

Analyse Data Animes

Par Mohamed Ali BAMBA





Score Éditorial Anime

Une plateforme de streaming d'animés souhaite mettre en avant rapidement des animés de très haute qualité sans attendre des mois de données d'engagement utilisateur.

Objectifs

1.

Identifier rapidement des animés "à forte valeur éditoriale" à partir de données limitées pour une plateforme de streaming.

2.

Pouvoir fournir à travers l'analyse de ces données des animés de qualités



None Count		0	None Count		0	None Count		0	None Count		0	None Count		0	None Count		0
59			44			Manga			Fin			Madhouse			19.7%		
Unique values			Unique values			Original			En cours			Studio Pierrot			13.1%		
						Other			En pause			Other			67.2%		
0 Naruto			Action / Shonen			Manga			220 Fin			Studio Pierrot			2002		
1 Trigun			Sci-Fi / Action			Manga			26 Fin			Madhouse			1998		
2 Dragon Ball Z			Action / Shonen			Manga			291 Fin			Toei Animation			1989		
3 Fullmetal Alchemist: Br...			Aventure / Military			Manga			64 Fin			Bones			2009		
4 Cowboy Bebop			Sci-Fi / Noir			Original			26 Fin			Sunrise			1998		
5 Sword Art Online			Isekai / Romance			Light Novel			25 Fin			A-1 Pictures			2012		
6 Re:Zero			Isekai / Psychologique			Light Novel			50 Fin			White Fox			2016		
7 Dr. Stone			Sci-Fi / Adventure			Manga			57 En cours			TMS Entertainment			2019		
8 One Piece			Adventure / Shonen			Manga			1090 En cours			Toei Animation			1999		
9 Blue Lock			Sport / Thriller			Manga			36 En cours			8bit			2022		
10 Haikyuu!!			Sport / Volley			Manga			85 Fin			Production I.G			2014		

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Etape 2 : Nettoyage des données

Cette partie consiste à supprimer les doublons, ligne et colonnes vides. Ainsi que les données qui ne seront pas nécessaires

- `df_clean = df.copy()`
- `df_clean = df_clean.dropna()`
- `df_clean['nom_colone'] = pd.to_datetime(df_clean['nom_colone'])`
- `df_clean = df_clean.drop_duplicates()`





Etape 3 : Enrichissement des données

Cette partie consiste à obtenir de nouvelles données basées sur les données déjà fournies afin d'approfondir l'analyse

```
df_clean['Ecart'] = df_clean['Note_Meilleur_Ep'] - df_clean['Note_Pire_Ep']  
df_clean['Regularite'] = 10 - df_clean['Ecart']  
df_clean['Score_Qualite'] = (df_clean['Note_Globale'] * 0.6) + (df_clean['Regularite'] * 0.4)  
display(df_clean)  
✓ [8] 290ms
```

