T-DAT-901 : Système de recommandation

Equipe : Dylan Marion, Kevin Kiyak, Jérémie Kalabasi, Mélanie Ipekci, Samy Lemaissi

Groupe 23

Date : 28/02/2021

Sommaire

I. Présentation

Contexte

L'application a pour but de proposer un produit à un client, à partir d'une méthode de recommandation nous pouvons offrir un produit à un client selon son profil. Nous disposons d'un fichier csv contenant des informations sur des produits et des clients.

Technologies

Pour répondre aux besoins fournis par les entreprises, nous avons choisis de développer la solution en langage python avec des librairies de spécialisation IA et Big Data.

Le programme à été développé sur un notebook et le rendu sera un logiciel en python lançable en ligne de commande sans fichier de données.

Données

Conformément au plan donné aux entreprises ayant partagé leurs données, nous respectons les données personnelles et de consommation. Nous utilisons les données dans le seul but de fournir les recommandations et gardons les informations strictement confidentielles.

Gestion de projet

Nous avons utilisé des outils de communications tels que Teams ou Google Drive pour partager avec le client mais aussi entre les membres de l'équipe (surtout en période de pandémie du Covid19). De même pour la méthodologie de travail, nous travaillons en agile et nous avons donc utilisé Trello.

Pour la partie développement, chaque développeur est libre sur son IDE et pour le développement en python, nous avons utilisé l'outil Jupyter Notebook et Google Collab.

II. Exploration des données

Préparation des données

Tout d’abord, le fichier d’origine est un csv d’environ 8 millions de lignes. La première étape a été de savoir comment lire ce fichier sans manquer d'espace mémoire, pour cela nous avons utilisé “chunk” pour limiter le nombre de lectures.

Ensuite nous avons passé du temps à observer les données qui pouvaient nous être utiles pour nos statistiques.

Puis lorsque nous avions eu une bonne compréhension des données, nous avons procédé à plusieurs étape de vérification :

- Suppression des caractères spéciaux
- Vérification des champs nulle/vide
- Formatage des données (float64 -> int32)
- Normalisation des données

L’objectif de ce pré-processus est de sortir des données parfaitement lisibles et utilisables.

Clustering

K-means permet d'analyser un jeu de données caractérisées par un ensemble de descripteurs, afin de regrouper les données "similaires" en clusters.

La similarité entre deux données peut être inférée grâce à la "distance" séparant leurs descripteurs ; ainsi deux données très similaires sont deux données dont les descripteurs sont très proches.

Ci-dessous nous avons utilisé cette algorithme pour k = 2

```
Top terms per cluster:
Cluster 0:
 maquillage
 soins
 du
 visage
 corps
 parfumage
 capillaires
 solaires
 multi
 familles
Cluster 1:
 hygiene
 visage
 solaires
 soins
 sante
 parfumage
 naturelle
 multi
 maquillage
 familles
```

Nous pouvons remarquer les termes les plus fréquents par cluster pour la catégorie "famille".

Segmentation

Afin de regrouper les clients les plus dépensiers, nous avons utilisé le “RFM score calculation”. C’est une méthode d'analyse de la qualité d'un client selon trois critères.

- La **récence** : date du dernier achat ou temps écoulé depuis.
- La **fréquence** : périodicité moyenne des achats sur la période considérée.
- Le **montant** des achats réalisés par ce client sur la période étudiée.

Customer Segmentation

RFM Score Calculations

RECENCY (R) : Days since last purchase
FREQUENCY (F): Total number of purchases
MONETARY VALUE (M) : Total money this customer spent

Segment	RFM	Description	Marketing
Best Customers	111	Bought most recently and most often, and spend the most	No price incentives, new products, and loyalty programs
Loyal Customers	X1X	Buy most frequently	Use R and M to further segment
Big Spenders	XX1	Spend the most	Market your most expensive products
Almost Lost	311	Haven't purchased for some time, but purchased frequently and spend the most	Aggressive price incentives
Lost Customers	411	Haven't purchased for some time, but purchased frequently and spend the most	Aggressive price incentives
Lost Cheap Customers	444	Last purchased long ago, purchased few, and spent little	Don't spend too much trying to re-acquire

Source: Blast Analytics Marketing

Voici les résultat obtenu, l’indice RFM de la colonne “RFM Score" nous permet d’estimer la qualité du client.

