

MOVIES



RECOMMENDATION ENGINE

UNE BASE DE DONNÉES PUBLIQUE D'INFORMATIONS SUR DES :

FILMS

Créer un moteur de recommandations de films, basé sur le contenu. Comment a-t-on traité ce sujet? Quelles approches va-t-on adopter ?

Sommaire :

1 - NETTOYAGE DATASET :

- Taux de remplissage
- Etudes de corrélation
- Taux de remplissage après le premier nettoyage
- Traitement des valeurs manquantes :
 - Variables quantitatives (remplacement par la moyenne ...)
 - Variables catégorielles

2 - ANALYSE EXPLORATOIRE :

- Analyse univariée :
 - la répartition du IMDB SCORE par rapport aux films
 - Movie facebook like
- Analyse bivariée :
 - Note IMDB par genre de film

3 - PRÉPARTIONS DE DONNÉES

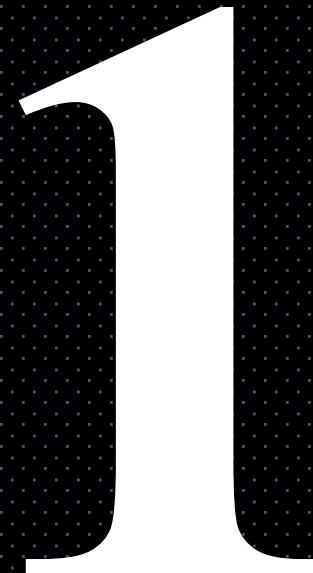
- Variables quantitatives :
 - Isoler les données numériques, qui ne demandent pas plus de traitement.
- Variables catégorielles :
 - Rendre binaire les données catégorielles OneHotencoding ...

4 - MOTEURS DE RECOMMANDATIONS

- Première recommandation (similarité de cosine)
- Distances
- KNN (**NearestNeighbors**)
- K-means Clustering
- Hierarchical Clustering

NETTOYAGE DATASET :

Présentation du Sujet
Traitement des données



5043

FILMS

28

VARIABLES

	color	director_name	num_critic_for_reviews	duration	director_facebook_likes	actor_3_facebook_likes	actor_2_name	actor_1_facebook_likes	gross
movie_title									
	Avatar	Color	James Cameron	723.0	178.0	0.0	855.0	Joel David Moore	1000.0 760505847.0
Pirates of the Caribbean: At World's End		Color	Gore Verbinski	302.0	169.0	563.0	1000.0	Orlando Bloom	40000.0 309404152.0
Spectre	Color	Sam Mendes		602.0	148.0	0.0	161.0	Rory Kinnear	11000.0 200074175.0
The Dark Knight Rises	Color	Christopher Nolan		813.0	164.0	22000.0	23000.0	Christian Bale	27000.0 448130642.0
Star Wars: Episode VII - The Force Awakens	Nan	Doug Walker		Nan	Nan	131.0	Nan	Rob Walker	131.0 Nan

Figure 1

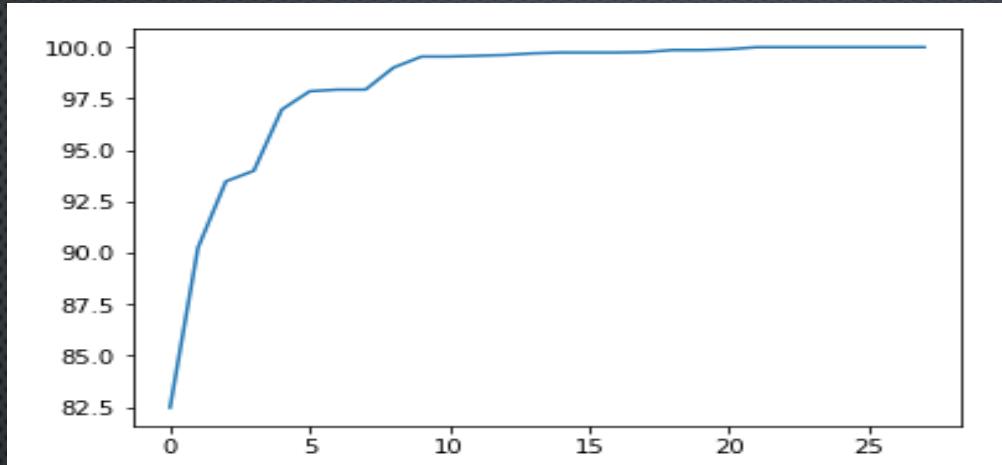


Figure 2 : Remplissage avant toutes opérations de nettoyages

- TX de Remplissage Total = 98.09 %
- Supprimer les doublons
- TX de Remplissage Total (Sans doublon) = 98.072 %

