|  |
| --- |
| Université Lille 1 – Master 1 |
| RDF – TP n°08 |
| Classification non-supervisée – Algorithme des k-means |

|  |
| --- |
| BARCHID Sami – SLIMANI Anthony  10/03/2019 |

# Introduction

L’objectif de ce TP est d’apprendre à classifier des données multi-variées par clustering dans un contexte d’apprentissage non-supervisé. L’outil de clustering qui sera vu ici est **kmeans**.

Ce rapport est divisé en 2 parties :

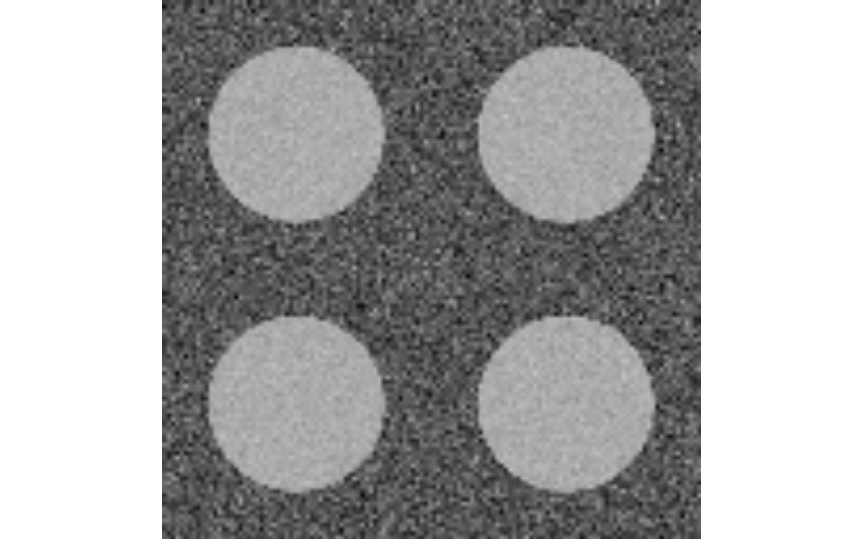
* **Segmentation d’une image en niveaux de gris :** application de la segmentation non-supervisée par clustering k-means sur une image en niveaux de gris.
* **Segmentation d’une image couleur :** application de la segmentation non-supervisée par clustering k-means sur une image en couleurs.

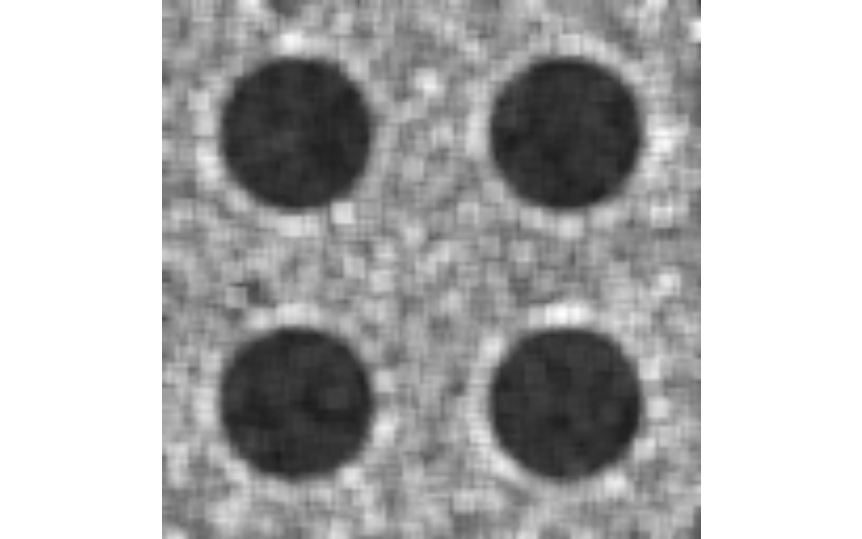
# Segmentation d’une image en niveaux de gris

Le but de cette partie est de binariser une image en niveaux de gris grâce à la méthode de clustering k-means.

