الكلية متعددة التخصصات - ورزازات
ⵜⴰⵖⵣⵓⵜ ⵜⴰⵅⵜⵣⴰⵎⵉⵏⵜ ⵏ ⵡⵓⵔⵣⴰⵣⴰⵜ
FACULTÉ POLYDISCIPLINAIRE DE OUARZAZATE

Master en Mathématiques Appliquées pour la Science de Données

Thème

Système de recommandation de films basé sur le filtrage collaboratif

Réaliser par :
Kaoutar IKKANE

Encadrer par :
M. Charaf HAMIDI
Mme. Salma GAOU

Année Universitaire
2023/2024

Résumé

Les systèmes de recommandations sont des systèmes automatiques qui permettent, par des algorithmes d'apprentissage automatique, de fournir à des utilisateurs des suggestions qui répondent à leurs exigences.

Le présent rapport explore une approche de recommandation de films en utilisant des techniques de filtrage collaboratif. Le projet repose sur l'analyse d'un ensemble de données comprenant des évaluations de films fournies par des utilisateurs.

Nous utilisons cette méthode basée sur les utilisateurs et une approche de factorisation matricielle à l'aide du module Surprise.

Abstract

Recommendation systems are automated systems that, through machine learning algorithms, provide users with suggestions that meet their requirements.

This report explores a movie recommendation approach using collaborative filtering techniques. The project is based on the analysis of a dataset containing movie ratings provided by users.

We employ this user-based method and a matrix factorization approach using the Surprise module.

Table des matières

Table des figures

1 Introduction générale

Le projet s'articule autour de l'utilisation d'une technologie de filtrage collaboratif pour analyser et recommander des films. Les données utilisées comprennent des informations sur les films, les utilisateurs et les évaluations des films par les utilisateurs. Tout d'abord, nous explorons les données à l'aide de bibliothèques telles que NumPy, Pandas, Matplotlib et Seaborn pour comprendre la répartition des notes et identifier les utilisateurs et les films les plus populaires.

Ensuite, nous examinons les caractéristiques des utilisateurs et des films et comptons le nombre de critiques pour chaque utilisateur et chaque film. Nous fixons un seuil basé sur un nombre minimum d'avis pour définir les "meilleurs" utilisateurs et films.

La phase de modélisation consiste à diviser les données en ensembles de formation et de test, puis à appliquer différents algorithmes de recommandation, notamment le filtrage collaboratif et la factorisation matricielle. Nous utilisons des outils tels que la bibliothèque Surprise pour implémenter ces algorithmes.

Enfin, nous évaluons les performances de notre modèle en calculant l'erreur quadratique moyenne (RMSE) pour mesurer l'exactitude de nos recommandations. Nous utilisons également des méthodes de recherche de grille pour ajuster les paramètres du modèle et améliorer ses performances.

En résumé, Ce sujet a été opté pour répondre à cette question : Comment créer un système de recommandation basés sur le filtrage collaboratif et de l'apprentissage automatique pour prédire avec précision les préférences des utilisateurs en fonction de leurs évaluations passées ?

2 Méthodologie

La méthdologie utilisées pour un système de recommandaton de films basé sur le filtrage collaboratif est présentée dans la Figure ci-dessous.

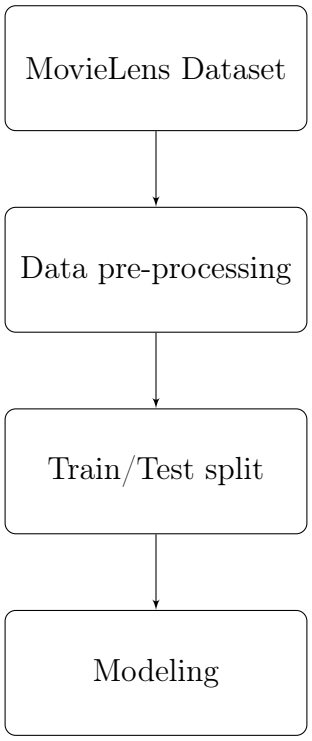

FIGURE 1 – La méthodologie du projet

2.1 La Collecte de données :

Les données sur les films et les notations sont chargées à partir de MovieLens Dataset l'ensemble de données est constitué des identificateurs des films, identificateurs des utilisateurs, et les évaluations données par les utilisateurs aux films regardés.