CLI_ID	recency	frequency	monetary_value	r_quartile	f_quartile	m_quartile	RFMScore
1490281	10	5	1050.62	2	3	3	233
13290776	12	9	2165.10	3	2	2	322
20163348	1	1	29.80	1	4	4	144
20200041	1	2	357.90	1	4	4	144
20561854	3	3	238.10	1	3	4	134

Analyse statistique

Pour la visualisation en 2d ou 3d des différentes figures, nous avons utilisé “matplotlib”.

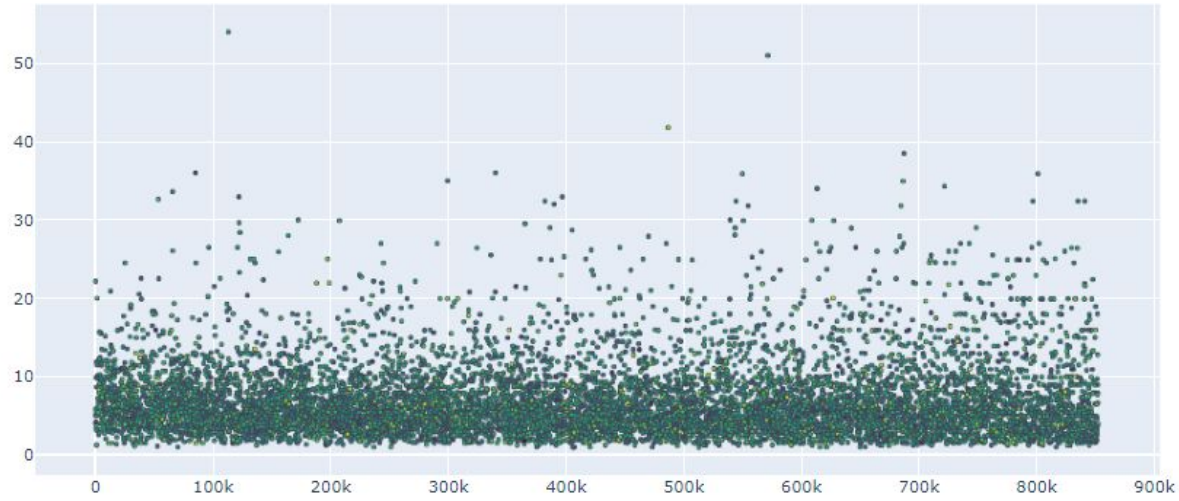
Analyse des dépenses des clients

Nous avons créé un nouveau data frame pour avoir le numéro d'id du client et le coût moyen d'un produit qu'il achète.

	index	CLI_ID	AVG_SPEND
0	0	1490281	3.680000
1	1	13290776	6.166667
2	2	20163348	7.450000
3	3	20200041	3.350000
4	4	20561854	2.350000
...	...	...	...
853509	853509	997048745	4.950000
853510	853510	997048751	5.760000
853511	853511	997048769	7.265000
853512	853512	997048777	3.878947
853513	853513	997385337	5.193478

853514 rows × 3 columns

La figure montre l'index d'un client en fonction du prix moyen des produits qu'il achète.

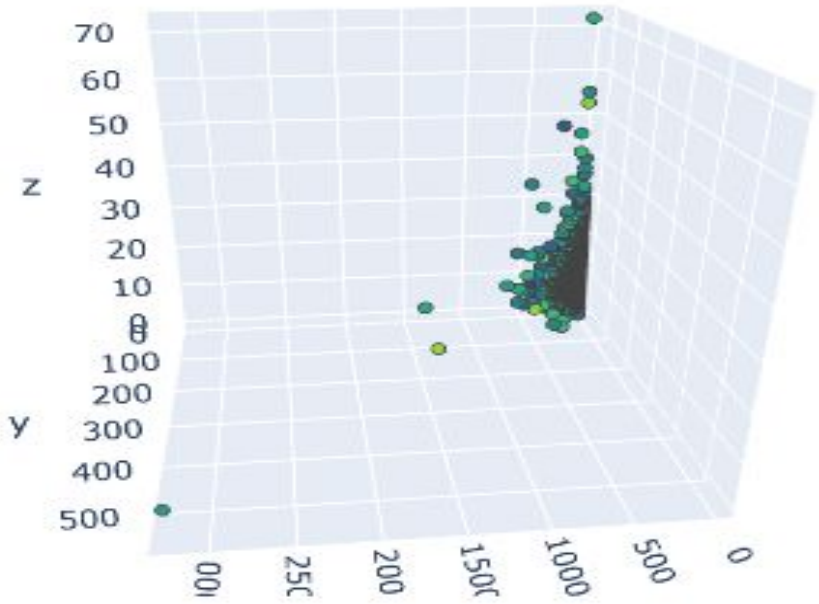


On construit un dataframe qui représente la somme des dépenses et le nombre de dépenses de chaque client.