Etape 4 : L'analyse statistique

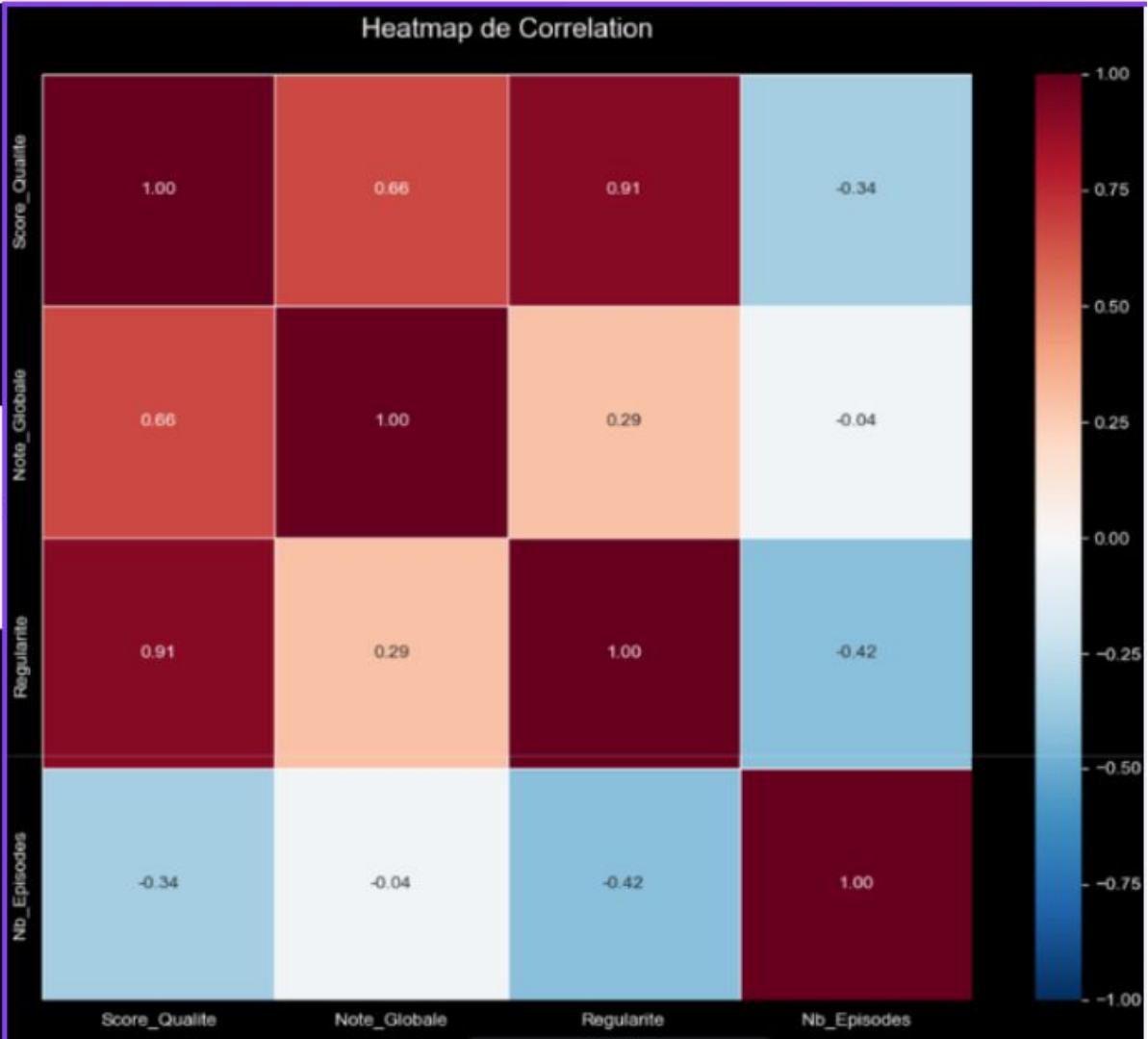
Cette partie consiste à supprimer les doublons, ligne et colonnes vides. Ainsi que les données qui ne seront pas nécessaires

- `df_clean = df.copy()`
- `df_clean = df_clean.dropna()`
- `df_clean['nom_colone'] = pd.to_datetime(df_clean['nom_colone'])`
- `df_clean = df_clean.drop_duplicates()`



Correlation

La corrélation compare l'évolution des paramètres les uns en fonction des autres



Etape 5 : Analyse et définition du Système de recommandation

```

import numpy as np

# 1. Regroupement des données (on utilise les noms exacts de son CSV)
resume_genres = df_clean.groupby('Anime').agg({
    'Score_Qualite': 'sum',
    'Nb_Episodes': 'sum',
    'Note_Globale': 'mean'
}).sort_values(by='Score_Qualite', ascending=False)

# 2. Définition des conditions de multiplication
conditions = [
    (resume_genres['Nb_Episodes'] <= 100),
    (resume_genres['Nb_Episodes'] > 100) & (resume_genres['Nb_Episodes'] <= 300),
    (resume_genres['Nb_Episodes'] > 300)
]

