Projet :
Classifier automatiquement des biens de consommation

OpenClassrooms - Formation Data Science
Candidat - Nicolas Roux

Présentation du cas

- **Objectif** : Fournir un POC démontrant la faisabilité d'un algorithme de classification d'objets à partir d'image et de description
- **Données** : Données clients
- **Stack** : Python - Jupyter Notebook

Synthèse

A partir d'une base de données composée **d'images** et de **descriptions textuelles** correspondantes, nous développons donc des algorithmes de classifications s qui permettent de **regrouper des produits en catégories cohérentes**.

Pour opérer avec des données visuelles et textuelles, ces algorithmes seront divisés en trois grandes parties :

1. Extraction de **caractéristiques d'image**
2. Extraction de **caractéristiques de texte**
3. Entraînement de modèles de **clustering** à des fins de classifications

Deux méthodologies sont comparées : la première utilisant une méthodologie plus "**conventionnelle**" reposant sur des algorithmes précédent le réel avènement du data science, avec une analyse textuelle basée sur la fréquence des mots et une analyse d'image basée uniquement des caractéristiques locales de points d'intérêts identifiés, et une seconde basée sur des **réseaux de neurones pré-entraînés**.

Préparation des données

Texte (BOW-TFIDF)

Preprocessing

1. Transformation en minuscule
2. Suppression des stopwords anglais
3. Suppression autres mots/groupes de caractères superflus

Image

Data augmentation

Les images disponibles sont multipliées en y appliquant des transformations, dont la rotation, l'augmentation du contraste ou de la luminosité, afin d'augmenter le nombre d'observations disponibles avec la même description

Image scaling (cnn)

Afin d'accélérer l'extraction de caractéristiques, les tailles des images sont réduites pour la méthodologie basée sur les réseaux de neurones.

Méthodologie 1 - "Conventionnelle"

Preprocessed Text

Bag of words

Un dictionnaire est constitué de tous les mots du corpus et pour chaque description, contient le nombre de fois qu'un mot est apparu

TF-IDF

Pour chaque mot (colonne), on prend ensuite cette fréquence et la multiplie par l'inverse de la proportion de documents dans lesquels il apparaît

Image

SIFT

L'algorithme SIFT détecte et définit des caractéristiques d'image qui sont invariantes par rapport à la rotation ou la taille de l'image.

Concatenation

Les caractéristiques textuelles sont ajoutées aux caractéristiques visuelles de telle sorte à avoir un vecteur unique représentant la combinaison de chaque images et de sa description

Réduc. Dim

On simplifie la table de données en gardant un nombre de dimensions permettant d'expliquer 90% de la variance des données

Clustering

Les vecteurs sont ensuite passé dans un algorithme de clustering pour identifier des clusters distincts et ainsi avoir un algorithme de classification

Méthodologie 2 - Réseaux de neurones

Preprocessed Text

BERT

En faisant usage de transfert d'apprentissage, on utilise BERT, une approche bi-directionnelle, pour produire une vectorisation du contenu de chaque description.

Image

VGG16

En faisant usage de transfert d'apprentissage, le VGG16 produit des caractéristiques d'images en prenant le vecteur produit en avant-dernière couche.

Concatenation

Les caractéristiques textuelles sont ajoutées aux caractéristiques visuelles de telle sorte à avoir un vecteur unique représentant la combinaison de chaque images et de sa description

Réduc. Dim

On simplifie la table de données en gardant un nombre de dimensions permettant d'expliquer 90% de la variance des données

