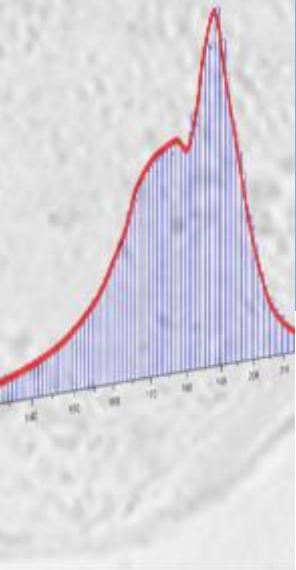




Cours de segmentation d'images

Projet K-means



Master M2TI - Paris V
2016-2017

John Chaussard
LAGA – Université Paris 13
chaussard@math.univ-paris13.fr

Le problème

On possède une image couleur, et on souhaite **diminuer le nombre de couleurs sans perdre l'information** contenue dans l'image (garder les formes générales de départ).



16 millions de
couleurs



128 couleurs



32 couleurs



8 couleurs

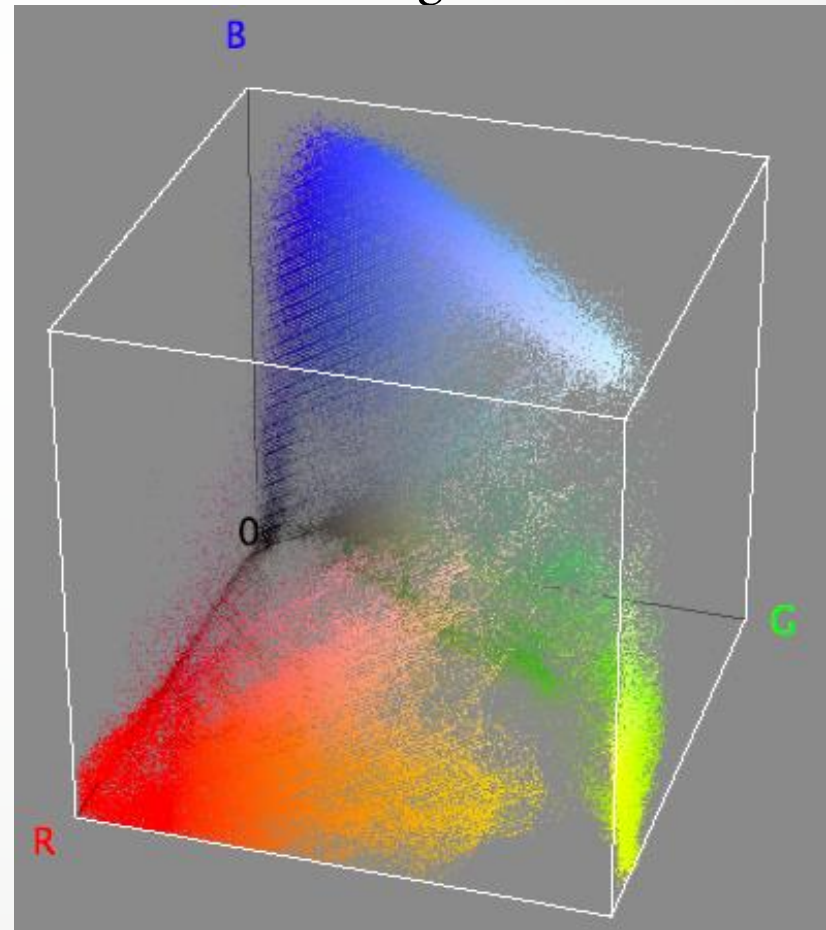
Ce type de réduction permet de réduire la quantité de données dans l'image qui pourra ensuite être analysée par d'autres algorithmes. Ceci permet aussi de compresser l'image en y diminuant la quantité d'information.

Projeter les points d'une image couleur

Comme expliqué précédemment, chaque pixel d'une image couleur est en réalité un ensemble de trois valeurs (une intensité en rouge, une en bleu et une en vert).

