Cardi Julien
Ferroni Sandro
Moyo-Kamdem Auren
Promo : 2025
SCIA-G

EDA
Projet : Exploratory Data Analysis

Version 1 : 21/11/2024

Introduction

Dans le cadre de notre cursus scolaire et de notre majeure SCIA-G, axée sur l'intelligence artificielle, nous avons eu l'opportunité de suivre des cours d'Exploratory Data Analysis (EDA). Ces cours visent à nous permettre d'examiner, d'analyser et d'interpréter des jeux de données complexes afin d'en extraire des informations significatives, tout en mettant ces connaissances en pratique à travers ce projet.

L'objectif de ce dernier est de mettre en pratique trois grandes thématiques abordées en cours avec une partie sur l'analyse des motifs et des règles d'associations notamment, une deuxième sur l'échantillonnage de motifs d'espace de sortie et une troisième partie sur l'analyse des groupes d'utilisateurs.

Pour ce projet, nous avons travaillé sur des données de transactions Amazon, principalement axées sur les achats et les notations d'utilisateurs concernant des articles de « Sports et plein air ».

Dans ce rapport, nous commencerons par analyser nos données et expliquer le nettoyage effectué. Ensuite, nous aborderons la première partie, centrée sur l'analyse des motifs et des règles d'association, avant de traiter l'extraction et l'analyse des motifs à forte utilité. Enfin, nous conclurons par l'analyse des groupes d'utilisateurs et de leurs achats.

Le code du projet est disponible sur le dépôt GitHub suivant :
https://github.com/CirSandro/EDA

1. Analyse des données

Pour réaliser notre travail, nous devions sélectionner le dataset de notre choix à partir du lien suivant : https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2/
Ce lien contient les évaluations des utilisateurs sur une multitude de produits sur le site d'Amazon, collectées entre mai 1996 et octobre 2018. Ces données sont triées par type de produits, et chaque type possède plusieurs fichiers contenant les informations correspondantes.

Nous avons choisi de travailler sur le dataset "Sports and Outdoors". Dans un premier temps, nous avons sélectionné les données issues du fichier "5-core", contenant 2 839 940 avis, ce qui est largement suffisant pour nos travaux.

Ce dataset comportait plusieurs champs :

- overall: note attribuée à l'item par l'utilisateur (sur 5)
- verified: certifie les achats vérifiés par les utilisateurs
- reviewTime: indique la date de notation
- reviewerID: ID de l'utilisateur
- asin: ID de l'item
- reviewerName: nom de l'utilisateur
- reviewText: texte accompagnant la notation du produit
- summary: résumé de la notation
- unixReviewTime: timestamp unix représentant la date et l'heure de l'avis
- style: style du produit (taille, couleur)
- vote: si l'avis a été noté par d'autres utilisateurs
- image: image accompagnant l'avis

Dans un premier temps, nous avons fait le choix de conserver uniquement les avis certifiés "verified", ce qui a réduit la taille de notre dataset à 2 611 805 avis.
Ensuite, nous avons décidé de ne conserver que les colonnes nous paraissant utiles, à savoir : 'overall', 'reviewerID', 'asin' et 'reviewTime'. Pour une meilleure compréhension, nous les avons renommées respectivement : 'rating', 'user', 'item' et 'reviewTime'.

Cependant, il nous manquait le prix des items, information qui pourrait être utile notamment dans la partie 3 de ce projet. Nous avons donc décidé d'importer également le fichier *metadata*, comportant 962 876 produits. Voici les colonnes qui nous intéressent :

- main_cat: représentant la catégorie principale du produit
- price: indiquant le prix de l'item
- asin: l'ID du produit

Nous avons donc conservé uniquement les produits ayant comme main_cat "Sports and Outdoors", la catégorie qui nous intéresse. Nous avons ensuite converti les prix, initialement sous forme de chaîne de caractères (format "$xx.xx"), en type *float*, et supprimé les valeurs mal définies ou sous un format non récupérable (par exemple, « xx.xx-xx.xx » indiquant une fourchette de prix).

Nous avons ainsi fusionné nos deux datasets, en conservant uniquement les avis pour lesquels nous connaissions le prix de l'item.