Traiter les valeurs manquantes :

Variables catégorielles :

- Remplacer les valeurs manquantes par des chaînes de caractères vides

Variables numériques (2 cas) :

Cas 1 : Remplacer les valeurs manquantes par 0 :

- Actor_1_facebook_likes - num_critic_for_reviews
- facenumber_in_poster - movie_facebook_likes

Cas 2 : Remplacer par la moyenne :

- Title years
- Duration

□ Etudes de corrélations :

- 'cast_total_facebook_likes' ≈ 'actor_1_facebook_likes' ($\alpha = 0.95$)
- 'facenumber_in_poster', 'aspect_ratio' sont décorrélées de l'ensemble des autres variables ($\alpha < 0.2 \forall$ couple)

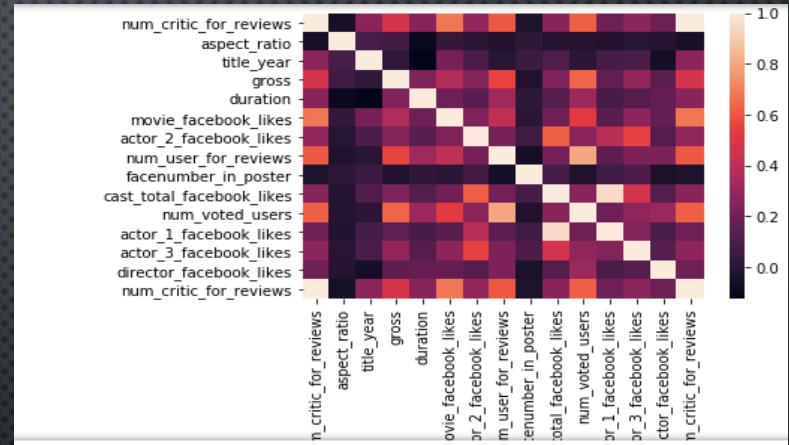


Figure 3 : Matrice de corrélations variables numériques

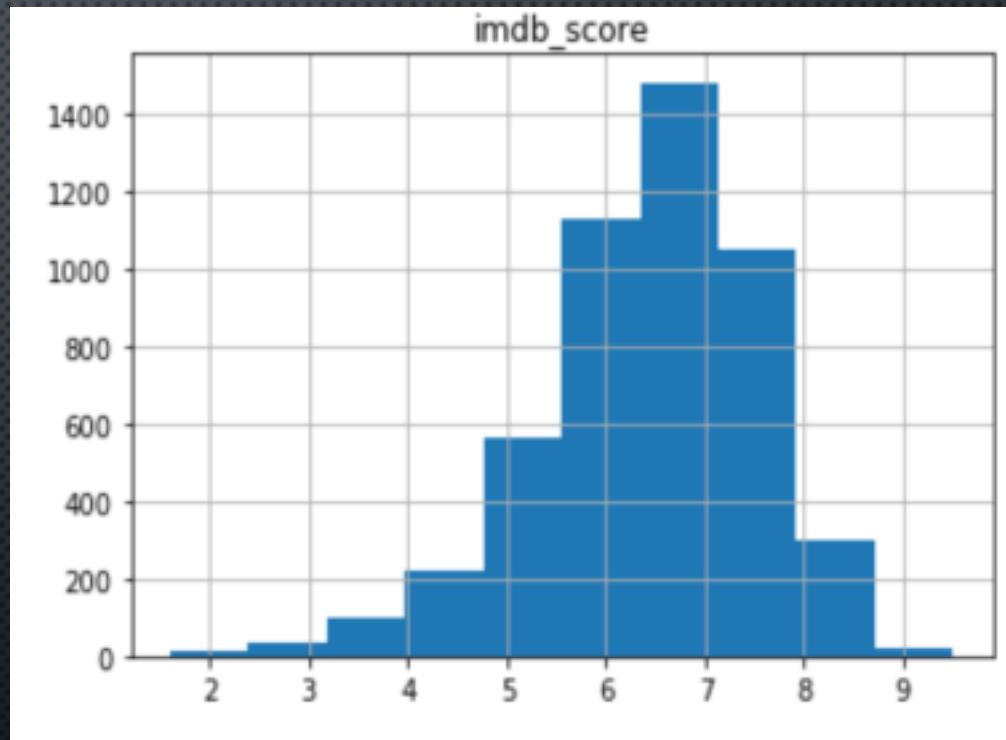
ANALYSE EXPLORATOIRE -

Traitement des variables catégorielles :