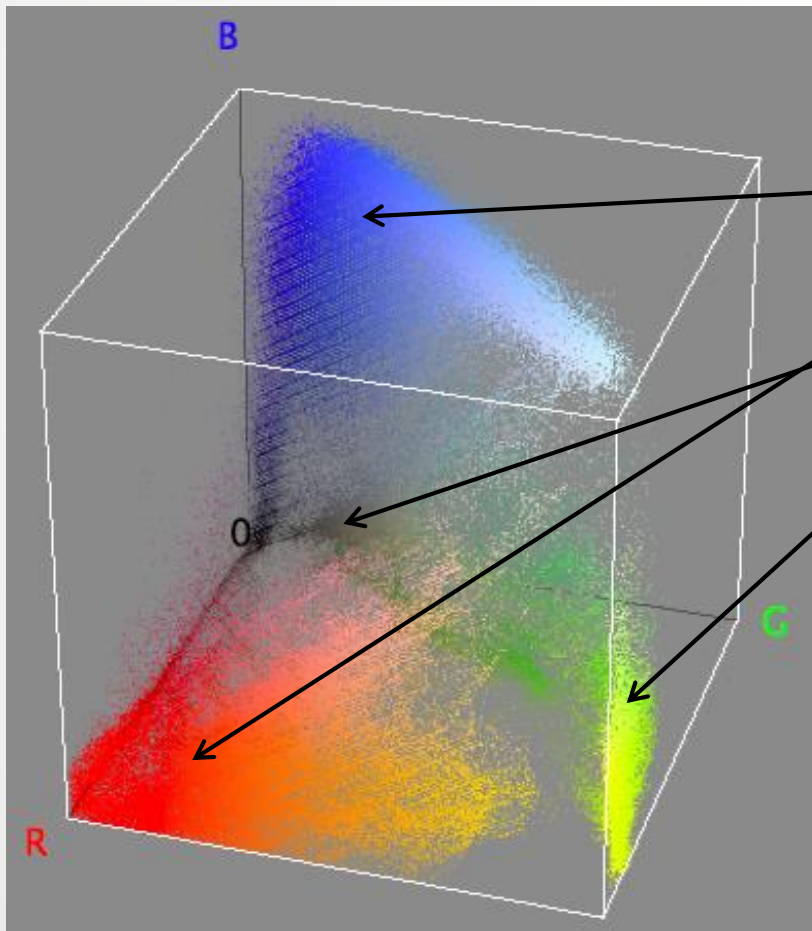
Chaque pixel possède donc trois valeurs pouvant être associées à des coordonnées dans l'espace.

On peut projeter chaque pixel dans un repère 3d et voir le résultat...



Projeter les points d'une image couleur

Que voyez-vous apparaître dans cette représentation ?



Des groupes de pixels apparaissent, correspondant à des éléments de l'image :

- . Le **plumage bleu**
- . Le **plumage/bec rouge**
- . Le **fond**
- . Le **plumage vert/jaune**

La représentation en histogramme conserve une informations concernant les groupes de pixels de couleurs proches.



Regrouper les points « proches »

Une stratégie pour diminuer le nombre de couleurs serait de réunir les points proches dans des mêmes groupes : **chaque pixel dans un même groupe se verrait ensuite attribuer la même valeur.**

On utilisera pour cela l'algorithme des K-means.

L'algorithme des K-means est un **algorithme d'apprentissage non supervisé**, c'est à dire qu'il permet de trouver une structure dans des données « non étiquetées ».