```

```

# Coefficients = 0.8 (0-100), 0.05 (100-300), 0.75 (>300)
multiplicateurs = [1.0, 0.05, 0.75]

# 3. Calcul de la recommandation finale en base SQL
# On applique la multiplication et on multiplie par 10 pour le format
resume_genres['Recommandation_Fourcast'] = (resume_genres['Score_Qualite'] * np.select(conditions, multiplicateurs))
* 10

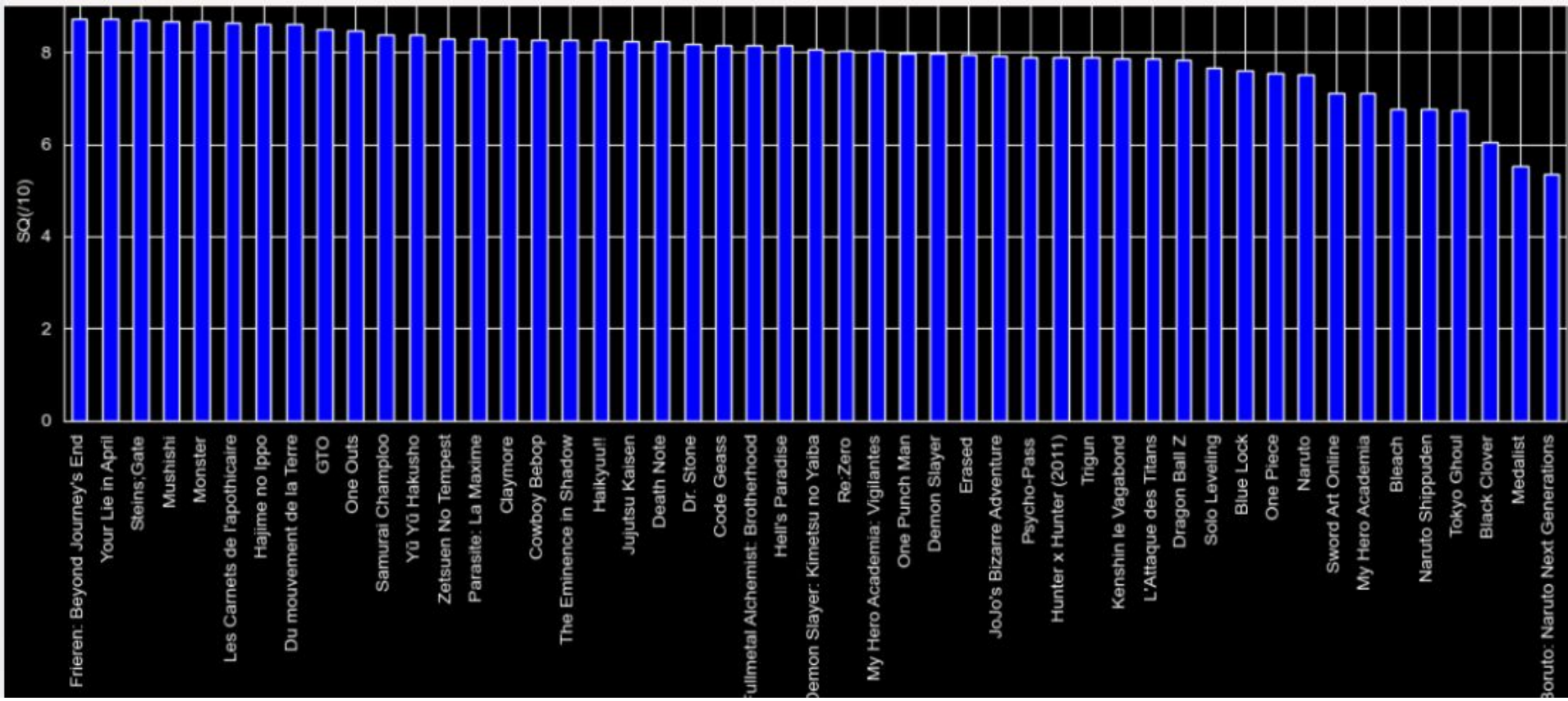