	index	CLI_ID	SUM_SPEND	NB_PRODUCTS	AVG_SPEND
0	0	1490281	18.40	5	3.680000
1	1	13290776	55.50	9	6.166667
2	2	20163348	7.45	1	7.450000
3	3	20200041	6.70	2	3.350000
4	4	20561854	7.05	3	2.350000
...	...	...	...	...	...
853509	853509	997048745	19.80	4	4.950000
853510	853510	997048751	28.80	5	5.760000
853511	853511	997048769	72.65	10	7.265000
853512	853512	997048777	73.70	19	3.878947
853513	853513	997385337	119.45	23	5.193478

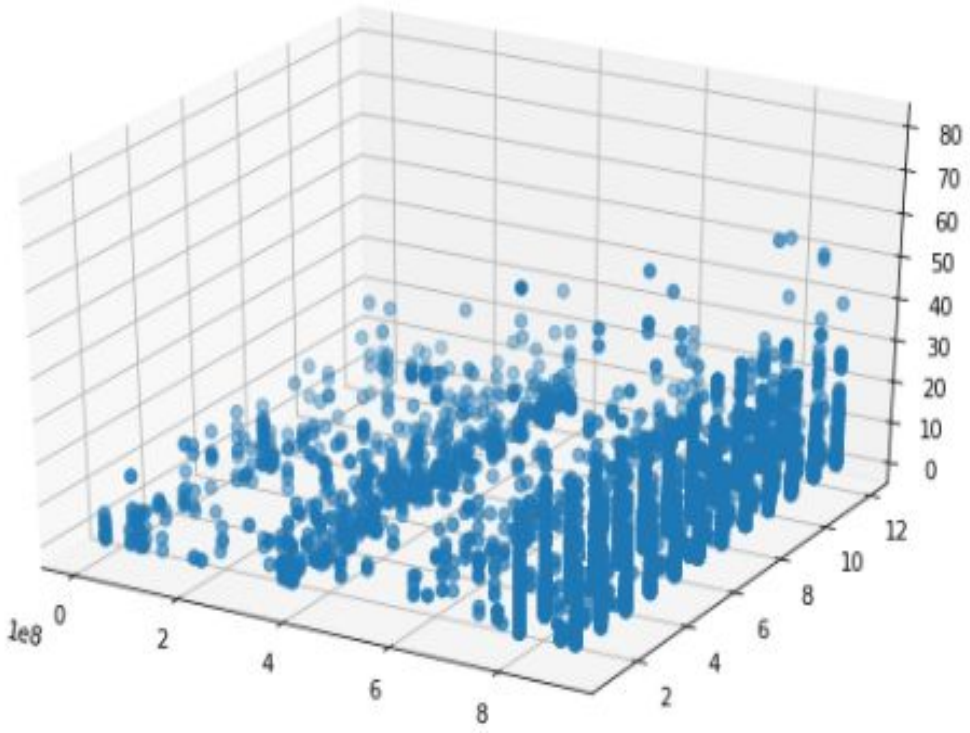
[853514 rows x 5 columns]

Cette figure montre la somme d'argent dépensé par un client en fonction du nombre de produit qui lui même est en fonction du prix moyen de dépense en 3D.



