h e g

Haute école de gestion
Genève

Comment utiliser des algorithmes pour proposer des abonnements aux utilisateurs du bar à café mia&noa en fonction de leurs profils et utilisations

Travail de Bachelor réalisé en vue de l'obtention du Bachelor HES

par :

Mikael BOLENS

Conseiller au travail de Bachelor :

Michel DERIAZ

Genève, le 24 septembre 2021

Haute École de Gestion de Genève (HEG-GE)

Filière informatique de gestion

Hes·so GENÈVE
Haute Ecole Spécialisée de Suisse occidentale

Déclaration

Ce travail de Bachelor est réalisé dans le cadre de l'examen final de la Haute école de gestion de Genève, en vue de l'obtention du titre Bachelor of Science HES-SO en informatique de gestion.

L'étudiant a envoyé ce document par email à l'adresse remise par son conseiller au travail de Bachelor pour analyse par le logiciel de détection de plagiat URKUND, selon la procédure détaillée à l'URL suivante : https://www.urkund.com .

L'étudiant accepte, le cas échéant, la clause de confidentialité. L'utilisation des conclusions et recommandations formulées dans le travail de Bachelor, sans préjuger de leur valeur, n'engage ni la responsabilité de l'auteur, ni celle du conseiller au travail de Bachelor, du juré et de la HEG.

« J'atteste avoir réalisé seul le présent travail, sans avoir utilisé des sources autres que celles citées dans la bibliographie. »

Fait à Genève, le 24 septembre 2021

Mikael Bolens

Remerciements

Je remercie grandement Maren Knief et Sebastian Knief, fondateurs de mia&noa, qui m'ont permis d'effectuer ce travail de Bachelor. Leur soutient a été essentiel pour mener ce projet à bien et leur disponibilité m'a permis d'avancer et de comprendre leur besoin. Je suis ravi d'avoir eu la chance de collaborer avec une entreprise qui a pour objectif d'arrêter le gaspillage.

Je souhaite également remercier Michel Deriaz qui m'a permis de d'effectuer cette recherche à mi-temps pour que je puisse continuer mon travail actuel et effectuer mon service militaire obligatoire. Son soutient m'a permis d'aller de l'avant et ses conseils avisés ont été indispensables.

Finalement, je remercie chaleureusement Anne-Marie Bolomey ainsi que mon père, Nicolas Bolens, qui ont pris le temps de relire mon travail pour corriger l'orthographe.

Résumé

Mia&noa est une entreprise jeune et innovante qui cherche une solution pour pouvoir analyser la consommation de ses clients dans le but de proposer de nouveaux plans tarifaires. Une analyse des transactions ainsi que plusieurs propositions d'amélioration seront étudiées. Les algorithmes d'analyse demandent une quantité importante de données pour effectuer des prédictions viables. Il est important de ne pas négliger l'évolution rapide de l'entreprise qui doit gérer de plus en plus de données.

Le but de cette recherche est de trouver une stratégie pour regrouper les utilisateurs selon leurs comportements afin de créer des offres qui apporteraient une plus-value aux clients de mia&noa. L'utilisation d'une application permet une multitude de possibilités ainsi que des nouveaux moyens de communication. Cependant, le bar à café doit être accessible pour tous les utilisateurs, même sans smartphone. Les modifications apportées ne doivent donc pas être une complexité supplémentaire pour ne pas nuire à l'expérience de l'utilisateur.

La recherche ainsi que le code produit auront pour objectif de catégoriser les consommations des clients pour proposer une nouvelle expérience tarifaire sous forme d'abonnement café ou de promotion. L'ambition est de satisfaire le consommateur en le surprenant sur l'utilisation du bar à café tout en facilitant et diminuant au maximum le temps nécessaire à la commande.

Toutes les étapes essentielles à la création d'un data warehouse en local sont expliquées et détaillées pour pouvoir effectuer les analyses nécessaires. Les jeux de données produits sont ensuite analysés par plusieurs algorithmes afin d'étiqueter les utilisateurs selon leurs habitudes de consommation.

La fin de la recherche est consacrée à l'analyse de données en continu afin que mia&noa puisse analyser l'évolution des comportements des utilisateurs et la réussite ou non de leur stratégie tarifaire. Les services cloud actuels offrent de nombreux outils pour le traitement du big data et la comparaison entre plusieurs services sont analysés afin de proposer la solution la plus adaptée à l'entreprise.

Liste des tableaux

Liste des figures

1. Introduction

Le 8 juillet 2021, mia&noa a mis à ma disposition un fichier Excel contenant la liste des transactions depuis le 29 juin 2021. Cette liste est mise à jour régulièrement, il faudra donc un système d'analyse réutilisable pour pouvoir affiner en continu les résultats obtenus. Pour créer un algorithme capable de proposer un abonnement qui corresponde aux habitudes des consommateurs, il est indispensable de comprendre les données qui nous sont fournies. Seules les données qui auront été jugées pertinentes seront utilisées par l'algorithme pour proposer l'abonnement adéquat.

Une phase d'analyse préliminaire est donc indispensable. Cette phase sera nécessaire pour déterminer à quel moment du processus on intègre la proposition d'un abonnement et comment modifier le processus actuel.

Une première analyse doit être également effectuée sur le fichier source. Celui-ci contient énormément d'informations, ce qui complique grandement la distinction de groupe. Pour avoir une vision graphique et filtrée plusieurs approches sont envisageables, le but étant d'obtenir un jeu de données filtré et pertinent.

Une fois que les jeux de données les plus pertinents seront obtenus, l'objectif de ce travail est d'en trier des tendances pour pourvoir catégoriser les habitudes de consommation des clients. Un algorithme de prédiction basé sur la probabilité n'est pas envisageable car les plans tarifaires n'existent pas et ceux-ci doivent être basés sur les tendances à analyser. Un algorithme de clustering sera donc utilisé. Ce type d'algorithme a la particularité d'avoir un modèle d'apprentissage non supervisé (Brownlee 2020), ce qui signifie qu'il ne se fonde pas sur des prédictions basées sur l'apprentissage, mais qu'il analyse une tendance sur la totalité des données.

Un objectif de ce travail comporte la compréhension et l'utilisation des algorithmes de clustering.

Pour terminer ce travail et donner satisfaction aux mandants, le code permettant l'analyse et l'algorithme de clustering sera transmis à mia&noa. Toutes les propositions concernant les plans tarifaires basés sur une stratégie de prix ne sont fournies que dans un but expérimental. Une fois le travail terminé, je céderai tous les droits à mia&noa, qui sera en mesure d'utiliser le code fourni ou pas. La proposition d'intégration sera faite sous forme de maquette pour faciliter une intégration potentielle dans l'application en production.

2. Analyse préliminaire

Le fichier fourni à analyser est un fichier possédant l'extension .xlsx. Le 15 juillet 2021, 6644 enregistrements étaient disponibles, chacun possédant 15 attributs. Le nombre d'enregistrements est mis à jour régulièrement, toutefois le nom des attributs ne changera pas.

2.1 Analyse des attributs :

A. Timestamp
   - Cette colonne contient le timestamp de la transaction. *Les données sont* ***continues***.

B. Customer ID
   - Cette colonne contient l'ID unique d'un utilisateur. *Les données sont* ***discrètes***

C. Account
   - Cette colonne contient le type de compte de l'utilisateur. *Les données sont* ***discrètes.***

D. Account vertical
   - Cette colonne contient le type de compte de l'utilisateur. C'est la vision verticale du type de compte. *Les données sont* ***discrètes.***

E. Transaction
   - Cette colonne indique le type de transaction. *Les données sont* ***discrètes*** *et les valeurs possibles sont "Brevage order", "Coffee Credit added" ou null.*

F. Discount
   - Cette colonne contient la réduction en pourcentage pour la transaction. *Les données sont* ***continues***.

G. Brevage Groupe
   - Cette colonne contient le type de boisson. *Les données sont* ***discrètes.***

H. Drink
   - Cette colonne contient la boisson choisit pour la transaction. Elle est la sous classe de "Brevage Groupe" *Les données sont* ***discrètes.***

I. Size
   - Cette colonne contient la taille de la boisson choisit. *Les données sont* ***discrètes.*** *Les valeurs possibles sont "regular", "small", thermos" ou null.*

J. Milk type
    - Cette colonne contient le type de lait choisit. *Les données sont **discrètes.** Les valeurs possibles sont "regular", "small", "thermos" ou null.*

K. Milk level
    - Cette colonne contient la quantité de lait choisit. *Les données sont **discrètes.** Les valeurs possibles sont "a lot of", "little", "no", "standard" ou null.*

L. Sugar lever
    - Cette colonne contient la quantité de sucre choisit. *Les données sont **discrètes.** Les valeurs possibles sont les mêmes que pour la quantité de lait.*

M. Location
    - Cette colonne contient la location de la transaction. *Les données sont **discrètes.** Les valeurs possibles sont "Chêne-Bourg", "HEG", "Event", "Online"*

      Les transactions Online concernent le chargement en crédit d'un utilisateur et sont liées à la valeur *"Coffee Credit added"* de l'attribut *"Transaction".*

      Les autres valeurs seront donc liées à la valeur *"Brevage order"* de l'attribut *"Transaction".*

N. Payement type
    - Cette colonne contient le type de payement de la transaction. *Les données sont **discrètes.***

      Les valeurs possibles sont *"Bank Transfer", "Coffee Credit", "Master", "Paypal", "Postcard", "Twint", "Visa", "Free Credit" ou null.*

O. Amount
    - Cette colonne contient la somme de la transaction. *Les données sont **continues***.

2.1 Choix des logiciels

Le fichier source étant un fichier Propriétaire Microsoft, mon objectif est de pouvoir utiliser ce même logiciel pour visualiser les données de façon graphique. Pour effectuer cela, plusieurs options sont envisageables. Excel est un logiciel puissant permettant un certain nombre d'opérations. Des graphiques peuvent directement être générés à partir d'Excel et il est également capable d'effectuer des calculs complexes à l'aide de formules. Un premier inconvénient concerne toutes les colonnes contenant des valeurs qualitatives. En effet, le but est de pouvoir générer un jeu de données pouvant être analysées par un algorithme de clustering. Les colonnes qualitatives discrètes doivent donc être traitées pour qu'elles puissent être analysées. Un autre problème réside dans le filtrage des données. Le fichier source est complet mais il est difficile de l'interpréter tel quel pour tirer des tendances directement depuis le logiciel Excel.

Une multitude de solutions d'analyse de données est disponible. Ces services sont mis à disposition par les plus grandes entreprises comme Google, Amazon, Facebook ou d'autres entreprises plus petites. Ces entreprises proposent des solutions permettant d'avoir un suivi des données en direct. De plus, il est possible de générer des graphiques pour avoir une lecture plus facile des données. Pour cette recherche, seul un fichier Excel est disponible. Configurer un environnement complet en ligne serait disproportionné. Microsoft propose des solutions correspondant aux besoins de la recherche.

Pour effectuer de telles analyses, une solution serait donc de persister les données des transactions dans une base de données. Pour effectuer cette opération, le logiciel Excel propose une solution. Il n'est cependant pas envisageable de persister les données sans effectuer une transformation au préalable en raison de la complexité du fichier.

Le fichier source contient les transactions mais celles-ci ne sont visibles que dans une seule dimension et cela permettrait seulement la création d'une table « transaction ». Microsoft offre un grand nombre de logiciels et de services. Un avantage est que l'entreprise s'assure de la maintenance de ses logiciels et facilite la compatibilité entre ceux-ci. Cet avantage est également un inconvénient dans une certaine mesure. Le fait que l'entreprise développe un ensemble de solutions, lorsque vous utilisez un de leurs logiciels, il vous sera difficile de sortir de l'environnement de Microsoft.

Pour visualiser et garder Excel comme outil d'analyse préliminaire, la première étape sera faite essentiellement à l'aide d'outils fournis par Microsoft. Une première analyse sera effectuée en enregistrant ces données dans une base de données fournie par Microsoft SQL SERVER en local. Étant donné que ces données doivent être transformées, une étape transitoire sera nécessaire pour convertir les données fournies (format .xlsx) et les enregistrer dans la base de données.

La transformation des données est donc indispensable. Le module 625-2 Business intelligence enseigné par David Billard en 2020 présentait le logiciel Pentaho Data Integration. La version communautaire est gratuite et open source. Le logiciel propose également une version d'entreprise basée sur un abonnement payant.

Ce logiciel est très complet et possède plusieurs applications. Ce qui nous intéresse dans le cadre de cette recherche, c'est qu'il permet une manipulation de données via des étapes précises. La transformation se fait via un schéma à une ou plusieurs entrées et une ou plusieurs sorties. Chaque étape transitoire correspond à une modification qui sera appliquée à un moment précis. La représentation graphique améliore la compréhension et rend le logiciel plus facile d'utilisation.

2.2 Logiciels

Pentaho Data integration distribué par Hitachi. Version 9.1

SQL server Management Studio distribué par Microsoft. Version 18.7.1

Visual Studio Entreprise 2019 distribué par Microsoft. Version 16.10.3

Microsoft Excel pour Microsoft 365 MSO distribué par Microsoft

2.3 Transformation des données

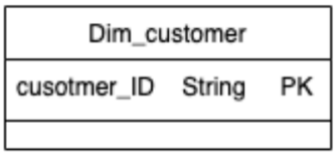
Pour stocker les données dans une base de données Microsoft Server SQL afin de pouvoir les visualiser, Excel possède une fonction pour effectuer des requêtes SQL et d'afficher les résultats sur un fichier croisé dynamique. Pour les visualiser selon différente dimension, chaque attribut discret sera isolé et les valeurs possibles seront stoker dans la table correspondante.

Grace à l'outil Pentaho Data integration, une modélisation en étoile des tables logique est possible. Le but étant d'avoir une table de fait et des tables de dimension. Chaque

table de dimension possède sa clé primaire et la table de fait sera au centre du schéma. La table de fait aura plusieurs colonnes contenant les valeurs et les clés étrangère associé aux tables de dimension.

Pour créer une classe de dimension, le processus est le même pour la majorité des attributs. La première dimension à être créé est celle des clients. Celle-ci nous permettra d'identifier la consommation de chaque utilisateur. Le fichier fournit par mia&noa étant anonymisé, seul l'identifiant de l'utilisateur sera enregistré. Cette id sera également la clé primaire de la table.