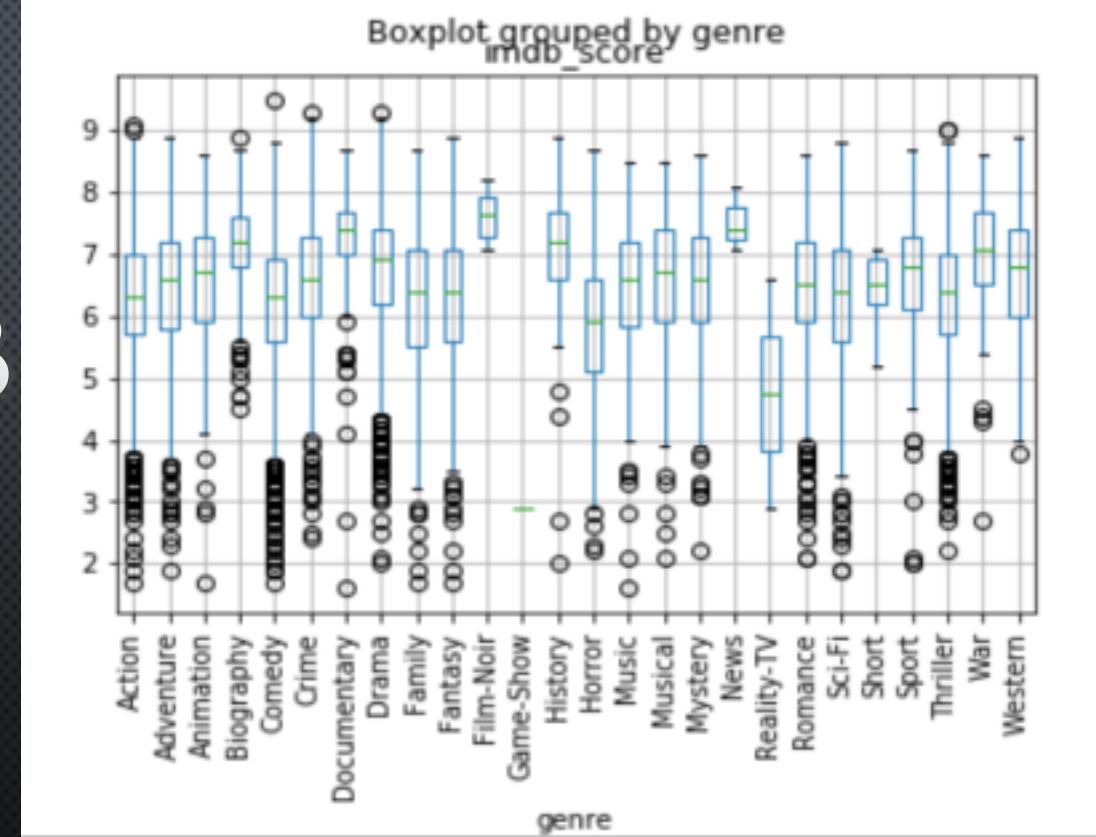
- Noms des réalisateurs
- Acteurs
- Genres
- Content Rating
- Pays