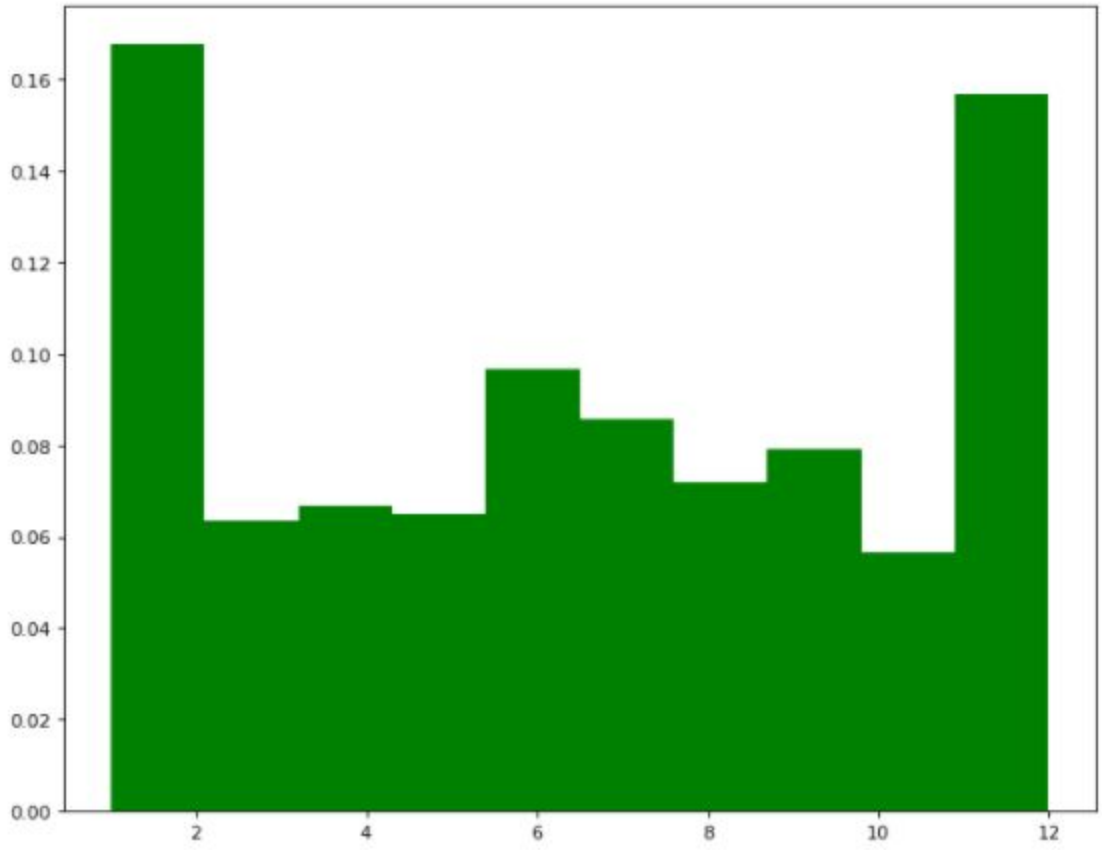
Analyse des mois les plus rentable

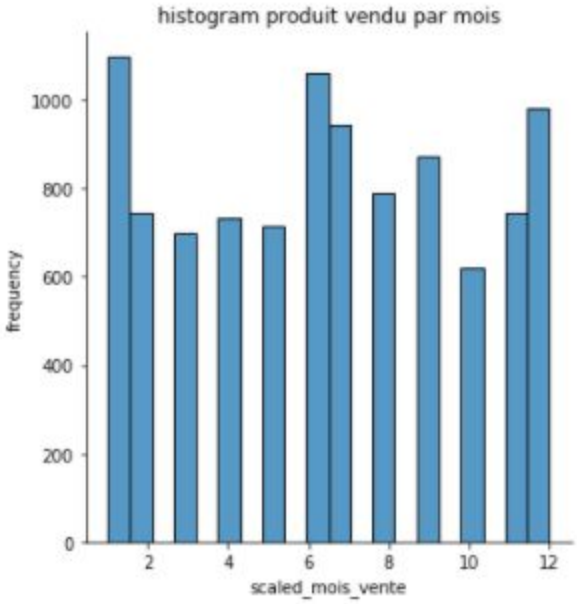
Ci-dessous une représentation en 3d des mois les plus rentable, on peut constater une hausse des produits acheté pour le mois de décembre



On remarque que la période des fêtes de fin d'années est très rentable.