Enfin, nous avons constaté que nos avis étaient répartis entre 2003 et 2018, avec très peu de données avant 2011. Nous avons donc décidé de ne conserver que les avis datant de 2011 et au-delà, afin de réduire légèrement la taille de notre dataset, qui se présente désormais sous cette forme :

- rating: note attribuée à l'utilisateur pour l'item donné
- user: ID de l'utilisateur
- item: ID du produit
- price: prix du produit

Ce dataset comporte ainsi 1 459 442 avis, 312 284 utilisateurs différents et 50 075 produits distincts.

2. Partie 1: Motifs et règles d'association

Pour notre analyse des données Amazon Sports & Outdoors, notre première étape a été le nettoyage des données en supprimant les doublons des paires utilisateur-produit. Nous avons ensuite transformé le dataset en format transactionnel en regroupant les items par utilisateur pour faciliter l'analyse des motifs d'achat.

Pour identifier les motifs fréquents, nous avons utilisé l'algorithme Apriori. Le choix du seuil de support minimal a été réalisé de manière empirique : nous avons d'abord testé avec un seuil de 0.01, mais cela ne donnait pratiquement aucun résultat. En diminuant progressivement ce seuil, nous avons constaté qu'une valeur de 0.002 offrait un bon compromis, nous permettant d'extraire 130 motifs fréquents significatifs. L'analyse de ces motifs a révélé que 67% sont des achats uniques, 16% des paires de produits, et 17% des groupes de trois produits ou plus. Le produit B00FA2RLX2 est clairement le plus populaire avec un support de 0,007038, suivi par B00BMSGU9Y (0,005117) et B0012Q2S4W (0,005101).

Face au nombre important de motifs, nous avons décidé d'utiliser l'algorithme LCM pour extraire uniquement les motifs fermés. Cette approche nous a permis de passer de 130 à 100 motifs, simplifiant considérablement notre analyse tout en gardant l'information essentielle.

Pour l'analyse des règles d'association, nous avons fixé le seuil de confiance à 0,5. Cette valeur nous a permis d'identifier des associations pertinentes, notamment trois règles parfaites avec une confiance de 1.0. La plus intéressante étant l'association {B00PD8JOTW, B016UQXB26} → {B016UQXB5I, B00N3XXXCS}, avec un lift remarquable dépassant 440. Ces associations fortes suggèrent des opportunités intéressantes pour des recommandations de produits ou des stratégies de ventes croisées.

Nous avons ensuite voulu comprendre si différents types d'utilisateurs avaient des comportements d'achat distincts. Nous avons choisi de séparer notre population en deux groupes selon leurs notes : les utilisateurs donnant des notes élevées (≥4 étoiles) et ceux donnant des notes plus basses (<4 étoiles). Cette segmentation a révélé des différences dans les préférences d'achat : les utilisateurs donnant des notes élevées achètent plus fréquemment le produit B00FA2RLX2 (support : 0,007602), tandis que ceux donnant des notes plus basses ont une préférence pour le produit B00BMSGU9Y (support : 0,007658) et ont tendance à réaliser leurs achats en groupes.

Pour la compression des données, nous avons exploité les motifs fermés identifiés précédemment. Nous avons développé une fonction qui compare chaque transaction à ces motifs, ce qui nous a permis de réduire significativement la taille des données tout en conservant la possibilité de retrouver l'information originale. Cette approche nous offre un stockage plus efficace tout en gardant la qualité de nos analyses.

3. Partie 2: Extraction et Analyse des Motifs à Forte Utilité

1. Introduction

Dans cette partie, l'objectif est de travailler sur la base de données entière obtenue après prétraitement afin d'explorer et d'analyser :

1. Les **motifs fréquents à la demande**, en étudiant la pertinence des contraintes de taille.
2. Les **motifs à forte utilité**, en intégrant les utilités sur les items grâce à l'algorithme générique **QPlus**.
   Enfin, un **système d'extraction de motifs à la demande** est proposé et validé à travers une étude qualitative basée sur un échantillon de motifs.