Figure 1 : Table customer



(Créé avec draw.io)

Les étapes pour la transformation des données *"*customer*"* founis par le fichier Excel via penthao sont les suivantes :

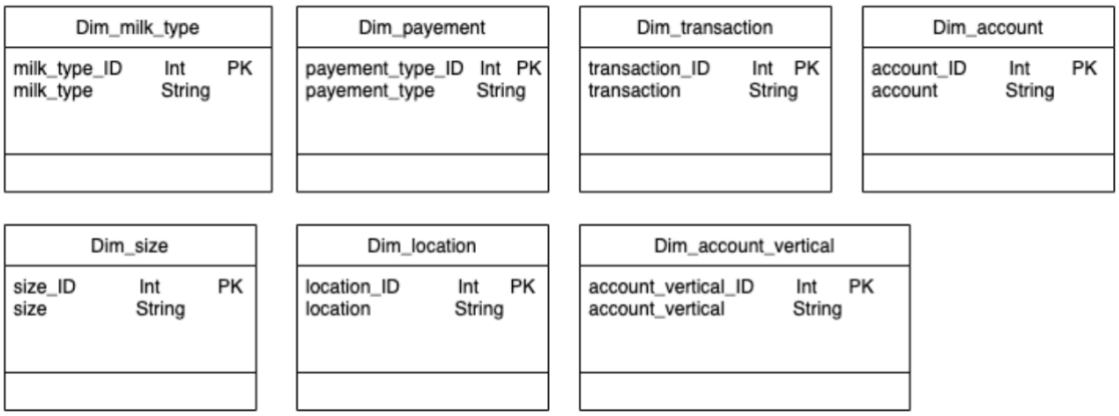
1. Choisir le fichier Excel à traiter
2. Sélectionner la colonne *"Customer ID",* du fichier Excel
3. Trier la sélection pour pouvoir identifier chaque ID
4. Supprimer les doublons pour avoir un set de customers
5. Enregistrer les données traitées dans la base de données SQLSERVER.

Figure 2 : Schéma penthao customer

(Capture d'écran penthao)

La prochaine étape consiste à créer toutes les tables de dimension qui ne dépendent d'aucune autre table logique.

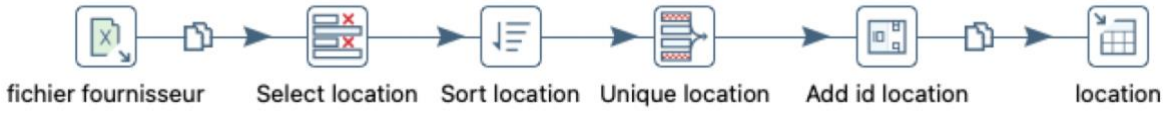
Figure 3 : Tables des dimensions



(Créé avec draw.io)

Pour les colonnes : "Account", "Account Vertical", "Transaction", "Size", "Milk Type", "Location" et "Payment Type", la transformation est similaire à celle du client. Seul une étape supplémentaire est ajoutée à la fin du processus pour créer manuellement une clé primaire de type Integer.

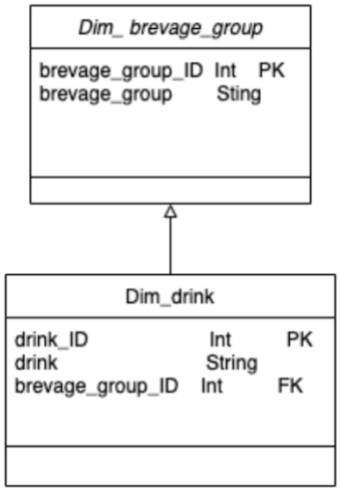
Figure 4 : Schéma penthao location



(Capture d'écran penthao)

L'analyse préliminaire nous a montré que les colonnes "Drink" et "Beverage Group" avaient une corrélation. La table Dim_drink étant la sous-classe de Dim_beverage_group, elle possèdera la clé étrangère pour créer la relation entre les deux tables.

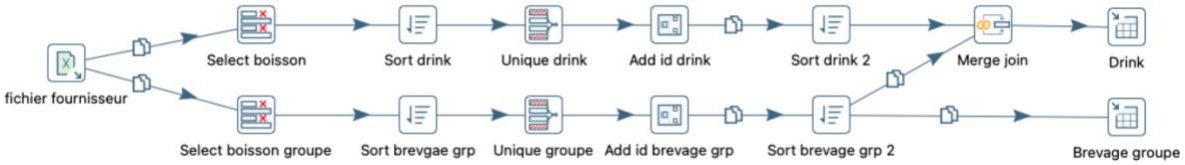
Figure 5 : Table drink et brevage group



(Créé avec draw.io)

Pour effectuer cette transformation, une étape *"*merge*"* est nécessaire. Celle-ci ajoutera l'id créé de la table *"Dim_brebage_group"* correspondant à la boisson de la transaction.

Figure 6 : Schéma penthao drink et bravge group



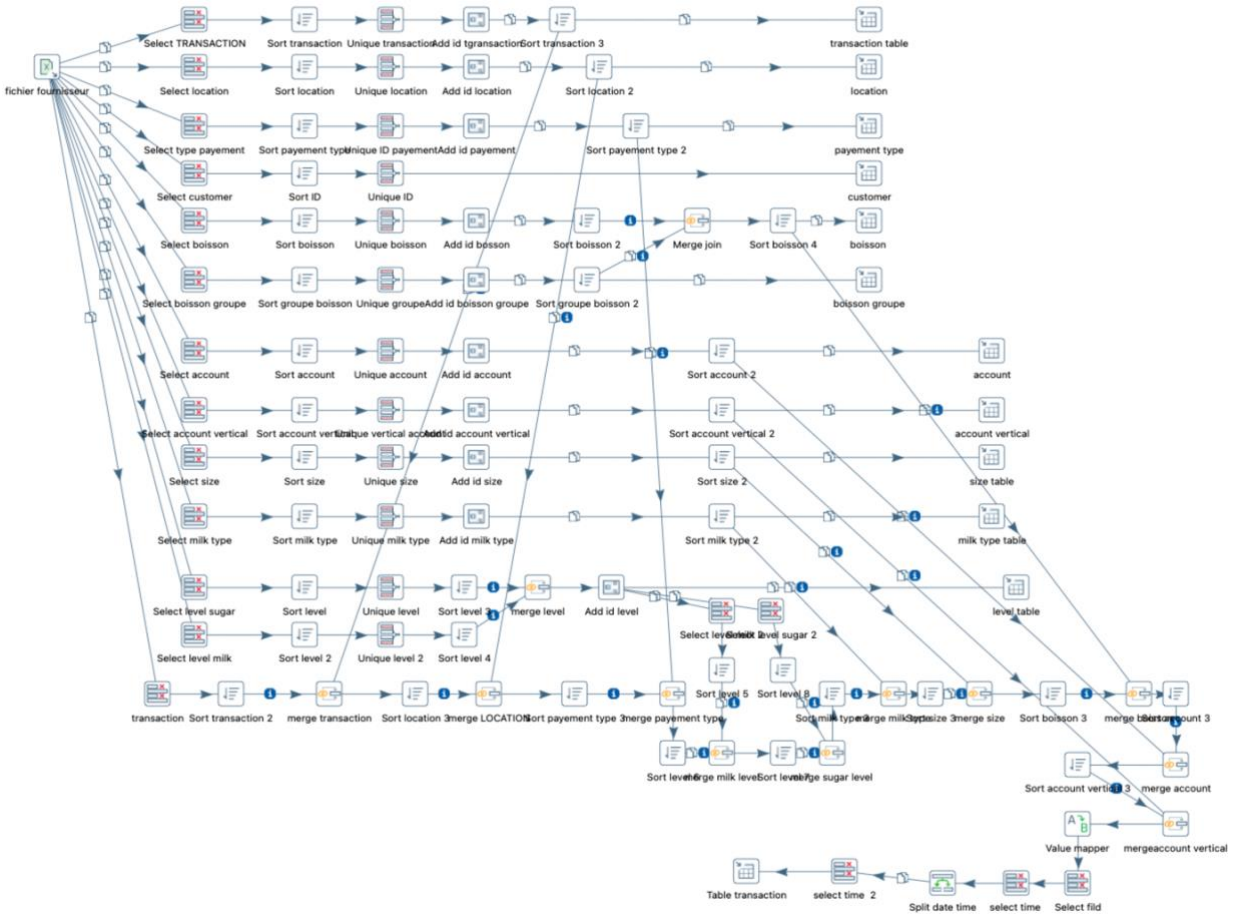
(Capture d'écran penthao)

A partir de ces deux nouvelles dimensions, il sera possible de créer une hiérarchie lors de la recherche. Cette hiérarchie nous permettra d'effectuer une recherche sur toutes les transactions concernant un groupe de boisson ou alors sur chaque boisson individuellement.

Il a également été constaté que les attributs ″Milk level″ et ″ Sugar level″ possédaient les mêmes valeurs discrètes possibles. Une table Dim_level sera donc créée et sera utilisée par les deux attributs pour définir la quantité de lait ou de sucre.

Le schéma de transformation final enregistre donc la table de fait avec toutes les clés étrangères indispensables à la modélisation en étoile des tables logiques. Cependant, deux tables essentielles seront rajoutées au schéma. Dim_Time et Dim_Date seront des tables permettant d'avoir une vision temporelle des ventes. Chaque transaction possède un timestamp contenant la date et l'heure de la transaction ; cet attribut sera donc utilisé pour créer la clé étrangère liant la transaction à un calendrier interne au serveur Microsoft. La table Dim_Time doit, quant à elle, être créée manuellement. Une table permettant d'avoir une dimension sur 24H sera générée et la clé primaire sera l'heure au format hh :mm :ss.

Figure 7 : Schéma penthao final



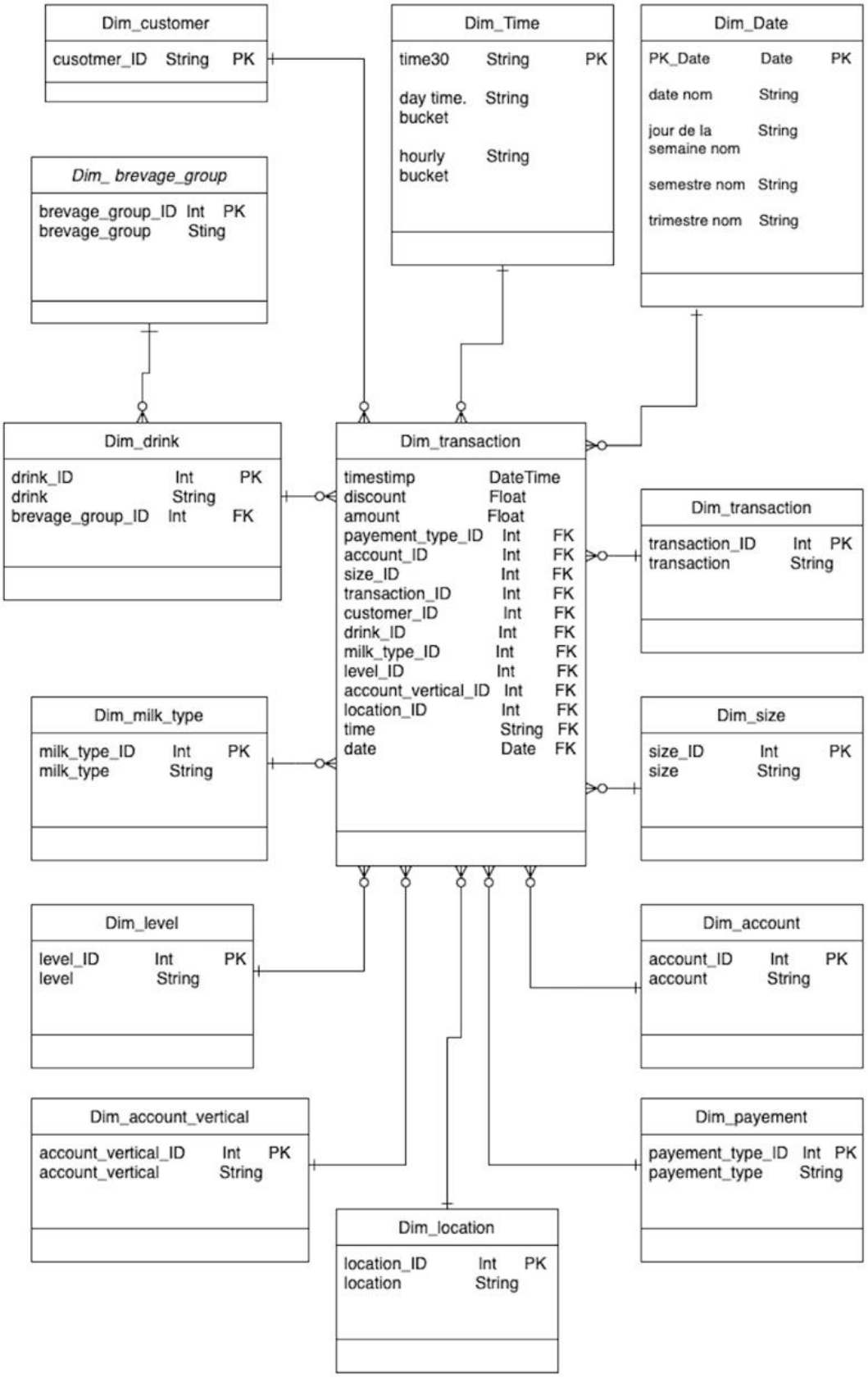
(Capture d'écran penthao)

Une fois la transformation finale terminée, tous les enregistrements sont maintenant disponibles dans la BDD SQLSERVER. Le logiciel Visual Studio sera alors nécessaire pour créer un cube au sein de notre Data Warehouse. Ce cube est créé à partir d'une vue qui comportera toutes les tables précédemment enregistrées. Pour avoir le détail lors de la recherche, les dimensions sont modifiées pour afficher son attribut en plus de son ID qui est maintenant un nombre entier.

Les outils de Microsoft sont très complets et potentiellement source d'erreur si mal utilisés. Le but de cette analyse préliminaire est de générer des jeux de données pouvant potentiellement distinguer des groupes de comportement.

La transformation permet de créer maintenant le cube suivant la modélisation ci-dessous. L'utilisation d'Excel et la génération d'un tableau dynamique seront utilisées pour générer les jeux de données voulues.

Figure 8 : Modélisation en étoile



(Créé avec draw.io)

2.4 Résultat de l'analyse préliminaire

Plusieurs tableaux ont pu être générés, possédant des données intéressantes. La génération de tableaux de façon dynamique est maintenant un grand avantage pour pouvoir filtrer et effectuer des calculs simples sur les différentes transactions. Les premiers tableaux générés sont ceux donnant une vision temporelle des ventes.

Tableau 1 : Transactions par heure

Étiquettes de lignes	Tbl Transaction Nombre
	581
Beverage Order	
00:00-00:59	2
01:00-01:59	9
02:00-02:59	3
03:00-03:59	1
04:00-04:59	3
05:00-05:59	5
06:00-06:59	76
07:00-07:59	204
08:00-08:59	419
09:00-09:59	601
10:00-10:59	714
11:00-11:59	281
12:00-12:59	335
13:00-13:59	619
14:00-14:59	332
15:00-15:59	446
16:00-16:59	399
17:00-17:59	206
18:00-18:59	162
19:00-19:59	110
20:00-20:59	38
21:00-21:59	21
22:00-22:59	28
23:00-23:59	10
Coffe Credit added (manual)	**1064**
Total général	**6669**

(Créé depuis l'Excel dynamique)

Le premier tableau (Tableau 1) regroupe les transactions du fichier source par tranche d'heure. Un premier filtre a été appliqué pour se concentrer sur les transactions *"Beverage order".* Les transactions sans catégorie et les transactions *"Coffee Credit added"* ne sont pas prises en compte car l'heure ne représente pas le comportement de l'utilisateur sur un achat volontaire. En effet, sur les transactions sans catégorie 99% sont des payements *"Free credit".* Étant donné que le client ne paye pas son café, la vision sera faussée et il sera plus compliqué de savoir s'il aurait vraiment payé son café à cet instant.

