DÉCEMBRE 2022

Génération de données synthétiques

Projet Tuteuré

ISIS INGÉNIEURS CASTRES

accenture

Rédigé par :
Kawtar Hamdi
Julie Hoarau
Nelson Rogers

REMERCIEMENTS

Nous tenons d'abord à remercier Mme. Baya Dhouib, M. Richard Vidal et Mme. Laetitia Kameni pour cette opportunité de collaboration avec Accenture en intelligence artificielle. Nous n'étions pas sûr de pouvoir trouver un sujet qui nous convenait, et nous avons été très satisfait du projet proposé. Ils nous ont offert l'opportunité de découvrir les enjeux sur lesquels travaille Accenture Labs et nous ont accordé leur confiance.
Nous les remercions pour la mission qu'ils nous ont confiée et pour tout le soutien qu'ils nous ont apporté.

Nous remercions notre école, ISIS Castres, de nous avoir permis de réaliser un tel projet tuteuré et l'ensemble du personnel de l'école qui a contribué à la recherche de sujets.

Merci également à Mme. Imen Megdiche qui nous a suivi pendant ce projet, malgré le fait qu'elle attende un enfant qui doit bientôt arriver. Nous vous souhaitons le meilleur pour cette nouvelle arrivée dans votre famille.

GLOSSAIRE

Données synthétiques : Les données synthétiques sont des données générées artificiellement par un modèle d'IA entraîné sur un ensemble de données réelles. Par exemple la collecte de données utilisateurs ou de données de santé.

Data conditioning : Le data conditioning optimise le mouvement et la gestion des données afin de les protéger et d'augmenter leur productivité. Le data conditioning utilise des techniques spécialisées conçues pour acheminer, optimiser et protéger les données stockées ou les données lors de leur déplacement dans un système informatique.

Intelligence artificielle : L'intelligence artificielle est la simulation des processus de l'intelligence humaine par des machines, en particulier des systèmes informatiques.

Machine Learning : C'est un ensemble de techniques donnant la capacité aux machines d'apprendre automatiquement un ensemble de règles à partir de données. Contrairement à la programmation qui consiste en l'exécution de règles prédéterminées. [1]

Deep Learning: C'est une technique de machine learning reposant sur le modèle des réseaux neurones: des dizaines voire des centaines de couches de neurones sont empilées pour apporter une plus grande complexité à l'établissement des règles. [1]

GAN : Les Generative Adversarial Networks, ou GAN, sont une approche de la modélisation générative utilisant des méthodes de deep learning, telles que les réseaux de neurones convolutifs.

Copule Gaussienne : C'est un modèle statistique et probabiliste qui utilise une fonction de distribution cumulative (multivariable). Ce modèle permet de décrire la distribution conjointe de l'ensemble des variables contenues dans nos données (les différentes colonnes des tables). Ainsi, ce modèle nous permet de générer des données possédant les mêmes caractéristiques statistiques que les données initiales.

Column Shape : La forme d'une colonne (*en anglais* "column shape") décrit sa distribution globale. Plus le score est élevé, plus les distributions des données réelles et synthétiques sont similaires.

Column Pair Trends: La tendance entre deux colonnes (Column pair trends) décrit comment elles varient l'une par rapport à l'autre, par exemple la corrélation. Plus le score est élevé, plus les tendances sont similaires.

Column Coverage: Cette propriété applique les métriques RangeCoverage aux données numériques et CategoryCoverage aux données catégorielles. Cela permet d'obtenir un score de couverture pour chaque colonne. Le score final de la propriété est la moyenne de toutes les colonnes.

Column Boundaries: Cette propriété applique la métrique BoundaryAdherence aux colonnes numériques uniquement. Le score final est la moyenne de toutes les colonnes.