---

2. Motifs fréquents à la demande et pertinence des contraintes de taille

2.1 Extraction des motifs fréquents

L'algorithme Apriori a été utilisé pour identifier les motifs fréquents. Un seuil de support minimal de 0,002 a été fixé après expérimentation pour équilibrer la quantité et la pertinence des motifs extraits.

2.2 Étude de l'impact des contraintes de taille

Nous avons étudié l'impact des **contraintes de taille maximale** (paramètre max_len) sur les motifs obtenus. Trois configurations ont été testées :

- **Sans contrainte de taille**
- **Taille maximale de 3 items**
- **Taille maximale de 2 items**

Résultats obtenus :

- **Sans contrainte de taille :**
  - Nombre total de motifs : **130**
  - Distribution par longueur :
    - Longueur 1 : 87 motifs
    - Longueur 2 : 21 motifs
    - Longueur 3 : 15 motifs
    - Longueur 4 : 6 motifs
    - Longueur 5 : 1 motif
- **Avec une taille maximale de 3 :**
  - Nombre total de motifs : **123**
- **Avec une taille maximale de 2 :**
  - Nombre total de motifs : **108**

Analyse :

- Les contraintes de taille permettent de réduire la complexité et le nombre de motifs extraits, ce qui peut simplifier leur interprétation.
- En revanche, l'absence de contraintes permet d'explorer des motifs plus complexes, bien qu'ils soient souvent moins fréquents.

Conclusion sur les contraintes de taille :

- Elles sont pertinentes pour des analyses ciblées nécessitant des résultats simples et exploitables.
- Cependant, lorsque des relations complexes sont recherchées, il est préférable de ne pas les imposer.

---

3. Extraction de motifs à forte utilité avec l'algorithme QPlus

3.1 Justification de l'approche

L'intégration des utilités permet de découvrir des motifs particulièrement pertinents dans un contexte métier en prenant en compte des métriques comme la popularité ou la satisfaction des utilisateurs. Pour cela, l'algorithme **QPlus** a été utilisé dans différentes configurations :

1. **Avec ou sans utilité moyenne comme seuil dynamique**
2. **Avec ou sans contraintes de taille**
3. **Traitement en mémoire (in-memory) ou par fichiers (on-disk)**

3.2 Approche adoptée

Étapes principales :

1. **Calcul des utilités des motifs :**
   - L'utilité d'un motif est calculée comme la somme des utilités des items qui le composent.
   - Intégration possible de contraintes comme la taille maximale des motifs.
2. **Filtrage des motifs :**
   - Deux options ont été testées :
     - Utilité minimale fixée par l'utilisateur.
     - Seuil dynamique basé sur l'utilité moyenne des motifs.
3. **Gestion de la mémoire :**
   - Pour des datasets volumineux, une approche **on-disk** a été implémentée, divisant les données en fragments pour éviter les limitations mémoire.

Résultats expérimentaux :

- **Avec utilité moyenne et contrainte de taille :**
  - Motifs identifiés tels que **(B004X55L9I, 38026.84)** ont montré une utilité significative tout en respectant les contraintes.
- **Sans contrainte de taille :**
  - Des motifs complexes comme **(B00FA2RLX2, 80405.27)** ont été découverts, révélant une forte utilité.

3.3 Calcul des utilités pondérées

Une pondération a été introduite pour équilibrer la popularité (nombre de votes) et la qualité perçue (moyenne des notes).

- **Formule de pondération logarithmique :**

$$Utilitépondérée = \log(1 + votes) \times moyennedesnotes$$

Avantages :

- Évite les biais disproportionnés envers les items massivement évalués.
- Identifie des motifs capturant à la fois l'intérêt des utilisateurs et leur satisfaction.

---

4. Extraction de motifs à la demande

4.1 Justification de l'approche

L'extraction à la demande permet aux utilisateurs de définir des critères spécifiques (ex. seuil d'utilité minimale) pour filtrer les motifs. Cette approche est particulièrement utile dans les contextes où :

1. Les données sont volumineuses.
2. Les besoins varient selon les cas d'usage.

Avantages de l'extraction à la demande :

- **Flexibilité :** L'utilisateur peut ajuster les critères en fonction des objectifs.
- **Efficacité :** Réduction de la charge computationnelle grâce au filtrage dynamique.
- **Pertinence :** Les résultats sont plus alignés avec les besoins métiers.