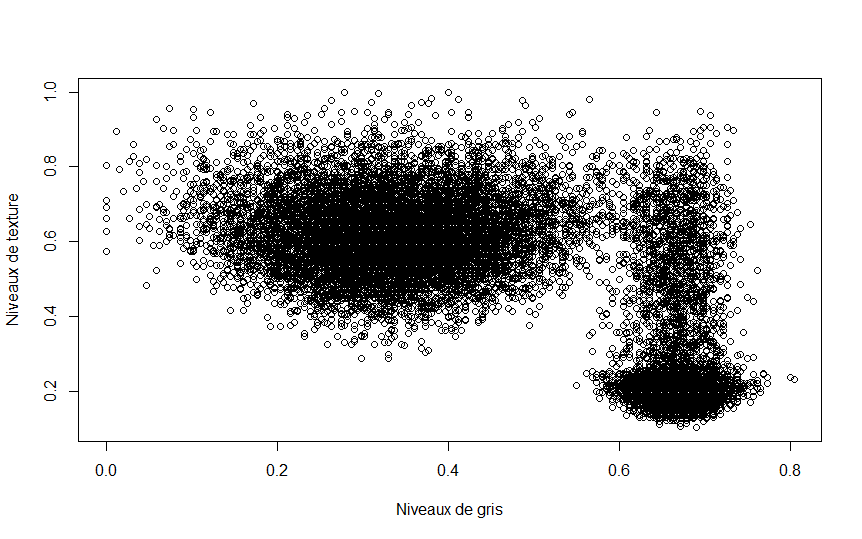
### Présentation de l’image étudiée

L’image étudiée dans cette partie est composée de deux attributs : les niveaux de gris et les niveaux de texture sur un voisinage de 5 pixels. Ce sont les deux attributs que nous avions déjà utilisés auparavant dans le TP de segmentation et binarisation.

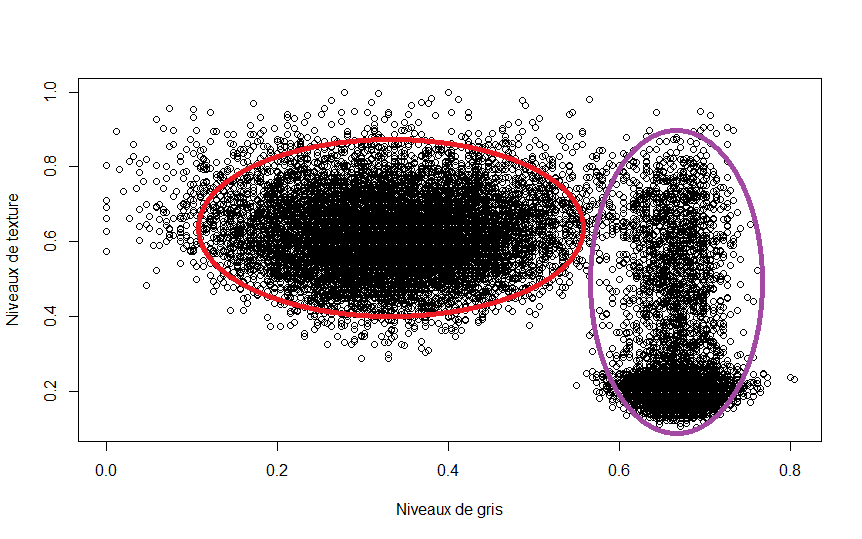
 Figure 1 – NIVEAUX DE GRIS DE L’IMAGE ÉTUDIÉE

 Figure 2 – NIVEAUX DE TEXTURE DE L’IMAGE ÉTUDIÉE

Nous sommes alors capables d’afficher les observations représentant les pixels dans le plan de projection des deux attributs choisis. La figure suivante représente les nuages obtenus :

 FIGURE 3 – REPRÉSENTATION DES PIXELS DANS LE PLAN DE PROJECTION DES DEUX ATTRIBUTS CHOISIS

Nous remarquons facilement que les points de l’image se séparent en deux nuages :

FIGURE 4 – IDENTIFICATION EMPIRIQUE DES DEUX NUAGES DE POINTS IDENTIFIÉS

* Un nuage dont le niveau de gris est plus élevé et le niveau de texture en majorité plus bas (entouré en mauve). Les pixels de ce nuage représentent les pixels qui composent le fond de l’image puisque ceux-ci ont un niveau de gris bas et varient beaucoup en niveau de texture (c’est pour ça que le fond de l’image est plus foncé que les objets et est composé de pixels de différentes valeurs de niveaux de gris).
* Un nuage dont le niveau de gris est moins élevé et le niveau de texture plus haut (entouré en rouge). Les pixels de ce nuage représentent les pixels qui composent les 4 disques puisque ceux-ci ont un niveau de gris élevé et varient peu en niveau de texture (ils sont quasiment tous gris clairs de façon homogène).