2.2 Nettoyage et prétraitement des données :

Le processus de nettoyage et de prétraitement des données a été réalisé en plusieurs étapes. Tout d'abord, la sélection des colonnes pertinentes, seules les colonnes 'movieId', 'userId', 'rating' sont choisies car elles contiennent des informations importantes concernant l'approche utilisée dans ce projet. Ensuite, l'exploration des données, nous avons généré un histogramme représentant la distribution des évaluations attribuées aux films. Cette visualisation offre un aperçu significatif de la répartition des différentes valeurs d'évaluations possibles. L'axe des x représente les différentes valeurs d'évaluations, tandis que l'axe des y représente le nombre de films ayant reçu chaque évaluation. évaluations :

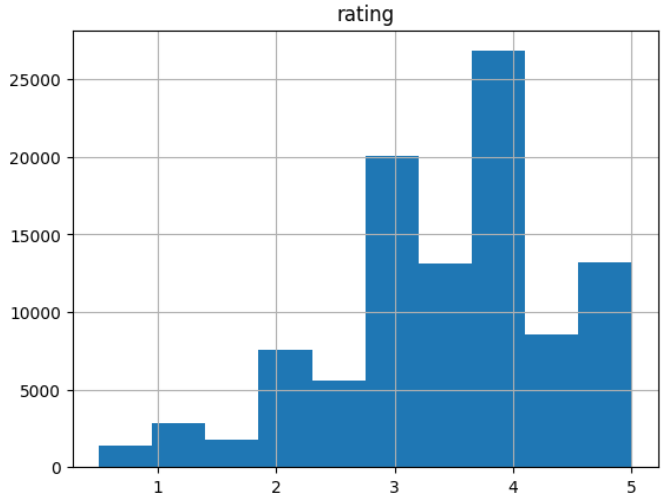

FIGURE 2 – histogramme des notations

Nous avons examiné la popularité des utilisateurs en fonction de la somme totale de leurs évaluations attribuées. La représentation graphique ci-dessous met en évidence cette popularité en utilisant un histogramme, où l'axe des x représente les identifiants des utilisateurs (userId) et l'axe des y représente la somme des évaluations (ratings) attribuées par chaque utilisateur :

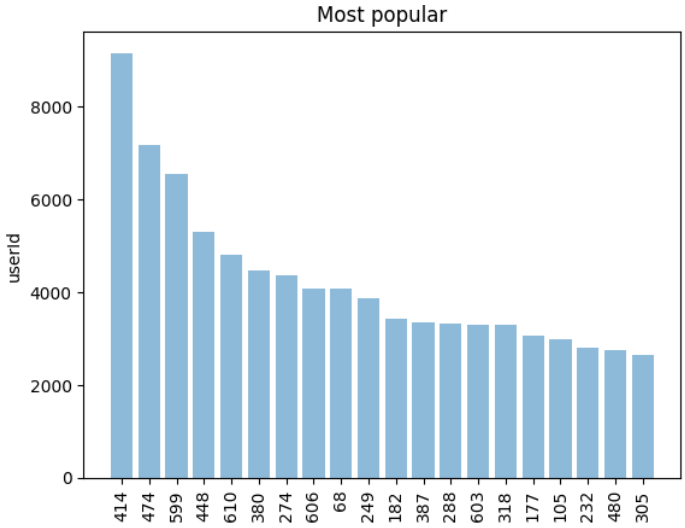

FIGURE 3 – Utilisateurs plus actifs

Des statistiques et des informations essentielles sur le nombre d'utilisateurs uniques, de films uniques et de notations sont également affichées, ce qui permet de mieux comprendre la structure des données.

Le processus inclut également le filtrage des utilisateurs et des films en fonction d'un seuil du nombre de notations, permettant de comprendre la distribution des données et les relations entre les variables, en se concentrant sur un sous-ensemble de données plus significatif.