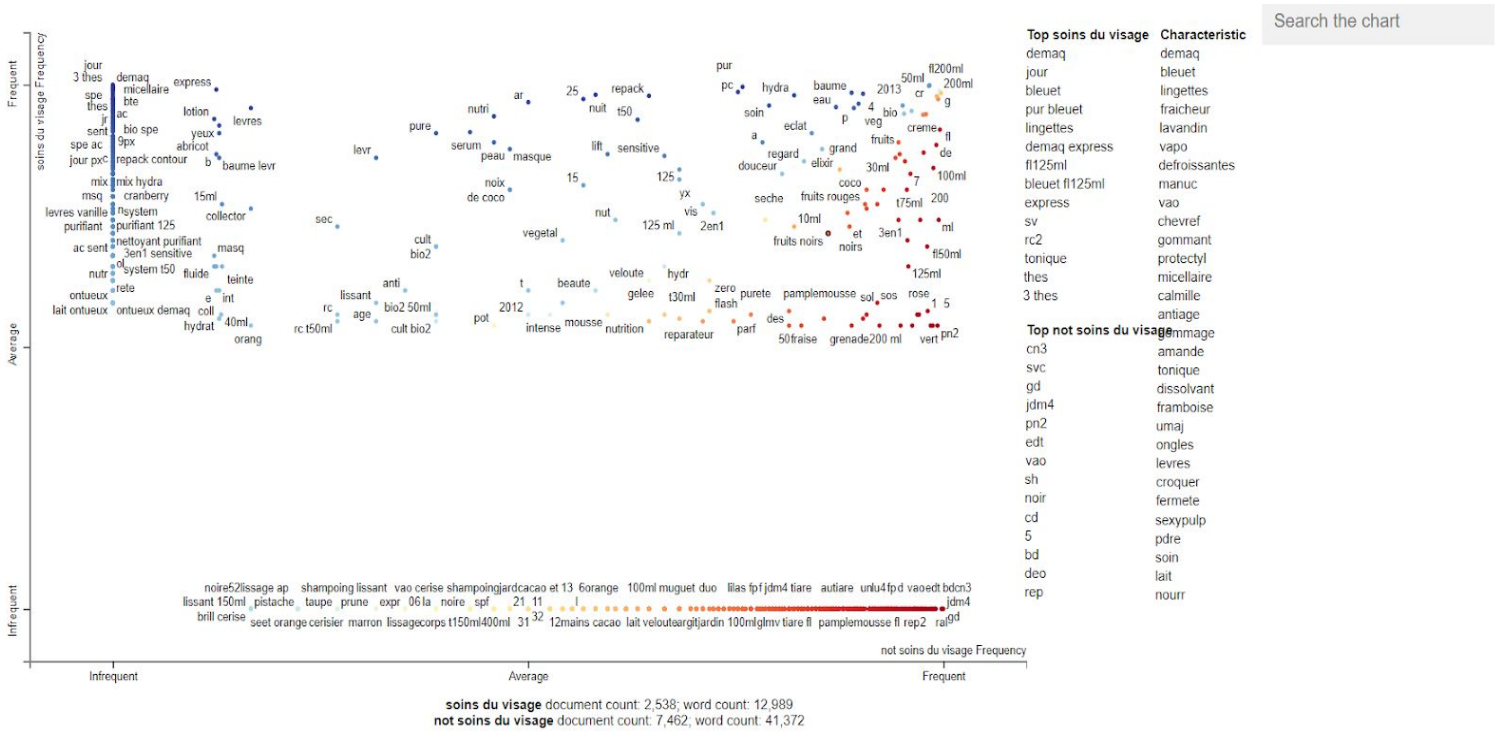
Nous pouvons voir une autre figure des fréquences d'achat par mois, on peut constater une augmentation en été.





Analyse syntaxique

Afin d'aller plus loin dans la visualisation des mots nous avons utilisé du **NLP** (Natural langage processing) avec "scatter texte", qui est un outil pour trouver des termes distinctifs dans les corpus et les afficher dans un nuage de points HTML interactif. Les points correspondant aux termes sont étiquetés de manière sélective afin qu'ils ne se chevauchent pas avec d'autres étiquettes.



Ci-dessus nous pouvons voir les mots les plus fréquents et moins fréquents en fonction de l'intensité de la couleur. Sur la droite nous pouvons voir les termes les plus fréquents pour la catégorie "soins du visage".

III. Recommandation

Content based method

Cette technique se focalise sur les caractéristiques des produits afin de recommander aux utilisateurs de nouveaux produits qui auront des propriétés similaires aux produits avec lesquels ils ont déjà interagi.

Afin de traiter au mieux les données, nous avons procédé à plusieurs étapes :

- Retirer les caractères spéciaux
- Vérifier si des données manque
- Réduire la mémoire

L'étape suivante consiste à créer un "bag of words" un sac de mots, nous avons donc créé une colonne bag of word et changé l'index par le libellé du produit.