# 4. Affichage : On va sélectionner 100 les colonnes voulues pour l'affichage final
colonnes_fixes = ['Score_Qualite', 'Nb_Episodes', 'Note_Globale', 'Recommandation_Fourcast']

display(resume_genres[colonnes_fixes].style.format(
    'Score_Qualite': '{:.1f}',
    'Nb_Episodes': '{:g}',
    'Note_Globale': '{:.2f}',
    'Recommandation_Fourcast': '{:.1f}'
))
# 100 lines

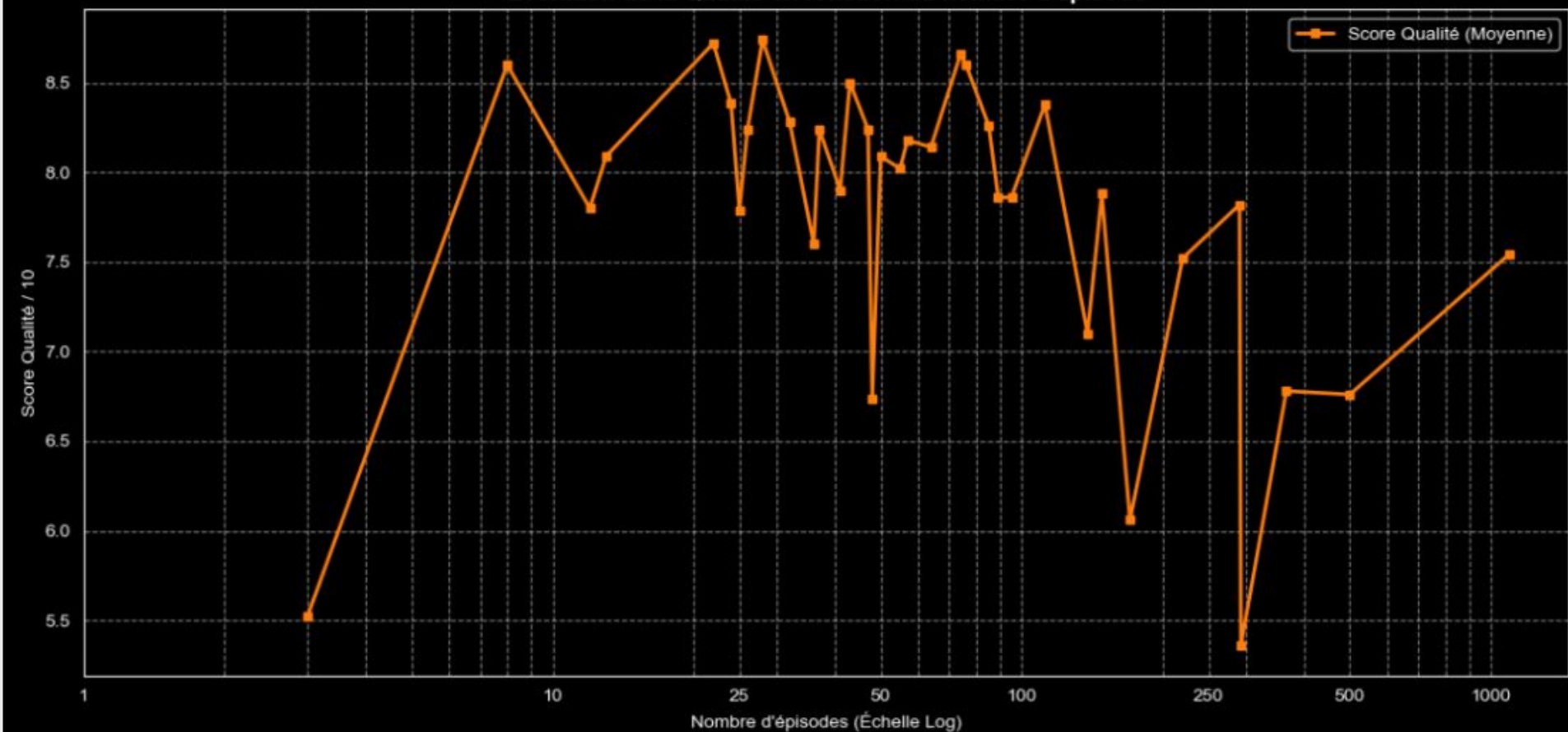
```