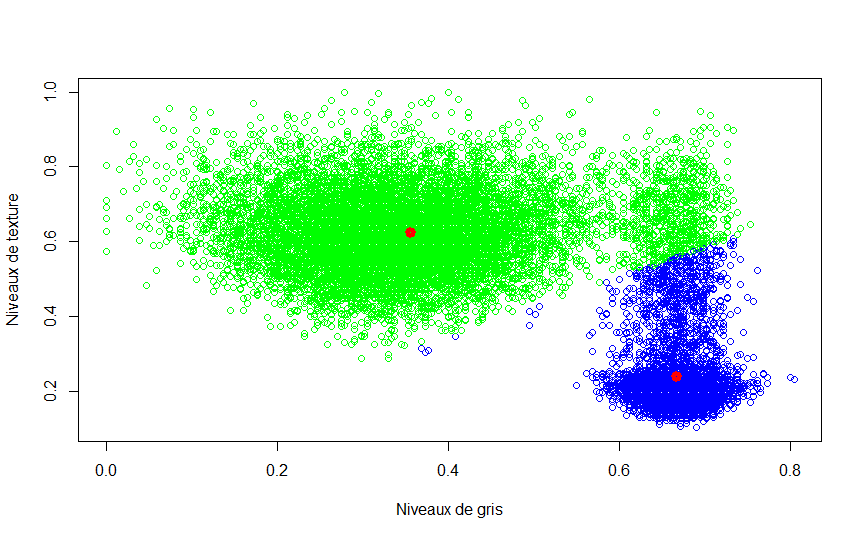
### Classification par k-means

Nous allons donc appliquer la classification non-supervisée par k-means sur l’image présentée, qui va classifier chaque pixel de l’image suivant sa distance avec un des centres de gravité des classes que nous avons définies.

Afin d’appliquer le clustering par k-means, l’utilisateur doit donc donner deux choses :

* Le nombre de classes à retrouver. Il faut donc choisir un nombre de classe pertinent pour obtenir une bonne classification.
* La condition d’arrêt du clustering. Ici, ce sera un nombre d’itération maximal, c’est-à-dire le nombre de fois qu’il va recalculer l’assignation des points aux classes et mettre à jour les centres de gravité des classes. Il est donc nécessaire de choisir un nombre d’itération au bout duquel le clustering par k-means a convergé, c’est-à-dire que la position des centres de gravité de va plus changer énormément entre deux itérations.

Comme nous avons identifié empiriquement deux nuages de points (le fond et les formes de l’image), nous avons décidé de compter deux classes. Nous avons aussi décidé d’un nombre d’itérations maximale de 30. La classification par k-means nous donne alors la figure suivante :

FIGURE 5 – CLASSIFICATION CLUSTERING PAR K-MEANS

* La première classe est en vert.
* La deuxième classe est en bleu
* Les centres de gravité des classes sont affichés en rouge.

Nous constatons que le clustering par k-means a classé les pixels de telle manière à respecter en grande partie les nuages que nous avions définis empiriquement. Cependant, nous remarquons que certains pixels ont été mal classés. Nous pourrons donc nous attendre à avoir une légère erreur de classification.