Le deuxième tableau se base sur le même filtre que le premier. Celui-ci montre le nombre de transactions selon les différents mois. Cette vision peut être très pratique pour avoir une visualisation rapide et globale de l'évolution des ventes dans le temps. De plus, il est maintenant possible de rajouter des dimensions rapidement ; il est donc envisageable d'ajouter la dimension *"Drink"* pour savoir précisément quelle boisson a été commandée avec le nombre de transactions correspondantes.

Tableau 2 : Transactions par mois

Étiquettes de lignes	Tbl Transaction Nombre
	581
Beverage Order	
juin.2020	1
juil.2020	140
août.2020	189
sept.2020	160
oct.2020	210
nov.2020	417
déc.2020	165
janv.2021	226
févr.2021	386
mars.2021	1013
avr.2021	722
mai.2021	759
juin.2021	535
juil.2021	101
Coffe Credit added (manual)	**1064**
Total général	**6669**

(Créé depuis l'Excel dynamique)

Les précédents tableaux sont des exemples simples qui ont pour seul objectif une visualisation facilitée du tableau Excel source. Une multitude de tableaux peuvent être maintenant visualisés et il sera plus facile de traiter les données par un algorithme de clustering. En effet, le fait de filtrer les dimensions ainsi que de fournir un fichier CSV facilitera le traitement des données par l'algorithme.

3 Algorithme de clustering

3.1 Qu'est-ce qu'un algorithme de clustering

Les algorithmes de clustering utilisent une technique d'apprentissage non supervisé. Il n'y a pas de jeux de tests utilisables au préalable pour que l'algorithme puisse apprendre et ainsi utiliser une technique de prédiction comme un arbre de probabilité. L'intérêt d'un tel algorithme est de réussir à regrouper des données ayant des propriétés similaires. L'algorithme KNN est un algorithme utilisant la technique des voisins proches pour effectuer sa prédiction. Son processus de regroupement est basé sur les distances euclidiennes calculées par l'algorithme. Son fonctionnement est proche du principe de clustering, cependant celui-ci est basé sur un apprentissage qui sera utilisé pour la prédiction. Il a été étudié lors du module 625-2 Data mining enseigné par Kalousis Alexandros en 2020. Ce type d'algorithme pourra être utilisé une fois que les abonnements seront mis en place et qu'un certain nombre de ceux-ci aura été vendu. Un abonnement pourra être attribué aux personnes ayant des attributs spécifiques en commun et en déduire qu'une personne ayant ces caractéristiques sera intéressée par le même abonnement.

Ces algorithmes peuvent avoir plusieurs utilités. Par exemple, l'analyse d'images d'animaux pour différencier les espèces afin de pouvoir catégoriser d'autres images d'animaux dont l'espèce est inconnue. Il y a donc plusieurs d'algorithmes utilisant différentes approches de regroupement des données. L'objectif sera de trouver un algorithme pouvant tirer des tendances selon le comportement des usagers de mia&noa.

3.2 Comment choisir le bon algorithme

Il est difficile de vérifier la pertinence des regroupements. Ceux-ci sont faits de manière mathématique et chaque approche de regroupement donnera un résultat différent sans que l'un soit meilleur que l'autre. L'objectif est de pouvoir proposer un abonnement qui pourrait correspondre à un maximum de clients. Plusieurs algorithmes seront testés pour essayer de distinguer de manière visuelle le comportement des usagés. Scikit-learn est une bibliothèque libre utilisée avec le langage Python. Étant destinée à l'apprentissage

automatique, elle possède plusieurs algorithmes de clustering. Ceux-ci seront utilisés et les résultats visualisés de façon graphique.

L'utilisation d'Excel durant l'étape préliminaire nous permet de générer facilement des fichiers .csv. Un jeu de tests sera alors utilisé et il sera identique pour chaque algorithme testé.

4 Procédure d'analyse

Premièrement, il a été envisagé d'utiliser le "machine learning" pour construire un modèle de prédiction afin de classer les utilisateurs selon leurs habitudes de consommation. Le principe aurait été de créer un jeu de données représentatif des consommations des utilisateurs afin d'entraîner un algorithme. Le modèle construit par celui-ci représente les différents clusters correspondant aux habitudes des clients. L'idée serait de créer un abonnement adapté aux différents groupes d'utilisateurs et d'analyser la performance du modèle.

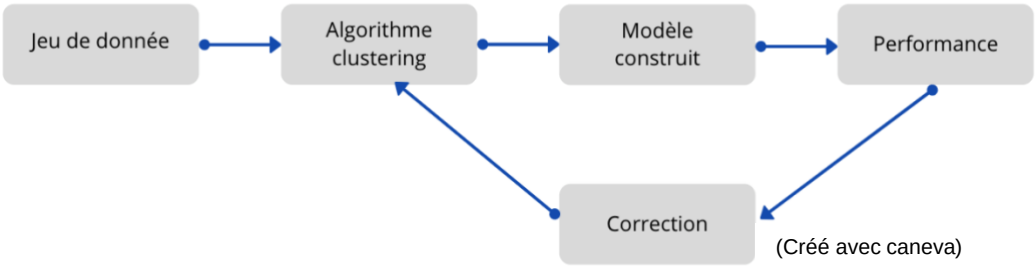
Pour quantifier la performance du modèle et obtenir des résultats pertinents, plusieurs groupes d'utilisateurs tests se verraient mis à disposition.

L'algorithme classerait les utilisateurs selon les abonnements créés au préalable. Le premier groupe aurait le choix entre tous les plans tarifaires disponibles. Il sera donc possible de calculer le pourcentage de réussite. Une distance euclidienne peut être calculée entre les différents clusters. Si l'utilisateur d'un cluster est mathématiquement éloigné d'un autre et qu'il choisit l'abonnement correspondant à ce cluster, l'erreur sera plus importante qu'un autre cluster plus proche. Ceci permettra de quantifier l'erreur du modèle.

Un deuxième groupe se verrait proposer l'abonnement correspondant à la prédiction du modèle et il sera possible de comparer le nombre d'utilisateurs ayant choisi le plan tarifaire à celui de ceux ayant refusé.

L'analyse des résultats permettra d'effectuer des modifications sur les hyperparamètres de l'algorithme. Celui-ci créera un nouveau modèle de prédiction qui sera à son tour testé selon le même processus.

Figure 9 : Étape pour le machine learning



(Créé avec caneva)

Une fois que les données satisfaisantes seront obtenues, chaque abonnement correspondra à une ou des caractéristiques de consommation. Ces caractéristiques seraient utilisées pour catégoriser chaque utilisateur existant dans le but de lui proposer le plan tarifaire le plus adapté.

Le machine learning est très compliqué et j'ai été rapidement confronté aux contraintes d'une telle étude. En effet, la phase de vérification et d'amélioration demande beaucoup de temps et d'investissement. Ce type de code n'ayant jamais été abordé en cours, une longue phase d'apprentissage personnel serait également nécessaire. De plus, la modification de l'application n'est pas comprise dans le temps dédié dans le cadre de cette recherche, contrairement à la proposition d'abonnement qui sera la finalité du projet. Pour pouvoir mener à bien la recherche, seule une partie du processus d'analyse sera effectuée. En effet, une étude spécifique au machine learning serait nécessaire pour pouvoir maîtriser un tel sujet.

4.1 Objectif

L'objectif est donc de proposer des plans tarifaires suivent trois axes de direction.

1. Utilisation des données temporelles pour pouvoir proposer une réduction à tous les utilisateurs durant les heures creuses.
2. Proposition d'un abonnement spécifique aux habitudes de consommation
3. Proposition d'un plan tarifaire pour la boisson ou le groupe de boisson préféré de l'utilisateur.

L'utilisation d'un algorithme de clustering sera seulement utilisée pour distinguer un groupe de comportement. Le modèle construit ne sera donc pas entrainé dans le cadre de cette recherche mais la construction de graphiques pourra être représentative de la situation actuelle.

5 Analyse

5.1 Analyse du premier jeu de donné avec un algorithme de clustering

Le premier jeu de données à être analysé sera simple. Il contiendra le nombre de transactions de type *"Beverage order"* et *"Coffee Credit added"* pour chaque utilisateur. Ce jeu de données a pour objectif de comparer les utilisateurs et leur fidélité. La fréquence avec laquelle un utilisateur crédite son compte comparée à la fréquence d'achat communiquera des informations sur la fidélité d'un client, partant du principe qu'un utilisateur qui recharge son compte avec une plus grosse somme aura tendance à consommer davantage.

5.2 Langage de programmation

Plusieurs langages peuvent être utilisé pour analyser des données. En data mining, le R propose de nombreuses fonctionnalités d'analyse ainsi que de nombreuses librairies, il est cependant difficile à maîtriser et son utilisation est principalement destinée aux calculs de statistiques. Python est également un langage puissant comptant des librairies libres d'utilisation. A la différence de R, python est utilisé par un plus grand nombre d'utilisateurs. En effet, python se classe 2ème dans le top 20 des langages de programmation les plus utilisés du trimestre 2021 selon RedMond. Il est beaucoup plus intuitif et la documentation autour des algorithmes de clustering est très bien fournie.

Pour analyser les données, un petit script python sera créé. Le script recevra en entrée le fichier .csv préalablement créé à l'étape préliminaire pour fournir en sortie un ou plusieurs graphes colorés représentatifs des clusters.

5.3 Génération des graphes

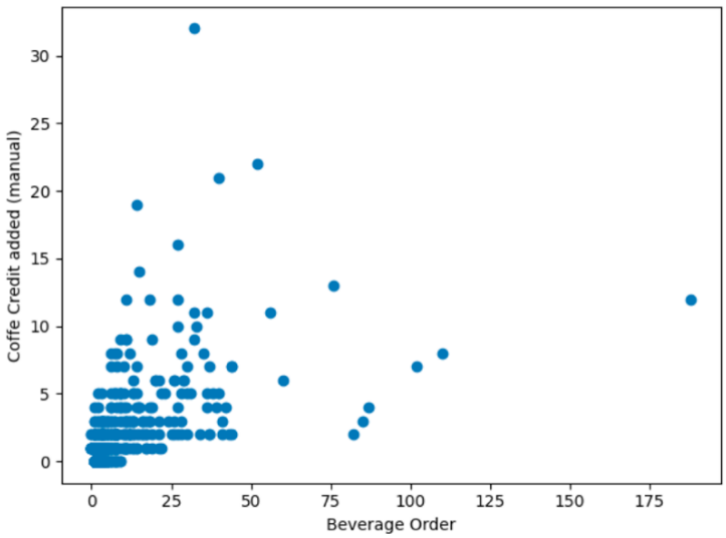
La première étape est de charger le fichier .csv en mémoire via la librairie open source pandas. Cette libraire fournit les fonctions nécessaires au traitement de fichier CSV pour en créer une structure de données.

```
import pandas as pd

def runAnalyse():
    # charge le fichier CSV dans la variable data
    data = pd.read_csv("path/to/file.csv", sep=';', usecols=[1, 2])
    # le fichier source peut contenir un valeur null quand
    aucune transaction n'a été effectué. La valeurs null
    seront remplacé par 0
    data = data.fillna(0)
    X = data.values
```

La variable X contient maintenant une liste de point qu'il est possible de représenter sur un graphe à deux dimensions. L'axe X représente de nombre total de commande d'un client et l'axe Y le nombre de fois qu'il à recharger son compte.

Figure 10 : Visualisation des données

Cette représentation nous montre facilement les extrêmes du fichier mais il est difficile de distinguer des groupes. Le fichier contient 874 points dont la majorité se trouve entre 0-5 crédit ajouté et 0-25 consommations. La dimension de temps n'est pas prise en compte dans ce graphe.



(Créé avec la libraire matplotlib)

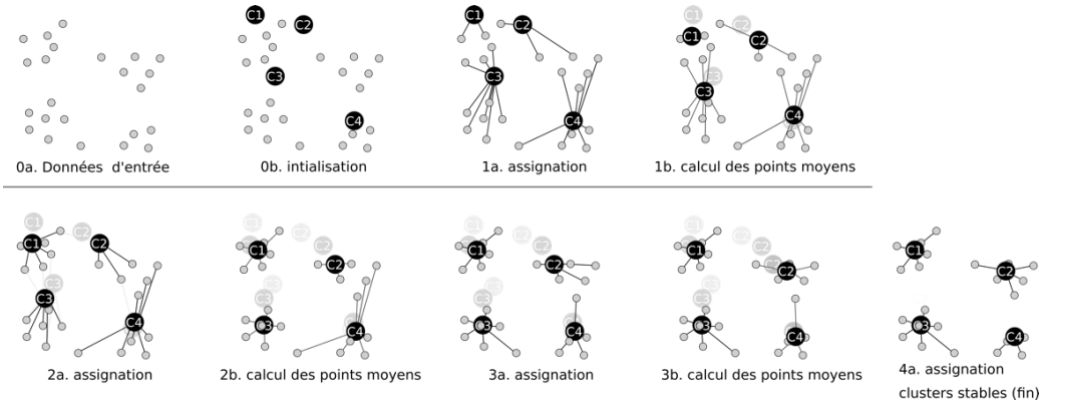
L'utilisation d'algorithmes de clustering peut colorer cette représentation. Ceux-ci peuvent faire ressortir des tendances qui sont invisibles au premier abord, comme la condensation de points. En effet, plusieurs points peuvent avoir la même place sur le graphe et seul un point ressortira. Les algorithmes disponibles dans la librairie sklearn fonctionnent selon différents principes. Pour utiliser et se familiariser avec ces algorithmes, le tutoriel créé par J. Brownlee été d'une grande utilité.

Le premier qui sera testé est l'algorithme k-mean. Il est très populaire et son principe est simple à comprendre. Le schéma est coloré en "k" groupes qui représentent les différents clusters

5.4 K-mean

Pour utiliser l'algorithme k-mean, il est indispensable de définir en premier lieu le nombre de clusters à créer. En effet, la première étape de cet algorithme est de placer un nombre prédéfini de centres qui seront choisis aléatoirement parmi tous les points du graphe. Ensuite, chaque point restant sera coloré de la couleur du centre le plus proche. Une fois cette étape réalisée, le centre de gravité de chaque cluster est calculé par l'algorithme et celui-ci sera utilisé comme nouveau centre. L'étape précédente consiste à colorer tous les points du graphe du centre le plus proche. Une fois que tous les points sont de nouveau colorés, le centre de gravité de chaque groupe est de nouveau calculé et l'attribution des couleurs répétée, jusqu'à ce que les différents centres de gravité ne changent plus.

