IA : prétraitement des données

Françoise Bouvet

francoise.bouvet@ijclab.in2p3.fr

Les différentes étapes du prétraitement

1. Visualiser et examiner les données pour détecter des corrélations
2. Isoler la(les) variable(s) cible(s) et l'encoder
3. Détecter les données manquantes et y pallier
4. Sélectionner les bonnes variables d'entraînement ou en créer d'autres
5. Normaliser les variables d'entraînement continues
6. Séparer l'ensemble en sous-ensembles (entraînement/validation/test)

Préambule : représentation des données

Entrée X

$$\mathbf{X} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \dots & x_n^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \dots & x_n^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^{(m)} & x_2^{(m)} & x_3^{(m)} & \dots & x_n^{(m)} \end{bmatrix}$$

Sortie Y

$$\mathbf{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

m : nombre d'échantillons
n : nombre de paramètres qui caractérisent chaque échantillon

Ensemble de données : m échantillons

Exemple

Prédire le risque d'attaque cardiaque à partir de l'âge, du sexe, de la pression artérielle, de la glycémie, du taux de cholestérol et du rythme cardiaque.

- L'ensemble de données contient 200 patients -> m = 200
- n = 6
- y prend une des 5 valeurs suivantes : 1, 2, 3, 4, 5 (de risque faible à élevé)

Données catégorielles

- Encodage ordinal
  - Attribution d'une valeur entière à chaque classe
  - Génère un ordre artificiel entre les catégories

  Ex : encoder 3 couleurs : bleu -> 1 ; vert -> 2 ; rouge -> 3

- Encodage « One-hot »
  - La catégorie est représentée par un vecteur de 0, sauf une valeur à 1
  - La taille du vecteur est égale au nombre de classes

  Ex : bleu -> $\begin{pmatrix}1\\0\\0\end{pmatrix}$; vert -> $\begin{pmatrix}0\\1\\0\end{pmatrix}$; rouge -> $\begin{pmatrix}0\\0\\1\end{pmatrix}$

Données continues (1)

Beaucoup d'algorithmes de ML sont très sensibles aux valeurs absolues des valeurs d'entrée alors que l'information cruciale se situe plutôt au niveau des valeurs relatives

- **normalisation** des données en entrée

Centrage et normalisation

$$x_i^m \leftarrow \frac{x_i^m - \mu_i}{\sigma_i}$$

où :

- $X = \{x^m\}, m \in \{1, M\}, x^m \in R^N$
- $\mu_i = \frac{1}{M} \sum_{m=1}^{M} x_i^m$
- $\sigma_i^2 = \frac{1}{M} \sum_{m=1}^{M} (x_i^m - \mu_i)^2$

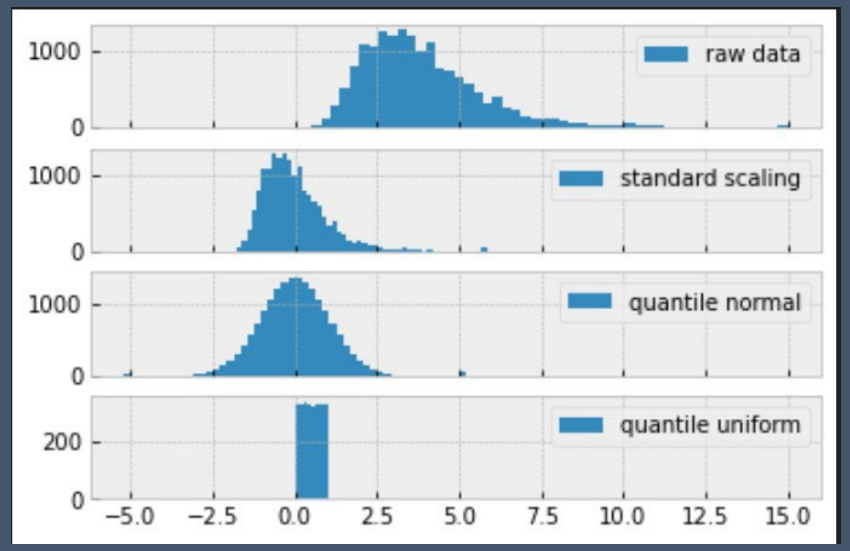
Cas particulier des données **« sparse »** (beaucoup de 0)