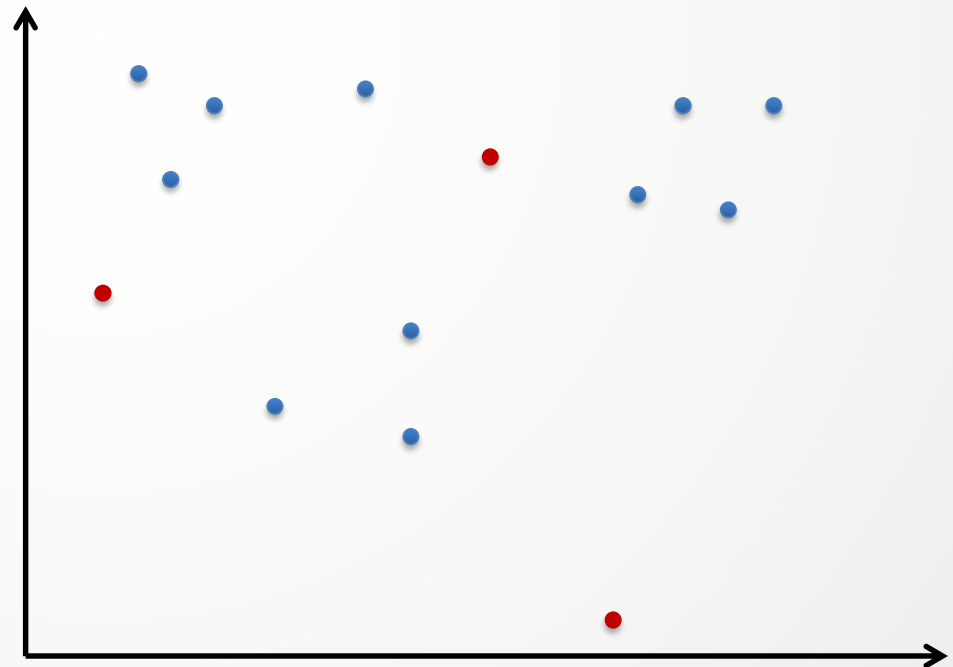
C'est un des algorithmes de base de l'intelligence artificielle, permettant aux ordinateurs d'apprendre des règles à partir d'exemples.

L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un **exemple en 2d** : imaginons que nous ayons des points (en bleu) dans un plan, et que nous souhaitons faire des groupes.

L'utilisateur doit décider de la **valeur de K**, qui sera le **nombre de groupes** à constituer (ici, on choisira $K=3$).

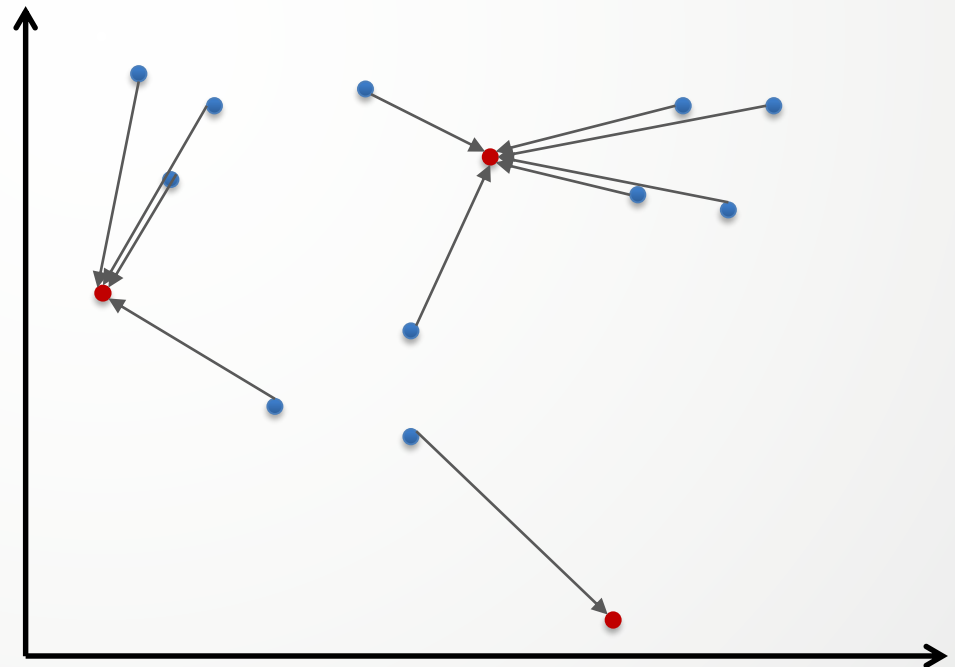
On commence par placer K points, appelés **graines** (ici en rouge), au hasard dans le plan.



