

# Cours – Lundi (Semaine 2) : Introduction aux Données Médicales et leur Traitement

## Objectifs du jour

- Comprendre les **types de données médicales**.
  - Identifier les **problèmes liés à leur gestion**.
  - Savoir **prétraiter** ces données pour les rendre exploitables en IA.
  - Manipuler des **jeux de données médicaux** comme **MIMIC-III** et **MedMNIST**.
- 

## 1. Types de données médicales

Type	Description
<b>EHR</b> (Electronic Health Records)	Dossiers médicaux électroniques : historique, traitements, résultats de labo, etc.
<b>DICOM</b>	Format standard pour les <b>images médicales</b> (IRM, radios, échographies).
<b>Notes cliniques</b>	Observations écrites des soignants, souvent en texte libre.
<b>Autres</b>	Données temps réel (télémédecine), génétique, capteurs, etc.

---

## 2. Problèmes fréquents

- **Sensibilité** : données strictement protégées (RGPD, HIPAA).
  - **Accessibilité difficile** : données éparpillées dans différents systèmes.
  - **Qualité faible** : erreurs, formats variés, données manquantes.
  - **Hétérogénéité** : texte, images, signaux physiologiques, etc.
- 

## 3. Prétraitement des données

Étapes importantes avant l'entraînement d'un modèle IA :

- **Nettoyage** :
  - Supprimer doublons et valeurs aberrantes.
  - Uniformiser les formats (dates, unités, etc.).
- **Gestion des valeurs manquantes** :
  - Imputation (remplissage par moyenne, médiane).
  - Suppression des lignes incomplètes si nécessaire.
- **Normalisation** :
  - Mettre les données sur une même échelle (souvent 0 moyenne, écart-type 1).



## 4. Données à explorer

### MIMIC-III

- Dossier médical complet de patients en soins intensifs.
- Idéal pour les tâches de prédiction médicale.

### MedMNIST

- Base d'images médicales (ex : IRM, rayons X).
  - Parfait pour s'entraîner à la **vision par ordinateur en santé**.
- 



### Exemple de code de prétraitement

```
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Charger les données
data = pd.read_csv('mimic-iii-data.csv')

# Gérer les valeurs manquantes
data.fillna(data.mean(), inplace=True)

# Normaliser
scaler = StandardScaler()
data_normalized = scaler.fit_transform(data)

# Affichage
print(data_normalized[:5])
```

---



### À retenir

- Les données médicales sont **variées** mais **sensibles**.
  - Avant tout apprentissage automatique, un **prétraitement rigoureux** est essentiel.
  - Les jeux de données **MIMIC-III** et **MedMNIST** offrent une **base concrète** pour l'expérimentation.
-