---

5. Étude qualitative des motifs extraits

5.1 Objectif de l'étude

Analyser un échantillon de motifs (taille ≥ 1000) pour comprendre leur distribution, identifier des tendances, et valider la pertinence des résultats.

5.2 Méthodologie

1. **Sélection de l'échantillon :**
   - Un échantillon aléatoire de **1000 motifs** a été extrait des motifs à haute utilité.
2. **Analyse statistique :**
   - Calcul de la moyenne, de la médiane et de l'écart-type des utilités.

5.3 Résultats

- **Nombre de motifs analysés :** 1000
- **Statistiques descriptives :**
  - Moyenne des utilités : **1164.14**
  - Médiane des utilités : **593.30**
  - Écart-type : **1625.32**

Interprétation :

- La moyenne élevée indique une tendance générale positive.
- La différence entre la moyenne et la médiane suggère la présence de motifs à très haute utilité influençant la moyenne.
- L'écart-type élevé révèle une forte dispersion, confirmant une grande variabilité dans les motifs.

Avantages de l'approche QPlus

- **Utilités pondérées :**
  - Intégration de la popularité et de la qualité perçue des items pour un calcul d'utilité plus robuste.
- **Seuils adaptatifs :**
  - Identification de motifs pertinents sans dépendre d'un seuil fixe.
- **Scalabilité :**
  - Traitement efficace des ensembles de données volumineux grâce à l'approche on-disk.

Conclusion

L'intégration des utilités dans l'extraction des motifs offre une méthode robuste et adaptable pour l'analyse de données transactionnelles.

Points clés :

1. Les contraintes de taille simplifient les résultats, bien qu'elles limitent l'exploration de motifs complexes.
2. L'algorithme **QPlus** a démontré sa robustesse, en intégrant des paramètres comme l'utilité moyenne et les contraintes mémoire.
3. L'étude qualitative a validé la pertinence des motifs extraits.

Axes d'amélioration :

- **Optimisation des ressources :** Utiliser des infrastructures avec plus de RAM ou des outils distribués comme Spark.
- **Seuils adaptatifs :** Ajuster dynamiquement le seuil minimal pour un meilleur équilibre entre volume et pertinence.

Améliorations algorithmiques : Explorer des méthodes comme **FP-Growth** ou **ECLAT** pour une meilleure efficacité en mémoire.

4. Partie 3: Groupes d'utilisateurs

Dans cette dernière partie, nous allons analyser des groupes d'utilisateurs partageant des similarités dans leur manière d'acheter des articles.

La première étape a consisté à définir ce qu'est un groupe d'utilisateurs en fonction des attributs disponibles dans le dataset.
Notre première idée a été de suivre une approche similaire à celle vue en cours sur les systèmes de recommandation, en appliquant un algorithme KNN basé sur les utilisateurs ayant des goûts similaires pour les produits, à l'aide d'une matrice de corrélation. Cependant, nous avons abandonné cette idée après avoir eu une autre compréhension du projet et constaté des résultats peu satisfaisants.
Nous avions expérimenté avec des valeurs de *k* allant de 2 à 500. Les performances les plus élevées ont été obtenues pour *k = 2* (avec un *silhouette_score* de 0,1) et *k = 100* (avec un *silhouette_score* de -0,15) (avec silhouette_score= (([distance d'un point avec autre cluster plus proche] - [distance d'un point avec son cluster])/[max de ces 2 valeurs])). Ces scores étant proches de 0, ils indiquent que les points sont, en général, très proches des frontières entre les groupes, ce qui rend les clusters peu distincts et peu exploitables pour notre analyse.