2

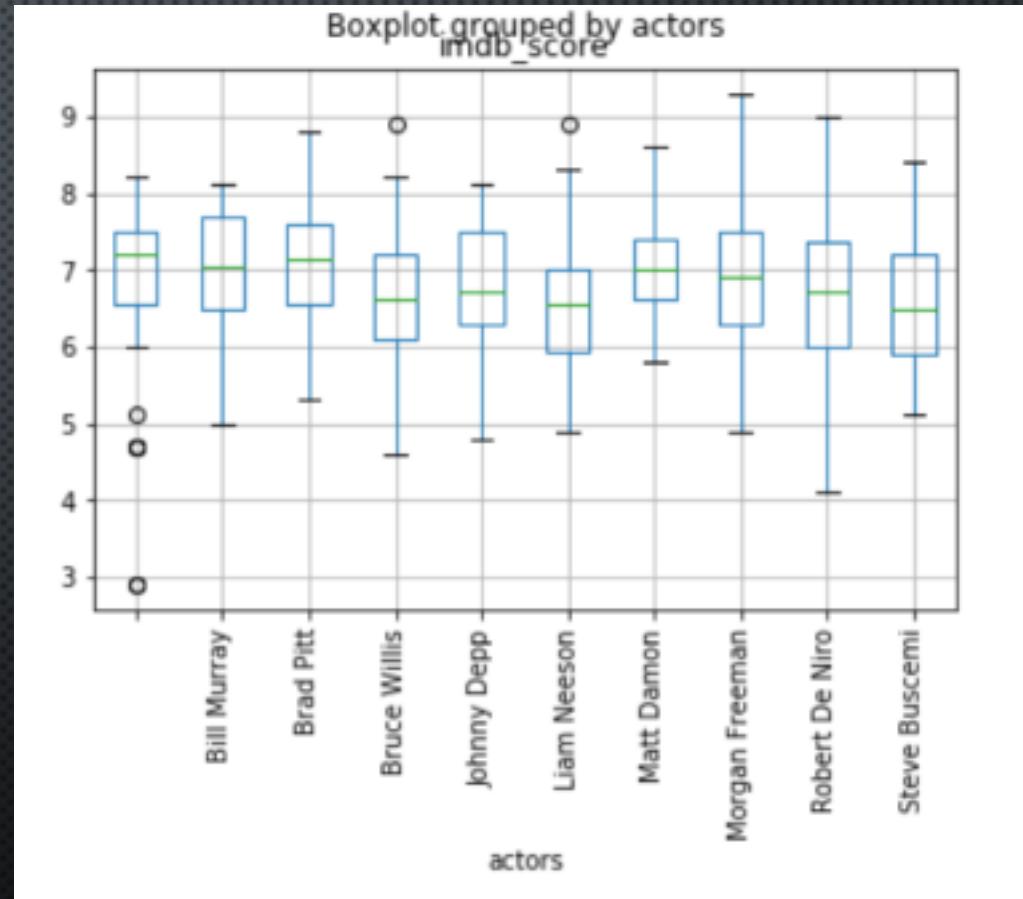
SCORE IMDB



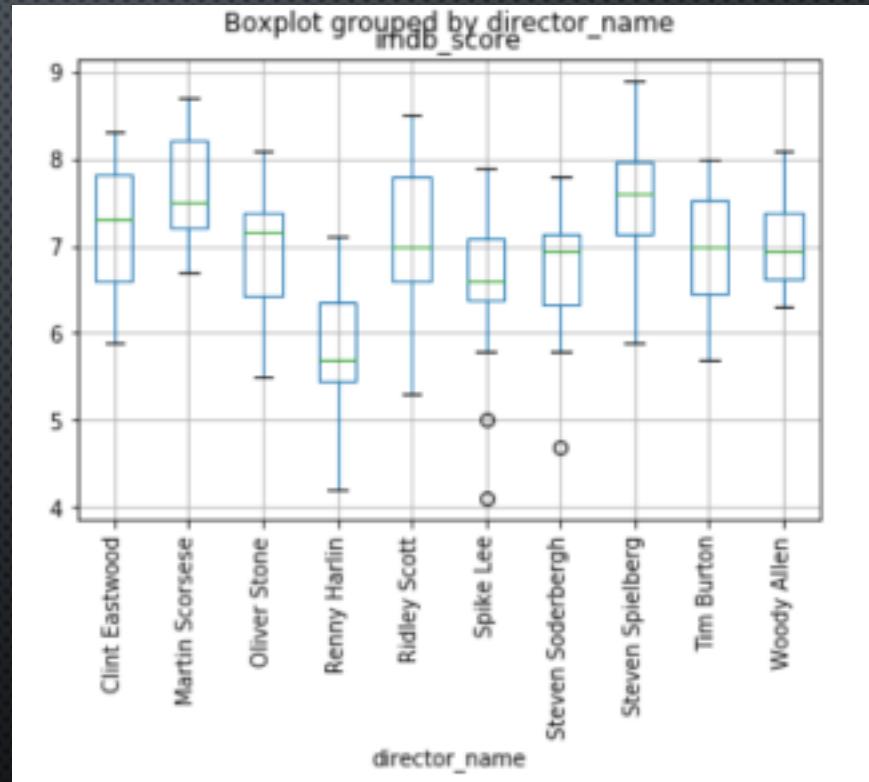
SCORE IMDB PAR GENRE



SCORE IMDB PAR ACTEUR



SCORE IMDB PAR RÉALISATEUR



PRÉPARATION DE DONNÉES :