Clustering

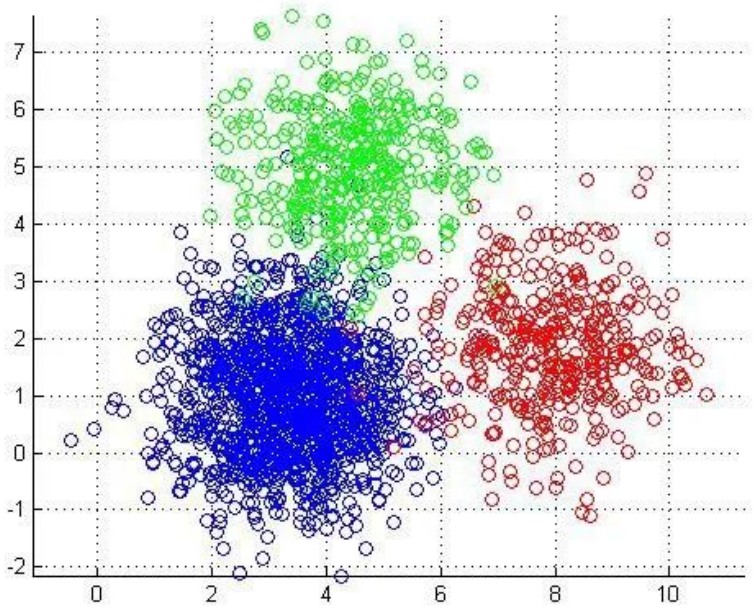
Les vecteurs sont ensuite passé dans un algorithme de clustering pour identifier des clusters distincts et ainsi avoir un algorithme de classification

Clustering 1 - KMeans Clustering

Algorithme de clustering par excellence, KMeans présente un avantage important, particulièrement dans des domaines du marketing car conceptuellement simple et **facile à expliquer** :

1. Sélection du **nombre de clusters**
2. **Attribution aléatoire** des observations aux différents clusters
3. **Ré-évaluation** de l'appartenance des observations aux différents clusters selon une métrique de distance
4. Répétition de l'étape 3 jusqu'à ce qu'une limite d'itérations soit atteinte ou qu'un **minimum locale** soit atteint.

L'algorithme est relativement simple et compatible avec des données très importantes. La méthode nécessite cependant de préciser le nombre de clusters et l'attribution aléatoire initiale des observations dans des clusters mènent à des résultats variants en fonction de la "seed".

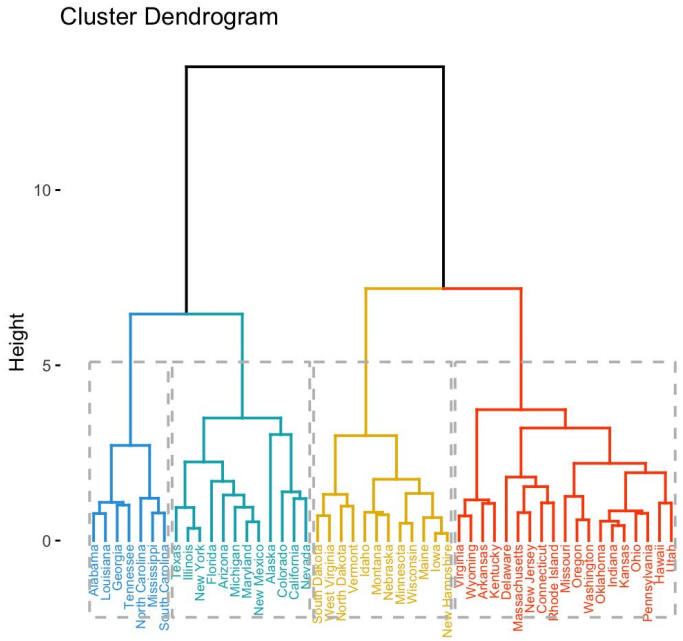


Clustering 2 - Regroupement hiérarchique

Le regroupement hiérarchique est une approche de clustering qui construit itérativement des clusters à partir du **regroupement de clusters**.