LIBELLE	CLI_ID	bag_of_words
gd jdm4 pamplemousse fl 200ml	1490281	hygiene hygdouchejardinmonde hygjdm
gd jdm4 pamplemousse fl 200ml	1490281	hygiene hygdouchejardinmonde hygjdm
cr jr parf bio spe ac sent 50ml	1490281	soinsduvisage viscjourjeunesspecifique visjeun...
eau micellaire 3 thes fl200ml	1490281	soinsduvisage visdemaqaaar visaaardemaqlotion
gd jdm4 tiare fl 200ml	1490281	hygiene hygdouchejardinmonde hygjdm
...	...	...
deo parfume 100ml evidence	903871816	hygiene hygdeoparfums hygparfumee
crayon sourcils chatain cn3 1 1g	903871816	maquillage maqyeuxsourcils maqyeuxclassique
crayon regard noir cn3 1 3g	903871816	maquillage maqyeuxcrayons maqyeuxclassique
deo parfume 100ml evidence	903871816	hygiene hygdeoparfums hygparfumee
deo parfume 100ml evidence	903871816	hygiene hygdeoparfums hygparfumee

Ensuite nous avons utilisé "**cosine similarity**” de **scikit-learn** pour mesurer la similitude entre deux vecteurs en calculant le cosinus de l’angle entre deux vecteurs.

```
cosine_sim = cosine_similarity(count_matrix)
cosine_sim
```

```
array([[1.        , 1.        , 0.        , ..., 0.        , 0.33333333,
        0.33333333],
       [1.        , 1.        , 0.        , ..., 0.        , 0.33333333,
        0.33333333],
       [0.        , 0.        , 1.        , ..., 0.        , 0.        ,
        0.        ],
       ...,
       [0.        , 0.        , 0.        , ..., 1.        , 0.        ,
        0.        ],
       [0.33333333, 0.33333333, 0.        , ..., 0.        , 1.        ,
        1.        ],
       [0.33333333, 0.33333333, 0.        , ..., 0.        , 1.        ,
        1.        ]])
```

Enfin, nous avons implémenté une fonction de recommandation. Ci-dessous le résultat pour un produit dont le libellé est “gd jdm4 pamplemousse fl 200ml”. On peut voir que le système nous propose d'autres produits similaires correspondant à un “gel douche”.

```
recommendation(bestSeller[0])
```

```
['gd jdm4 grenade fl200ml',
 'gd jdm4 pamplemousse fl 200ml',
 'gd jdm4 orange fl 200ml',
 'gd jdm4 grenade fl200ml',
 'gd jdm4 the vert fl200ml',
 'gd jdm4 the vert fl200ml',
 'gd jdm4 grenade fl200ml',
 'cd jdm riz du laos fl 200ml',
 'gd jdm4 tiare fl 200ml',
 'gd jdm4 orange fl 200ml']
```

Nous avons essayé d’utiliser la méthode de recommandation collaborative filtering, mais sans succès car cette méthode repose sur les préférences du client. Nous aurions pu créer une colonne préférence en fonction de la fréquence d’achat d’un produit.

Système de recommandation

Nous avons mis en place un programme capable de prendre en entrée l'identifiant d'un client et proposer une liste de produits en utilisant la méthode de content-based.

Le principe est simple l'utilisateur entre son identifiant, le système récupère son profil ainsi que tous les produits achetés et renvoie le libellé du produit qu'il a le plus acheté en dernier. Ce produit est ensuite envoyé à la fonction de recommandation qui recommande une liste de produits similaires.

IV. Conclusion

Bilan

Le projet d'un point de vue général est très intéressant, il nous a permis de découvrir ou d'améliorer certaines pratiques nécessaires pour la création d'un projet en IA.

De plus, nous avons découvert d'autre façon de recommander un produit, même si nous en avons implémenté qu'une, nous avons étudié la méthode "collaborative filtering".

Malgré un manque de temps qui nous aurait permis d'approfondir nos recherches et d'améliorer certains de nos résultats, nous sommes en mesure de créer un système capable de recommander une liste de produits.

Perspectives

Concernant les axes d'améliorations, nous avions pensé à la création d'un affichage "user friendly" qui propose de visualiser le profil d'un client et prédire ses futures achats en fonction des données analysées.