Figure 11 : Fonctionnement de k-Mean



(Mquantin 2017)

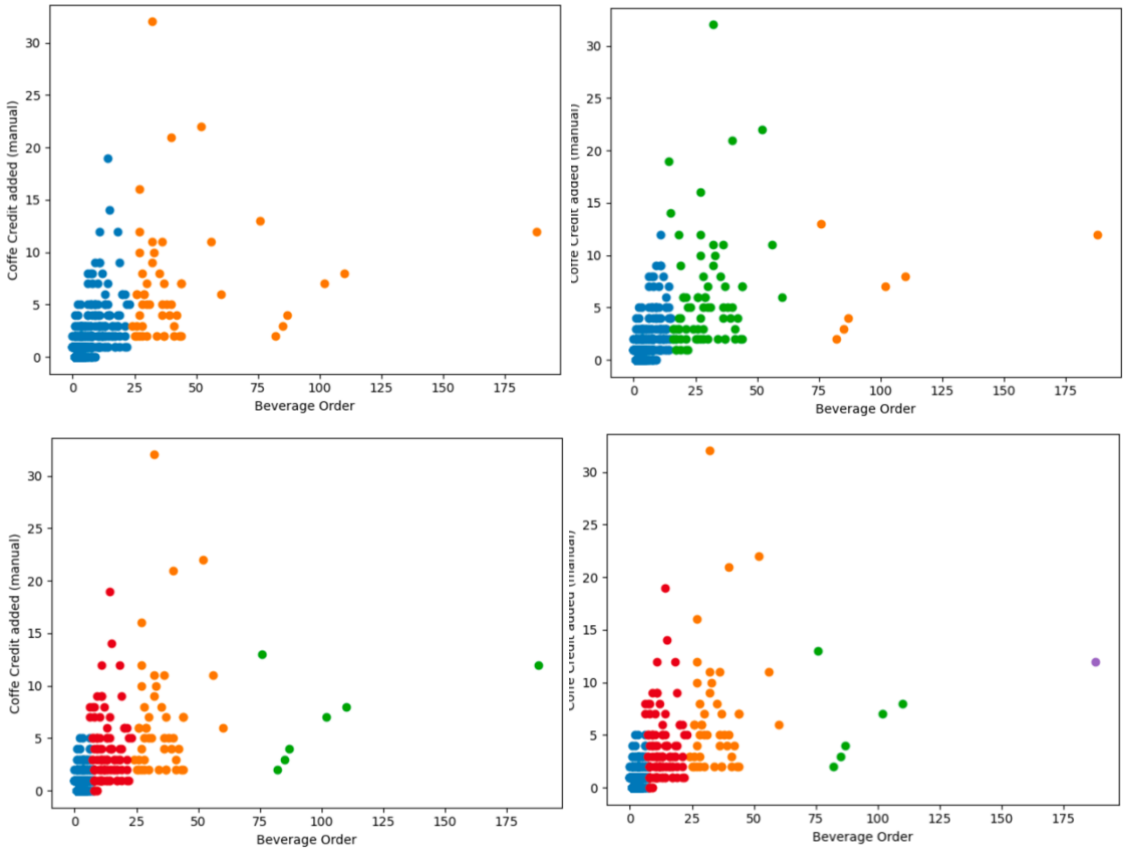
Le code correspondant à l'utilisation de l'algorithme k-mean :

```
# définition du nombre de cluster à assigner
model = KMeans(n_clusters=5)
# transmet les données à l'algorithme
model.fit(X)
# chaque point se voit attribuer une couleur
prediction = model.predict(X)
# selection d'un set des clusters
clusters = unique(prediction)
# création du scatter plot pour la représentation graphique
for cluster in clusters:
    # on séléctionne les index de tous les points correspondant au cluster
    row_ix = where(prediction == cluster)
    # création du scatter des points correspondant au cluster
    pyplot.scatter(X[row_ix, 0], X[row_ix, 1])
# Affichage du graphe avec les légendes des axes X et Y
pyplot.xlabel(data.columns[0])
pyplot.ylabel(data.columns[1])
pyplot.show()
```

Le problème avec cet algorithme est la définition du nombre de clusters à chercher. En effet, aucun groupe ne se distingue. Le code sera donc répété en changeant la variable n_clusters de 2 à 5 dans le but de trouver le schéma le plus pertinent. Il est également prévisible que cet algorithme ne soit pas pertinent en ce qui concerne la concentration des points. Tous les points ayant les mêmes coordonnées seront par définition de la même couleur et n'affecteront pas la surface de la zone du centre de gravité. Cette surface étant utilisée pour calculer le nouveau centre, la vision ne sera pas représentative de la concentration mais de la disposition des points sur le graphe.

Les quatre graphes générés ayant un nombre de clusters de 2 à 5 sont répartis de la manière suivante :

Figure 12 : Résultats K-mean



(Créé avec la libraire matplotlib)

Les résultats obtenus sont en corrélation avec ce qui été prévisible avec cet algorithme. En effet, nous pouvons imaginer le centre de gravité de chaque cluster et les points les plus proches font partie du même cluster. Les clients les plus fidèles se trouveront dans la partie inférieure droite du graphe. En effet, les clients ayant effectué beaucoup de commandes de boissons tout en ayant une faible quantité de transactions de type *"Coffee Credit added"* auront ajouté une somme importante permettant l'achat d'un nombre plus conséquent de boissons. De plus, mia&noa a actuellement pour politique de récompenser les utilisateurs procédant à un virement important sur leur compte en offrant des crédits gratuits. Par conséquent, plus l'utilisateur chargera son compte d'un montant important, plus il aura un nombre de crédits gratuits et donc se trouvera dans la partie inférieure droite du graphe. Le résultat de la coloration à quatre clusters montre bien que le groupe vert correspond à ce type d'utilisateur.

Cette coloration est utile pour la compréhension du fonctionnement de l'algorithme. La coloration en deux dimensions permet une visualisation graphique mais n'est pas efficace pour déceler les préférences ou d'analyser de façon précise les habitudes de consommation de l'utilisateur afin de lui proposer un plan tarifaire adapté.

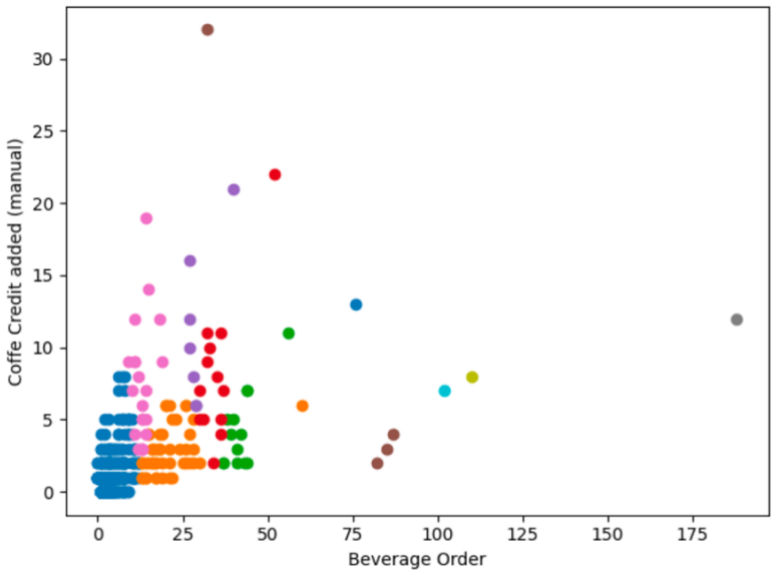
5.5 Mean shift clustering

Mean shift est un algorithme de nature hiérarchique. Pour organiser les données, il procède à la recherche d'un noyau appelé *"centroïde"* dans le but de classer les individus aux alentours de ce centre en calculant la moyenne des points. Le principe de mean shift consiste à créer une fenêtre glissante pour trouver une ou plusieurs zones denses de points autour des centroïdes. Selon la documentation officielle sklearn, les individus sont filtrés lors d'une étape de post-traitement pour former l'ensemble final des centroïdes.

Le grand avantage d'utiliser un algorithme comme Mean shift est qu'il n'y a pas besoin de déterminer au préalable un nombre K de clusters contrairement à l'algorithme k-mean. La complexité de cet algorithme est cependant bien plus élevée O(n(carré)), le rendant moins performant pour les opérations de data mining.

```
# définition de meanshift
model = MeanShift()
# transmet les données à l'algorithme
prediciton = model.fit_predict(X)
# selection d'un set des clusters
clusters = unique(prediciton)
# création du scatter plot pour la représentation graphique
for cluster in clusters:
    # on séléctionne les index de tous les points correspondant au cluster
    row_ix = where(prediciton == cluster)
    # création du scatter des points correspondant au cluster
    pyplot.scatter(X[row_ix, 0], X[row_ix, 1])
# Affichage du graphe avec les légendes des axes X et Y
pyplot.xlabel(data.columns[0])
pyplot.ylabel(data.columns[1])
pyplot.show()
```

Figure 13 : Visualisation mean shift



(Créé avec la libraire matplotlib)

DBSCAN

DBSCAN est également un algorithme très connu. Contrairement à l'algorithme k-mean, il utilise l'estimation de la densité locale. Chaque groupe créé sera représentatif du cluster. Son utilisation peut être très variée, comme pour identifier des objets dans une image. L'algorithme a besoin de recevoir en paramètre le nombre minimum d'individus pour créer un cluster min_sample et eps (epsilon), qui corresponde à la distance nécessaire entre deux de ces points.

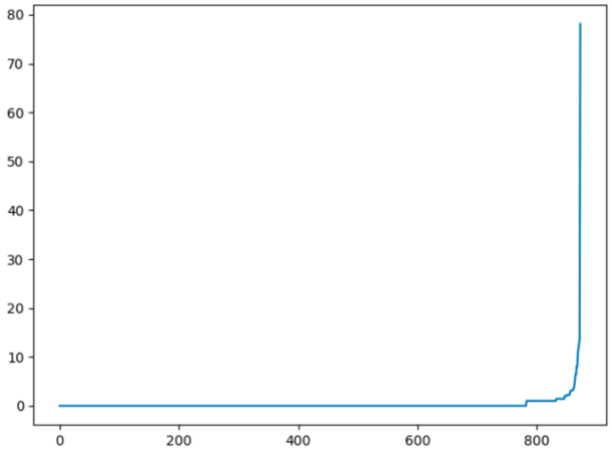
Cet algorithme possède deux avantages. Les deux paramètres cités ci-dessus sont les seuls que l'algorithme utilisera pour procéder au regroupement des individus. Il n'a pas besoin d'avoir au préalable un nombre de clusters passé en paramètre car l'algorithme procédera à la sélection du nombre de clusteurs optimale en se basant sur la concentration des points. Ceci le rend moins rigide. De plus, les valeurs extrêmes ou anormales peuvent être gérées par l'algorithme.

Avant d'appliquer l'algorithme sur le jeu de données, les paramètres doivent être définis pour obtenir le meilleur partitionnement possible. Le tutoriel d'Aurelia Fellous nous explique que la librairie Scikit-learn permet de déterminer les voisins les plus proches de chaque observation ainsi que les distances grâce à une classe NearestNeighbors. L'objectif sera de passer en paramètre une valeur eps contenant 90% des individus ayant une distance inférieure à la valeur choisie.

```
#Utilisation de la classe en indiquant le nombre de voisin pour calculer la
distance
neigh = NearestNeighbors(n_neighbors=2)
# transmet les données à la classe NearestNeighbors
nbrs = neigh.fit(X)
# calcule des distances
distances, indices = nbrs.kneighbors(X)
# triage des distances
distances = np.sort(distances, axis=0)
# selection de la liste  pour la représentrer graphiquement
distances = distances[:, 1]
# affectation des distances et affichage du graphe
pyplot.plot(distances);
pyplot.show()
```

Le résultat obtenu nous indique qu'il y a 783 points sur 874 ayant une distance inférieure à 1. Cela représente 89.5%. La valeur eps qui a été retenue est égale à 0.1. Une fois la valeur optimale déterminée, il est maintenant possible de colorer notre graphe en utilisant DBSCAN.

Figure 14 : Visualisation des distances



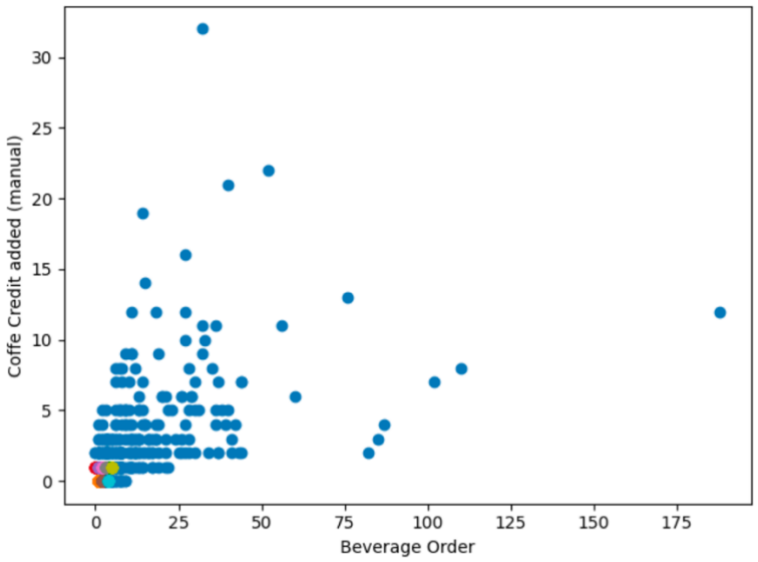
(Créé avec la libraire matplotlib)

Le code correspondant à l'implémentation de DBSCAN

```
# définition de DBSCAN avec ses paramètres
model = DBSCAN(eps=0.1, min_samples=10)
# transmet les données à l'algorithme
prediction = model.fit_predict(X)
# selection d'un set des clusters
clusters = np.unique(prediction)
# création du scatter plot pour la représentation graphique
for cluster in clusters:
    # on séléctionne les index de tous les points correspondant au cluster
    row_ix = np.where(prediction == cluster)
    # création du scatter des points correspondant au cluster
    pyplot.scatter(X[row_ix, 0], X[row_ix, 1])
# Affichage du graphe avec les légendes des axes X et Y
pyplot.xlabel(data.columns[0])
pyplot.ylabel(data.columns[1])
pyplot.show()
```

La coloration obtenue est différente de celle de k-mean.

Figure 15 : Visualisation DBSCAN



(Créé avec la libraire matplotlib)

DBSCAN montre une forte concentration des instances dans la zone 0 – 5 boissons consommées. De plus, plusieurs clusters se sont dessinés malgré le manque de clarté. Les données observées représentent les consommations de plus d'une année. Cette donnée montre donc qu'il y a un grand nombre d'utilisateurs qui ne sont pas très actifs ou sont récents et n'ont pas encore utilisé leur compte. Les points ne faisant pas partie d'un cluster sont donc potentiellement les clients les plus anciens ou les plus fidèles. En effet, une distance plus élevée du point (0 ;0) montre une activité plus importante. Malheureusement, DBSCAN ne distingue aucun groupe parmi les individus ayant une grande activité mais cela nous permet d'affiner la compréhension des données.

