



JEUX GRATUITS - STEAM



PROJET APP OBJET

Yousra ZAABAT

2024 - EPSI

STEAM : POURQUOI CE CHOIX?

Type de données collectées:

- date de publication
- évaluation
- prix
- compatibilité et capacités nécessaires
- langues disponibles

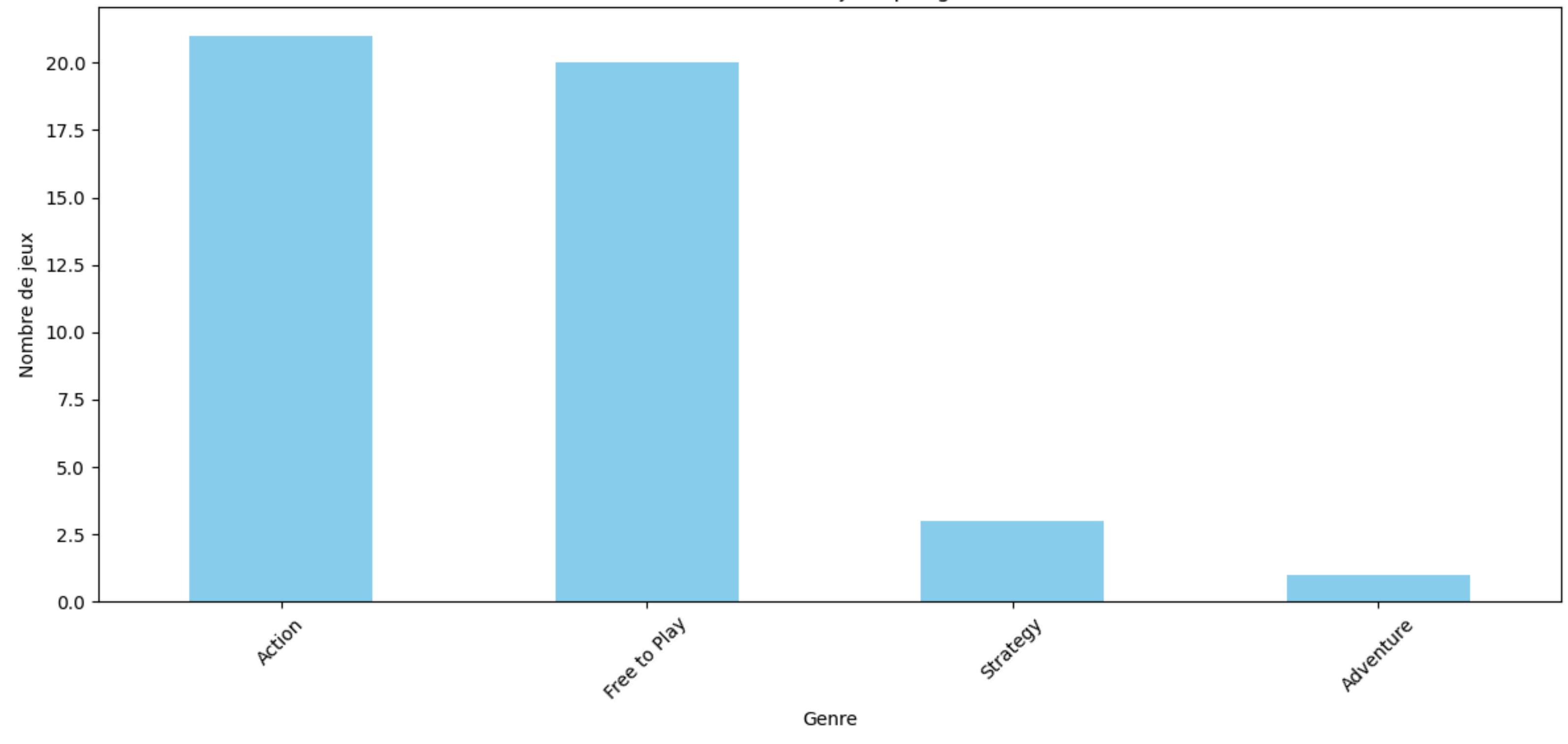
Choix de ces données:

- indiquent une tendance (consommation/temps)
- fiables et à jour (STEAM a accès à toutes les données)

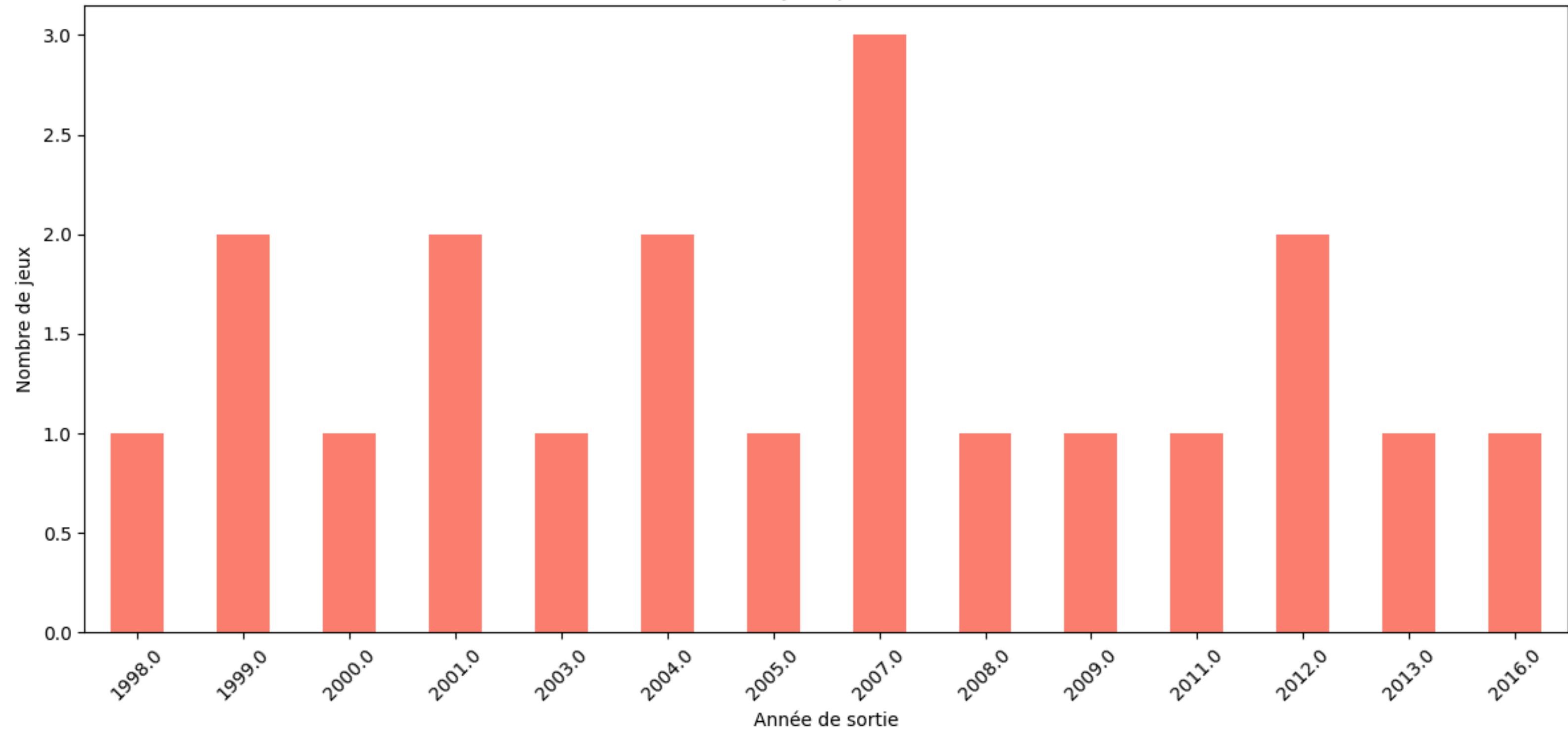
PROBLEMATIQUE

En quoi les caractéristiques des jeux sur Steam, telles que les genres, les dates de sortie, les langues supportées, et les exigences en matière d'espace de stockage, sont-elles diversifiées ?