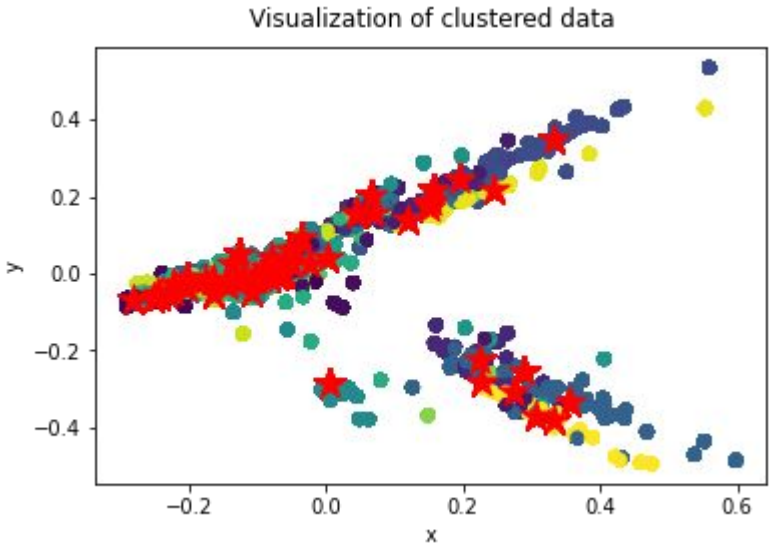
Au début de l'algorithme, toutes les observations sont un cluster séparé et ceux-ci sont progressivement regroupés en fonction de leur proximité l'un à l'autre.

Les avantages de cette méthode comptent la **réplicabilité** de la méthode et une **compatibilité** avec la plupart des métriques de distance / similarité, au prix d'avoir un coût computationnel important.



