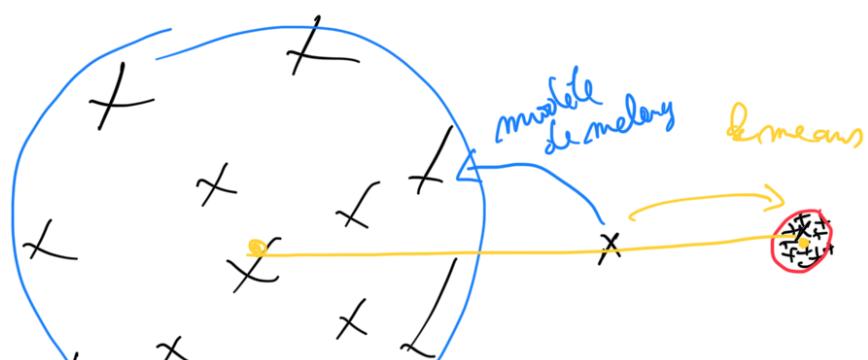


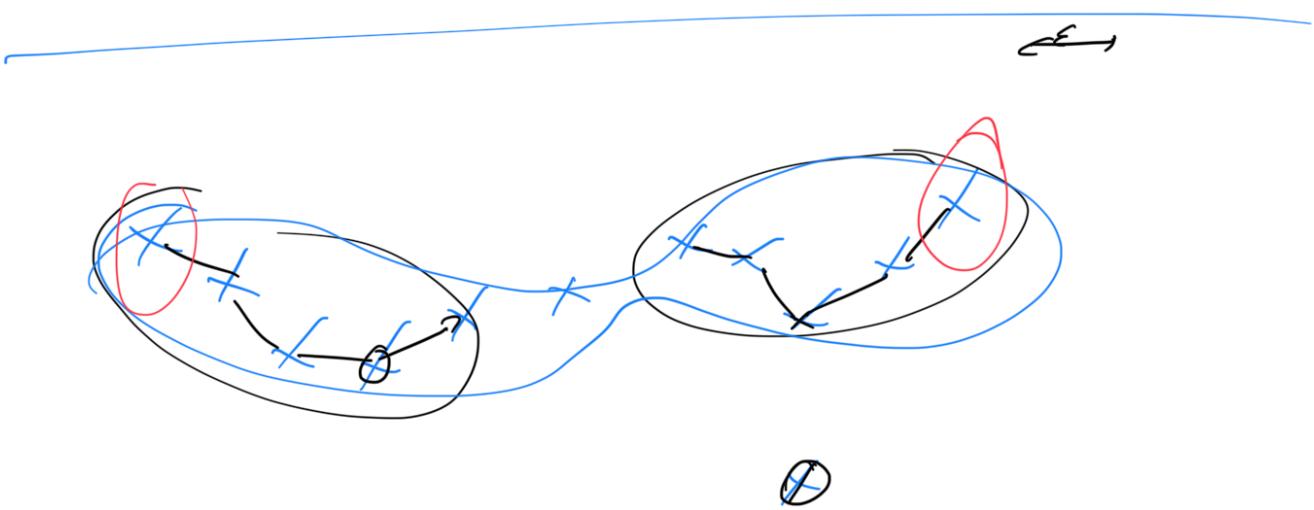
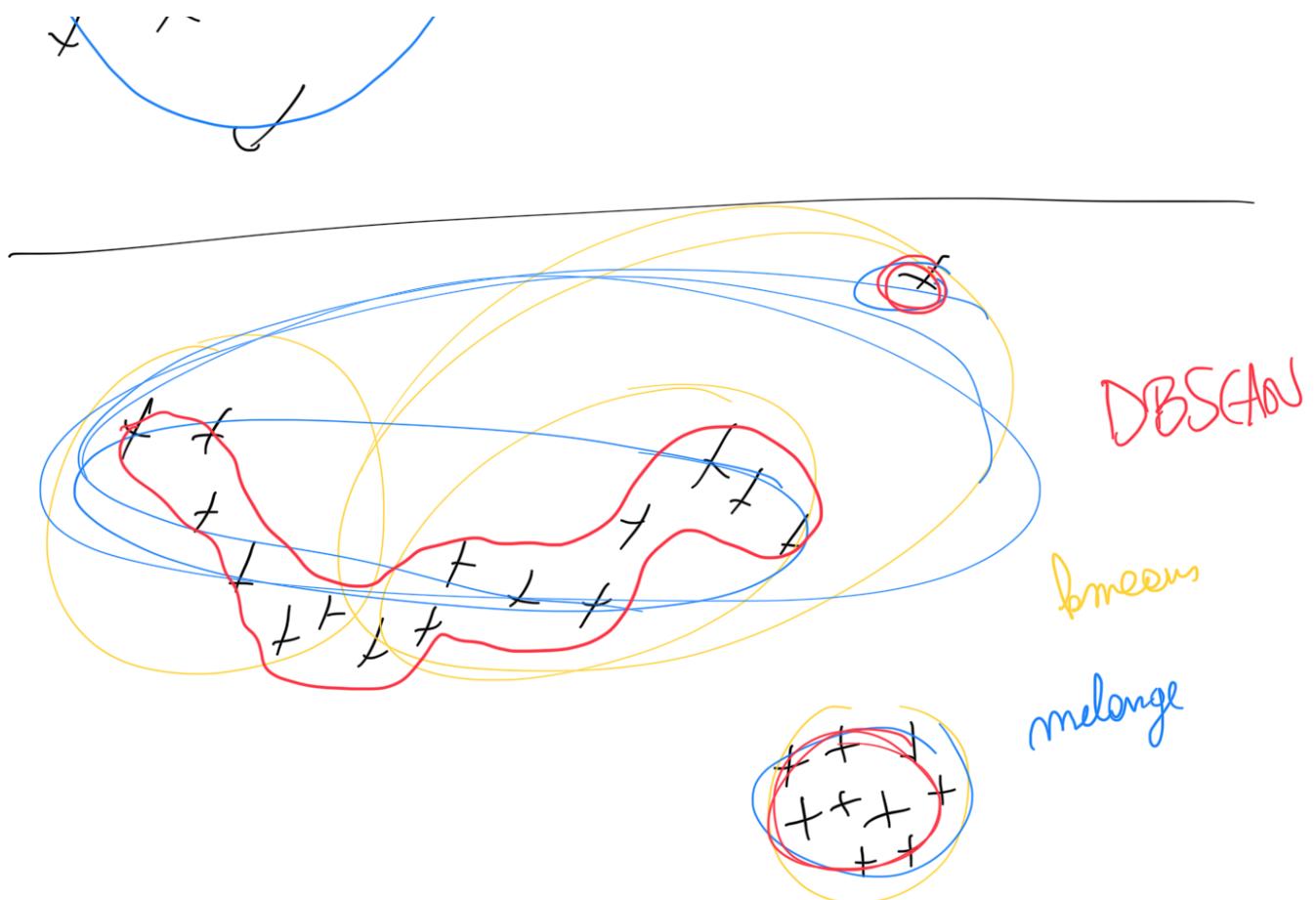
modèle de mélange de loi normale, $p=100$