- Variables catégorielles
- Variables quantitatives

3

Variables catégorielles :

- Binairiser les variables catégorielles pour pouvoir appliquer nos algorithmes et calculer la distance. Méthodes : **OneHotEncoding** , **Get_dummies** ...

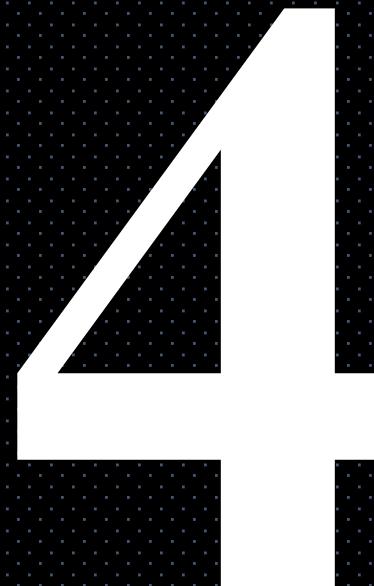
	genre_Action	genre_Adventure	genre_Animation	genre_Biography	genre_Comedy	genre_Crime	genre_Documentary	genre_Drama	genre_Family
0	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0
5	1	1	0	0	0	0	0	0	0
6	0	1	1	0	1	0	0	0	1

Variables quantitatives :

- Standardiser les variables X in [0 : 1] afin d'homogénéiser nos variables, pour appliquer nos algorithmes.

RECOMMENDATIONS DE FILMS :

- Distances
- KNN (**NearestNeighbors**)
- K-means Clustering
- Hierarchical Clustering



4917

FILMS

276

VARIABLES

DISTANCES

Euclidian

$$d(\mathbf{p}, \mathbf{q}) = d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

MINKOWSKI

$$D(X, Y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}$$

Manhattan

$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^n |p_i - q_i|,$$

CHEBYSHEV

$$d(A, B) = \max_{i \in [[0, n]]} (|A_i - B_i|).$$

DISTANCES

Index	euclidian	manhattan	chebyshev	minkowski
('Avatar \xa0',)	The Lord of the Rings: Th...	Man of Steel	Titanic	Titanic
("Pirates of the Caribbean...")	Pirates of the Caribbean: De...			
('Spectre \xa0',)	Skyfall	Skyfall	Skyfall	Skyfall
('The Dark Knight Rises\...")	Inception	Inception	American Hustle	American Hustle
('Star Wars: Episode VII ---")	Crop Circles: Quest for Tru...	Crop Circles: Quest for Tru...	Ghost Hunters	Crop Circles: Quest for Tru...
('John Carter \xa0',)	Green Lantern	Green Lantern	Green Lantern	Green Lantern
('Tangled \xa0',)	Enchanted	Enchanted	Enchanted	Enchanted
('Avengers: Age of Ultron...")	The Avengers	The Avengers	The Avengers	The Avengers
('Harry Potter and the Half-...")	Harry Potter and the Priso...			
('Batman v Superman: Dawn...")	Man of Steel	Man of Steel	The Hunger Games	The Hunger Games
('Superman Returns\xa0',)	2012	2012	X-Men	The Matrix Reloaded
('Quantum of Solace\xa0',)	The World Is Not Enough			
("Pirates of the Caribbean...")	Pirates of the Caribbean: At...			
('The Lone Ranger\xa0',)	The Green Hornet	Pirates of the Caribbean: On...	The Patriot	The Debt

KNN(NearestNeighbors) RECOMMANDATION :

- THE LORDS OF SALEM
- THE CRAFT
- DON'T BE AFRAID OF THE DARK
- PARANORMAL ACTIVITY: THE MARKED ONES
- QUEEN OF THE DAMNED

NB : K = 6 puisque le KNN donne toujours comme premier résultat le film lui même.