En outre, des étapes de prétraitement supplémentaires comprennent le calcul de la somme des évaluations attribuées par chaque utilisateur, l'identification des utilisateurs les plus populaires, la catégorisation des films en fonction de leurs évaluations (faible, moyen, élevé), ainsi que le calcul des moyennes et des tops des films les mieux notés. Ces étapes préparent les données pour l'analyse ultérieure et la création de modèles de recommandation.

2.3 La division des données :

Une fois que les données ont été prétraitées et transformées dans un format pouvant être utilisé pour l'analyse, elles doivent être divisées en un ensemble d'entraînement et un ensemble de test, nous avons alloué 70 % à l'entraînement et 30 % au test. Concernant l'ensemble d'entraînement est utilisé pour entraîner le modèle d'apprentissage automatique, tandis que l'ensemble de test est utilisé pour évaluer les performances du modèle.

2.4 Algorithmes utilisés :

Dans ce projet, deux algorithmes de filtrage collaboratif sont utilisés, comprenant le KNNWithMeans "KNN : K-Nearest Neighbors" et SVD "Singular Value Decomposition" pour créer le modèle de recommandation. Ces algorithmes sont spécifiés à l'aide de la bibliothèque Surprise, qui est une bibliothèque Python dédiée aux systèmes de recommandation.

2.4.1 KNNWithMeans (k-NN with Means) :

Cet algorithme utilise l'algorithme k-NN (k-Nearest Neighbors), il s'appuie sur la similarité entre utilisateurs , prend en compte les moyennes pour normaliser les recommandations, et utilise les K voisins les plus proches pour générer des prédictions.
Les principaux aspects de l'algorithme sont comme suit :

- ◇ Moyennes : KNNWithMeans prend en compte les moyennes des évaluations des utilisateurs (ou items) pour normaliser les recommandations. L'idée est de corriger les biais potentiellement présents dans les évaluations individuelles en considérant la tendance générale des évaluations d'un utilisateur ou d'un item.

- ◇ K voisins : L'algorithme utilise les K voisins les plus proches pour générer des prédictions. K représente le nombre de voisins considérés lors de la recommandation. Plus K est élevé, plus l'algorithme considère de voisins, mais cela peut également entraîner une augmentation de la complexité.

- ◇ Prédiction : Une fois les voisins identifiés, KNNWithMeans combine leurs évaluations pondérées pour générer une prédiction de l'évaluation qu'un utilisateur donnerait à un item particulier.

Le modèle est configuré avec les paramètres suivants :

◇ Nombre de voisins : à considérer "k=10"

◇ Nombre minimum de voisins : requis pour la prédiction "min_k=6"

◇ Corrélation de Pearson : La mesure de similarité entre utilisateurs basée sur la corrélation de Pearson. " sim_options={'name' : 'pearson_baseline', 'user_based' : True} "
La figure ci-dessous présente une visualisation de cette similarité sous forme d'une matrice de corrélation (heatmap). On a limité la heatmap aux dix premiers utilisateurs, et les valeurs dans le triangle supérieur ont été masquées pour éliminer la redondance.