L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

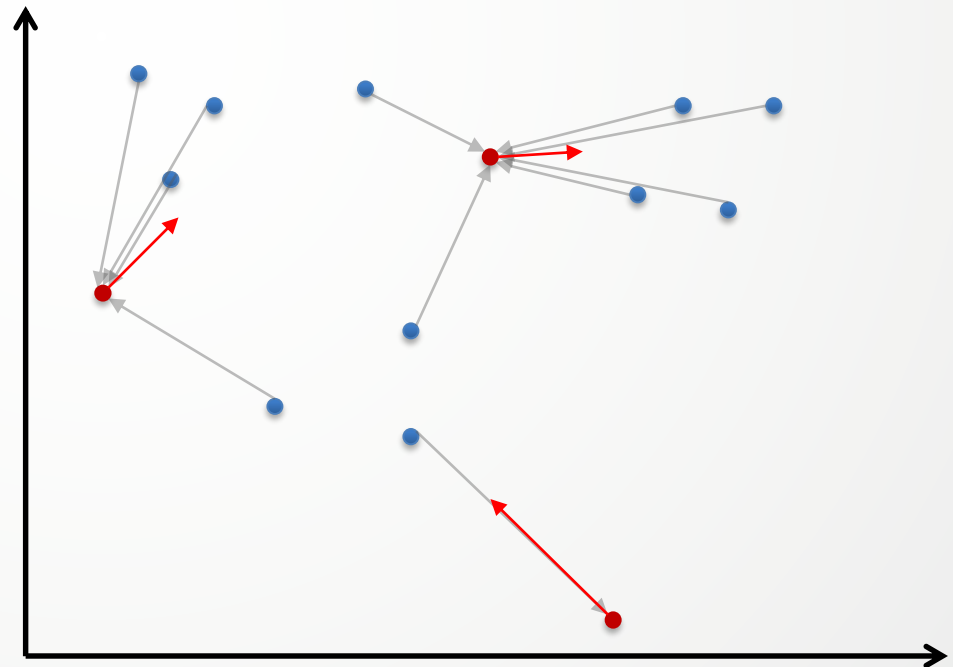
Phase 1 : On « relie » alors chaque point (bleu) de l'image à la graine (rouge) qui lui est la plus proche.



L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

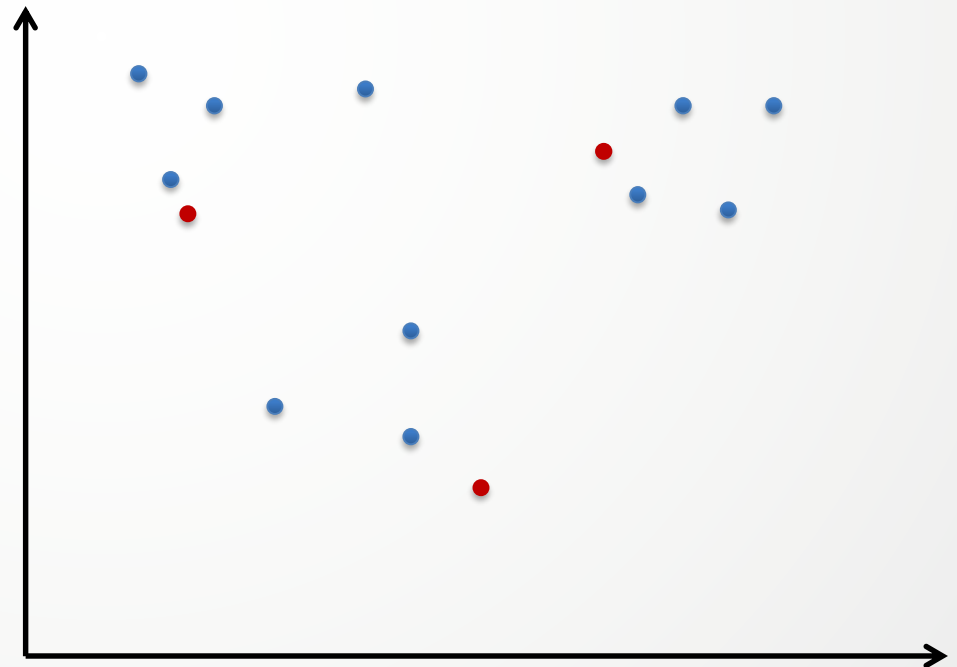
Phase 2 : On déplace chaque graine « vers » le barycentre des points avec lesquels elle est reliée.



L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

Phase 2 : On déplace chaque graine « vers » le barycentre des points avec lesquels elle est reliée.