L'Excel dynamique nous permet de confirmer cette hypothèse. En effet, en visualisant les données sur la dimension *customer_ID* nous pouvons faire l'observation suivante :

- 856 utilisateurs ont effectué une ou plusieurs commandes de boisson de type *"Brevage order".* 439 ont effectués une seule commande soit 51.3% des utilisateurs. En ajoutant la dimension *Payement type* nous obtenons le tableau suivent.

Tableau 3 : Nombre d'utilisateur avec une seule commande

	Coffee Credit	Free Credit	Total général
		159	159
Beverage Order		6	433	439

Ce tableau est pertinent en deux points. Premièrement 433 sur 439 soit 98.6% des individus ayant effectué une seule transaction de type *"Beverage order"* ont fait une transaction gratuite (*"Free Credit"*). Cela signifie que seuls 49.4% des utilisateurs ont effectué plus d'une transaction ou alors une transaction payante.

Concernant l'axe Y *"Coffee Credit added"* du graph, 361 utilisateurs ont effectué plus d'une transaction d'ajout de crédit. 162 de ces utilisateurs, soit 44% ont effectué une seule transaction.

DBSCAN permet donc une première constatation pouvant être utilisée comme proposition. Le grand nombre d'utilisateurs pouvant potentiellement être actifs ne doit pas être négligé. Le but étant de les fidéliser, la proposition doit être attrayante pour le client et profitable pour mia&noa.

Pour récapituler, les trois algorithmes utilisés ont une approche différente pour former les clusters. De plus, le jeu de données utilisé contenait seulement deux dimensions, *"Beverage order"* et *"Coffee Credit added".* Ce jeu de données a été utilisé pour pouvoir se familiariser avec les algorithmes ainsi que pour visualiser de façon globale la fidélité des utilisateurs. En effet, pour être capable d'identifier des groupes de comportement, un utilisateur qui a seulement utilisé son crédit de café gratuit perturbera l'algorithme. Plus de la moitié des clients dans le fichier source ont donc actuellement effectué une seule transaction. Pour identifier des comportements de consommation, seuls les utilisateurs ayant effectué plus de trois transactions seront comptés. De plus, la fréquence à laquelle un utilisateur réapprovisionne son compte n'est pas forcément très pertinente par rapport à ses habitudes de consommation. Le prochain jeu de données généré aura plus de deux dimensions et le nombre de transactions de type *"Coffee Credit added"* ne sera pas pris en compte. Le jeu de données comportera toutes les catégories de boissons comme dimension. Seules les transactions classées *"Beverage order"* sont prises en compte mais il n'y aura aucune différence entre les transactions offertes (*"Free Credit"*) et celles payantes (*"Coffee Credit"*) car l'objectif est de catégoriser les préférences des consommateurs et non la façon dont ils effectuent leurs payements.

5.6 Analyse du deuxième jeu de donné avec un algorithme de clustering

Pour générer le nouveau fichier à analyser, l'étape préliminaire est de nouveau indispensable. Cette fois-ci, les valeurs de la colonne « *Drink* » seront utilisées comme dimension et chaque utilisateur aura en détail le nombre de transactions pour chaque boisson. Pour générer ce fichier il faut à nouveau appliquer un filtre afin d'obtenir seulement les transactions de type *"Beverage order".* Une fois ce filtre appliqué, chaque ligne correspond à l'ID d'un client et les colonnes auront comme valeur toutes les boissons disponibles. Il est important que l'utilisateur ait effectué plus de trois transactions pour pouvoir identifier une préférence de consommation de celui-ci. Un autre filtre sera donc appliqué sur le jeu de tests afin d'écarter les utilisateurs ayant uniquement utilisé leur crédit gratuit. Le fichier final contient 306 individus et ce sont ceux-ci qui seront analysés.

Ce jeu sera testé par deux des trois algorithmes précédemment utilisés. Il n'y a pas d'indication préliminaire au nombre de clusters à chercher. Seuls les algorithmes n'ayant pas besoin de ce paramètre sont intéressants. La représentation visuelle en plus de trois dimensions est difficile. Étant donné que l'algorithme procède au regroupement des utilisateurs, il étiquette chaque cluster avec un numéro. Les individus composant le cluster auront donc une étiquette correspondant au cluster auquel ils appartiennent.

Pour pouvoir visualiser les résultats ainsi qu'identifier les différents groupes, le code est similaire à celui utilisé pour l'analyse à deux dimensions. Cependant, nous n'avons plus l'utilité de la librairie "matplotlib" qui était destinée à la génération des graphes. Une classe Data a été ajoutée pour copier chaque cluster dans un fichier CSV. Son constructeur contient le chemin d'accès au fichier csv utilisé pour l'analyse. La classe contient une fonction *getInfoCSV(ids)* qui permet la génération du fichier csv contenant les informations d'une liste de clients passée en paramètre. Celle-ci sera utilisée pour analyser un cluster.

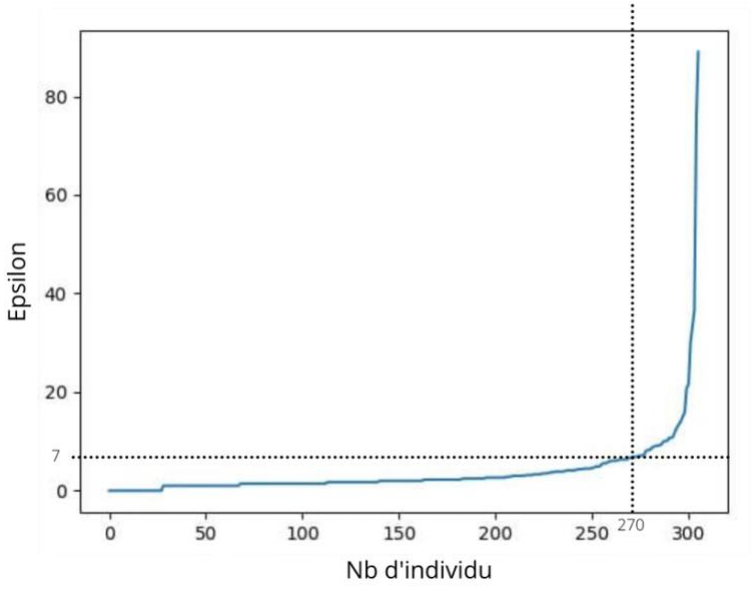
Code correspondant à la classe Data :

```
class Data:

    def __init__(self, path):
        self.data = pd.read_csv(path, sep=';')
        self.data = self.data.fillna(0) # remplace les valeurs null par 0

    def getInfoCSV(self, ids, pathCSVwrite, colonne):
       ecrire = open(pathCSVwrite,'w',newline='')  #Ouverture du fichier
       ecrire.write(";".join(colonne) + "\n")      CSV en écriture
       for i in ids:
         tmp = self.data.iloc[self.data.index[self.data['id'] == i]].values
            tmp = list(map(int,tmp[0]))
            print(tmp)
            ecrire.write(";".join(list(map(str,tmp))) + "\n")
```

5.7 Résultat DBCAN

Pour observer la concentration d'individus avec l'algorithme de DBSCAN, la première étape est la même qu'auparavant, les hyperparamètres doivent être définis. Le nombre d'individus minimum pour créer un groupe a été défini à deux. En effet, le jeu de données à analyser contient 306 individus ayant effectué plus de trois transactions. Un groupe d'individus trop important risque de limiter le nombre de clusters qui pourront être détectés par l'algorithme. Le paramètre epsilon est défini de la même manière que pour le premier jeu de données.

Figure 16 : Visualisation des distances par boisson



(Créé avec la libraire matplotlib)

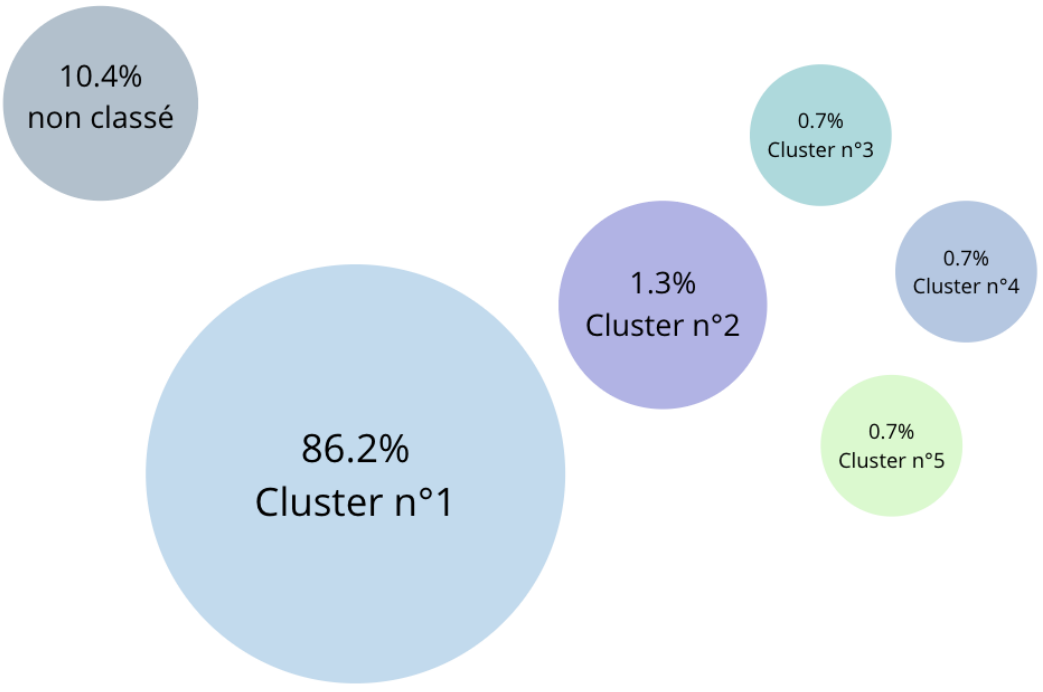
En fixant epsilon à 7, 90% de la population est pris en compte dans l'analyse ce qui correspond à la valeur minimum pour que l'algorithme puisse opérer de façon optimale.

Un fois les hyperparamètres définis, chaque cluster utilisera la classe précédemment créée pour persister les clusters en fichier CSV.

```
# définition de DBSCAN avec ses paramètres
model = DBSCAN(eps=7, min_samples=2)
# transmet les données à l'algorithme
prediction = model.fit_predict(X)
# selection d'un set des clusters
clusters = unique(prediction)

cpt = 0
# création du scatter plot pour la représentation graphique
for cluster in clusters:
    # on séléctionne les index de tous les points correspondant au cluster
    row_ix = where(prediction == cluster)
    # création du scatter des points correspondant au cluster
    print(data.values[row_ix, 0])
    print("NEXT \n ")
    Data('allDrink.csv').getInfoCSV(data.values[row_ix, 0][0], 'resultDB'+
    str(cpt) +'.csv')
    cpt = cpt+1
```

5.7.1 Cluster

Figure 17 : Visualisation des clusters DBSCAN



(Créé avec caneva)

Cinq clusters ont été détectés par l'algorithme sans compter les 10.4% d'individus qui ont été écartés. L'algorithme a placé 86.2% des individus dans un seul cluster. Cela représente 264 personnes. Malheureusement, avec un tel cluster il est difficile de tirer des conclusions pertinentes car celui-ci regroupe la majorité des consommateurs. La moyenne de transactions par utilisateur est de 9.3. L'utilisateur ayant effectué le plus de transactions est celui possédant l'id 2008 avec 41 transactions.

Les quatre clusters restants sont quant à eux beaucoup plus petits et lisibles. Le ″cluster n°2″ contient quatre individus qui ont effectué entre 26 et 44 transactions. Les quatre ont principalement commandé des **Cappuccinos** ou **Choco-Shot Macchiatos.**

Le cluster ″cluster n°3″ contient deux individus qui ont commandé principalement ou exclusivement des **Espressos Macchiatos.** Le premier en a commandé 16 sur 22 transactions effectuées et le deuxième la totalité des 15 transactions.

Le ″cluster n°4″ contient deux individus qui ont principalement commandé des **Flat Whites** et le ″cluster n°5″ est aussi composé de deux individus mais ayant principalement commandé des **Cafés** et **Cappuccinos.** La moyenne de transaction entre les quatre individus composant les deux clusters est de 39 transactions.

Ces résultats montrent qu'une grande majorité des utilisateurs ayant effectué plus de trois commandes aiment consommer différentes boissons. Seuls des groupes composés d'une infime partie de la population ont été détectés par l'algorithme. Ceux-ci ont cependant une forte préférence pour une boisson et cette information peut être utilisée dans le cadre d'un plan tarifaire adapté.

5.8 Résultat MeanShift

La complexité de MeanShift est un désavantage lorsque la quantité de données est importante. La première analyse montre qu'il est envisageable de l'utiliser en l'état ; en effet, le jeu de données à analyser est limité dans les données. Celles-ci vont cependant évoluer dans le temps et le nombre de transactions peut augmenter très rapidement selon le nombre de nouveaux utilisateurs qui s'inscriront dans l'avenir. L'utilisation d'un Data Warehouse est donc essentielle pour pouvoir filtrer les données et les analyser avec les dimensions souhaitées.

Le code pour implémenter MeanShit est donc très similaire à celui de DBSCAN

```
# définition de MeanShift
model = MeanShift()
# transmet les données à l'algorithme
prediction = model.fit_predict(X)
# selection d'un set des clusters
clusters = unique(prediction)

cpt = 0
# création du scatter plot pour la représentation graphique
for cluster in clusters:
    # on séléctionne les index de tous les points correspondant au cluster
    row_ix = where(prediction == cluster)
    # création du scatter des points correspondant au cluster
    print(data.values[row_ix, 0])
    print("NEXT \n ")
    Data('allDrink.csv').getInfoCSV(data.values[row_ix, 0][0],
    'resultMeanShift'+ str(cpt) +'.csv')
    cpt = cpt+1
```

Les clusters définis par MeanShift sont plus précis. En effet, cet algorithme analyse la densité des points et continuera de créer des clusters jusqu'à ce qu'il converge et trouve la solution optimale. Par principe, plus le nombre de données est important, plus les résultats seront pertinents. Cependant, le problème de complexité est à prendre en compte car plus les données seront importantes, plus le temps de calcul sera long.

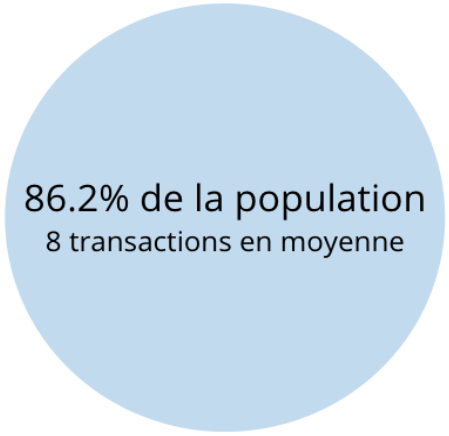
Étant donné que MeanShif crée ses clusters sans paramètre au préalable, un utilisateur seul peut faire partie d'un cluster. Une fois les données analysées 21 clusters numérotés de 0 à 20 ont été détectés par l'algorithme.