SOMMAIRE

INTRODUCTION

Aujourd'hui, les avancées technologiques provoquent une numérisation de plus en plus importante de toutes nos données. Ce phénomène est notamment observable par l'utilisation de plus en plus fréquente de l'intelligence artificielle (IA), par exemple. En effet, l'IA est l'une des clés pour résoudre les problèmes auxquels nous faisons face au sein de la société actuelle. L'IA est notamment présente dans le cadre commercial et utilisée à des fins de stratégie et d'analyse.

Les données prennent alors une place de plus en plus importante dans le monde moderne. Le développement de modèles d'IA performants fait que la demande pour des données de haute qualité en grande quantité ne cesse d'augmenter.

Toutefois, la collecte et le traitement des données en question ne sont pas toujours des tâches faciles. En effet, de nombreux problèmes que nous cherchons à résoudre nécessitent l'utilisation de données sensibles ou rares.

Apparaît alors le questionnement autour du type de traitement que nos données personnelles vont subir. Comment assurer un traitement respectueux du caractère sensible des données manipulées tout en ayant des données en quantité et qualité suffisamment importantes pour le développement de modèles d'IA performants ?

C'est dans ce questionnement que s'est fondé notre projet tuteuré. En effet, nous avons réalisé un travail de recherche dans le la génération de données synthétiques. Nous nous intéressons aux données tabulaires et plus particulièrement aux données tabulaires relationnelles (multi-tables).

De nombreux travaux ont été fait pour la génération de données contenues dans une seule table tels que le développement des modèles de TGAN, CTGAN ou bien TabFairGAN. Notre travail a pour but de trouver des solutions à la génération de données selon un modèle relationnel tout en respectant certains critères d'anonymat. Pour cela, nous avons réalisé une analyse de l'état de l'art et exploré des pistes d'amélioration de ces concepts.

CONTEXTE DU PROJET

A. Contexte général

1. Présentation de Accenture [2]

Chaque année notre école ISIS propose un projet tutoré en partenariat avec des entreprises, le but est d'appliquer nos connaissances, nos cours sur un des projets proposés.
Le projet Synthétique a été proposé par Accenture qui est une entreprise internationale de conseil en technologie, elle a été créée le 1 Janvier 2001 sous le nom de Andersen Consulting. Accenture est le partenaire stratégique des entreprises et institutions françaises dans leur transformation technologique et humaine.

En chiffres, +10 000 collaborateurs en France, 40 secteurs d'activité et 80% des entreprises du CAC40 travaillent avec Accenture depuis plus de 10 ans.

2. Qu'est-ce que les données synthétiques ?

Les données synthétiques sont des données générées artificiellement par un modèle d'IA entraîné sur un ensemble de données réelles. Par exemple la collecte de données utilisateurs ou de données de santé. Son objectif est de reproduire les propriétés et les modèles statistiques d'un ensemble de données existantes, grâce à une modélisation de leur distribution probabiliste et à un échantillonnage.
Le but principal des données synthétiques est d'éliminer tout risque d'exposer des données critiques ou de compromettre la confidentialité et la sécurité des entreprises et de leurs clients.

Il existe trois types de données synthétiques :

1-Données factices/ données fictives :

Les données fictives sont des données qu'on peut générer aléatoirement mais les caractéristiques, les relations et les modèles statistiques qui se trouvent dans les données d'origine ne sont pas conservés. (il existe des méthodes ainsi que des sites qui permettent la génération des données fictives).

2-Données synthétiques générées à base de règles :

Les données synthétiques générées à partir de règles sont des données synthétiques générées par un ensemble prédéfini de règles.

3-Données synthétiques générées par l'intelligence artificielle IA :

Ce type de données sont générées par des algorithmes d'intelligence artificielle.
Le modèle d'IA est formé sur les données d'origine pour apprendre toutes les caractéristiques, relations et modèles statistiques. Après, cet algorithme d'IA est capable de générer des points de données entièrement nouveaux et de modéliser ces nouveaux points de données de manière à reproduire les caractéristiques, les relations et les modèles statistiques de l'ensemble de données d'origine.