Performance Conventionnelle - kmeans

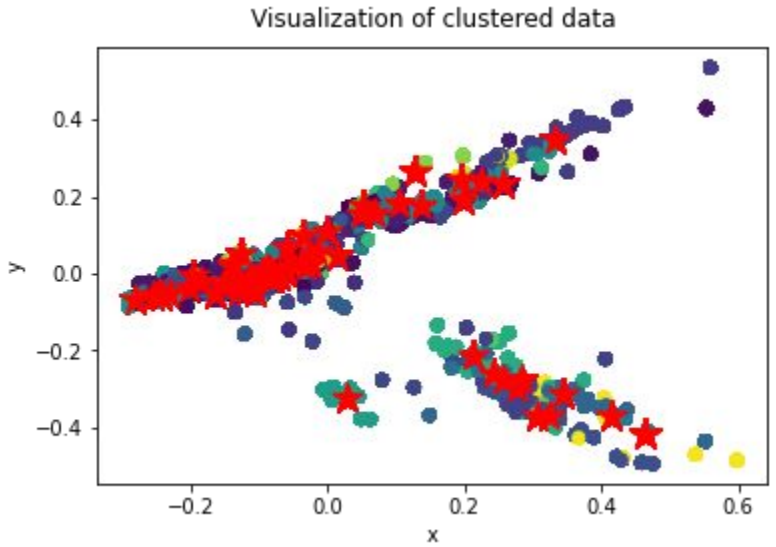
Clusters sur pca 2-dimensions



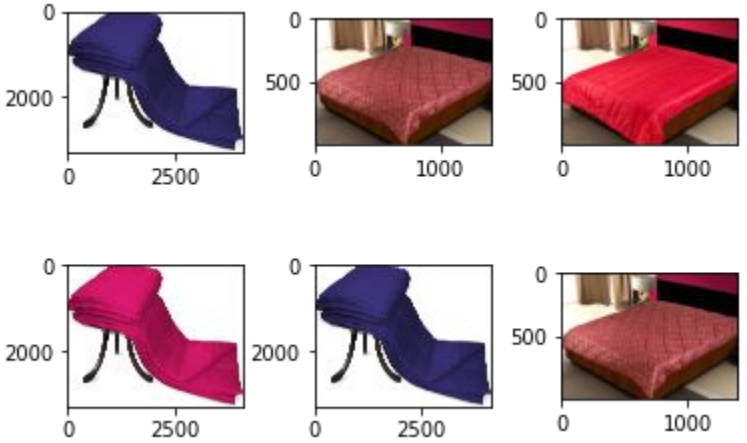
Sélection aléatoire d'un cluster

Performance Conventionnelle - Regroupement hiérarchique

Clusters sur pca 2-dimensions

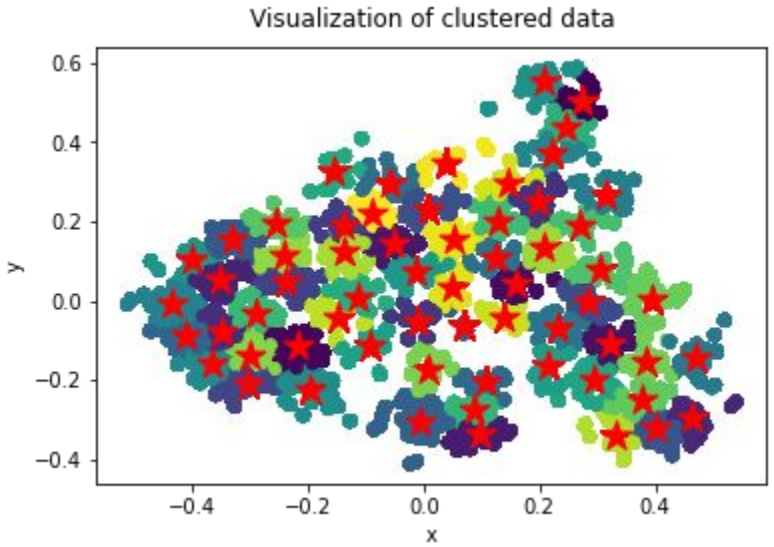


Sélection aléatoire d'un cluster

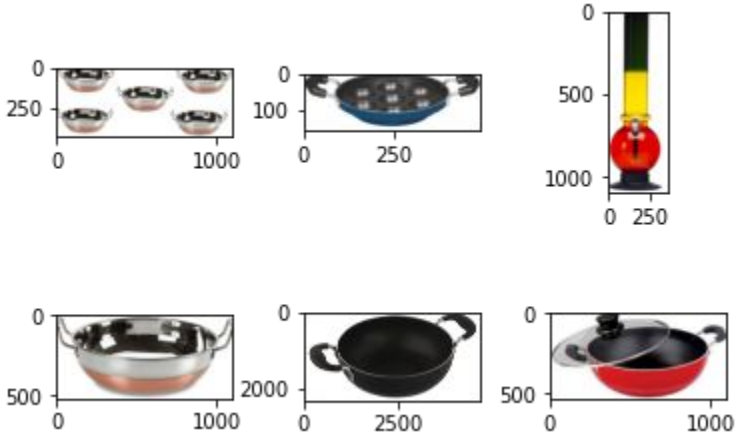


Performance réseaux neurones - kmeans

Clusters sur pca 2-dimensions

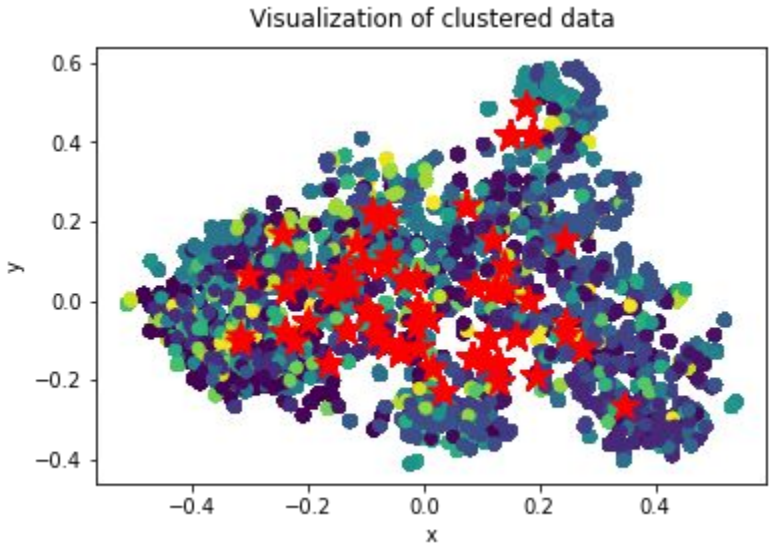


Sélection aléatoire d'un cluster

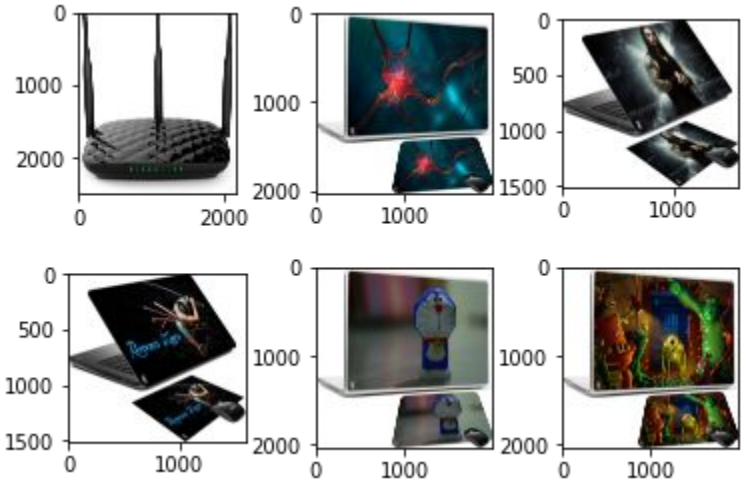


Performance réseaux neurones - Regroupement hiérarchique

Clusters sur pca 2-dimensions



Sélection aléatoire d'un cluster

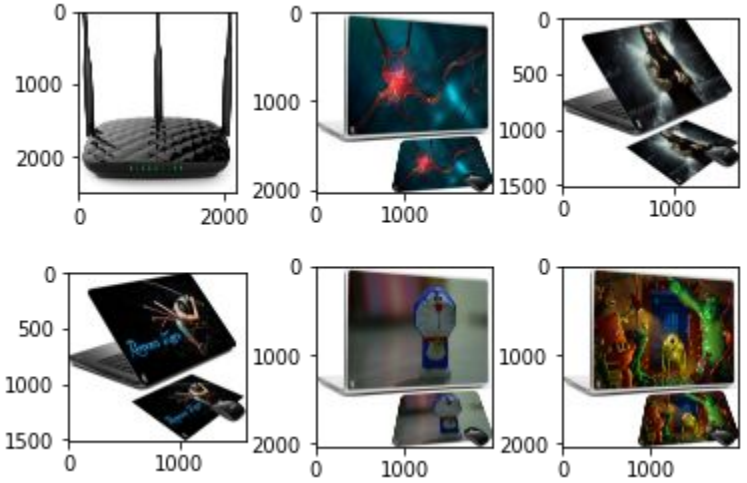


OBJ

Performance réseaux neurones - Regroupement hiérarchique

UMAP DIMENSIONALITY REDUCTION

Sélection aléatoire d'un cluster



OBJ

Comparaison des clusters

	Km conv	Ac conv	Km cnn	Ac cnn
Km conv	1.0	0.4362705611835603	0.17123512741648497	0.15623797100856737
Ac conv	0.4362705611835603	1.0	0.15925368943324592	0.14655392119149163
Km cnn	0.17123512741648497	0.15925368943324592	1.0	0.09920806068331031
Ac cnn	0.15623797100856737	0.14655392119149163	0.09920806068331031	1.0

Un ARI score deux à deux montre qu'on obtient des compositions de clusters différents. Parmi les potentielles causes sont :