Cowboy Bebop	8.3	26	8.80
Eminence in Shadow	8.3	32	8.20
Haikyuu!!	8.3	85	8.70
Jujutsu Kaisen	8.2	47	8.60
Death Note	8.2	37	8.60
Dr. Stone	8.2	57	8.30
Code Geass	8.1	50	8.70
Alchemist: Brotherhood	8.1	64	9.10
Hell's Paradise	8.1	13	7.70
Kimetsu no Yaiba	8.1	55	8.50
Re:Zero	8.0	50	8.20
Academia: Vigilantes	8.0	13	7.60
One Punch Man	8.0	24	8.50
Demon Slayer	8.0	55	8.50
Erased	7.9	12	8.30
Jo's Bizarre Adventure	7.9	26	8.00
Psycho-Pass	7.9	41	8.30
Hunter x Hunter (2011)	7.9	148	9.00
Trigun	7.9	26	8.20

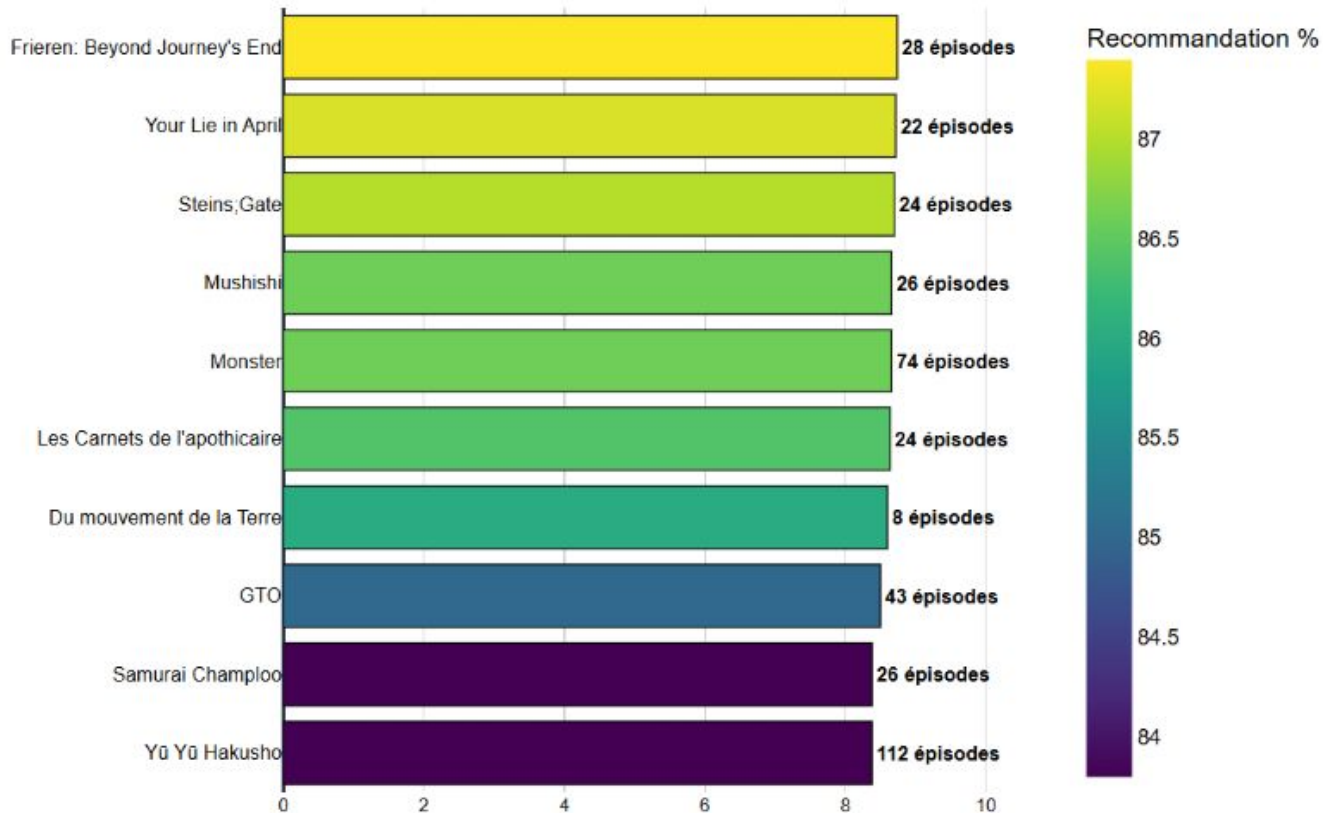
Etape 6 : Visualisation des données



Évolution de la Qualité en fonction du nombre d'épisode



TOP 10 ANIMES : CLASSÉS PAR RECOMMANDATION



Etape 7 : Storytelling

Heatmap

Diagramme

Système de
recommandation