3. Avantages des données synthétiques ?

a) Rapidité : accélération de l'accès aux données

Les données synthétiques peuvent compléter et remplacer les données réelles et ainsi permettre l'utilisation accélérée de l'ensemble des données traitées.

Le concept de donnée synthétique existe depuis les années 90 mais les nouveaux algorithmes génératifs ont grandement amélioré leur qualité.
Les premières startups créatrices de données synthétiques par machine learning sont apparues dans le marché des véhicules autonomes avec des sociétés comme Applied Intuition, Parallel Domain ou Cognata. L'apport des données synthétiques permettait de générer tous les cas possibles afin d'entraîner les algorithmes de pilotage, ce qui n'est pas possible dans le monde réel.
Au fur et à mesure, l'utilisation des données synthétiques s'est généralisée à d'autres domaines, comme la reconnaissance faciale, la vision par ordinateur…etc

Prenons l'exemple d'une institution financière. L'entreprise disposait d'une base de données inestimables, qui pouvait aider les décisionnaires à résoudre des problèmes métier. Toutefois, ces données étaient soumises à une protection et à un contrôle rigoureux, qu'y accéder relevait du parcours du combattant – même si elles étaient destinées à ne jamais sortir de l'entreprise. Par conséquent, l'équipe d'analystes a attendu six mois avant d'avoir accès à un faible volume de données, qu'elle devait au demeurant très rapidement exploiter. Et il lui a fallu six mois de plus pour accéder à des données mises à jour. Pour contourner cet obstacle, l'entreprise a donc créé des données synthétiques à partir de ses données d'origine. L'équipe peut maintenant mettre à jour et modéliser ses données sans interruption, ce qui lui permet d'extraire en continu de précieuses informations sur la façon d'améliorer les performances de l'entreprise (exemple tiré de [3]).

b) Scalabilité : résolution d'enjeux plus importants et plus variés

Les données synthétiques ont la possibilité d'augmenter la qualité de traitement et évoluer en grande quantité grâce à son aspect génératif.
La scalabilité est une résultante de la sécurité et de la rapidité. Davantage de données peuvent être analysées grâce à un accès rapide et sûr aux données, et, ainsi de résoudre une plus grande diversité de problèmes.

4. Qu'est-ce que des données anonymes ?

Les données anonymes sont les données à caractère personnel qui ont été dépersonnalisées. Il est donc impossible, en pratique, d'identifier la personne concernée.
Les données à caractère personnel sont les données telles que le nom, la date de naissance, le numéro de sécurité sociale, etc.

Depuis le 25 Mai 2018 le règlement général sur la protection des données (RGPD) établit des règles afin de protéger l'identité des personnes face aux traitement des données ainsi qu'aux circulations de ses informations personnelles.

B. Contexte juridique

1. Les réglementations

Dans le cadre des données sensibles telles que les informations personnelles identifiables (PII en Amérique du Nord, ou données à caractère personnel en Europe) ou les informations personnelles sur la santé (PHI, ou données sensibles de santé en Europe), les entreprises sont contraintes à respecter certaines réglementations.

En Europe, les entreprises doivent notamment respecter la RGPD sous peine d'amendes importantes. En France, il existe aussi la Commission Nationale de l'Informatique et des Libertés (CNIL) qui assure l'application de la loi concernant la collecte, le stockage et le traitement des données.

2. Les méthodes mises en place pour respecter la législation

Afin de pouvoir expérimenter librement avec le traitement des données dans le cadre du projet, nous avons choisi de générer nos propres données. Cette décision permettait également d'enlever la contrainte de devoir trouver un dataset libre d'accès assez complexe et ayant assez de lignes. Pour ce faire, nous avons utilisé https://filldb.info/, un générateur de données de test MySQL en ligne. Ainsi, nous étions capables de définir le schéma de données que nous souhaitons pour commencer avec des données de base convenables.