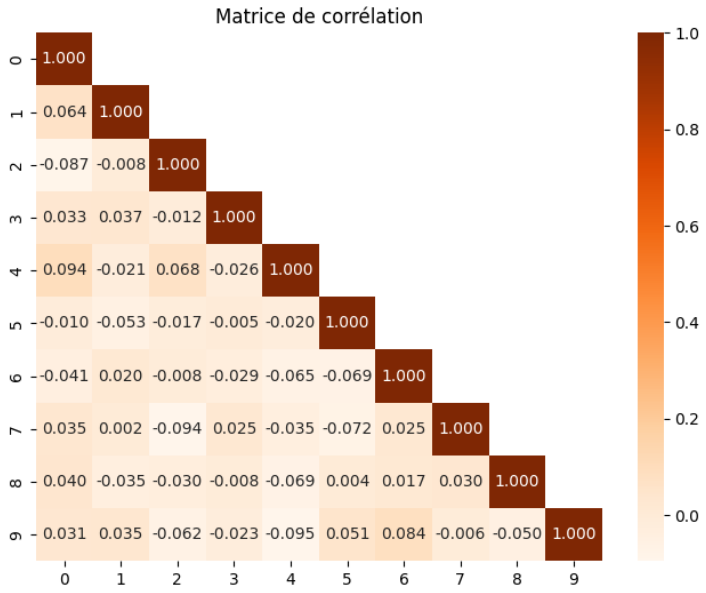

FIGURE 4 – Matrice de corrélation

2.4.2 Singular Value Decomposition (SVD) :

Le SVD est une méthode de décomposition matricielle qui consiste à décomposer une matrice de données originale en trois matrices distinctes : U , Σ , V^T. Dans le contexte du filtrage collaboratif, ces matrices sont interprétées comme suit :

◇ U : "Matrice utilisateur" Cette matrice représente la relation entre les utilisateurs et les facteurs latents. Chaque ligne de cette matrice correspond à un utilisateur, et les éléments de la ligne capturent la manière dont l'utilisateur est associé aux facteurs latents.

◇ Σ : "Matrice diagonale des valeurs singulières " Cette matrice contient les valeurs singulières, qui représentent l'importance relative des facteurs latents. Les valeurs singulières sont classées par ordre décroissant, et seules les premières valeurs sont souvent conservées car elles capturent l'essentiel de l'information.

◇ V^T : "Matrice item (film)" Cette matrice représente la relation entre les items (films dans ce cas) et les facteurs latents. Chaque colonne de cette matrice corres-

pond à un item, et les éléments de la colonne capturent la manière dont l'item est associé aux facteurs latents.

Le modèle est configuré avec les paramètres suivants :

- ◇ Nombre de facteurs latents : à utiliser pour la décomposition "n_factors=15"
- ◇ Terme de régularisation : pour contrôler l'ajustement du modèle "reg_all=0.02"

Ces paramètres ajustent la complexité du modèle pour éviter le surajustement et maximiser la capacité de généralisation.

3 Résultats et discussion

Après avoir collecter, explorer, analyser les données nous avons construit un modèle de recommandation en utilisant les deux algorithmes KNNWithMeans et SVD.

3.1 Les métriques de performances :

Pour évaluer la performance d'un modèle prédictif, notamment dans le contexte des modèles de recommandation,diverses métriques peuvent être utilisées dans le cadre de l'analyse du modèle, concernant notre projet nous avons utilisé le RMSE "Root Mean Squared Error" en français "Erreur Quadratique Moyenne Racine".

Le **RMSE** est une métrique d'évaluation qui quantifie l'écart moyen entre les évaluations réelles des utilisateurs et les évaluations prédites par le modèle. Plus précisément, pour chaque paire utilisateur-item dans l'ensemble de test, le modèle prédit une évaluation, puis le RMSE mesure la différence entre cette prédiction et la véritable évaluation.

3.2 Comparaison algorithmique et évaluation des performances :

Après avoir utilisé le RMSE pour évaluer la performance des deux modèles de recommandation, ous avons trouvé des résultats très similaires tel que la valeur du RMSE du KNNWithMeans est égale à 0,8349, et la valeur du RMSE du SVD est égale à 0,8359.
Les valeurs RMSE relativement faibles indiquent un écart moyen modéré entre les prédictions de ces modèles et les scores réels sur l'ensemble de tests, ce qui suggère une bonne capacité à faire des prédictions précises. La similitude des performances des deux modèles avec des valeurs RMSE très proches suggère qu'ils sont compétitifs en termes de précision prédictive.

3.3 Amélioration du modèle SVD par ajustement d'hyperparamètres :

L'ajustement d'hyperparamètres d'un modèle se réfère au processus d'optimisation des paramètres du modèle pour atteindre une performance optimale, dans notre cas l'ajustement est appliqué sur le SVD en utilisant les deux paramètres tels que le nombre de facteurs latents (n_factors) et le terme de régularisation (reg_all).