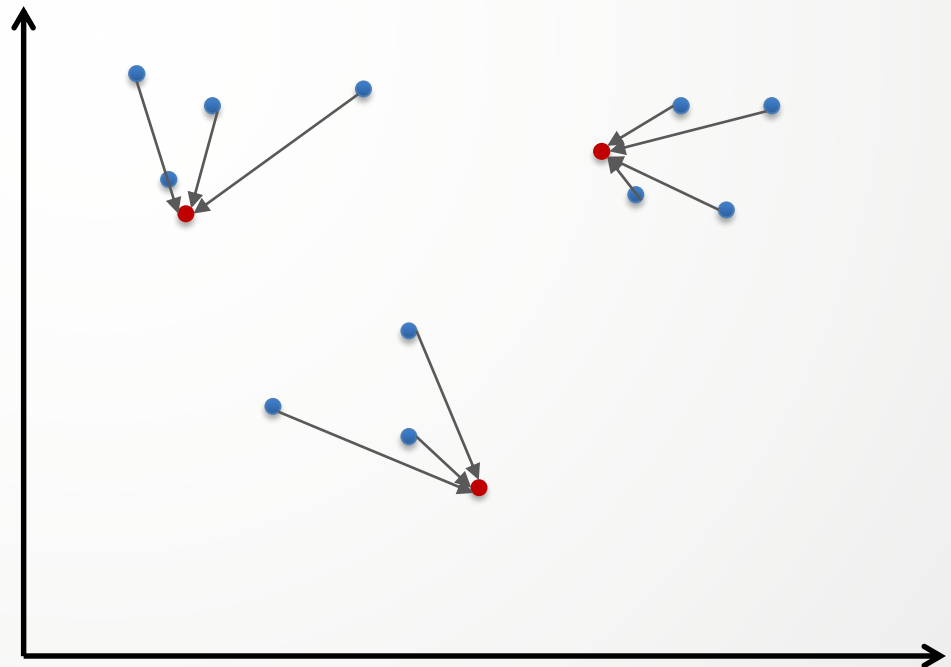


L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

On itère ces deux étapes jusqu'à ce que chaque graine ne bouge plus (beaucoup).

Phase 1 : relier les points aux graines



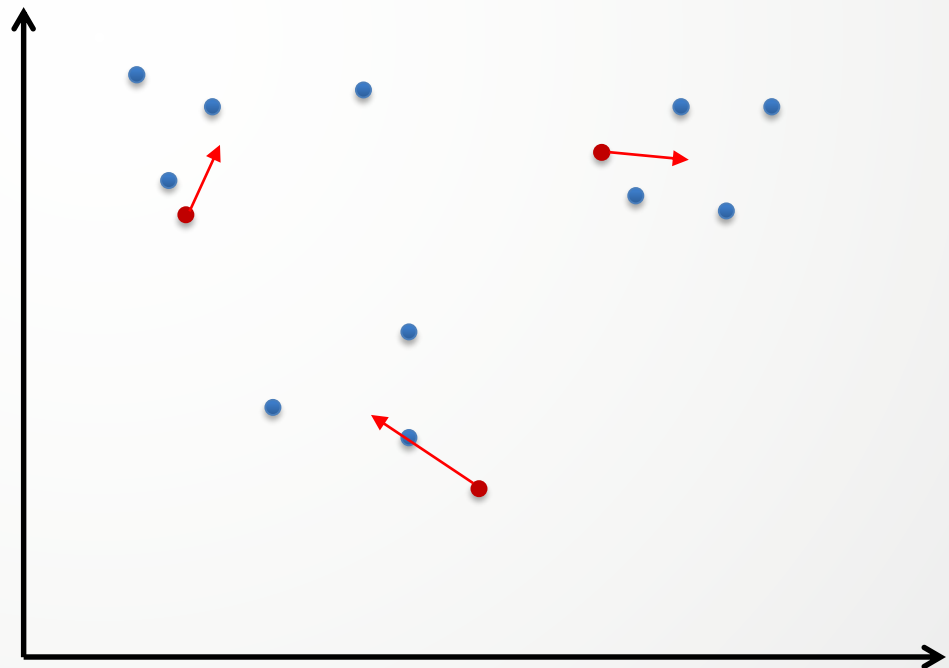
L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

On itère ces deux étapes jusqu'à ce que chaque graine ne bouge plus (beaucoup).