Toutefois, cette génération n'est pas parfaite, nous avons donc dû effectuer un traitement de ces données afin d'avoir des données cohérentes, ce que nous verrons dans la suite de ce rapport.

C. Acteurs impliqués

La méthode des personas consiste à créer une représentation fictive mais réaliste des utilisateurs du produit numérique ou du service. L'objectif principal est de permettre aux concepteurs et aux développeurs de se référer aux utilisateurs. Les personas les représentent concrètement.

L'objectif de la définition des personas est de rassembler en une simple fiche le maximum d'informations sur la cible afin de réussir à vous mettre à sa place, et anticiper ses attentes. Cette étape est essentielle à la définition de votre stratégie web.
On a créé les 3 principales personas qui gèrent le projet Synthetic au sein d'Accenture, ensuite la personas de notre tuteur et finalement 3 personas qui représentent l'équipe d'ISIS. (Voir Annexes)

RÉALISATIONS

A. Démarche projet

1. Outils utilisés

a) GanttProject [4]

Figure 1: Gantt Project

GanttProject est un outil libre de gestion de projet écrit en Java, ce qui permet de l'utiliser sur de nombreux OS tels que Windows, Linux, MacOS.

Le diagramme de Gantt est donc nommé d'après son inventeur, Henry Laurence Gantt (1861-1919). Il permet de visualiser de façon simple toutes les tâches planifiées d'un projet, ainsi que leurs dates d'échéance.

Dans un diagramme de Gantt, chaque tâche est représentée par une ligne. En colonne, nous trouvons la description de chaque tâche et en ligne se trouve l'échelle de temps en jours, semaine, mois… A l'extrême gauche de chaque ligne/tâche se trouve une flèche représentant la date de début de l'action, à l'extrême droite de cette même ligne se trouve la date de fin de la tâche.

b) MySQL [5]

Figure 2 : MySQL

Notre projet consiste à travailler avec une base de données. Parmi les logiciels qu'on a utilisés on trouve MySQL qui permet notamment de stocker nos données relationnelles. C'est-à-dire dans des tables, dont chaque table définit une caractéristique qui est divisée en lignes (ou enregistrements) ainsi qu'en colonnes (ou champs).

MySql est un logiciel gratuit avec une installation facile sur les Windows et les MAC, il a servi pour faire des requêtes, nettoyer notre base de données ainsi que tester la fonctionnalité de nos méthodes utilisées pour générer les données synthétiques.

c) Google Colab [6]

Figure 3 : Google Colab

Google Colab est un produit de Google Research, il permet de partager le code entre l'ensemble de l'équipe, chaque membre peut modifier, supprimer ainsi qu'ajouter son propre code et l'exécuter sur le navigateur. Il nous a servi essentiellement pour avancer, partager et tester sur nos propres machines.

Son utilisation est facile grâce à son environnement particulièrement adapté à l'analyse des données, les fonctions et les méthodes utilisées ainsi qu'au machine learning. Cependant, il présente des gros désavantages (sur la version gratuite du moins) tels que le le temps d'exécution et les runtimes se réinitialisant.

d) Microsoft Teams [7]

Figure 4 : Microsoft Teams

Notre travail a été en collaboration avec Accenture, on était en challenge avec la durée du projet et la réussite de notre génération de données synthétiques de qualité, on a privilégié un meeting tous les 15 jours avec Mme. Baya Dhouib pour présenter notre avancement et poser nos questions afin de trouver une solution.

Tous les meetings se sont passés sur Microsoft Teams qui est une application de collaboration conçue pour un travail en équipe qui permet à l'équipe de rester informée, organisée et connectée mais de partager aussi des documents sur l'espace de Chat.

Nous avions initialement prévu de réaliser les réunions sur Zoom et avions même préparé une licence, mais il s'est avéré que l'utilisation de Teams était plus adaptée du point de vue d'Accenture.

