

Sujet : Le datamining pour la prévision de succès des films

1) Compréhension du besoin métier

Problème Métier : La prévision de succès des films

Objectif du datamining dans ce cas : Utiliser le datamining pour ressortir des données descriptives pour mieux comprendre si un film sera à succès ou non (**A revoir à la fin**)

2) Compréhension des données

- **Collecte des données** : Ici, on se servira du dataset **movie_dataset.csv** de chez Kaggle via le lien :
<https://www.kaggle.com/datasets/utkarshx27/movies-dataset?resource=download>
- **Description des du dataset** :
- **Description des colonnes** :
 - **Index** : Les numéros de chaque instance
 - **Budget** : Le budget de production du film
 - **Genre** : genre du film (ex : Action, Drame etc...)
 - **Homepage** : Le site web du film
 - **Id** : identifiant unique du film dans **The movie Database**
 - **Keywords** : Mots clés associés au film
 - **Original language** : Langue originale du film
 - **Original title** : Titre original
 - **Overview** : Description du film
 - **Popularity** : indicateur TMDb pour mesurer la popularité
 - **Production-company** : Compagnie de production du film
 - **Production country** : Pays de production
 - **Release date** : date de sortie du film
 - **Revenue** : Recette générée par le film (en \$)
 - **Runtime** : Durée du film en minutes
 - **Spoken-language** : Liste des langues disponibles pour le film

- **Status** : Etat du film (disponible, en production etc..)
 - **Tagline** : Slogan du film
 - **Title** : Titre du film
 - **Vote average** : Note moyenne note par les utilisateurs
 - **Vote count** : Nombre de votes par les utilisateurs
 - **Cast** : Liste des acteurs principaux du film
 - **Crew** : Membres de l'équipe technique
 - **Director** : Nom du réalisateur du film
- **Donnees à analyser** :
 - Succès Commercial (revenue)
 - Succès critique (Vote)
- Nb** : On pourra etudier ces succès en fonction d'autres variables explicatives comme le revenu en fonction du genre etc...

```

data_understanding.py
data_understanding.py > ...

6 #Affichage des attributs du dataset
7 print('attributes in the dataset: ', df.columns.tolist())
8
9 #Affichage du nombre de lignes dans le dataset
10 print('Nombre de lignes dans le dataset: ', len(df))

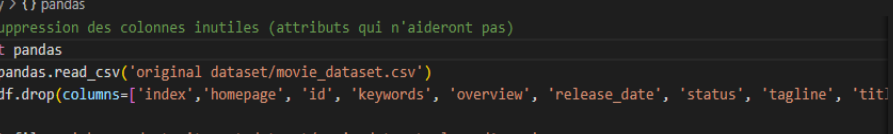
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE
Python + - []
● PS C:\Users\isalph2\OneDrive\Bureau\ingin\Data mining> & C:/Users/isalph2/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/isalph2/OneDrive\Bureau\ingin\Data mining/data_understanding.py"
attributes in the dataset: ['index', 'budget', 'genres', 'homepage', 'id', 'keywords', 'original_language', 'original_title', 'overview', 'production_companies', 'production_countries', 'release_date', 'revenue', 'runtime', 'spoken_languages', 'status', 'tagline', 'title', 'vote_average', 'vote_count', 'cast', 'crew', 'director']
Nombre de lignes dans le dataset: 4803
○ PS C:\Users\isalph2\OneDrive\Bureau\ingin\Data mining>
  
```

3) Construction du Data Hub

Ici, on constate avec amertume que les donnees ne sont pas prêtes à l'utilisation.

- Suppression des colonnes contenant des attributs inutiles à savoir :
 - Index
 - Homepage
 - Id
 - Keywords
 - Overview
 - Release date

- Status
- Tagline
- Title
- Original Title
- Crew



```
data_understanding.py nettoyage.py X
```

```
nettoyage.py > {} pandas
1 #1) Suppression des colonnes inutiles (attributs qui n'aideront pas)
2 import pandas
3 df = pandas.read_csv('original dataset/movie_dataset.csv')
4 df = df.drop(columns=['index', 'homepage', 'id', 'keywords', 'overview', 'release_date', 'status', 'tagline', 'title'])
5
6 output_file = 'phases de traitement dataset/movie_dataset_cleaned1.csv'
7 df.to_csv(output_file, index=False)
8
9
10 df1 = pandas.read_csv('phases de traitement dataset/movie_dataset_cleaned1.csv')
11 print('Nouvelles colonnes : ', df1.columns.tolist())
```