Phase 1 : relier les points aux graines

Phase 2 : déplacer les graines



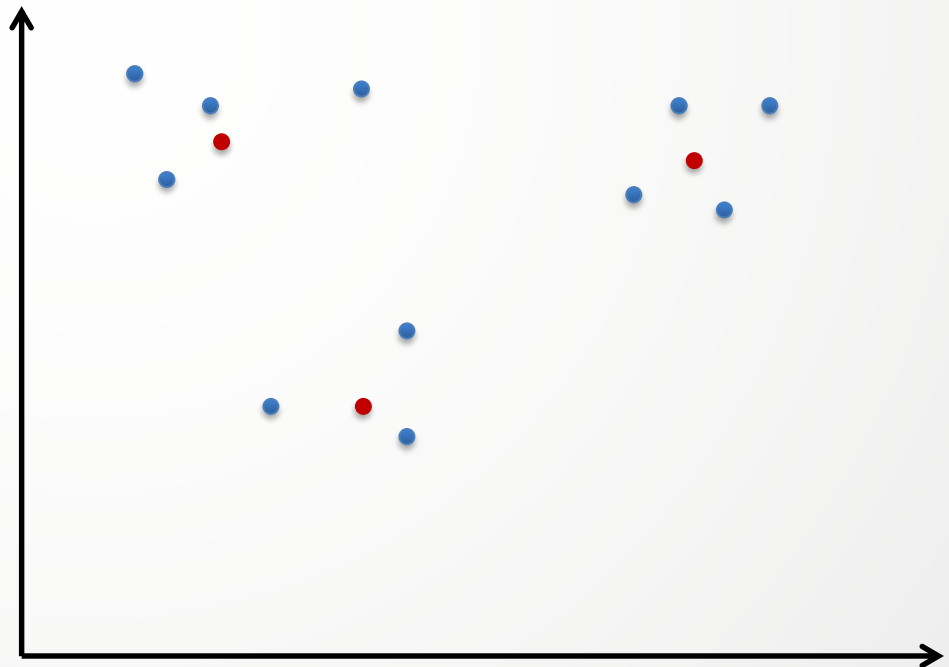
L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

On itère ces deux étapes jusqu'à ce que chaque graine ne bouge plus (beaucoup).

Phase 1 : relier les points aux graines

Phase 2 : déplacer les graines

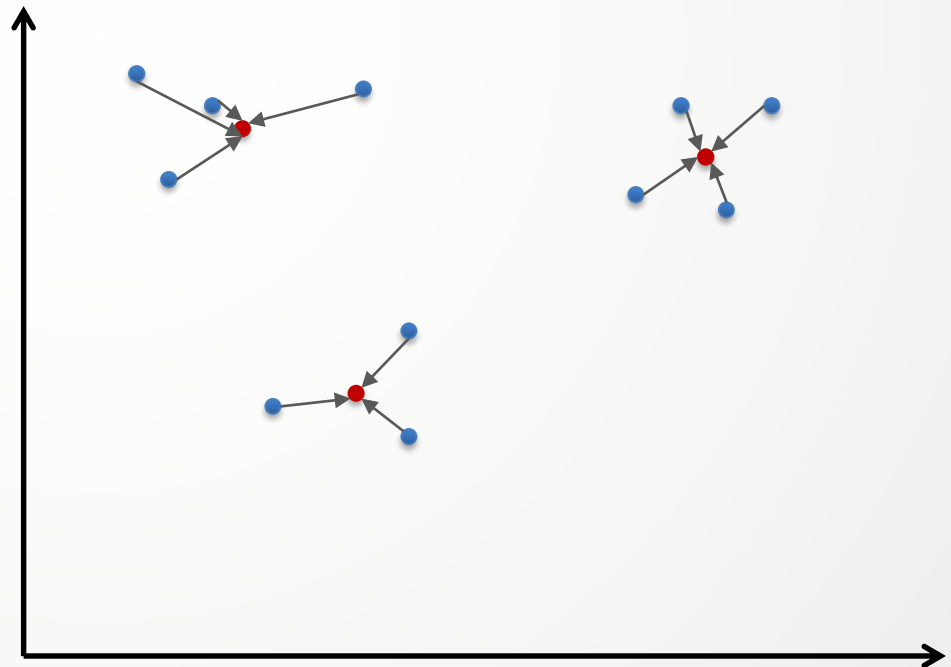


L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

Une fois que les graines ne bougent plus « beaucoup », on relie une dernière fois chaque point à la graine qui leur est la plus proche.