Nous avons alors adopté une méthode totalement différente, en segmentant les utilisateurs en fonction de leurs habitudes d'achat et de notation.
Pour ce faire, nous avons regroupé les utilisateurs selon trois critères : le nombre d'avis laissés, la moyenne des notes attribuées et la moyenne des prix des produits. Ces trois champs ont ensuite été divisés en quatre catégories : *Low*, *Medium*, *High* et *Very High*, en suivant plus ou moins la distribution des données. Voici les plages définissant ces catégories :

Catégorie	price_mean (en $)	price_count	rating_mean
Low	0 - 15	0 - 2.1	0 - 3.5
Medium	15 - 35	2.1 - 5.9	3.5 - 4.4
High	35 - 100	5.9 - 10	4.4 - 4.9
Very High	100 - 1000	10 - 200	4.9 - 5.1

Nous avons ainsi pu commencer à analyser nos groupes. Pour cela, nous avons croisé les différentes catégories pour former des groupes du type : (price_mean='Low', price_count='Medium', rating_mean='Low'), générant ainsi 64 groupes distincts.
Pour chacun de ces groupes, nous avons compté le nombre d'utilisateurs, puis dans la même idée calculé le coverage, défini comme le rapport entre le nombre d'utilisateurs dans le groupe et le nombre total d'utilisateurs, et enfin nous avons également calculer le prix moyen, le nombre moyen d'avis, ainsi que la moyenne des notes attribuées pour chaque groupe.

La prochaine étape a consisté à analyser ces groupes à l'aide de l'algorithme MOMRI (*Multi-Objective Maximization for Representative Itemsets*).
Dans un premier temps, nous avons stocké les informations des utilisateurs de chaque groupe sous forme de dictionnaire. Ces dictionnaires contenaient les utilisateurs avec leurs items et notes et les groupes avec leurs utilisateurs associés. Bien que cette étape ait été longue à exécuter, elle nous a permis de préparer efficacement l'ensemble des données pour les étapes suivantes.

Nous avons ensuite appliqué l'algorithme *alpha-MOMRI* sur nos différents groupes et les notations attribuées à chaque item. L'objectif était de maximiser trois métriques importantes pour les avis des groupes : le coverage, la diversité (des items notés) et le diamètres (dispersion des utilisateurs au sein d'un groupe).
Les groupes générés dans cette étape n'étaient pas nécessairement croisés comme précédemment. Ils variaient entre 1 et 3 sous-groupes, tout en respectant la contrainte qu'aucune sous-catégorie ne provenait de la même colonne. Nous avons appliqué l'algorithme avec un paramètre **alpha** de 1,15 et un **sigma** de 100 (sélectionnant uniquement les groupes contenant au moins 100 utilisateurs). Cela nous a permis d'obtenir 124 groupes non alpha-dominés.

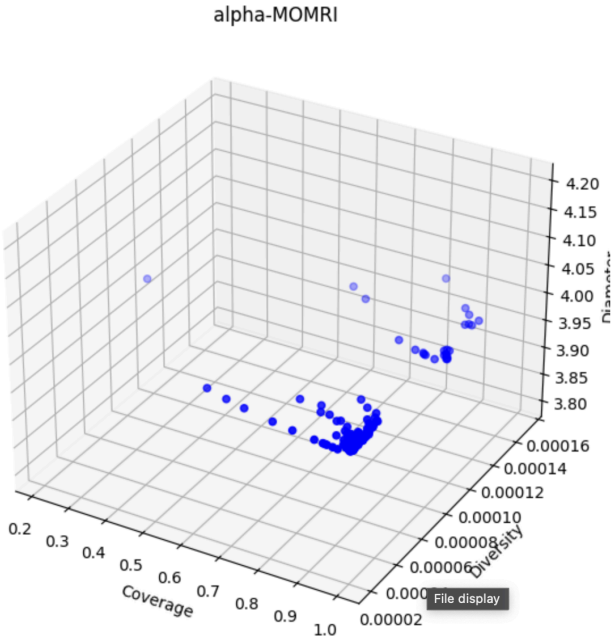
Dans une approche similaire, nous avons appliqué l'algorithme *heuristic-MOMRI* en utilisant l'algorithme de *Shotgun Hill Climbing*. Les mêmes paramètres ont été utilisés, à savoir un **alpha** de 1,15, un **sigma** de 100, et un nombre d'itérations fixé à 10. Cette méthode nous a permis d'obtenir 10 groupes.

