atelier-2b

February 12, 2024

1 Atelier 2 - Systèmes de recommandation

1.0.1 Question 5 (b)

En utilisant les approches I-I et U-U, quels films sont recommandés pour un utilisateur qui a voté 5 pour Clockwork Orange (no. 179) et pour Full Metal Jacket (no. 188)? (10 pts.)

[2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import tqdm.autonotebook as tqdm
import warnings
warnings.filterwarnings('ignore')

votes = pd.read_csv("data/votes.csv")
items = pd.read_csv("data/items.csv")
users = pd.read_csv("data/u.csv")

MUI = votes.pivot(index="user.id", columns="item.id", values="rating")
MUI_numpy = MUI.to_numpy()

def cosinus_matrices(A, B):
    A = A.copy()
    B = B.copy()
    A[np.isnan(A)] = 0
    B[np.isnan(B)] = 0
    axis = 1
    col_norm_A = np.linalg.norm(A, axis=axis, keepdims=True)
    col_norm_B = np.linalg.norm(B, axis=axis, keepdims=True).T
    W = A @ B.T / (np.abs(col_norm_A * col_norm_B) + 1e-10)
    W = W * (np.ones_like(W) - np.identity(W.shape[0]))
    return W

MUI_Query = np.zeros_like(MUI_numpy)
MUI_Query[0, 179] = 5
MUI_Query[0, 188] = 5

def Biais_mat(R):
```

```
    R_moy_u = np.nanmean(R, axis=1, keepdims=True)
    R_moy_i = np.nanmean(R, axis=0, keepdims=True)
    R_moy = (R_moy_u + R_moy_i) / 2
    return R_moy

MUI_numpy = MUI_numpy - Biais_mat(MUI_numpy)
MUI_Query = MUI_Query - Biais_mat(MUI_Query)
```

Approche Utilisateur-Utilisateur

[3]:
```
similariter = cosinus_matrices(MUI_numpy, MUI_Query)[:, 0]
np.argsort(similariter)[::-1]
items[" movie title "].iloc[np.argsort(similariter)[::-1][:10]]
```

[3]:
```
155                                 Reservoir Dogs (1992)
258                            George of the Jungle (1997)
766                                 Addiction, The (1995)
639      Cook the Thief His Wife & Her Lover, The (1989)
40                                   Billy Madison (1995)
414                        Apple Dumpling Gang, The (1975)
521                                     Down by Law (1986)
24                                   Birdcage, The (1996)
381    Adventures of Priscilla, Queen of the Desert, …
322                                    Dante's Peak (1997)
Name:  movie title , dtype: object
```

Approche Item-Item

[4]:
```
similariter = cosinus_matrices(MUI_numpy.T, MUI_Query.T)[:, 0]
np.argsort(similariter)[::-1]
items[" movie title "].iloc[np.argsort(similariter)[::-1][:10]]
```

[4]:
```
73         Faster Pussycat! Kill! Kill! (1965)
103                        Theodore Rex (1995)
246        Turbo: A Power Rangers Movie (1997)
34     Free Willy 2: The Adventure Home (1995)
102             All Dogs Go to Heaven 2 (1996)
111                                Flipper (1996)
137                D3: The Mighty Ducks (1996)
36                                 Nadja (1994)
109                Operation Dumbo Drop (1995)
33                 Doom Generation, The (1995)
Name:  movie title , dtype: object
```

1.0.2 Question 6 (b)

Je suis une femme ingénieure de plus de 52 ans. Quelle est la probabilité que j'aime Toy Story? Utilisez une approche Bayésienne dichotomique où le seuil pour aimer est un vote de 4 et plus et

où celui de l'âge est fixé à 52 ans. Présumez de l'indépendance des facteurs. (10 pts.)

[5]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import tqdm.autonotebook as tqdm

votes = pd.read_csv("data/votes.csv")
items = pd.read_csv("data/items.csv")
users = pd.read_csv("data/u.csv")

MUI = votes.pivot(index="user.id", columns="item.id", values="rating")
MUI_numpy = MUI.to_numpy()
```

$P(aime|age > 52, job = ingenieur) = \frac{P(age>52, job=ingenieur|aime)*P(aime)}{P(age>52, job=ingenieur)}$

[6]:
```
aime_toyStory = (MUI_numpy >= 4)[:, 0]
users_above_52_ingenieur = (users[" age "] > 52) & (users[" job "] ==␣
 ↪"engineer").to_numpy()

Prob_aime_toyStory = np.mean(aime_toyStory)
Prob_users_above_52_ingenieur = np.mean(users_above_52_ingenieur)
Prob_users_above_52_ingenieur_sachant_aime_toyStory = np.
 ↪mean(users_above_52_ingenieur[aime_toyStory])
Prob_aime_toyStory_sachant_users_above_52_ingenieur = (
    Prob_users_above_52_ingenieur_sachant_aime_toyStory*Prob_aime_toyStory
) / Prob_users_above_52_ingenieur
Prob_aime_toyStory_sachant_users_above_52_ingenieur
```

[6]:
```
0.4285714285714286
```

La probabilité qu'une personne aime Toy Story sachant que je suis une femme ingénieure de plus de 52 ans est de **0.43**.