2. Évolution du projet

Sur notre projet tutoré, on était face à un délai court pour finir notre projet, avec des réunions tous les 15 jours. Grâce au diagramme de Gantt on a mis l'ensemble des tâches réalisées, les réunions avec notre tuteur, notre avancement sur les tâches essentielles en spécifiant une date de début et la date de la fin.

L'organisation du projet reste un élément essentiel pour réussir le but final et délivrer le rapport dans les délais mis en place au départ.

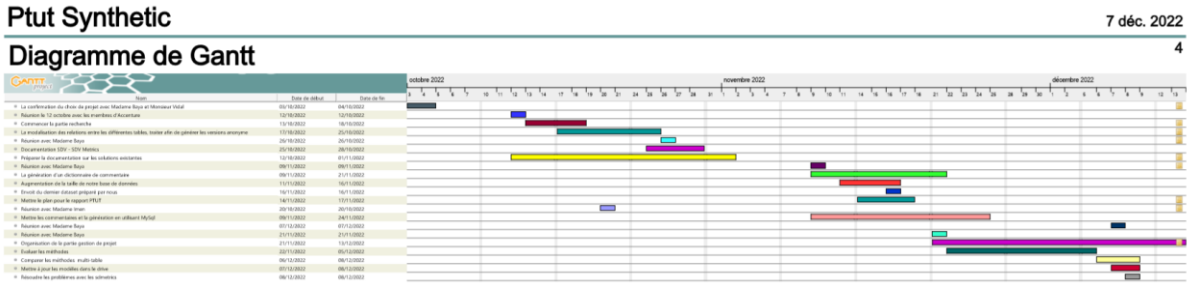

Figure 5 : Diagramme de Gantt du projet

Afin de bien organiser l'avancement du projet et de tout mettre en place, on a décidé de mettre un compte rendu pour chaque tâche mise en ligne sur le logiciel Gantt par exemple :

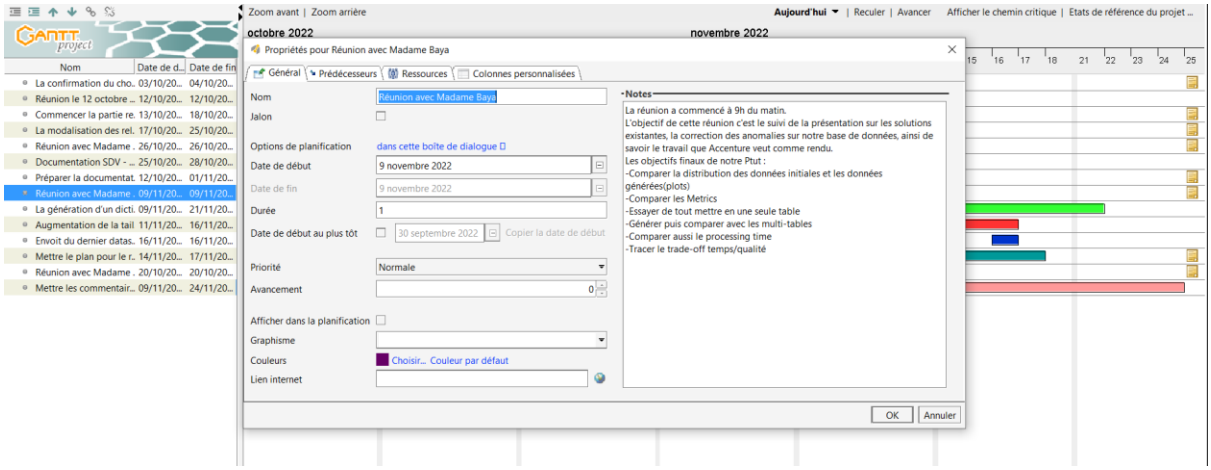

Figure 6 : Renseignement des tâches dans le diagramme de Gantt

Grâce à cette fonctionnalité on a pu ajouter un petit paragraphe qui englobe les objectifs ainsi que les missions prioritaires à faire pour chaque séance ou réunion.