- Normaliser sans centrer pour ne pas les rendre « denses » (mémoire)

Données continues (2)

Quantile -> float

Transformation non linéaire des données pour obtenir une distribution spécifique



Discretisation (binning) -> classe

Regroupe les valeurs en k classes (bins)

Techniques classiques :

- **Uniforme** : chaque bin a la même étendue en valeur
- **Quantile** : basé sur les percentiles, chaque bin a le même nombre de valeurs
- **Clustering** : des groupes sont identifiés et des exemples sont assignés à chaque groupe

Problèmes courants : manque de données

Solution usuelle : **augmentation de données**

- Transformations géométriques
  - Rotation
  - Translation
  - Distorsion
  - Modification du contraste
- Modèles génératifs
  - GAN
  - Transformers
  - Variational Auto Encoder

Problèmes courants : données non balancées

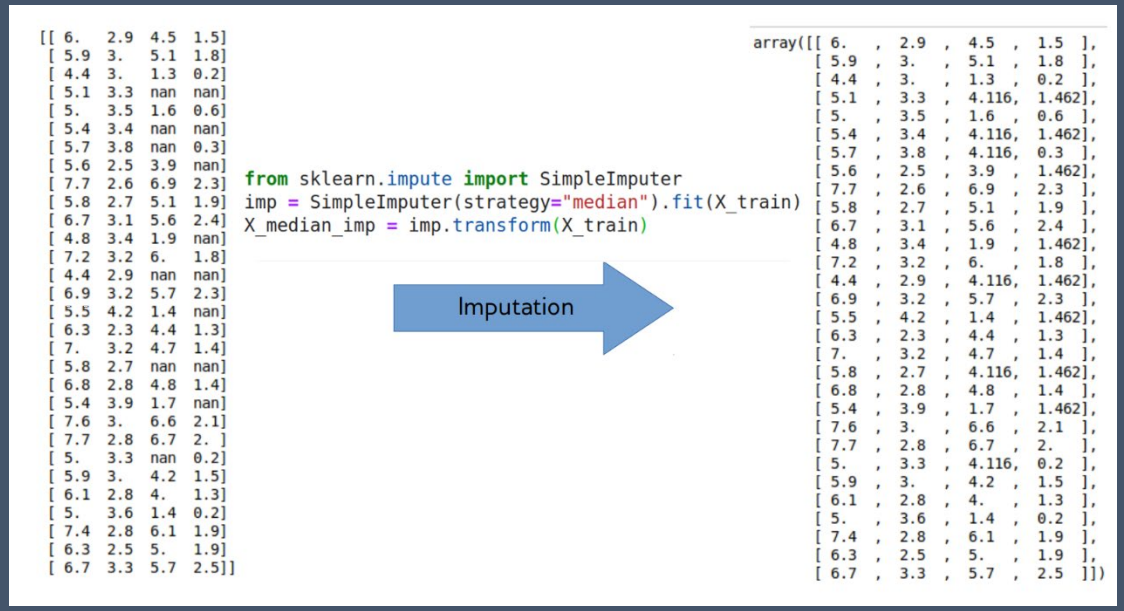
Imbalanced data : grande différence entre la quantité de données dans chaque classe

- Problème classique en **classification**
- Attention à la **métrique**
- Solution usuelle :
  - **Sous-échantillonnage** des classes majoritaires par suppression
  - **Sur-échantillonnage** des classes minoritaires par duplication / synthèse