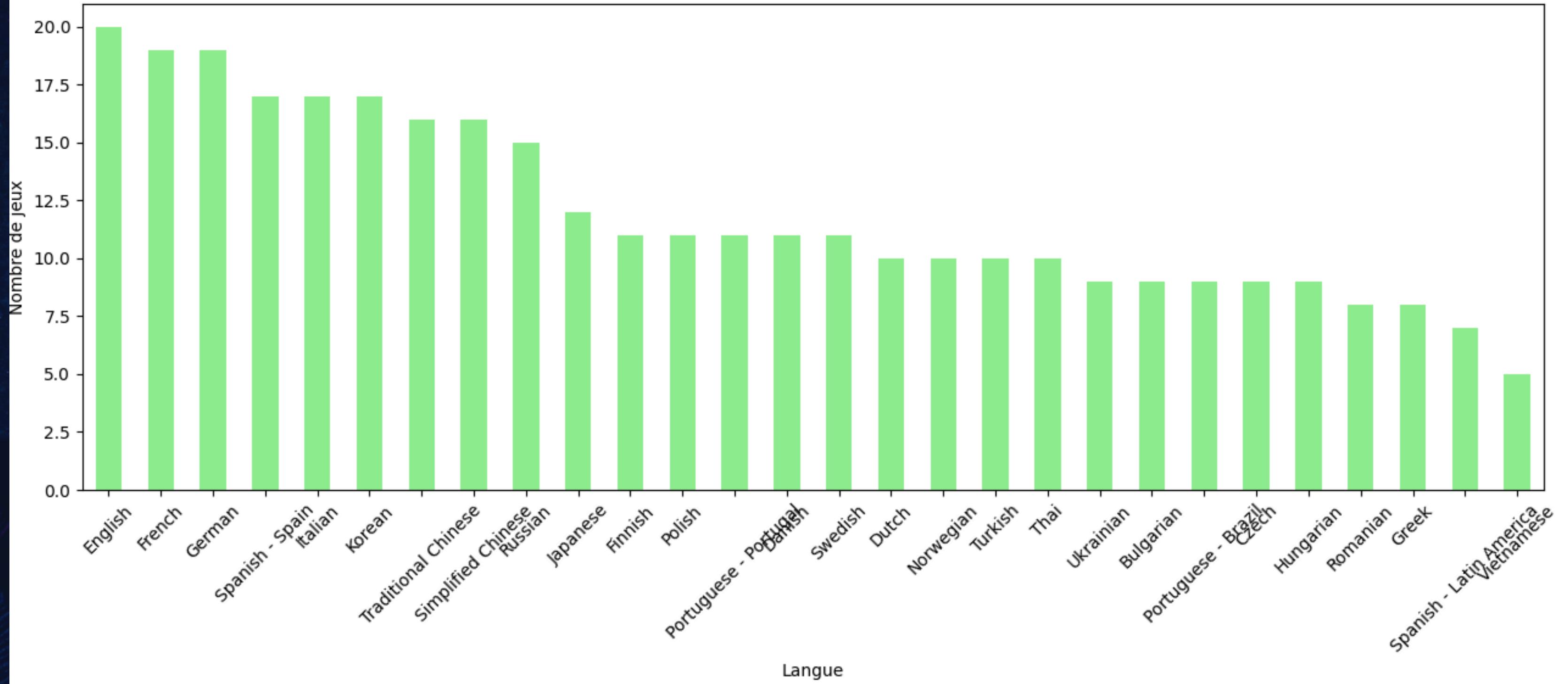
Distribution des jeux par genre



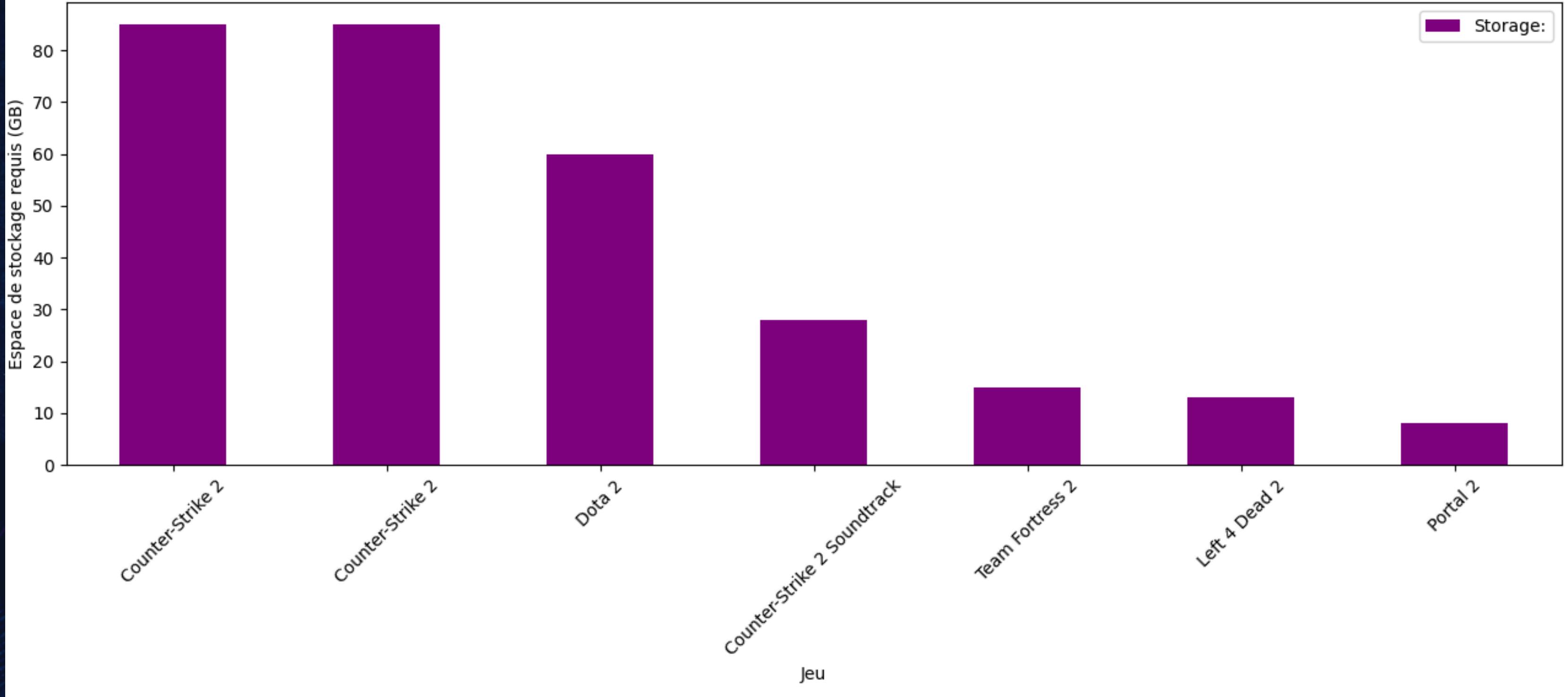
Distribution des jeux par année de sortie



Distribution des langues supportées par jeu



Espace de stockage requis par jeu (excluant > 6000 GB)