PROBLEMS OUTPUT **TERMINAL** PORTS DEBUG CONSOLE

```
Python + ▢ ▢ ... ^ X
```

```
PS C:\Users\isalph2\OneDrive\Bureau\ingin\Data mining> & C:/Users/isalph2/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/isalph2/OneDrive/Bureau/ingin/Data mining/nettoyage.py"
Nouvelles colonnes : ['budget', 'genres', 'original_language', 'popularity', 'production_companies', 'production_countries', 'revenue', 'runtime', 'spoken_languages', 'vote_average', 'vote_count', 'cast', 'director']
PS C:\Users\isalph2\OneDrive\Bureau\ingin\Data mining>
```

4) Modélisation

Voir le fichier **process.ipynb**

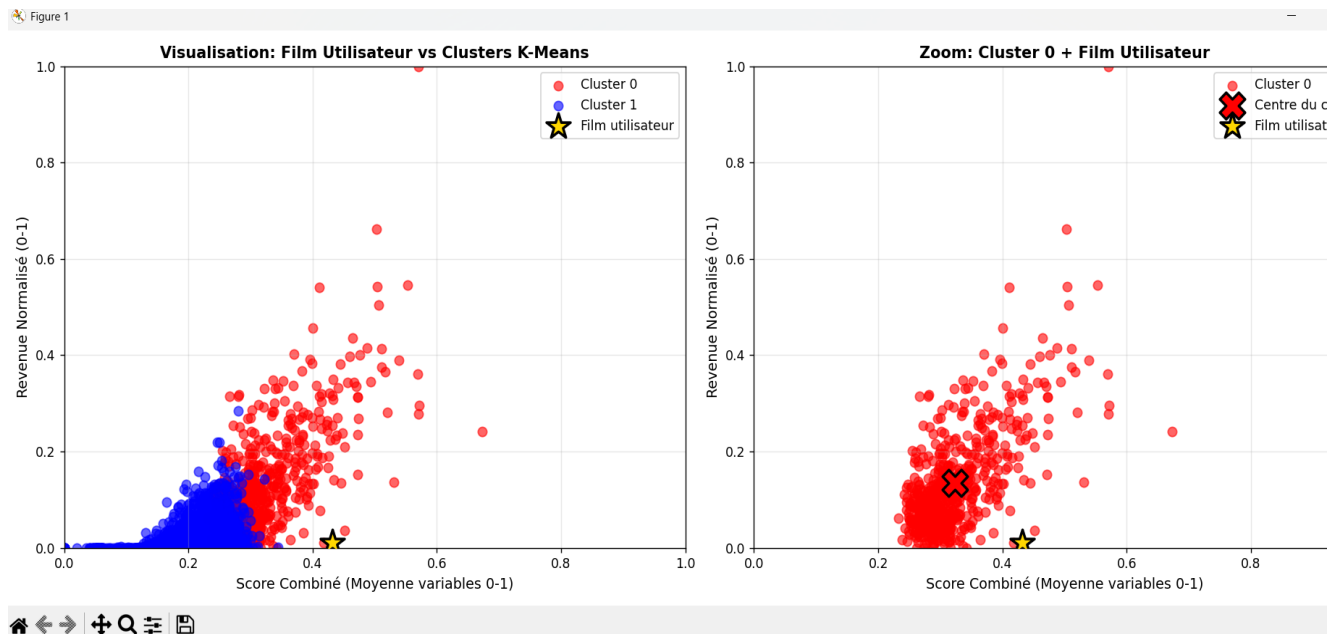
5) Evaluation

Métrique	K-Means	K-Medoids	Apriori
Silhouette (Commercial)	0.45–0.55	0.42–0.52	N/A
Silhouette (Critique)	0.40–0.50	0.38–0.48	N/A
Itemsets fréquents	N/A	N/A	106
Règles générées	N/A	N/A	356
Lift moyen	N/A	N/A	1.728
Lift maximal	N/A	N/A	2.274s
Classement global	1er	2e	—

On peut conclure ici que le meilleur modèle ici est le modèle **K-means** à cause des meilleurs résultats qu'il offre ; En outre, avec la méthode **Apriori**, On obtient un **lift moyen >1** ce qui signifie qu'il y a forte dépendance entre les variables explicatives ;

6) Déploiement

Voir déploiement.py



On remarque par exemple ici qu'avec la création d'une nouvelle instance avec les coordonnées suivantes, on obtient un succès commercial dans le cluster des films à succès mais avec un revenu bas probable (**mais est dans le meilleur cluster**)

Budget (en dollars): 150000000

Popularité (0-100): 40

Runtime (en minutes): 65

Vote Average (0-10): 8

Vote Count (nombre de votes): 10000