- Un nombre de clusters (ici 62) trop élevé ou trop bas
- Des clusters sous représentés (des catégories produits représentés par un nombre très bas de produits)
- Les comparaisons de clusters ont été faits sur une seule itération

Conclusion POC

Avec relativement peu de données, peu de retraitement de données et des modèles assez simples, et une approximation du nombre de clusters, on parvient à regrouper les observations en clusters qui contiennent manifestement des objets de types similaires.

Pour obtenir une seconde couche de vérification, nous pourrions comparer les nouveaux clusters avec la catégorisation initiale (estimée peu fiable) afin de voir si on retrouve les mêmes objets dans les mêmes clusters.

Une prochaine étape serait d'augmenter la taille de l'échantillon afin de se mettre dans une contexte plus réaliste.

Finalement, la création d'un moteur de classification devrait être mis en production.

Améliorations projet

1. Data augmentation :
   a. les images peuvent encore être augmentées afin d'enrichir les données d'entraînement
2. CNN :
   a. D'autres algorithmes de CNN pourrait être utilisés
   b. Utiliser des modèles pré-entrainés sur des images plus pertinentes
3. NLP :
   a. Pour la méthode utilisant tf-idf, une personnalisation de la liste des stop words pourrait être intéressante
   b. Utiliser d'autres modèles d'embeddings


MERCI