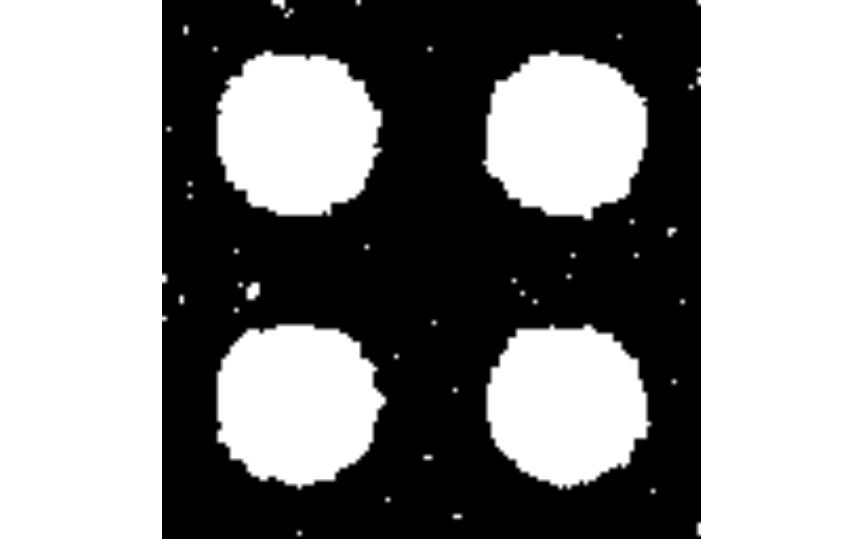
### Segmentation

Après avoir obtenu les classes et leurs centres de gravité grâce au clustering par k-means, nous pouvons alors effectuer la binarisation de l’image.

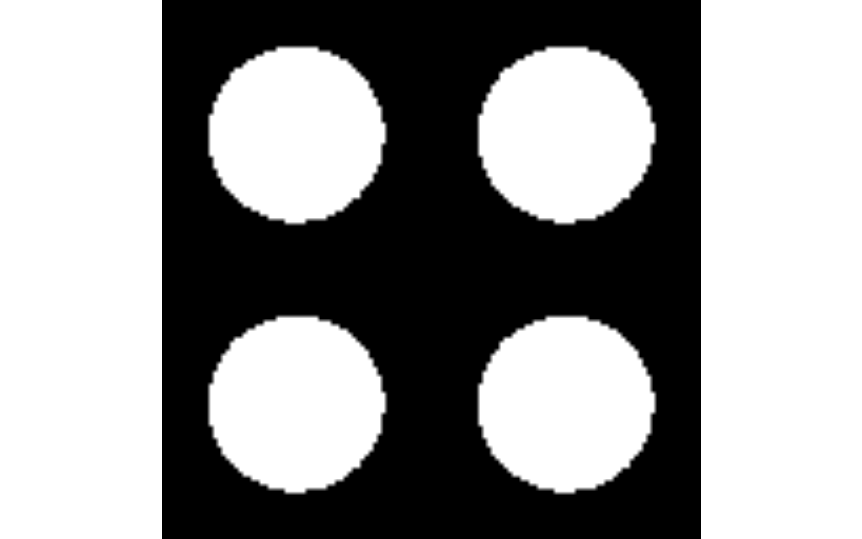
Pour ce faire, nous devons tout d’abord calculer, pour chaque pixel, la distance séparant l’observation associée au centre de gravité de chaque classe.

Après avoir récupéré ces distances, nous pouvons obtenir l’image segmentée qui résulte du minimum des deux distances.

Nous obtenons ainsi l’image segmentée suivante :

FIGURE 6 – IMAGE SEGMENTÉE

Nous disposons, pour ce TP, d’une image de référence permettant de mesurer le taux de bonne classification de l’image. La figure suivante représente l’image de référence :

FIGURE 7 – IMAGE DE RÉFÉRENCE

Le calcul du taux de bonne classification nous donne un résultat de 0.95 = **95% de bonne classification**.

Nous comprenons donc que le clustering par k-means est une technique en majorité efficace dans le cas de notre image pour segmenter une image dans un contexte d’apprentissage non-supervisé.

Nous observons aussi que, comme nous l’avions dit précédemment, nous remarquons de légères erreurs de segmentation dans l’image. Cela correspond aux points ayant été classés dans la mauvaise classe par rapport aux deux nuages que nous avons définis empiriquement. Ce problème s’explique du fait que le clustering par k-means est plus adapté sur des nuages sphériques. Or, dans notre cas, les nuages sont plus ovales que sphériques. De ce fait, certains points du nuage (les plus éloignés du centre de gravité) sont plus proches du centre de gravité de l’autre nuage de points. De ce fait, ils sont mis dans la mauvaise classe.

# Segmentation d’une image couleur