INFORMATIQUE
TECHNIQUE

Expert

Le Machine Learning avec Python

De la théorie à la pratique

Préface de **Patrick Albert**
Cofondateur d'ILOG et du HUB France IA

En téléchargement

code source

+ QUIZ

Version en ligne
OFFERTE !
pendant 1 an

Madjid KHICHANE
PhD en Intelligence Artificielle

eni

Table des matières

Les éléments à télécharger sont disponibles à l'adresse suivante :
http://www.editions-eni.fr
Saisissez la référence de l'ouvrage **EIMLPYT** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Partie 2 : Outils techniques de la Data Science - Python, NumPy, Pandas et Jupyter

Chapitre 2-2
Le langage Python

Chapitre 2-2
La bibliothèque NumPy

Chapitre 2-3
La bibliothèque Pandas

Chapitre 2-4
Travailler avec Jupyter

Partie 4 : Les grands algorithmes du Machine Learning

Chapitre 4-1
La régression linéaire et polynomiale

Chapitre 4-2
La régression logistique

Chapitre 4-3
Arbres de décision et Random Forest

Chapitre 4-4
L'algorithme k-means

Chapitre 4-5
Support Vector Machine

Chapitre 4-6

Analyse en composantes principales

Chapitre 4-7
Les réseaux de neurones

Partie 5 : Le Deep Learning et le traitement automatique du langage

Chapitre 5-1
Le Deep Learning avec TensorFlow

Chapitre 5-2

Le traitement automatique du langage

Annexe
La programmation orientée objet avec Python

Chapitre 4-4
L'algorithme k-means

1. Objectif du chapitre

Les chapitres précédents ont abordé des exemples de deux types d'algorithmes de Machine Learning : les algorithmes de régression et de classification. Ce chapitre porte sur l'algorithme k-means, appelé l'algorithme des k-moyennes en français, qui est un algorithme simple à comprendre et qui fait partie des algorithmes de clustering les plus connus et les plus utilisés.

L'algorithme k-means a été introduit par J. McQueen en 1967. C'est un algorithme non supervisé qui permet de répartir un ensemble de n observations en k clusters. L'objectif après l'application de l'algorithme k-means sur un jeu de données est que chaque cluster contienne des observations homogènes et que deux observations de deux clusters différents soient hétérogènes.

Les domaines d'application de l'algorithme k-means sont nombreux. Par exemple, il est très utilisé pour la segmentation des clients à des fins de marketing, ou encore pour l'isolation des motifs dans les images, car justement les images présentent souvent des régions homogènes, notamment en matière d'intensité lumineuse.

De manière générale, le succès de l'algorithme k-means et de ses versions réside dans sa simplicité et sa capacité à traiter des données de grande taille.

À la fin de ce chapitre, le lecteur aura abordé :

Le fonctionnement de k-means via des illustrations.

– Les étapes principales de l'algorithme k-means classique.
– L'application de l'algorithme k-means avec Scikit-learn.
– L'impact des valeurs extrêmes sur les performances de l'algorithme k-means.
– La recherche de la valeur optimale du paramètre *K* de l'algorithme k-means.
– Les avantages et les inconvénients ainsi que les variantes de l'algorithme k-means.

2. k-means du point de vue géométrique

Comme précisé au début de ce chapitre, l'algorithme k-means est très intuitif et simple à comprendre. Avant d'entrer dans les détails, il faut noter que k-means, comme tous les algorithmes de clustering, ne nécessite pas l'étiquetage des données, car c'est une procédure non supervisée.

De façon informelle, étant donné *n* observations à répartir sur *k*clusters, k-means choisit initialement, de manière aléatoire, *k* observations parmi les *n* observations, comme étant les centres des *k* clusters recherchés. Chacune des *n* observations sera associée au cluster dont le centre est le plus proche parmi les *k* centres choisis initialement. Une fois que toutes les observations sont associées à leurs clusters respectifs, le centre de chaque cluster est recalculé en fonction des observations qu'il contient. Puis, de nouveau, chacune des observations est associée au cluster dont le centre est le plus proche de cette observation par rapport à tous les centres des autres clusters. Ces opérations de recalcul des centres des clusters puis d'association des observations aux clusters les plus proches sont répétées jusqu'à ce qu'un critère d'arrêt soit atteint.

L'algorithme k-means utilise une fonction pour calculer les distances entre les observations et les centres des clusters. Ce calcul des distances peut être basé sur la distance euclidienne, la distance de Manhattan ou toute autre fonction permettant de mesurer la dissimilarité entre les observations.