Problèmes courants : données manquantes

Données manquantes : NaN

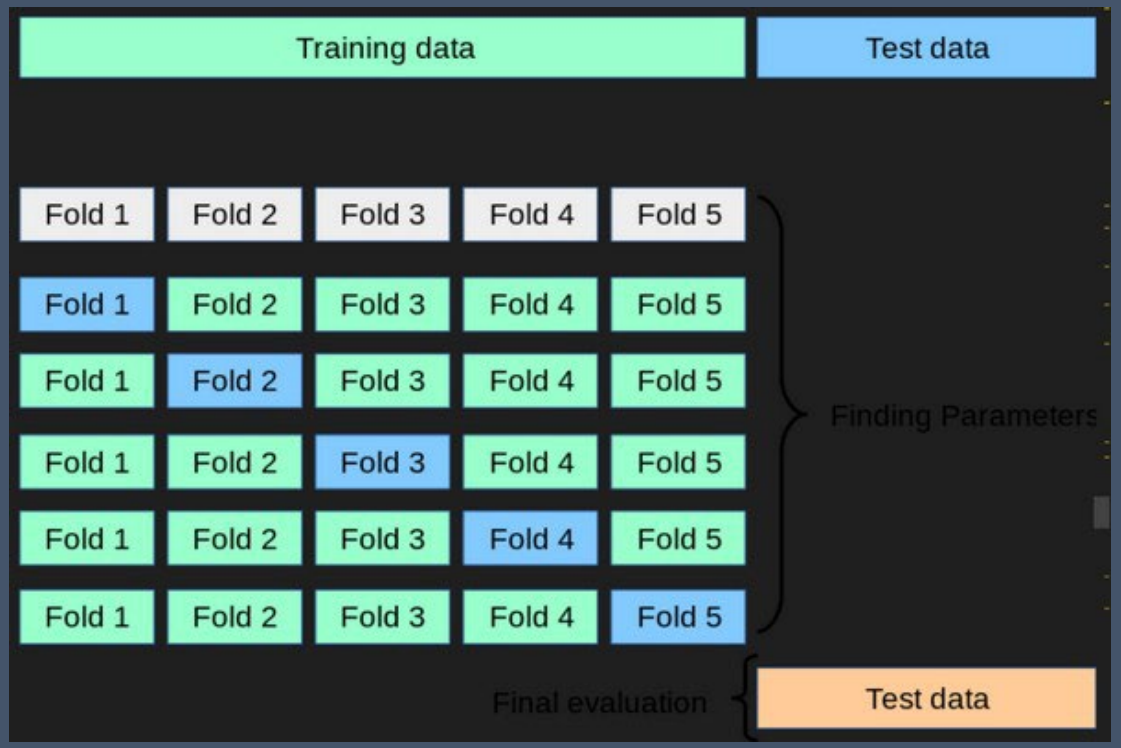
- **Ignorer** l'échantillon
- Faire de **l'imputation** de donnée
  - Technique statistique de remplissage de données manquantes à partir des autres valeurs
  - Ex : catégorie la plus présente, moyenne, médiane, valeur de l'échantillon précédent, etc



Problèmes courants : données bruitées

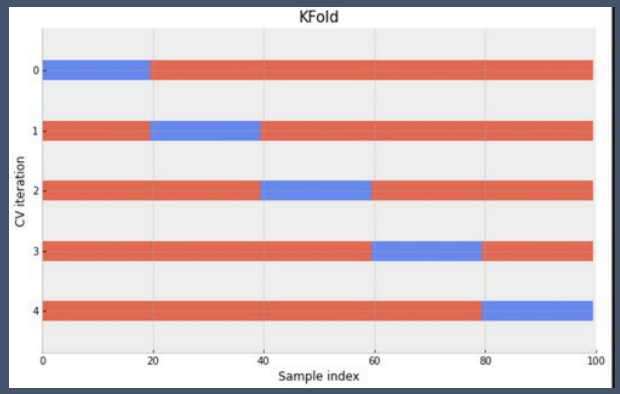
- Discrétisation (binning)
- Clustering
- Supprimer les valeurs aberrantes (outliers)

Cross-validation

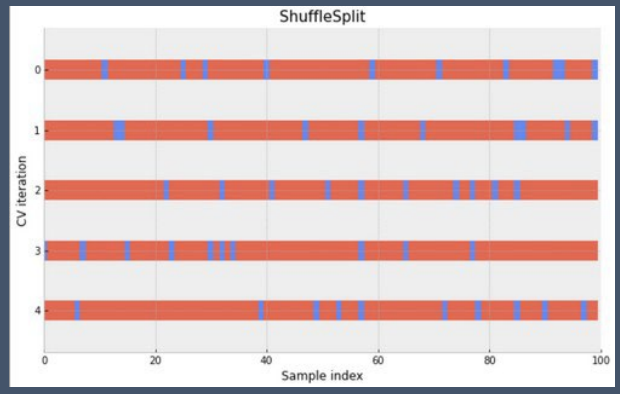


1. Entrainer K modèles avec des jeux de données différents
2. Au choix :
   - Prendre le meilleur des K modèles
   - Prendre le meilleur des K modèles et le ré-entraîner sur l'ensemble des données
   - Garder les K modèles et se fier à l'avis du plus grand nombre