K-MEANS

Principe : Étant donné un ensemble de points $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$, on cherche à partitionner les n points en k ensembles $\mathbf{S} = \{S_1, S_2, \dots, S_k\}$ ($k \leq n$) en minimisant la distance entre les points à l'intérieur de chaque partition

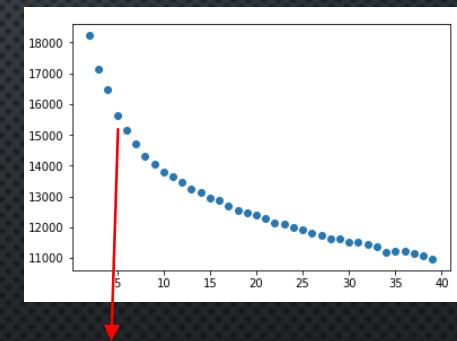
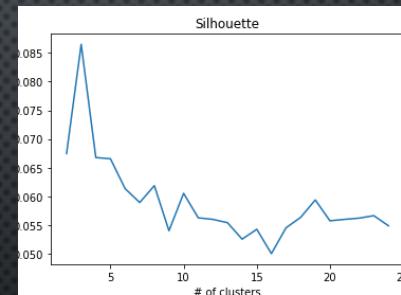
NB : Certes pour minimiser l'inertie intra-clusters, on doit choisir le nombre de clusters optimal ($N_{\text{clusters}} = 4$). Cependant cette approche ne répond pas à notre problème, avec $N_{\text{clusters}}=4$, on va avoir un nombre moyens très élevé de films dans chaque cluster, en effet choisir un nombre de clusters relativement élevé va nous permettre d'avoir un nombre qui est proche de 5 (nombre de films à recommandés) par cluster.

Problème : il y aura des clusters avec moins de 6 films et d'autres avec plus de 6 film

- Solution : utilisation de KNN pour les clusters avec plus de 6 films pour recommander les 5 plus proches, et garder les clusters inférieure ou égale 6

- Inconvénient avoir des recommandations avec moins de 5 films

Je choisi $N_{\text{clusters}} = 800$ ($\text{nb_moyen_film_cluster}=7$



$N_{\text{clusters}} = 4$

Meilleur coefficient de silhouette équivalent de $N_{\text{clusters}} = 4$

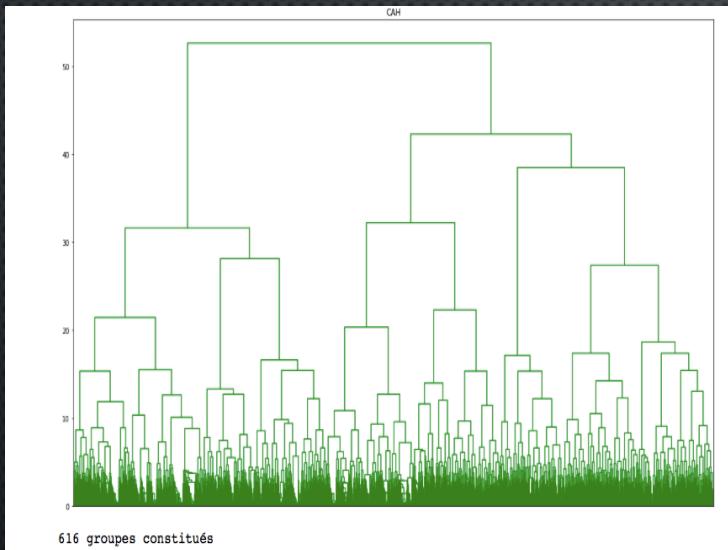
Figures 2 : Choix du nombre de clusters
: Observer l'inertie de notre algorithme ou le wcss (Within Cluster Sum of Squares)(méthode coude)

K-MEANS RECOMMANDATION :

- UNFAITHFUL
- INSTINCT
- UNDERWORLD AWAKENING
- DARK WATER
- COLLATERAL DAMAGE

CHA (ascendant hierarchical clustering) :

Définition : La classification ascendante hiérarchique est dite ascendante car elle part d'une situation où tous les individus sont seuls dans une classe, puis sont rassemblés en classes de plus en plus grandes.