Pour mieux comprendre cet algorithme de clustering, cette section déroule l'algorithme k-means sur un exemple simple. Soit six observations a, b, c, d, e et f à répartir sur deux clusters C_1 et C_2 ; supposons que la distance utilisée est la distance euclidienne classique. Ces six observations sont définies dans un espace à deux dimensions et leurs coordonnées sont indiquées dans le tableau suivant :

Axes	a	b	c	d	e	f
x	2	4	2	4	10	10
y	4	4	2	2	2	4

Figure 10-1 : un simple jeu de données avec leurs coordonnées en deux dimensions

Remarque

Pour rappel, la distance euclidienne entre deux observations A=(xA,yA) et B=(xB,yB) est calculée grâce à la formule :

$$\sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

Avant de faire un traitement quelconque sur les données, il est toujours intéressant de les visualiser lorsque c'est possible. Dans cet exemple, les données sont définies dans un espace à deux dimensions, donc elles peuvent être facilement visualisées sur deux axes comme dans la figure 10-2 ci-dessous.

Remarque

Même lorsque les données sont définies dans un espace à grande dimension, supérieur à deux ou à trois dimensions, il existe des méthodes qui permettent de les visualiser en deux ou trois dimensions, avec une perte d'informations qu'on espère minime. Ces méthodes sont appelées les méthodes de réduction de domaines. Le chapitre Analyse en composantes principales présente l'analyse du même nom, qui est l'une des méthodes de réduction de domaines les plus connues et qui permet d'avoir une vue en deux dimensions des données définies initialement dans un espace à grande dimension.

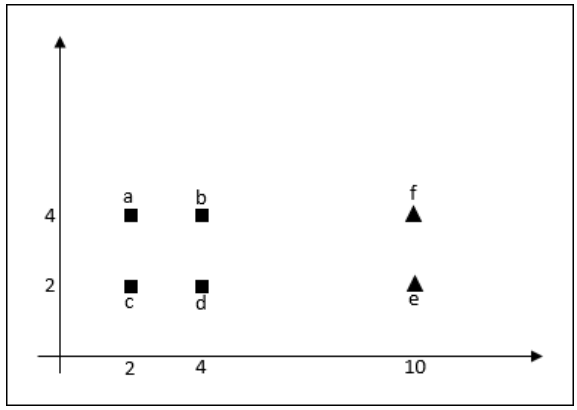

Figure 10-2 : représentation graphique en deux dimensions des données

Ce graphique montre clairement que les observations a, b, c et d, représentées par des carrés, sont très proches entre elles au sens de la distance euclidienne, par rapport aux deux observations e et f. Également, les deux dernières observations, représentées par des triangles, sont très proches entre elles.

Pour cet exemple, en se basant donc sur la distance euclidienne, un algorithme de clustering efficace proposerait certainement de répartir ces six observations dans les deux clusters C_1 et C_2 comme dans la figure 10-3 ci-dessous.

En effet, avec la distance euclidienne, cette répartition est optimale. La section suivante définit de façon plus formelle la notion de solution optimale pour un algorithme k-means et pour un nombre de clusters fixe.

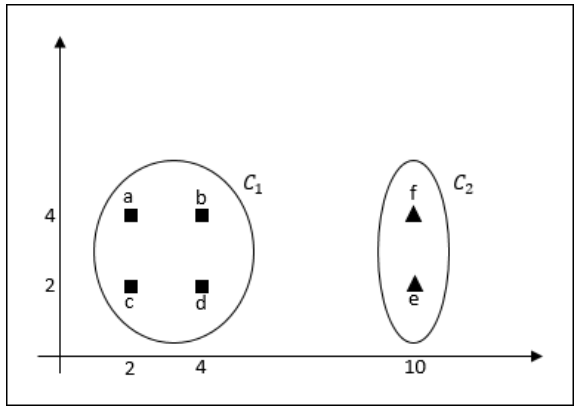

Figure 10-3 : répartition des six points dans les deux classes C_1 et C_2

En suivant les étapes classiques de l'algorithme k-means, le résultat optimal de la figure 10-3 peut être obtenu comme suit :

1. L'algorithme k-means commence initialement par sélectionner de façon aléatoire deux observations parmi les six observations disponibles. Les deux observations ainsi sélectionnées vont être considérées comme les centres des deux clusters recherchés C_1 et C_2. Ici, nous supposons que l'algorithme k-means recherche un nombre de clusters qui est égal à 2.

