# Devoir TP

***Directives:*** *Le dernier jour pour rendre votre travail est le 05/02/2023. Il est possible de traiter le sujet en binôme maximum. Le rendu est sous forme d'un fichier pdf (avec le nom ou les deux noms) généré par un fichier rmarkdown. Il est indispensable d'inclure les sorties résultats de votre code dans le fichier pdf pour pouvoir l'évaluer. 2 points sont réservés à la qualité de présentation comme l'utilisation de ggplot2 pour les figures.* ***Un délai supplémentaire est possible sur demande !!***

**Exercice 1 (clustering spectral: un peu de visualisation)**

1. Créer un échantillon de taille $n = 100$ suivant le mélange Gaussien suivant :

$$p_1\mathcal{N}(m_1, \Sigma) + p_2\mathcal{N}(m_2, \Sigma) + p_3\mathcal{N}(m_3, \Sigma),$$

avec $p_1 = p_2 = p_3 = \frac{1}{3}$, $m_1 = \begin{pmatrix} -3 \\ 0 \end{pmatrix}$, $m_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, $m_3 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}$ et $\Sigma = \begin{pmatrix} 0.2 & 0 \\ 0 & 0.1 \end{pmatrix}$.

2. Tracer le nuage de points obtenu. Les classes sont-elles (visuellement) bien séparées ?
3. Construire une matrice de similarité $S$ à l'aide de la fonction de similarité gaussienne de paramètre $\sigma = 1$.
4. Calculer la matrice d'adjacence $W_{\text{full}}$ du graphe entièrement connecté associé à la matrice $S$ obtenue dans la question précédente. Quel est le degré moyen du graphe (la moyenne des degrés des sommets du graphe) ?
5. Tracer le graphe à l'aide du package igraph.[1]
6. En choisissant pour seuil $\varepsilon$ le quantile empirique d'ordre 0.75 des indices de similarité, construire la matrice d'adjacence $W_\varepsilon$ du graphe du $\varepsilon$-voisinage, et tracer le graphe correspondant. Quel est le degré moyen du graphe ?.
7. En prenant $k = 2\lfloor \log n \rfloor$, construire la matrice d'adjacence $W_{\text{knn}}$ des k-plus proches voisins mutuels et tracer le graphe associé. Quel est le degré du graphe moyen ?
8. Constuire les matrices Laplaciennes normalisées des graphes obtenus dans les questions 4, 6 et 7 ainsi que leurs valeurs propres.
9. Tracer les 10 premières valeurs propres pour chaque matrice Laplacienne. Que remarque-t-on ?
10. Appliquer l'algorithme de partitionnement spectral normalisé pour chaque graphe de similarité, en prenant 3 classes. Comparer les résultats obtenus.

**Exercice 2 (Clustering de données catégorielles)**

Le clustering de données catégorielles peut se faire de deux façons:

- Utiliser une métrique adaptée à ce type de données pour effectuer un clustering géométrique.
- Utiliser un modèle de mélange pour données catégorielles.

On souhaite comparer ces deux approches.

1. Créer une fonction `rbinarymixture` qui génère indépendamment $n$ vecteurs de $p$ variables catégorielles à trois modalités selon le modèle génératif suivant:

$$Z_i \sim \mathcal{M}(1/2, 1/2) \text{ et } X_j | Z_{ik} = 1 \sim \mathcal{M}(1/3 + (-1)^k\varepsilon, 1/3, 1/3 + (-1)^{k+1}\varepsilon).$$

Cette fonction prend pour arguments $n$, $p$ et $\varepsilon$. Elle retourne une liste composée de la partition et des covariables.

[1] **Indication** : on utilisera la fonction graph.adjacency, puis la fonction simplify pour éviter les boucles for. Pour améliorer la lisibilité, on pourra pondérer la largeur des arêtes du graphe par leur poids, en modifiant la valeur E(p)$width, si p est le nom donné à la sortie de la fonction graph.adjacency.

2. Générer un échantillon avec $n = 100$, $p = 10$, $\varepsilon = 0.2$.

3. Effectuer le clustering de cet échantillon (uniquement sur les covariables) avec une CAH utilisant le critère de Ward et la métrique de Manhattan. Calculer l'ARI entre la partition obtenue en 2 classes et la vraie partition.

4. Faire le clustering de ces données en utilisant un modèle de mélange à l'aide du package VarSelLCM.

5. Implémenter la fonction `singlesimulation` qui prend comme argument une liste générée par la fonction `rbinarymixture` et qui donne les valeurs des ARI obtenus par la méthode CAH pour données qualitatives et par clustering par modèles de mélange pour données catégorielles.

6. Générer $N = 100$ échantillons avec les paramètres $n = 100$, $p = 10$ et $\varepsilon = 0.2$. Appliquer à chaque échantillon la fonction `singlesimulation`.

7. Conclure sur la méthode à utiliser.