L'ajustement d'hyperparamètres commence par une recherhe sur une grille (*Grid Search*). Cela consiste à spécifier une liste de valeurs possibles pour chaque hyperparamètre, puis à évaluer le modèle pour toutes les combinaisons possibles de ces valeurs. Dans notre projet, cela est réalisé avec la classe GridSearchCV de la bibliothèque Surprise.

FIGURE 5 – Ajustement d'hyperparamètres

Notre objectif était d'observer l'impact de cet ajustements sur la performance du SVD, la figure ci-dessus montre cela à partir des valeurs du RMSE avant et après.

3.4 Limitations et Perspectives :

— Limitations :
Bien que notre approche du système de recommandation basé sur le filtrage collaboratif, elle comporte des limitations et des défis inhérents. Tout d'abord, le biais dans les évaluations des utilisateurs représente un défi potentiel, car le modèle pourrait reproduire ces préférences subjectives. Le problème du "Cold Start", où le modèle peut avoir des difficultés à recommander pour de nouveaux utilisateurs ou films, est également présent, nécessitant des approches spécifiques.

— Perspectives :
Pour améliorer la performance du modèle, nous pourrions envisager plusieurs avenues :

◇ Utilisation de Modèles Avancés : Explorer des modèles plus avancés de recommandation, tels que les méthodes de factorisation matricielle ou les réseaux de neurones, pour capturer des relations plus complexes entre utilisateurs et films.

◇ Utilisation de Modèles Hybrides : Intégrez des modèles hybrides qui combinent le filtrage collaboratif avec d'autres méthodes, comme le filtrage basé sur le contenu. Cela peut améliorer la précision des recommandations en prenant en compte à la fois les préférences des utilisateurs et les caractéristiques des films.

4 Conclusion générale

Ce projet a été conçu dans le but de créer un système de recommandation de films basé sur les préférences des utilisateurs, en utilisant des techniques de filtrage collaboratif et de factorisation matricielle. Voici quelques points clés issus de notre analyse et de la mise en œuvre des modèles :

- ◇ **Méthodologie :**
  La méthodologie utilisée dans ce projet comprend plusieurs étapes importantes. La première consiste à obtenir les données nécessaires du MovieLens Dataset, qui comprend les identifiants des films, les utilisateurs et les évaluations attribuées. Les données subissent un processus de nettoyage en plusieurs étapes après la collecte. La sélection des colonnes appropriées et l'exploration des données font partie de cela.
  Une fois les données nettoyées, elles sont divisées en ensembles d'entraînement et de test. Cette division permet de préparer les données pour l'entraînement du modèle et son évaluation.
  L'étape suivante implique l'utilisation de deux algorithmes de filtrage collaboratif, à savoir KNNWithMeans et SVD. Ces algorithmes, configurés avec des paramètres spécifiques, sont appliqués pour créer un modèle de recommandation basé sur la similarité entre utilisateurs et la décomposition matricielle.

- ◇ **Résultats et discussion :**
  La section des résultats et discussions met en lumière les aspects cruciaux de l'évaluation du modèle de recommandation, comprenant une analyse approfondie des métriques de performances utilisées pour évaluer l'efficacité du modèle. Ces métriques fournissent des indications quant à la précision et à la pertinence des recommandations générées.
  Une comparaison détaillée des performances des algorithmes KNNWithMeans et SVD. Cette évaluation permet de déterminer quel algorithme offre les résultats les plus satisfaisants en termes de recommandations précises.
  Une étude approfondie sur l'optimisation du modèle SVD en ajustant ses hyperparamètres. Cette démarche vise à maximiser la performance du modèle en évitant le surajustement et en améliorant sa capacité de généralisation.
  Enfin, la section souligne les limitations potentielles du modèle actuel et explore des perspectives d'amélioration. Cela inclut des considérations sur les données utilisées, les éventuels biais, et les opportunités pour développer des fonctionnalités futures.

En résumé, ce projet a fourni des informations précieuses sur la création de systèmes de recommandation de films. Bien que le modèle affiche des performances prometteuses, il reste encore des améliorations à apporter pour mieux prédire les préférences des utilisateurs et fournir des recommandations plus personnalisées.