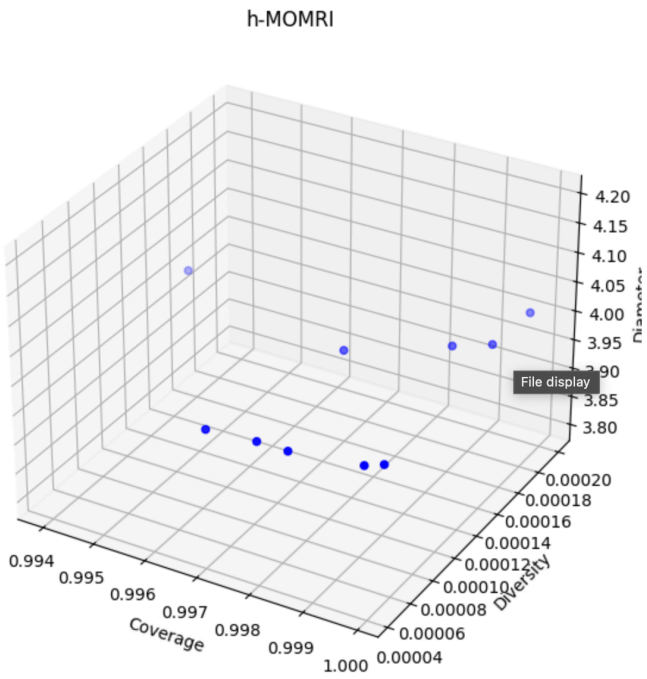
Enfin, il ne restait plus qu'à analyser nos résultats. À cette fin, nous avons synthétisé les informations obtenues dans le tableau suivant. À noter que les valeurs correspondent aux moyennes calculées sur l'ensemble des groupes retournés par chaque algorithme.

Algorithme	Temps (en S)	Coverage	Diversity	Diameter
alpha-MOMRI	406.5	0.947	0.000047	4.0
h-MOMRI	276.2	0.998	0.000105	4.0

Ainsi, on constate qu'alpha-MOMRI est plus long que l'heuristique, et que, malgré un plus petit nombre de groupes, nous couvrons moins de données en moyenne. Cela peut s'expliquer par le fait que les groupes utilisés pour heuritic-MOMRI pourraient être comparés à des "groupes de groupes". La diversité est également à l'avantage de h-MOMRI, avec une valeur supérieure à alpha. En revanche, le diamètre reste le même dans les deux cas, probablement parce que, dans chaque groupe, nous obtenons en moyenne toujours quelqu'un attribuant la note maximale et quelqu'un attribuant la note minimale.

Nous avons également réalisé un affichage visuel en 3D des valeurs obtenues par les différents groupes :





Conclusion

En somme, ce projet nous a permis de mettre en pratique les connaissances acquises dans le cadre de l'analyse exploratoire de données. Nous avons exploré des techniques avancées telles que l'extraction de motifs fréquents, les règles d'association et la segmentation des comportements utilisateurs.

Bien sûr, notre projet pourrait être amélioré de plusieurs façons. Par exemple, dans les deux premières parties, on pourrait essayer un *min_support* plus élevé permettant un filtrage plus efficace des données, mais aussi un plus petit pour obtenir plus de motifs, cependant nous avons été limités par la mémoire RAM disponible. Pour la troisième partie, nous aurions pu approfondir l'utilisation de l'algorithme MOMRI sur notre approche basée sur les KNN centrés sur les utilisateurs, et tester différentes valeurs de paramètres comme *alpha* pour observer leur impact sur les résultats.

Malgré ces limites, nous avons réussi à approfondir notre compréhension et notre maîtrise d'outils et d'algorithmes clés, notamment *Apriori*, *QPlus*, et *MOMRI*. Nous avons également appris à manipuler des datasets complexes nécessitant des prétraitements rigoureux, tout en explorant des approches innovantes pour la compression des données et le développement de systèmes d'extraction à la demande.

Ce projet constitue donc une base solide pour de futurs travaux, renforçant notre capacité à analyser des données volumineuses, interpréter des résultats pertinents, et maîtriser des outils essentiels pour résoudre des problématiques concrètes dans le domaine de la data science.

Cardi Julien
Ferroni Sandro
Moyo-Kamdem Auren
SCIA-G
ING3
PROMO 2025