B. Activités réalisées et résultats obtenus

1. Mise en place d'un jeu de données initial

Bien que nous fussions initialement partis sur l'utilisation de données de santé, cela posait un problème pour l'obtention de données intéressantes pour la création d'un modèle d'intelligence artificielle. En effet, les données de santé sont des données sensibles, selon le RGPD, et il est difficile d'obtenir des données intéressantes en quantité suffisante.

Nous avons donc fait le choix d'utiliser des données issues d'un modèle de données de e-commerce qui présente une complexité intéressante pour assurer un modèle robuste. En effet, les modèles de données des sites de e-commerce présentent une variété de types et une multitude de relations entre les tables qui est intéressante dans le cadre notre recherche.

La génération effectuée avec [8] nous a permis d'obtenir un jeu de données relationnel avec certaines tables ayant jusqu'à 10 000 lignes selon le modèle suivant :

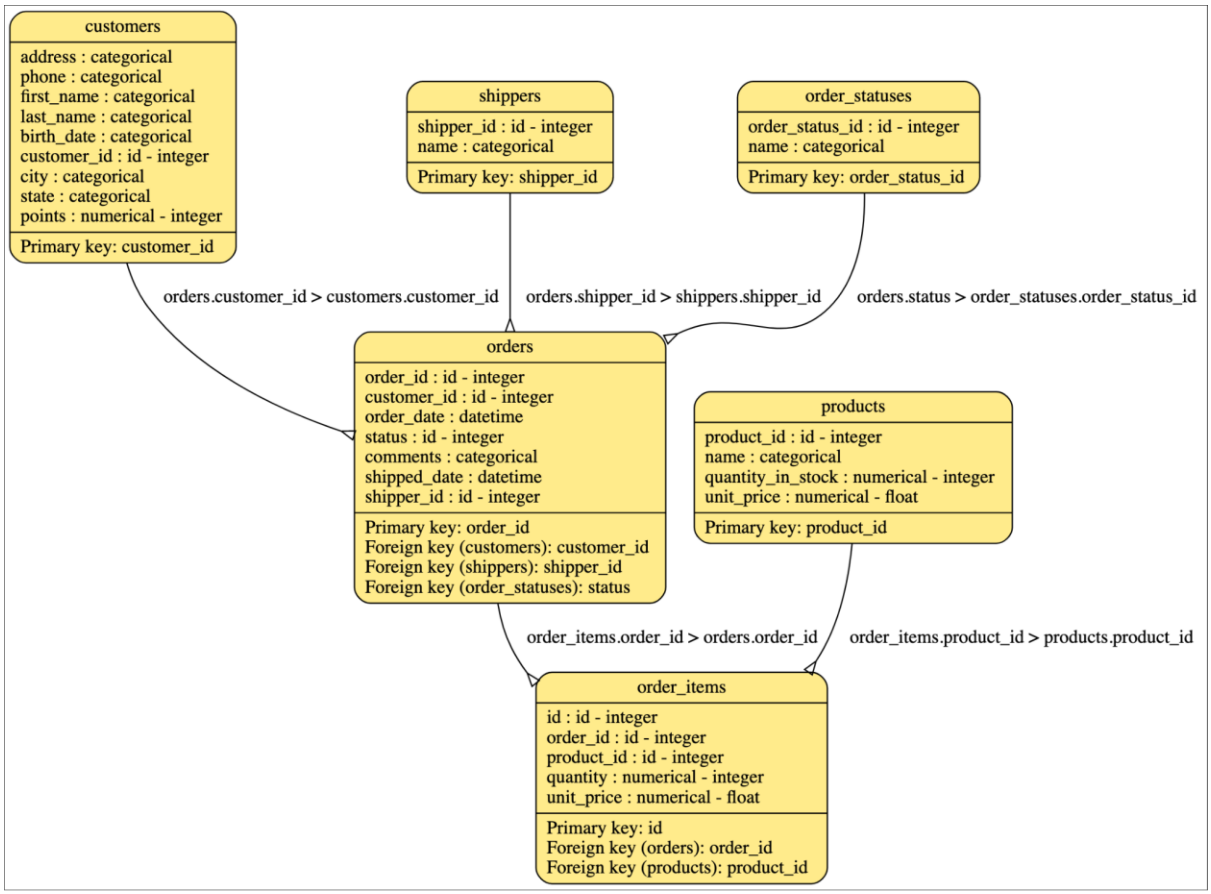

Figure 7 : Modèle de données du projet généré avec les métadonnées de SDV

Les données générées, bien que dans le bon format, n'étaient pas cohérentes d'une colonne à une autre. En effet, ce problème se remarquait notamment au niveau des dates. Par exemple, certaines commandes pouvaient présenter une date d'envoi en 2010, une date de commande en 2013 et une date de naissance client en 2018. Ces valeurs sont bien évidemment absurdes. Il a donc fallu corriger les incohérences dans les données avant de procéder à la génération de données synthétiques.

Les transformations suivantes ont été réalisées sur notre base de données MySQL en utilisant des requêtes SQL :

- Modification des dates d'un même enregistrement pour assurer la cohérence inter-colonnes et inter-tables.
- Modifications des références au statut (1-en cours de traitement, 2-envoyée, 3-livrée) d'une commande pour assurer la cohérence entre le statut et les colonnes concernant l'envoi.
- Assurer que si une commande a été envoyée, il y a une date d'envoi renseignée
- Insertion de valeurs nulles pour éviter un dataset trop parfait