   Dans cet exemple, supposons que les deux observations a et d sont sélectionnées aléatoirement. Ces deux observations vont être considérées comme les centres respectifs des clusters C_1 et C_2. L'algorithme k-means calcule les distances entre chacune des six observations avec les centres a et d. Les résultats sont reportés dans le tableau ci-dessous :

Les centres des clusters	a	b	c	d	e	f
Centre de C_1=a=(2,4)	0	2	2	2.8284	8.2462	8
Centre de C_2=d=(4,2)	2.8284	2	2	0	6	6.3245

Figure 10-4 : distances entre les observations et les centres de C_1 et C_2

2. Une fois que k-means dispose de toutes les distances entre toutes les observations et les deux centres a et d, il procède à l'association entre les observations et les clusters. Par exemple, l'observation e va être associée au cluster C_2, puisqu'elle est plus proche du centre de C_2 que du centre de C_1. À la suite de cette étape, les deux clusters C_1 et C_2 vont être constitués comme suit $C_1 = \{a, b, c\}$ et $C_2 = \{d, e, f\}$

   Lorsqu'une observation est à la même distance des clusters C_1 et C_2, alors elle est affectée à l'un de ces deux clusters de manière aléatoire. Dans notre exemple nous avons affecté de manière arbitraire les deux observations b et c au cluster C_1.

   La figure suivante présente graphiquement les deux clusters C_1 et C_2 obtenus à la suite de cette étape :

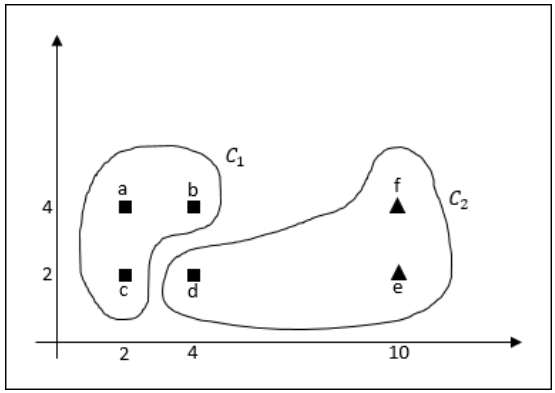

   Figure 10-5 : affectation des observations aux clusters C_1 et C_2 après la première itération

3. Lors de la première étape, l'algorithme k-means a choisi de façon aléatoire a et d en tant que centres des deux clusters recherchés. À l'étape 2, les deux clusters C_1 et C_2 ont été concrètement construits.

À cette étape, l'algorithme k-means calcule le centre du cluster C_1 en utilisant les observations a, b et c et calcule le centre du cluster C_2 en utilisant les observations d, e et f.

Ainsi : le centre de $C_1 = \left(\frac{1}{3}(2+2+4), \frac{1}{3}(4+4+2)\right) = (2.66, 3.33)$

le centre de $C_2 = \left(\frac{1}{3}(4+10+10), \frac{1}{3}(2+2+4)\right) = (8, 2.66)$

Les centres des deux clusters C_1 et C_2 sont représentés avec le symbole étoile dans la figure 10-6 ci-dessous :

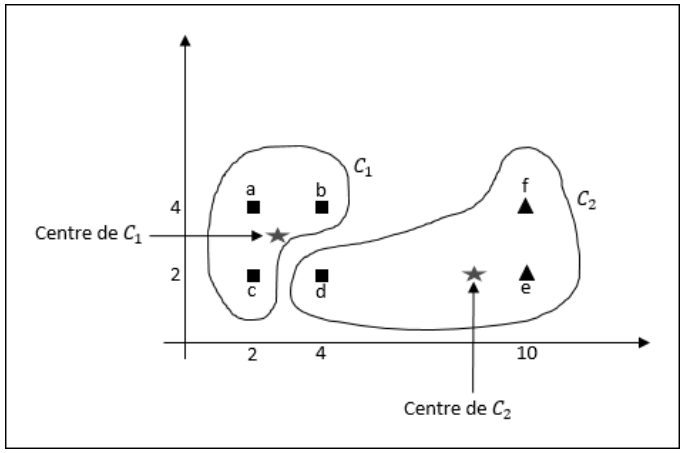

Figure 10-6 : les centres des clusters C_1 et C_2

4. À cette étape, les six observations sont réparties de nouveau sur les deux clusters C_1 et C_2 en se basant sur les nouveaux centres calculés à l'étape précédente. Les distances entre les observations et les centres de C_1 et C_2 sont reportées dans le tableau suivant :

Les centres des clusters	a	b	c	d	e	f
Centre de C_1= (2.66,3.33)	0.9404	1.4981	1.4847	1.8879	7.4595	7.3505
Centre de C_2= (8,2.66)	6.1478	4.2184	6.0361	4.0540	2.1060	2.4074

Figure 10-7 : distances entre les observations et les centres de C_1 et C_2