5.8.1 Cluster

- 14 clusters sont composés d'un seul individu soit 0.33% de la population.
- 1 cluster est composé de 2 individus soit 0.7% de la population.
- 3 clusters sont composés de 3 individus soit 1% de la population.
- 1 cluster est composé de 8 individus soit 2.6% de la population.
- 1 cluster est composé de 23 individus soit 7.5% de la population
- 1 cluster est composé de 250 individus soit 81.7% de la population.

Cluster n°0

Figure 18 : Cluster n°0 de DBSCAN



(Créé avec caneva)

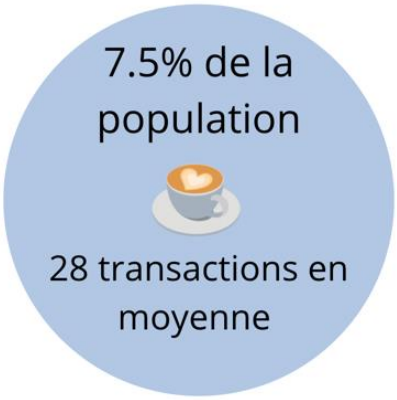
Le premier cluster ressemble au cluster principal de BDSCAN. En effet, ce cluster est composé de la majorité des individus de l'analyse. Il est difficile d'en tirer des tendances car celui-ci ne donne pas plus d'informations que le fichier source qui est analysé.

Malgré la difficulté de tirer des tendances, on peut constater que la moyenne de transactions de ce cluster est de seulement huit transactions, ce qui est un chiffre particulièrement bas. On peut en conclure que l'algorithme a regroupé les individus ayant effectué le moins de transactions.

Cluster n°1

Le deuxième cluster est quant à lui beaucoup plus petit car il est composé de 23 individus ayant tous une forte préférence pour le **Cappuccino.** Les individus de ce cluster ont effectué entre 13 et 44 transactions. Le nombre total de transactions de ce cluster est de 658 dont 464 était pour des Cappuccinos. Cela représente plus de **70%** des transactions du cluster.

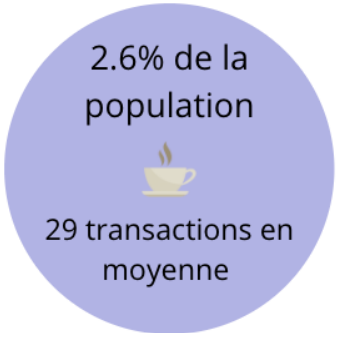
Figure 19 : Cluster n°1 DBSCAN



(Créé avec caneva)

Cluster n°3

Figure 20 : Cluster n°3 DBSCAN



(Créé avec caneva)

Le troisième cluster est plus petit que le deuxième car il est composé de seulement huit individus. Cependant la préférence est d'autant plus forte. En effet, ce groupe d'individus a une préférence très marquée pour le **café**. 212 transactions ont été effectuées dans ce cluster dont 180 étaient une commande de café. Cela représente **85%** des transactions.

Le reste des clusters sont composés d'une à trois personnes. Parmi les clusters composés de trois personnes, le cluster n°2 montre une très forte préférence pour la boisson **Flat White** avec **85%** des transactions effectuées dans le cluster (97 transactions sur 115). Le cluster n°10 présente une similarité avec le cluster n°1. Les individus le composant ont une très forte préférence pour le **Cappuccino.** La moyenne de transactions est de 48, ce qui est plus élevé que pour le cluster n°1. 118 transactions sur les 144 étaient destinées à un cappuccino ce qui représente plus de **80%**.

Un seul cluster est composé de deux personnes qui ont une forte préférence pour le **Choco-Shot Macchiato** (62 transactions sur 72).

Les 14 clusters restants sont donc composés d'une seule personne. Ceux-ci sont disponibles en annexe à la fin du document. Chaque individu ayant été isolé par l'algorithme possède une identité particulière. Ces utilisateurs ont également effectué plus de transactions que la moyenne. L'individu du cluster n°17 a commandé 188 transactions à lui seul ! Il est probable que cet utilisateur soit l'un des représentants travaillant pour mia&noa.

6 Proposition d'abonnement

Le principe d'abonnement sous forme de plan tarifaire a pour but principale d'éviter à l'utilisateur la contrainte de payement inattendu. Une discussion avec plusieurs utilisateurs m'a permis de constater que le montant disponible n'est que rarement consulté. Les utilisateurs créditent leur compte seulement lorsque qu'ils sont à court de crédit au moment de commander une boisson. Les bar à café sont placé dans des écoles et gares actuellement ce qui implique qu'une personne être potentiellement pressé. L'optimisation des commandes ainsi que l'assurance de n'être jamais à court de crédit est un point qui peut plaire aux utilisateurs.

Un abonnement illimité n'est pas envisageable en l'état, étant donné que les utilisateurs utilisent massivement le crédit gratuit, entre autres grâce à la possibilité de se créer plusieurs comptes avec plusieurs adresses électroniques. La possibilité de commander des cafés illimités implique la possibilité pour l'utilisateur de distribuer des cafés à son entourage. Chaque abonnement devra donc correspondre aux habitudes de consommation et il sera encadré par des conditions d'utilisation.

6.1 Résultat de l'analyse

6.1.1 Général

La boisson la plus consommée parmi les transactions *"Beverage order"* est le Cappuccino avec 1471 transactions. Cela représente 29.3% de toutes les transactions *"Beverage order"* confondues. Le café vient en deuxième position avec 542 transactions. Il est également remarquable que la boisson Choco-Shot Macchiato est un grand succès avec 458 transactions.

La moyenne de transactions parmi les individus ayant effectué plus de trois transactions est de **14.3** et **154** utilisateurs ont déjà effectué cinq ou plus transactions en un mois. Cela représente 50% des individus ayant effectué plus de trois transactions.

6.1.2 Clustering

L'analyse des données montre que plusieurs clusters ayant différentes préférences ont été détectés. Le premier cluster de meanshif a une moyenne de transactions inférieure à la moyenne totale et représente plus de 86% de la population. Cela est intéressent car ce sont des individus qui ont tout de même effectué 8 transactions en moyenne, ce qui signifie qu'ils ont apprécié le service du bar à café mais ne l'utilisent pas de façon régulière. Une offre spécifique aux individus possédant les mêmes caractéristiques de ce cluster peut être utilisée pour créer un plan tarifaire. L'offre qui sera adaptée à ce groupe d'utilisateurs doit être orientée pour fidéliser l'utilisateur. Mia&noa offre une nouvelle façon de consommer et participe activement à la limitation des déchets et à la consommation responsable. Cette vision doit être comprise par l'utilisateur qui distinguera la corrélation entre l'utilisation du bar à café et la participation aux actions écoresponsables.

Les autres clusters regroupés par les algorithmes de meanshift ou DBSCAN sont quant à eux très clairs sur les préférences des utilisateurs. Le nombre d'utilisateurs les composant est cependant limité. Les plans tarifaires peuvent être adaptés aux préférences du client et mia&noa possède une information clé pour satisfaire son client. Les moyennes de transactions de ces clusters sont supérieures à la moyenne générale. Étant donné que les utilisateurs ont déjà utilisé l'application, ils sont également informés de la vision de l'entreprise. Il faut donc les convaincre qu'un plan tarifaire est avantageux pour eux en les récompensant dans le but de les fidéliser. Un utilisateur fidèle est

également un très bon moyen pour promouvoir l'entreprise. Celui-ci peut potentiellement parler du bar à café et de ses bénéfices à son entourage.

6.2 Analyse des plans tarifaires

Mia&noa se veut innovateur sur de nombreux points. En effet, l'absence de distribution de gobelets à café jetables et l'utilisation d'une application pour commander son café sont des éléments innovateurs. Des individus mal informés peuvent être dans la confusion, il est donc indispensable de communiquer et d'informer l'utilisateur. En effet, ce modèle de consommation est en parfaite adéquation avec les intentions de réduction des déchets. L'entreprise est innovatrice dans ses processus, un plan tarifaire innovant sera par conséquent en adéquation avec les fondamentaux de l'entreprise.

Un plan tarifaire doit être avantageux pour les deux parties. L'utilisateur ne doit pas se sentir mis sous pression et obligé de consommer. L'idée serait que l'utilisateur paye un abonnement qui lui permettrait de consommer ses boissons préférées sans devoir se soucier du crédit sur son compte.

Actuellement, l'individu est invité à créditer son compte pour pouvoir générer le QR code du café souhaité. Plus l'utilisateur chargera son compte, plus il recevra de crédit gratuit (avec un maximum de 10 CHF). Le problème avec cette méthode est que l'utilisateur risque d'avoir en permanence quelques francs restants dans son compte.

6.2.1 Proposition n°1

Mia&noa offre à l'utilisateur un crédit gratuit l'incitant à essayer le bar à café. Cette offre est une réussite vu le grand nombre d'utilisateurs ayant utilisé ce crédit gratuit, 433 sur 856 ayant effectué une transaction. Étant donné que ces utilisateurs ont effectué une seule transaction, ceux-ci ne sont pas encore fidèles mais intéressés aux offres. De plus, ils sont désormais inscrits et possèdent l'application sur leur téléphone.

Ma première proposition serait d'envoyer une notification à tous les utilisateurs ayant plus de 3-4 semaines d'ancienneté et n'ayant jamais crédité leurs comptes. Cette notification serait une offre limitée dans le temps offrant une boisson du même groupe que la boisson déjà consommée, à condition que l'utilisateur crédite son compte ou qu'il procède à l'achat d'une boisson. Les utilisateurs ayant créé plusieurs comptes pour recevoir plusieurs fois le crédit gratuit ne recevront l'offre qu'une seule fois. A cette fin, seuls les comptes connectés recevront la notification et celle-ci ne sera envoyée qu'une seule fois. L'idée est que cette notification soit envoyée une fois que les nouvelles

stratégies seront mises en production. En effet, la notification peut motiver l'utilisateur à ouvrir l'application de mia&noa. Une fois l'application lancée, les nouveaux plans tarifaires seront possibles et l'offre temporaire pourra potentiellement convaincre le client d'essayer un de ces plans.

Ce groupe d'utilisateurs pourra ainsi être suivi. Des statistiques d'utilisation pourront être faites une fois la notification envoyée. Il sera possible de distinguer les comptes ayant profité de l'offre de ceux qui seront restés inactifs. Pour cela, la data wharehouse est l'outil indispensable qui permet d'analyser les données en continu. La suite de cette recherche sera consacrée à l'analyse de données en continu.

6.2.2 Proposition n°2

Actuellement, il est possible de choisir sa boisson préférée et de pouvoir générer son QR code de façon simplifiée. Cela implique cependant d'avoir du crédit disponible. Avant la proposition d'un nouveau plan tarifaire, le processus de génération d'un QR code pourrait être légèrement simplifié. En effet, l'achat unique pourrait être possible directement depuis la sélection de la boisson en procédant au débit automatique du moyen de payement enregistré. Le but est d'offrir la possibilité aux utilisateurs occasionnels de profiter d'une boisson sans avoir à se soucier du montant qui restera sur leur compte ou de devoir créditer ce dernier. Le lait est une matière première composant une partie des boissons. Etant donné que le gaspillage est proscrit dans la politique de mia&noa, il est possible d'envisager de réduire le prix des boissons composées de lait lorsque que la date limite de consommation est atteinte. Pour que cela soit efficace, la génération simplifiée d'un QR code pourrait également être fait sans devoir s'inscrire préalablement.

6.2.3 Proposition n°3

Les trois boissons ayant composé des clusters sont Cappuccino, Choco-Shot Macchiato, café. Un cluster supplémentaire a fait sortir qu'un petit groupe avait une forte préférence pour le Flat White. Les individus composant ces clusters ont tous une forte préférence pour une boisson particulière. Pour inciter l'utilisateur à souscrire à un plan tarifaire et ainsi faciliter les prévisions de consommation pour mia&noa, le client pourrait se voir offrir sa boisson préférée ou créditer son compte s'il souscrit à un nouveau plan tarifaire. Cette information peut également être utilisé pour indiquer à l'utilisateur qu'avec un plan tarifaire, sa boisson préférée serait à un prix préférentiel.

Figure 21 : Exemple d'une fenêtre pop-up d'offre promotionnelle

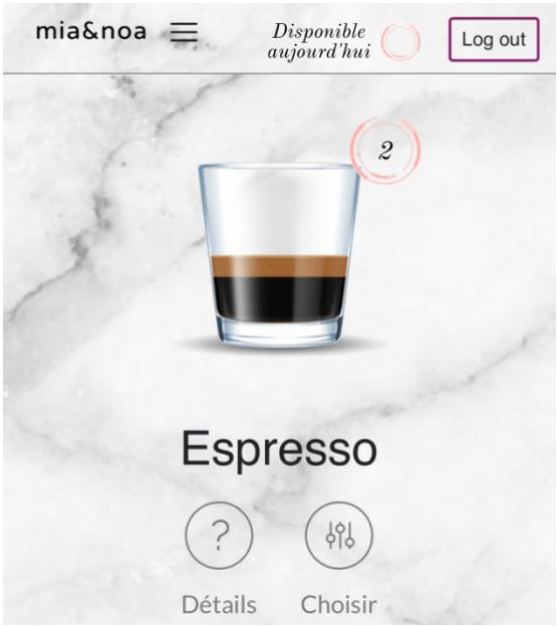
(Créé avec caneva)

6.2.4 Proposition n°4

6.2.4.1 Un abonnement sans accumulation

L'utilisateur régulier peut être intéressé à payer ses consommations de façon mensuelle. Un abonnement sans accumulation donnerait à l'utilisateur un certain crédit par semaine. A la place d'afficher le crédit sous forme monétaire, celui-ci pourrait être visible directement sur la page de sélection du café. Si ce crédit n'est pas utilisé dans la semaine, il ne sera pas disponible la semaine suivante. L'utilisateur devra choisir au préalable les boissons souhaitées mais le prix de l'abonnement correspondra au type de boisson choisi. L'analyse nous montre qu'un nombre considérable d'utilisateurs a consommé 5 ou plus boissons en un mois. La moyenne est de 9 commandes de boissons par mois parmi cette population. Un abonnement offrant trois boissons par semaine engloberait la grande majorité de ces utilisateurs mais le choix du nombre appartiendrait à l'utilisateur.

Figure 22 : Screen "2 Espresso disponibles"



(Créé avec caneva)

6.2.4.2 Un abonnement avec accumulation

L'abonnement avec accumulation aurait comme particularité d'accumuler les boissons non consommées d'une semaine à une autre. Celui-ci donne à l'utilisateur la garantie de ne pas perdre un crédit non consommé mais l'abonnement sera moins avantageux pour le client.