➢ Méthode utilisée en **Machine Learning**

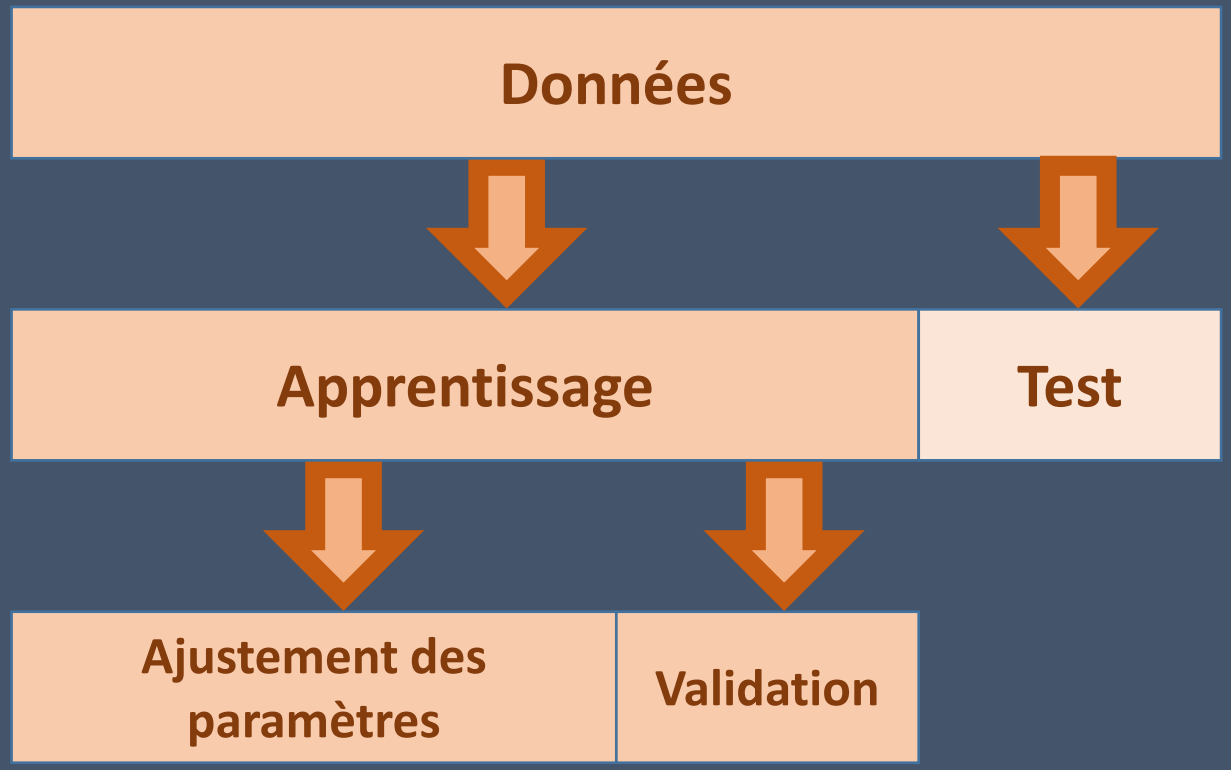


Séparation en KFold



Séparation aléatoire

Séparation des données : test / apprentissage / validation



- L'échantillon d'apprentissage est divisé en 2 : une partie pour ajuster les paramètres du réseau ; une partie « validation » pour évaluer les performances du réseau à chaque itération.
- L'échantillon test sert uniquement à donner les performances finales du modèle.

Typiquement : 60%/20%/20% ; 70%/20%/10%

➢ Méthode utilisée en **Deep Learning**