# Projet Machine Learning
## M2-Datascale 2023

Aissatou BALDE & Aïda SOW

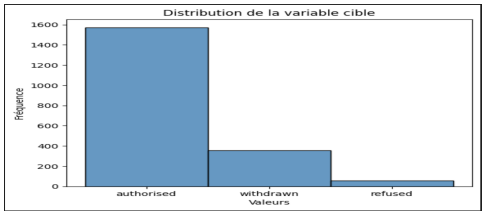
## Contexte & Problématique

Les données utilisées dans ce projet nous viennent de l'Agence Européenne des médicaments. Ce dataset comprend un ensemble de données relatives à des médicaments avec leurs caractéristiques  et au statut qui leur est attribué par l'Agence Européenne des médicaments.
La problématique que nous voulons résoudre est de savoir si un médicament produit sera autorisé ou non à la vente en se basant sur un nombre de critères donnés. En effet, l'Agence Européenne des médicaments est l'autorité qui valide la mise en marché d'un médicament sur la base d'un certain nombre de critères. Ces critères représentent les caractéristiques énumérées dans le dataset.

- En apprentissage supervisé, nous allons prédire la variable *authorisation_status* qui dit si un médicament est autorisé à la vente  ou non ou s'il est renvoyé pour révision
- En apprentissage non-supervisé, notre tâche sera d'interpréter les classes retrouvées au regard de l'apprentissage supervisé sur la variable *authorisation_status*

## Apprentissage supervisé

Pour l'apprentissage supervisé, nous sommes dans un problème de classification multi-classe. Notre variable cible est *authorisation_status* qui présente la distribution suivante:




Cette distribution présente un problème de classe imbalance. En effet la modalité, "authorised" est très majoritaire par rapport à  "refused" et "withdrawn". Ainsi pour

choisir notre modèle de classification, nous allons utiliser les modèles de : **Régression logistique**, **Random Forest** et **XGBoost**. Pour chaque modèle, nous allons d'abord effectuer l'entraînement sur la variable cible en conservant le "déséquilibre" des classes et ensuite effectuer l'entraînement sur les données corrigées avec la technique de l'oversampling (utilisation du SMOTE).
Pour comparer les modèles, nous allons utiliser la précision, le rappel et la F1-Score par classe comme nos métriques.
Dans le rapport nous allons comparer les modèles sur l'entraînement des données améliorées avec la technique oversampling.
Après avoir appliqué l'oversampling et préparé nos données en les divisant en ensembles d'entraînement et de test, nous avons entraîné les trois modèles sur l'ensemble d'entraînement:

1. **Régression Logistique**

Pour la régression logistique, après optimisation du modèle, nous avons obtenu les paramètres suivants pour le modèle:

```
▾                    LogisticRegression
LogisticRegression(C=0.1, max_iter=1000, multi_class='ovr')
```

Avec ces paramètres, les résultats obtenus après test du modèle sur les suivants:

```
              precision    recall  f1-score   support

  authorised       0.93      0.92      0.92       468
     refused       0.99      1.00      0.99       463
   withdrawn       0.93      0.93      0.93       485
```

Les performances du modèle sont bonnes, avec une précision élevée pour les classes "refused" et "authorised", ce qui signifie qu'il a bien identifié ces catégories. Le rappel élevé pour toutes les classes indique que le modèle a également réussi à rappeler la majorité des instances positives.

2. **Random Forest Classifier**

Pour le random Forest, les paramètres obtenus après optimisation du modèle sur les données reéquilibrées sont les suivants :

```
▾                 RandomForestClassifier
RandomForestClassifier(max_depth=40, n_estimators=50)
```

Avec ces paramètres, les résultats obtenus sont les suivants pour la phase de test:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| authorised | 0.94 | 0.96 | 0.95 | 468 |
| refused | 1.00 | 1.00 | 1.00 | 463 |
| withdrawn | 0.96 | 0.94 | 0.95 | 485 |

Les résultats de classification montrent de bonnes performances pour les trois catégories, avec une précision de 94 %, 100 % et 96 % respectivement pour les catégories "authorised", "refused" et "withdrawn". De plus, le rappel élevé de 96 %, 100 % et 94 % indique que le modèle a bien réussi à capturer les instances de ces catégories, ce qui se traduit par des scores F1 élevés de 95 % pour "authorised" et "withdrawn" et de 100 % pour "refused".

3. **XGBoostClassifier**

Pour ce modèle de classification, nous obtenons après optimisation les résultats suivants:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.96 | 0.97 | 0.97 | 468 |
| 1 | 1.00 | 1.00 | 1.00 | 463 |
| 2 | 0.97 | 0.97 | 0.97 | 485 |

Le modèle a produit des résultats prometteurs avec des scores élevés de précision, de rappel et de score F1 pour les trois classes, indiquant une bonne capacité à prédire les différentes catégories. C'est le modèle qui offre le plus grand f1-score et precision pour la classe "refused".