Pour qu'un tel plan tarifaire soit efficace, il est indispensable de suivre attentivement l'évolution de l'utilisation de l'abonnement. Un utilisateur ayant un abonnement sans accumulation pourrait payer un crédit de cinq cafés par semaine. Le prix de l'abonnement devrait être minutieusement calculé. En effet, le prix de la boisson variant selon sa composition, le prix de l'abonnement devra donc correspondre aux boissons choisies par l'utilisateur.

6.2.4 Proposition n°5

6.2.4.1 La mise en place d'un système de points qui récompense l'utilisateur pour chaque dépense effectuée.

Pour respecter le fait que le prix d'une boisson à une autre varie, offrir une boisson de façon globale n'est pas logique. Une façon de résoudre la question est d'offrir un nombre de points correspondant au montant dépensé ; le nombre de points requis pour avoir une boisson serait proportionnel à son prix. Il serait envisageable d'ajouter une section "Special offer" sur la page d'accueil qui comporterait toutes les boissons accompagnées du nombre de points requis.

Cette proposition est en adéquation avec la 2ème proposition qui propose de faciliter l'achat unique d'une boisson sans devoir créditer son compte au préalable. Ces offres ciblent principalement les consommateurs occasionnels qui seront challengés pour avoir un maximum de points et ainsi obtenir leur boisson préférée gratuitement. De plus, si l'utilisateur commence à consommer de façon régulière, il pourra correspondre aux habitudes de consommation d'un cluster et ainsi devenir un client fidèle, intéressé à souscrire un plan tarifaire.

Figure 23 : Exemple screen "Offre spéciale"

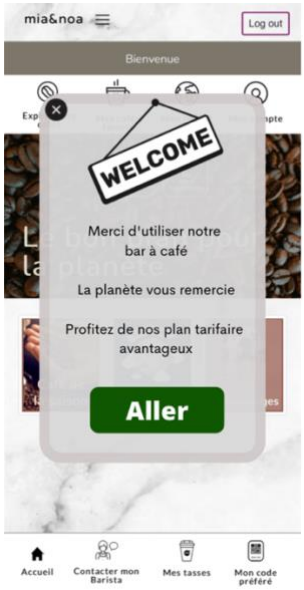
(Créé avec caneva)

6.3 Proposition d'abonnement lors de l'inscription.

La simplicité est un point clé pour mia&noa. Il n'est donc pas envisageable d'obliger l'utilisateur à remplir un questionnaire pour pourvoir lui proposer une offre adaptée car cela compliquerait l'inscription. Deux options sont envisageables de mon point de vue pour amener l'utilisateur à la page d'inscription à un plan tarifaire :

- La première serait une fenêtre « pop-up » qui apparaitrait une fois l'inscription terminée. Cette fenêtre pourrait être fermée à tout moment mais serait dotée d'un bouton donnant accès au choix des plans.

Figure 24 : Exemple Screen "plan tarifaire après l'inscription"



(Créé avec caneva)

- La seconde serait d'ajouter une option directement sur la page de d'inscription. L'utilisateur peut choisir de s'inscrire avec ou sans plan tarifaire.

Figure 25 : Exemple Screen "plan tarifaire à l'inscription"

(Créé avec caneva)

6.4 Proposition d'abonnement pour les utilisateurs existant.

Un utilisateur doit pouvoir souscrire à un nouveau plan tarifaire à tout moment. A mon sens, la page la plus adaptée pour y accéder serait la page "mon compte". Celle-ci propose actuellement la possibilité de charger son compte ou de le gérer. L'accès aux plans tarifaires se ferait de la même manière que les autres options. Pour inciter l'utilisateur à voir les différentes possibilités, un petit signe "new" peut être ajouté dans le but d'attirer l'œil.

Figure 26 : Exemple Screen "sélection d'un plan tarifaire"

(Créé avec caneva)

6.5 Plans tarifaires

Le prix du café est un élément important pour mia&noa. En effet, le lait est un élément onéreux, tout comme son substitut l'avoine. Pour que l'entreprise reste rentable avec les plans tarifaires adaptés au client, mon idée serait de diminuer la marge sur une boisson mais la quantité qui sera vendue sera plus facilement prédictible.

Le principe serait le suivant :

Un utilisateur souhaitant un plan tarifaire qui comprend une boisson par jour ouvrable pouvant contenir du lait ou son substitut.

L'algorithme prend en compte la boisson la moins rentable pour mia&noa et calcule la somme totale dans le cas où l'utilisateur utilise son plan tarifaire au maximum.

1ère étape : Compter le nombre de jours ouvrables pendant l'intervalle entre la date de souscription et la fin du deal afin de prédire le nombre total de boissons qui sera consommé et le prix correspondant.

2ème étape : Le prix du plan qui sera facturé correspondra au total obtenu à la première étape en appliquant une marge inférieure à celle appliquée à une boisson seule.

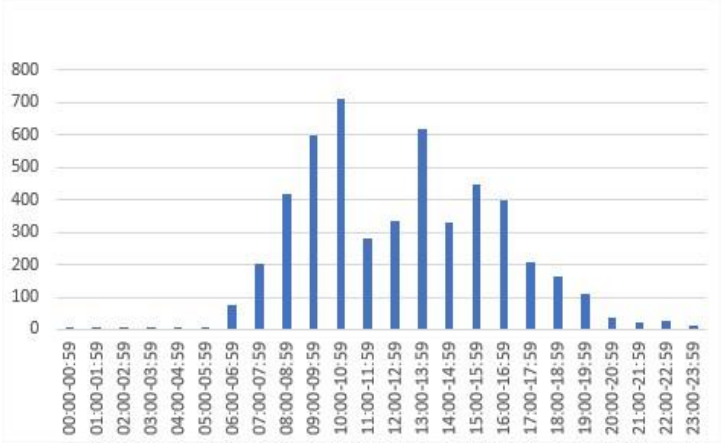
3ème étape : Analyser le prix facturé et les boissons consommés pour adapter la prochaine facture.

6.6 Offre promotionnelle

6.6.4 Happy hour

En accumulant la totalité des boissons commandées triées par tranche d'heure, on constate qu'une forte partie des utilisateurs consomme une boisson le matin et une autre après le repas de midi. Un pourcentage de réduction pourrait être offert dans les tranches d'heures creuses, soit de 11h00 -12h00, 18h00-19h00 et 19h00-20h00.

Figure 27 : Transactions par tranches d'heures



(Créé avec Excel)

6.6.5 Special week

L'idée se base sur la proposition N°5 qui propose un système de point de fidélité. Les semaines ayant comme prédiction une faible consommation pourraient offrir un nombre de points plus important qu'en temps normal. L'utilisateur sera donc récompensé de façon plus importante à chaque commande pour qu'il puisse profiter d'une boisson gratuite plus rapidement.

6.6.6 Special day

Une utilisation du concept "special day" serait de proposer une boisson particulière à un prix réduit. Les consommateurs réguliers faisant parti d'un cluster ayant une forte préférence pour une boisson pourrait également se voir offrir leur boisson préférée pour tout achat effectué.

6.7 Résumé des offres spéciales

Le but est de rendre les offres de l'application dynamiques. L'utilisateur doit être motivé et le fait de changer régulièrement les offres ou de les activer de façon temporaire le motivera à utiliser l'application afin de connaître les offres du moment.

Tous les utilisateurs doivent être pris en considération. Les consommateurs réguliers trouveront leur compte avec l'adaptation d'un plan tarifaire à payement mensuel, tandis qu'un utilisateur occasionnel pourra consommer son café en le payant directement tout en profitant d'un programme de fidélité ou d'offres promotionnelles. Bien entendu, un utilisateur possédant un plan tarifaire pourra également profiter des offres exceptionnelles, ce qui lui donnera un double avantage et majorera son sentiment de satisfaction

.

7 Analyse des données en continue

Pour garder un suivi continu sur les habitudes de consommation des utilisateurs dans le but de proposer les meilleurs abonnements possibles, l'infrastructure mise en place pour cette recherche n'est pas suffisante. L'infrastructure actuelle a été créée pour analyser un fichier Excel. Mia&noa stocke actuellement ses données dans une base de données SQL. Celle-ci contient cependant des données confidentielles. Pour continuer le projet, cette base de données devrait être migrée ou régulièrement copiée sur l'un des services proposants un data Warehouse.

Les bases de données SQL sont très populaires mais les technologies ont été adaptées pour l'analyse de bigdata en direct et sur l'automatisation des traitements. Plusieurs outils ont été développés comme Penthao utilisé lors de l'analyse préliminaire, qui peut être utilisé comme outil d'automatisation.

Ces outils offrent une palette de services mais les leaders du marché ont également créé des alternatives permettant l'analyse de bigdata.

L'utilisation d'un data warehouse mise en place dans le cloud offre de nombreux avantages. De plus, certains fournisseurs offrent bien plus qu'un hébergement des données car ils peuvent être accompagnés d'outils d'analyse dont les algorithmes de clustering les plus connus. L'accès aux données est très rapide et l'organisation de celui-ci est facilité. La sécurité des données est également essentielle et il est important de garantir la sécurité des données sensibles.

La suite de la recherche sera donc de comparer les leaders du marché et de comparer les services qu'ils offrent pouvant être utiles à l'analyse des données de mia&noa. Le nombre de données collecté est pour le moment limité, mais cela n'est que temporaire et l'optimisation de la gestion quotidienne des données produites et la minimisation des coûts est un point clé pour que l'entreprise puisse prospérer. L'utilisation d'un cloud est devenue impérative pour gérer ces transformations, ce qui rend la gestion beaucoup plus facile et ces méthodes de transformation rentables.

Il sera donc possible d'analyser les données de façon précise et instantanée. Cela implique l'analyse de la réussite ou non d'un plan tarifaire. Les méthodes agiles sont de plus en plus populaires et permettent de s'adapter à tous les types de situation.

Au fil des ans, les architectures de stockage de données ont rapidement changé et la majorité des fournisseurs de services notables proposent désormais des solutions basées sur le cloud. Cette migration a réduit les coûts initiaux pour les utilisateurs ainsi qu'amélioré l'évolutivité et les performances par rapport aux systèmes traditionnels de stockage de données.

Figure 28 : Snowflake vs Redshift vs BigQuery

(« Snowflake vs Redshift vs BigQuery: 11 Critical Differences » s. d.)

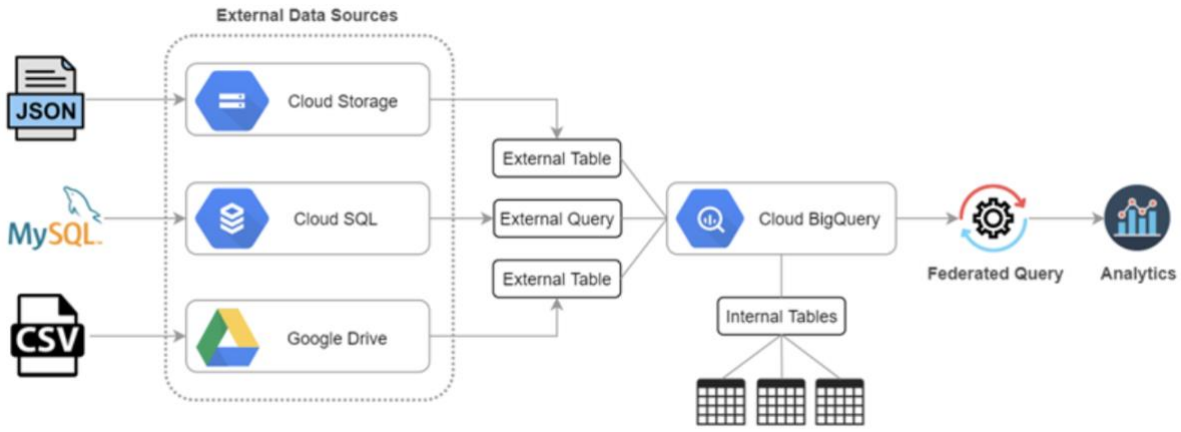
Pour proposer l'environnement le plus adapté à mia&noa, la comparaison entre les trois principales plateformes de data wharehouse disponibles sur le marché permettra de comprendre globalement leur fonctionnement. Google BigQuery, Redshift et Snowflake sont les fournisseurs de SaaS (Software as a Service) les plus notables en ce moment. Le lundi 9 août 2021, Sébastian Knief m'a informé qu'il avait l'intention d'utiliser Google BigQuery et qu'un test de migration est en train d'être effectué. Un script est en cours d'écriture pour que les données transmises à la base de données SQL actuelle soient copiées sur la plateforme BigQuery. La base de données SQL actuellement en production ne doit pas être corrompue. Pour cela les données seront redondantes entre les différentes bases de données. Étant donné que l'utilisation de BigQuery n'est qu'à l'étape de test, la comparaison entre deux autres plateformes peut être utile pour renforcer le sentiment de mia&noa d'utiliser les services de Google, ou alors les motiver à utiliser les services d'un de ses concurrents directs.

7.2 BigQuery proposé par Google :

Pour commencer, BigQuery est un data warehouse en mode "serverless". Cela signifie qu'il n'y a pas d'infrastructure à administrer. BigQuery permet le traitement d'ensembles de données équivalant à plusieurs pétaoctets de données. L'architecture du data warehouse est conçue pour exécuter des requêtes SQL et désignée pour analyser une grande échelle de données allant jusqu'à des milliards de lignes.

Les ressources nécessaires sont allouées automatiquement à chaque fois que c'est nécessaire. Cela exempte l'utilisateur de disposer d'une machine virtuelle ou d'instance.

Figure 29 : Illustration du pipeline d'analyse de BigQuery



(Mishra 2020)

La solution est idéale pour effectuer des regroupements ou tout type de calcul. C'est un outil puissant pour les analystes permettant des gérer les données d'une manière simplifiée. De plus, la tarification de ce service est à la demande ou forfaitaire ce qui le rend peu coûteux.

BigQuery rendrait la visualisation de toutes les transactions ou autres informations par domaine ou potentiel succursale de façon simplifié et très rapide.

7.3 Snowflake :

Ce service basé dans le cloud permet l'analyse de données sous forme de SaaS. Il est également alimenté par le protocole SQL et supporte les données complètes ainsi que les semi-structurées telles que JSON, XML, etc. Snowflake est une base de données orientée colonne. Les données sont transmises par colonne contrairement au base orienté objet qui retournerait la ligne d'une table. Snowflake n'intègre cependant aucun stockage, cela signifie qu'il s'appuie sur une base de données native d'un fournisseur cloud. Ce service possède plusieurs similitudes avec BigQuery, aucun de ces services nécessitant d'installation de logiciel ou de configuration au niveau hardware. A la différence de BigQuery, Snowflake n'offre pour le moment pas d'outil de machine learning intégré, mais il a l'avantage d'être une solution compatible avec tous les clouds provider.