FICHIER JSON

fichier import.py

```
import json
from url import tab # Importe le tableau depuis le fichier url.py

# Convertir le tableau en JSON
json_data = json.dumps(tab, indent=4, ensure_ascii=False)

# Écrire le JSON dans un fichier
with open('data.json', 'w', encoding='utf-8') as f:
    f.write(json_data)

print("Le tableau a été converti et écrit dans data.json.")
```

création du fichier
json à partir du
tableau

fichier data.json contenant les données

DATA VISUALISATION

fichier graph.py

```
import json
import matplotlib.pyplot as plt
import pandas as pd

# Charger les données JSON
with open('data.json', 'r', encoding='utf-8') as f:
    jeux_data = json.load(f)

# Convertir les données en DataFrame pandas pour une analyse plus facile
df = pd.DataFrame(jeux_data)

# 1. Distribution des jeux par genre
genre_counts = df.explode('Genre')['Genre'].value_counts()
genre_counts.plot(kind='bar', figsize=(12, 6), color='skyblue')
plt.title('Distribution des jeux par genre')
plt.xlabel('Genre')
plt.ylabel('Nombre de jeux')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

- matplotlib (graphiques)
- pandas (stockage de données)

FICHIER PYTHON DE WEBSRAPING

fichier url.py

```
import requests
from bs4 import BeautifulSoup
from datetime import datetime

links = []
url = 'https://store.steampowered.com/app/730/CounterStrike_2/'
response = requests.get(url)

if response.ok:
    soup = BeautifulSoup(response.text, 'html.parser')
    # Trouver toutes les balises <a> contenant des liens
    for a_tag in soup.find_all('a', href=True):
        # Extraire l'URL
        link = a_tag['href']
        # Vérifier si l'URL est complète et validee
        if link.startswith('https://store.steampowered.com/app/'):
            links.append(link)
```

Recupération des liens
de chaque jeu dans un
tableau "links"

On parcourt le tableau pour récupérer les informations de chaque jeu

```
for link in links:  
    url = link  
  
    response = requests.get(url)  
    soup = BeautifulSoup(response.content, 'html.parser')  
  
    # Trouver la balise div avec l'ID 'genresAndManufacturer'  
    genres_and_manufacturer_div = soup.find('div', {'id': 'genresAndManufacturer'})  
    game_area_div = soup.find('div', {'class': 'game_area_sys_req sysreq_content active'})  
  
    dictionnaire = {}  
  
    # Extraire les données de la balise  
    if genres_and_manufacturer_div:  
        data = genres_and_manufacturer_div.text.strip()  
  
        # Séparer les différentes lignes de données  
        lignes = data.split('\n')
```

Exemple de données extraites

Dictionnaires de tous les jeux stockés dans un tableau

```
# Extraire les données de la balise
if genres_and_manufacturer_div:
    data = genres_and_manufacturer_div.text.strip()

    # Séparer les différentes lignes de données
    lignes = data.split('\n')

    for ligne in lignes:
        # Séparer la clé et la valeur
        if ':' in ligne:
            cle, valeur = ligne.split(':', 1)
            dictionnaire[cle] = valeur

    if 'Genre' in dictionnaire and ',' in dictionnaire['Genre']:
        # Diviser la valeur en une liste en supprimant la virgule et l'espace
        genres = dictionnaire['Genre'].split(', ')

    # Mettre à jour la valeur de la clé 'Genre' dans le dictionnaire
    dictionnaire['Genre'] = genres

    # Convertir la chaîne de caractères en date
    if 'Release Date' in dictionnaire:
        date_str = dictionnaire['Release Date']
        date_obj = datetime.strptime(date_str, '%d %b, %Y').date()

        # Convertir la date au format 'dd/mm/yyyy' avec des slashes
        date_formatee = date_obj.strftime('%d/%m/%Y')

    # Mettre à jour le dictionnaire avec la date au format date
    dictionnaire['Release Date'] = date_formatee
```

- Balises retenues:
- 'genresAndManufacturer'
 - 'game_area_sys_req'
 - 'sysreq_content active'
 - 'game_language_options'

— CONCLUSION —



En quoi les caractéristiques des jeux sur Steam, telles que les genres, les dates de sortie, les langues supportées, et les exigences en matière d'espace de stockage, sont-elles diversifiées?

- Matériel (évolution technologique)
Environnement des joueurs/développeurs
- Les joueurs (réponse aux tendances)
Style de jeu

JEUX GRATUITS - STEAM



MERCI DE NOUS
AVOIR ECOUTEES