Dendrogramme, 616 groupes , t=3

Après essai pour découper mon dendrogramme afin de trouver un nombre moyen d'individus dans chaque clusters qui est proche de 5 j'ai reparti mes films sur 616 clusters(avec 616 je vais avoir des clusters avec au moyen 7 films : je voulais pas découper encore plus parce que sinon je vais avoir des clusters avec juste un seul film) cependant j'ai des clusters qui contienne plus que 5 et aussi des clusters avec moins de 5, j'ai utilisé le KNN pour avoir les film les plus proches dans les clusters avec plus de 6 films, j'ai gardé les clusters qui contienne moins de 6 films .

CHA RECOMMANDATION :

- BLAKHAT
- HOP
- KAGLE AYE
- HARRY POTTER AND THE ORDER OF THE PHOENIX
- FANTASTIC 4 : RISE OF THE SILVER SURFER

EVALUATION & CHOIX D'ALGORITHME :

Méthode : Pour choisir le meilleur algorithme par rapport à notre sujet, j'ai décidé de calculer le Matching rating moyenne , entre 3 recommandation exemplaire(vérités terrains) que j'ai construite à base des films de superhéros(films très proches entre eux) , série de films HARRY POTTER , X-MEN, et les différentes recommandations .

Model : films de superhéros qu'ils sont proches entre eux :

ID 27 Titre : Spider-Man 2

ID19 Titre : The Amazing Spider-Man

ID10 Titre : Superman Returns

ID16 Titre : Batman v Superman: Dawn of Justice

ID 28 Titre : Iron Man 3

Model1=[27,19,10,16,28]

Mode3 : série de film X_Men :

ID 29 Titre : X-Men: The Last Standid

ID 192 Titre : X-Men 2id

ID 110 Titre : X-Men Origins: Wolverineid

ID 218 Titre : The Wolverine

ID 486 Titre : X-Men

Model3 =[29,192,110,218,486]

Mode2 : série de films de Harry Potter :

ID 177 Titre : Harry Potter and the Prisoner of Azkabanid

ID 260 Titre : Harry Potter and the Chamber of Secretsid

ID 102 Titre : Harry Potter and the Goblet of Fireid

ID 101 Titre : Harry Potter and the Order of the Phoenixid

ID 8 Titre : Harry Potter and the Half-Blood Prince

Model2 =[177,260,102,101,8]]

▫ KNN : 20%

>

▫ CHA : 6.6 %

>

▫ K-MEANS : 0.0%

TAUX DE MATCHING MOYEN RETENU (ID aléatoire = 192) : 20% (KNN)

FLASK API : RECOMMENDER OF MOVIES:

- Création de l'api avec le framework Flask.
- Sur la page d'accueil une liste des film avec les "id" à choisir.
- Recherche lancée, le serveur cherche les 5 points les plus proches basés sur les résultats K-MEANS retourne ces données à un template pour l'affichage.
Le site est disponible à cette adresse:
<http://agrandik.pythonanywhere.com/>

FLASK API : RECOMMENDER OF MOVIES:

27

Non sécurisé | agrandik.pythonanywhere.com

film recommandation

Choisir un Id qui correspond à votre film

	Recommander	movie_title
0	Avatar	
1	Pirates of the Caribbean: At World's End	
2	Spectre	
3	The Dark Knight Rises	
4	Star Wars: Episode VII - The Force Awakens ...	
5	John Carter	
6	Tangled	
7	Avengers: Age of Ultron	
8	Harry Potter and the Half-Blood Prince	
9	Batman v Superman: Dawn of Justice	
10	Superman Returns	
11	Quantum of Solace	
12	Pirates of the Caribbean: Dead Man's Chest	
13	The Lone Ranger	
14	Man of Steel	
15	The Chronicles of Narnia: Prince Caspian	
16	Pirates of the Caribbean: On Stranger Tides	
17	Men in Black 3	
18	The Hobbit: The Desolation of Smaug	
19	The Amazing Spider-Man	
20	Robin Hood	
21	The Hobbit: The Desolation of Smaug	
22	The Golden Compass	
23	Titanic	
24	Captain America: Civil War	
25	Battleship	
26	Jurassic World	

Non sécurisé | agrandik.pythonanywhere.com

	movie_title
4590	Crop Circles: Quest for Truth
4711	Sisters in Law
4712	Ayurveda: Art of Being
4576	Starsuckers
4595	The Harvest/La Cosecha

CONCLUSION