Figure 30 : Snowflake illustration

(« Beyond "Modern" Data Architecture - Snowflake Blog » 2020)

Snowflake est une solution intéressante pour les entreprises qui utilisent les services de plusieurs clouds providers voulant effectuer du big data. Dans la situation de mia&noa qui est en plein développement et utilise une base de données SQL, la contrainte d'une compatibilité entre les différents clouds n'est pas déterminante pour cette entreprise. Contrairement à BigQuery, le service Snowflake n'est pas totalement Serverless, en effet il est nécessaire de choisir une configuration appropriée et précalculer les besoins aux heures de pointe, et la capacité des entrepôts de données doit être choisie lors du déploiement.

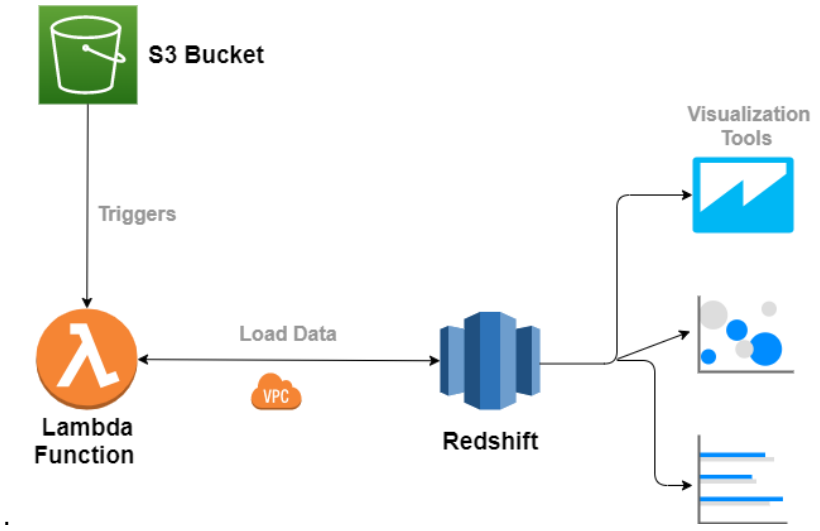
7.4 Redshift propsé par Amazon :

Amazon a créé le service de data warehouse Readshif et le gère complètement. Il est conçu pour gérer des données volumineuses à l'échelle du pétaoctet. Amazon Web Service (AWS) est, selon eux, la plateforme cloud la plus complète et la plus largement adoptée dans le monde entier. Cette plateforme propose plus de 200 services et possède des centres de données sur tous les continents.

Redshift est l'un des services proposés par AWS, celui-ci s'intègre donc parfaitement bien dans l'environnement d'Amazon. Contrairement à BigQuery qui est un data wharehouse en mode Serverless, Redshift est totalement managé par AWS. Une configuration préalable est donc indispensable pour utiliser le service d'Amazon.

AWS facture ses services à l'heure pour chacun des serveurs, même si ceux-ci ne sont pas utilisés. La vitesse des requêtes dépendra de la configuration préalable, il faut donc soigneusement préparer l'environnement pour utiliser Redshift.

Figure 31 : Illustration Redshift dans son environnement



(LaptrinhX 2020)

Redshift est un outil puissant ; cependant pour effectuer des requêtes rapides, le service vous demande impérativement d'indexer les données selon les critères d'analyse souhaités. Cela rend difficile l'analyse en live de données car il faudra mettre à jour les index en continu. Redshift requiert un plus haut niveau de compétence pour pouvoir l'utiliser de manière optimale, contrairement à Bigquery qui s'adapte parfaitement bien et de façon autonome.

La tarification est également un point clé car il faudra utiliser le service de façon optimale pour limiter les coûts d'analyse.

7.5 La solution

La solution choisie par Sébastian Knief est donc, à mon sens, un très bon choix. BigQuery étant une solution en mode serverless, cela dispense mia&noa le management des serveurs pour effectuer des analyses et par conséquent du personnel qualifié avec un salaire élevé. La capacité de stockage et de traitement du service fourni par google est équivalente à plusieurs milliards de lignes. Pour le moment seules les transactions des différents bars à café sont enregistrées mais BigQuery peut devenir un outil indispensable pour faire évoluer l'entreprise. Le traitement de big data permet l'analyse d'une donnée de façon bien plus précise. Il serait possible par exemple de regarder le nombre de connexions avant d'effectuer une commande, ou combien de café ont été consulté avant de choisir celui qui sera servi. Les données collectées peuvent provenir de plusieurs sources autre que l'application comme la page Facebook ou leur site internet.

Un profilage des personnes deviendra donc possible et cela en concevant les garanties de confidentialités et de sécurité.

Avant une migration complète, il est également possible d'effectuer une série de tests et d'utiliser le BigQuery au stade de développement. L'accès aux données étant simplifié et sécurisé, les principes de DevOPS peuvent facilement être appliqués en automatisant les analyses ainsi que les rapports d'activité.

BigQuery propose également des outils de machines learning ce qui le rend très utile dans le cadre de cette recherche. Les données peuvent être injectées de différentes manières et tous les outils de Google analytique son disponible. BigQuery peut fournir et traiter les données de façon très rapide avec un langage similaire à SQL. Le machine learning peut être utilisé pour attribuer un type de plan tarifaire pour un type de consommation afin de proposer un plan tarifaire ayant le plus de probabilité d'être choisie. La possibilité d'effectuer des prédictions sur la réussite d'un bar à café peut être également envisageable selon son emplacement à condition d'avoir un nombre important de données à analyser.

La gestion des données est simplifiée et plusieurs attributs peuvent être facilement corrélé aux transactions effectuées par les utilisateurs. Avec un tel service, il serait possible de distinguer la cause d'une une baisse de consommation dû aux conditions météorologiques qui empêchait l'accès au bar à café situé à l'extérieur. Il suffirait d'effectuer une recherche en effectuant une jointure sur une table contenant les

conditions météo à un timestamp donné. L'utilisation du machine learning pourrait également estimer quand est-ce qu'un utilisateur effectuera sa prochaine transaction.

Les données peuvent également être géré selon différents domaines. Mia&noa s'agrandit et les bars à café seront franchisé. L'analyse par domaine sera indispensable pour comprendre ce qui plaît le plus au client. Pour évoluer, l'entreprise installe de plus en plus de bars à café et ceux-ci doivent impérativement être disposés à des emplacements stratégiques. L'analyse précise des données peut être utilisée pour faire des statistiques de réussite selon l'emplacement dans le but d'utiliser ces statistiques pour convaincre de futur franchisé.

L'environnement proposé par Google permet donc d'utiliser des données provenant de plusieurs sources et de les traiter pour les utiliser pour tous les besoins métiers. Le marketing de l'entreprise pourra adapter sa stratégie mais également les techniciens qui pourront optimiser leurs déplacements en anticipent les services que les bars à café doivent subir.

8 Conclusion

La quantité de données est essentielle pour tirer des statistiques comportementales viable. Les différents algorithmes de clustering m'ont permis de classer les utilisateurs selon le nombre de transactions d'une ou deux boissons consommées mais également de distinguer le grand nombre d'utilisateur ayant uniquement profité de l'offre d'adhésion. Les algorithmes utilisés peuvent analyser toutes sorte de données comme des images, et ne sont donc pas spécifiques à la problématique. L'utilisation d'un data warehouse est indispensable pour effectuer ces différentes analyses et une solution cloud est parfaitement adaptée aux ambitions de mia&noa.

Les différentes propositions qui ont été formulées ont pour objectif de faciliter la commande de boisson et de payement tout en donnant aux clients la garantie de payer son café au meilleur tarif. L'obligation d'inscription est potentiellement un frein à l'achat d'une boisson, la proposition n°2 prend donc en compte cette catégorie d'utilisateur en leur offrant la possibilité de commander une boisson de manière simplifiée. Les gares ferroviaires sont un emplacement stratégique et l'optimisation du temps de commande est essentielle pour satisfaire les clients pressés. Un plan tarifaire permet aux clients de ne plus se soucier des payements d'une boisson, de raccourcir le temps de commande et également d'économiser.

Le data warehouse créé pour cette recherche a été indispensable pour pouvoir effectuer les différentes statistiques d'utilisation mais n'est pas à la hauteur des ambitions de mia&noa. En effet, la base de données est en local et donc difficilement transportable. De plus, seul 6669 transactions ont été analysées, ce qui permet l'utilisation d'algorithmes complexes. Les données sont cependant en constante évolution et le nombre d'utilisateurs augmente tous les jours. Les attributs analysés vont également évoluer avec le temps, et le temps de calcul pour effectuer les analyses sera de plusieurs jour, voire plusieurs années une fois que le nombre de données collectées correspondra à plusieurs million ou milliard de lignes.

Pour évoluer et continuer les analyses de manière efficace, le service BigQuery proposé par Google correspond parfaitement aux espérances de l'entreprise. Il est essentiel d'avoir une infrastructure pouvant traiter le big data pour que l'entreprise puisse s'agrandir rapidement sans affecter l'expérience utilisateur. L'environnement permettra entre autres d'avoir des analyses instantanées pour pouvoir réagir très rapidement. Cependant les modifications au niveau applicatif sont couteuses et doivent être minutieusement testées avant la mise en production.

Bibliographie

Analytics & Insights. « 4 Algorithmes d'Apprentissage Non Supervisé », 9 février 2019. https://analyticsinsights.io/apprentissage-non-supervise/.

Snowflake. « Beyond "Modern" Data Architecture - Snowflake Blog », 9 avril 2020. https://www.snowflake.com/blog/beyond-modern-data-architecture/?lang=fr.

Brownlee, Jason. « 10 Clustering Algorithms With Python ». *Machine Learning Mastery* (blog), 5 avril 2020. https://machinelearningmastery.com/clustering-algorithms-with-python/.

Data Analytics Post. « Clustering ». Consulté le 30 juillet 2021. https://dataanalyticspost.com/Lexique/clustering/.

Google Developers. « Common ML Problems | Introduction to Machine Learning Problem Framing ». Consulté le 7 juillet 2021. https://developers.google.com/machine-learning/problem-framing/cases?hl=fr.

CodeProject. « Create & Populate Time Dimension with 24 Hour+ Values », 25 août 2013. https://www.codeproject.com/Tips/642912/Create-Populate-Time-Dimension-with-Hourplus-Va.

Google Cloud. « Créer un modèle de clustering en k-moyennes pour la segmentation du marché à l'aide de BigQuery ML ». Consulté le 20 août 2021. https://cloud.google.com/architecture/building-k-means-clustering-model?hl=fr.

Ippon | Cabinet de conseil et expertise en technologies | Discovery to Delivery. « DATA WAREHOUSE CLOUD Redshift vs Snowflake », 24 mai 2019. http://blog.ippon.fr/2019/05/24/redshift-vs-snowflake/.

Cartelis. « Design d'un Data Warehouse - Zoom sur la modélisation en étoile », 20 juin 2019. https://www.cartelis.com/blog/data-warehouse-modelisation-etoile/.

« Différences entre BigQuery et Redshift ». Consulté le 9 septembre 2021. https://wiki.sfeir.com/googlecloudplatform/bigdata/bigquery/bigquery_vs_redshift/.

« Differences entre Snowflake et BigQuery ». Consulté le 9 septembre 2021. https://wiki.sfeir.com/googlecloudplatform/bigdata/bigquery/bigquery_vs_snowflake/.

François Husson. *k-means : méthode de partitionnement (cours 3/4)*. Consulté le 19 juillet 2021.

https://www.youtube.com/watch?v=3VLGFOMj8oI&ab_channel=Fran%C3%A7oisHusson.

Google Cloud Tech. *What is BigQuery?*, 2020. https://www.youtube.com/watch?v=d3MDxC_iuaw.

XLSTAT, Your data analysis solution. « K plus proches voisins (KNN) ». Consulté le 31 juillet 2021. https://www.xlstat.com/fr/solutions/fonctionnalites/k-nearest-neighbors-knn.

« K-moyennes ». In *Wikipédia*, 9 septembre 2020. https://fr.wikipedia.org/w/index.php?title=K-moyennes&oldid=174559088.

BDM. « Langages de programmation : évolution, tendances, communautés et emploi », 2 juin 2021. https://preprod.blogdumoderateur.com/langages-programmation-evolution-tendances-communautes-emploi/.

LaptrinhX. « How to Send a CSV File from S3 into Redshift with an AWS Lambda Function ». LaptrinhX, 11 décembre 2020. https://laptrinhx.com/how-to-send-a-csv-file-from-s3-into-redshift-with-an-aws-lambda-function-2872006355/.

Analytics & Insights. « Le Clustering: Définition et Top 5 Algorithmes », 1 mars 2019. https://analyticsinsights.io/le-clustering-definition-et-implementations/.

Formation Data Science | DataScientest.com. « Machine Learning & Clustering: Focus sur l'Algorithme DBSCAN », 29 juillet 2020. https://datascientest.com/machine-learning-clustering-dbscan.

OpenGenus IQ: Computing Expertise & Legacy. « Mean Shift Clustering Algorithm », 6 avril 2019. https://iq.opengenus.org/mean-shift-clustering-algorithm/.

MeasureSchool. *Big Query Live Training - A Deep Dive into Data Pipelining*, 2020. https://www.youtube.com/watch?v=UPMH11BqvGs.

Mishra, Soumendra. « How to Integrate External Data Sources with BigQuery ». *Google Cloud - Community* (blog), 12 septembre 2020. https://medium.com/google-cloud/how-to-integrate-external-data-sources-with-bigquery-9e126d5751ea.

Mquantin. *Français : Illustration du déroulement de l'algorithme des k-means*. 27 juillet 2017. Travail personnel. https://commons.wikimedia.org/wiki/File:K-means.png?uselang=fr.

OpenClassrooms. « Partitionnez vos données avec DBSCAN ». Consulté le 2 août 2021. https://openclassrooms.com/fr/courses/4379436-explorez-vos-donnees-avec-des-algorithmes-non-supervises/4379571-partitionnez-vos-donnees-avec-dbscan.

Pentaho Documentation. « Pentaho Data Integration », 21 janvier 2020. https://help.pentaho.com/Documentation/9.0/Products/Pentaho_Data_Integration.

Stitch. « Pentaho Data Integration (Kettle) vs. Stitch - Compare Features, Pricing, Services, and More. » Consulté le 28 juillet 2021. https://www.stitchdata.com/vs/pentaho/.

Amazon Web Services, Inc. « Qu'est-ce qu'AWS ? » Consulté le 9 septembre 2021. https://aws.amazon.com/fr/what-is-aws/.

Snowflake Inc. *What is Snowflake? 8 Minute Demo | Snowflake Inc.*, 2020. https://www.youtube.com/watch?v=xojAXXRo_S0.

Learn | Hevo. « Snowflake vs Redshift vs BigQuery: 11 Critical Differences ». Consulté le 5 septembre 2021. https://hevodata.com/learn/snowflake-vs-redshift-vs-bigquery/.

« Social Network for Programmers and Developers ». Consulté le 20 septembre 2021. https://morioh.com/p/0a754552186a.