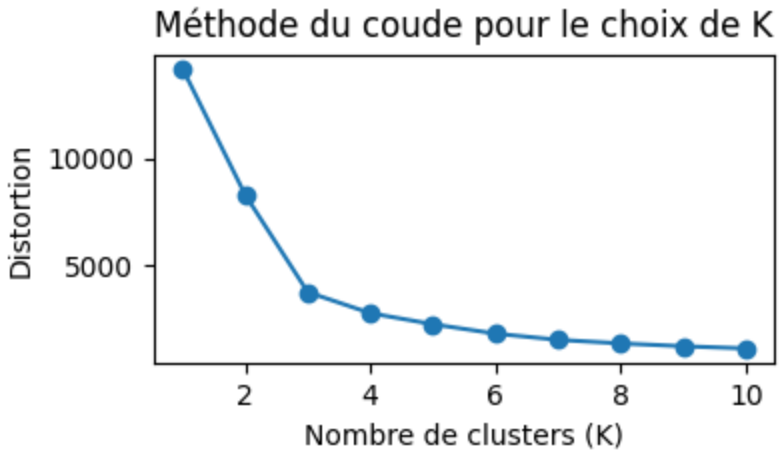
Les résultats de notre comparaison ont révélé que XGBoost a surperformé la Régression Logistique et RandomForestClassifier en termes de précision et de F1-score.En outre, nous avons souligné l'importance de l'optimisation des hyperparamètres pour améliorer les performances des modèles sous-performants. La sélection du modèle final a été basée sur une analyse approfondie des métriques.

# Apprentissage non-supervisée

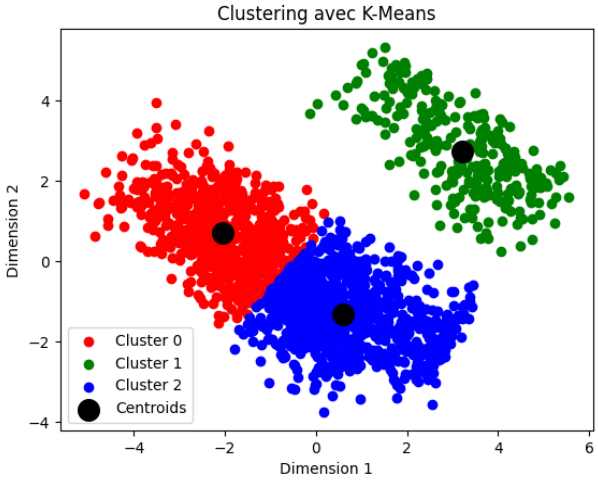
En effectuant un apprentissage non supervisé multiclasse, nous essayons de découvrir des structures, des relations ou des groupes dans notre ensemble de données sans avoir d'étiquettes(sans utiliser notre target) de classe spécifiques pour guider le modèle. Il existe plusieurs modèles de Machine learning. Mais pour cette problématique, nous allons utiliser K-means qui est l'un des algorithmes de clustering les plus populaires. Il divise les données en K clusters, où K est un paramètre que vous spécifiez. Il attribue chaque exemple au cluster le plus proche en fonction de la distance entre les points et les centroïdes des clusters.

Pour mettre en oeuvre le modèle k-Means, nous avons suivis les étapes suivantes:

1. **Transformation des données:** on utilise une technique de réduction de la dimensionnalité comme l'analyse en Composantes Principales (PCA) pour réduire la dimensionnalité de nos données normalisées avec deux composantes principales, ce qui peut faciliter le clustering.
2. **Choix du nombre de clusters:** on sélectionne un nombre de clusters (K) qui correspond au nombre de classes que l'on souhaite identifier. Cependant, on peut utiliser des méthodes telles que la méthode du coude (Elbow Method) pour confirmer le choix de K.




3. **Exécution du k-Means**



4. **Evaluation du clustering:** on utilise la métrique Indice de Silhouette qui évalue à quel point les points d'un cluster sont similaires les uns aux autres par rapport aux points d'autres clusters. Il varie de -1 (mauvais regroupement) à +1 (regroupement bien défini). Une valeur proche de zéro indique un chevauchement entre les clusters. Nous avons obtenu un indice de 0.22 qui indique que les clusters sont assez bien définis, c'est d'ailleurs confirmé par la visualisation des clusters faite plus haut.

# Conclusion

Ce projet de machine learning a été une expérience enrichissante, combinant à la fois des approches supervisées et non supervisées pour analyser un ensemble de données de l'Agence Européenne des médicaments lié à l'autorisation de médicaments.

Dans la partie supervisée, la classification des médicaments en "authorised," "refused" et "withdrawn" a été réalisée. Le choix d'un modèle a été fait avec succès, démontrant des performances solides et fiables. Cela pourrait être d'une grande utilité dans le domaine de la réglementation pharmaceutique pour la prise de décision.

D'autre part, la partie non supervisée a permis de découvrir des tendances cachés dans les données, offrant une perspective intéressante pour l'optimisation des processus de réglementation. Cette approche complète a renforcé notre compréhension de la gestion des médicaments et des autorisations, montrant comment le machine learning peut être un atout précieux dans ce domaine.

En résumé, ce projet a ouvert de nouvelles opportunités pour l'analyse de données pharmaceutiques et la prise de décisions éclairées.

**Lien Github:**

L'analyse exploratoire et les modèles sont disponibles dans le repository github suivant:
https://github.com/Aida73/M2DS-Drugs-Authorization-Classification.git