On regroupe ensemble les points rattachés à une même graine.

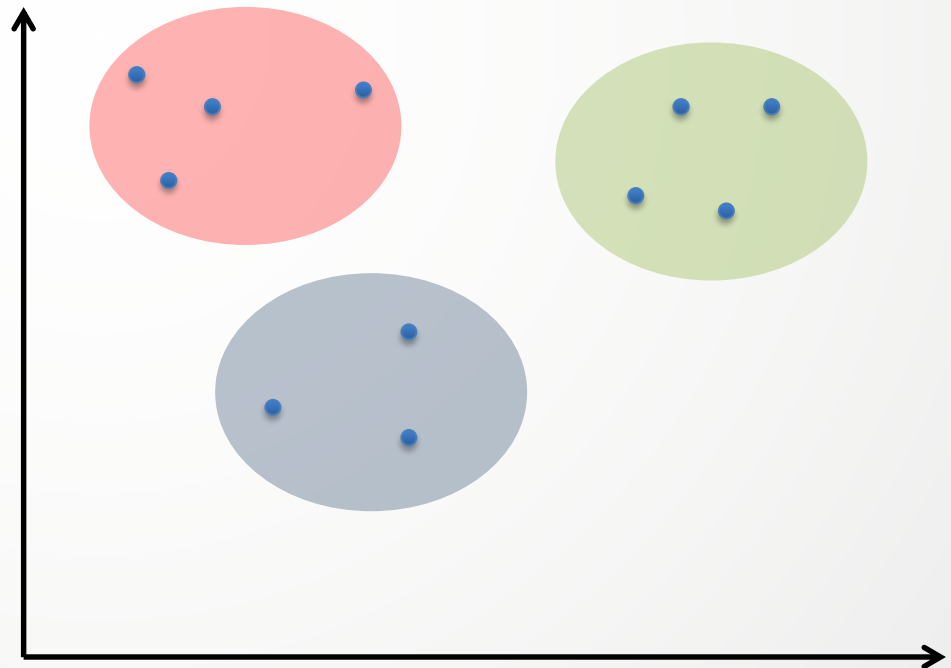


L'algorithme des K-means

Pour décrire l'algorithme des K-means, on utilisera un [exemple en 2d](#) : imaginons que nous ayons des points dans un plan, et que nous souhaitons faire des groupes.

Une fois que les graines ne bougent plus « beaucoup », on relie une dernière fois chaque point à la graine qui leur est la plus proche.

On regroupe ensemble les points rattachés à une même graine.



Colorier l'image

Dans le cas d'une image couleur, chaque point représente un pixel de l'image.

A la fin de l'algorithme, on remplace la couleur de chaque pixel par la couleur (qui est la coordonnée) de la graine à laquelle il est rattaché.

Résultats

Plus on diminue le nombre de graines (et donc le nombre de groupes), et plus on perd de l'information dans le résultat par rapport à l'image d'origine.



16 millions de
couleurs



128 couleurs



32 couleurs



8 couleurs

Quelques éléments d'implémentation

Certains éléments restent à éclaircir dans l'algorithme.

- . Le placement initial des graines

Une stratégie consiste à relier chaque graine à des pixels choisis au hasard, de façon à ce que chaque graine soit reliée à au moins un pixel.

Ensuite, on place les graines sur le barycentre des pixels qui lui sont attachés.

- . Se déplacer « vers » le barycentre

A chaque tour, les graines se déplacent vers le barycentre. On choisira un paramètre λ (entre 0 et 1) qui permettra de contrôler cette vitesse de déplacement. Il est raisonnable qu'à chaque tour, les graines ne se déplacent pas directement sur les barycentres.