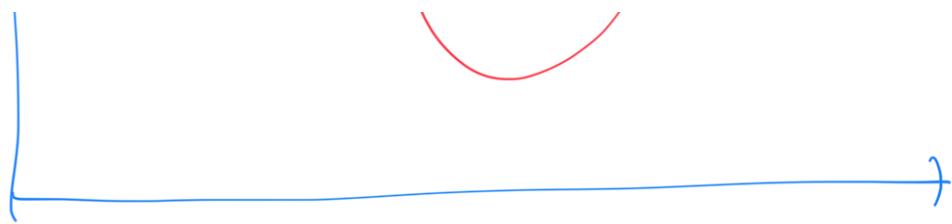
$$\text{cluster } k \sim \mathcal{N}(\gamma_k, \Sigma_k)$$

$$\gamma \in \mathbb{R}^{100}$$

$$\Sigma \in \mathbb{R}^{100 \times 100}$$







$$Y \begin{pmatrix} y_{11} & y_{12} & \dots & y_{1c} \\ | & | & \dots & | \\ y_m & y_{m2} & \dots & y_{mc} \end{pmatrix} = \beta \begin{pmatrix} x_{11} & \dots & x_{1p} \\ | & \dots & | \\ x_{m1} & \dots & x_{mp} \end{pmatrix}$$

$\hookrightarrow C_1^Y, C_2^Y, \dots$

PLS 2 \approx proche de l'analyse canonique