La génération initiale ne permettant pas d'ajouter de vrais commentaires aux commandes, nous avons cherché un dataset libre d'accès pour le joindre à notre jeu de données. Nous avons ensuite assigné de manière aléatoire un certain nombre de commentaires aux différentes commandes.

Une autre contrainte au niveau des données a dû être respectée du fait de certaines limitations présentées pas Synthetic Data Vault (SDV), la bibliothèque que nous utilisions pour la génération de données synthétiques. En effet, cette bibliothèque ne supporte les lignes orphelines, c'est-à-dire les lignes ayant une valeur nulle en tant que clé étrangère.

Cette limitation était importante car la table orders comprenait une clé étrangère vers la table shippers. Cette référence pourrait être nulle dans le cas où une commande n'a pas encore été assignée à un livreur (statut 1-en cours de traitement + pas de livreur attribué).
Pour résoudre ce problème, nous avons créé un livreur fictif appelé dummy qui permettait d'éviter le problème en l'assignant aux commandes n'ayant pas de livreurs.

2. Évaluation de l'existant

Il existe aujourd'hui de nombreux outils de génération de données synthétiques (Faker,Mimesis,Gretel.AI, SDV..), parmi ceux-ci, nous avons choisi les outils qui répondent au mieux à la problématique de génération de données synthétiques et donc anonymes: Gretel.Ai et SDV. Ces outils s'appuient principalement sur les GAN pour la génération de données.

Les Generative Adversarial Networks, ou GAN, sont une approche de la modélisation générative utilisant des méthodes de deep learning, telles que les réseaux de neurones convolutifs. [9]

La modélisation générative est une tâche d'apprentissage non supervisée en machine learning qui consiste à découvrir et à apprendre automatiquement les patterns dans les données d'entrée de telle sorte que le modèle puisse être utilisé pour générer ou produire de nouveaux exemples qui auraient plausiblement pu être tirés de l'ensemble de données d'origine.

Les GAN sont constitués d'un générateur et d'un discriminateur, chacun étant un réseau de neurones, qui adversaires. En effet, le générateur va essayer de générer de nouveaux exemples pour essayer de duper le discriminateur et le discriminateur va essayer de différencier les exemples tirés du jeu de données initial des données générées.

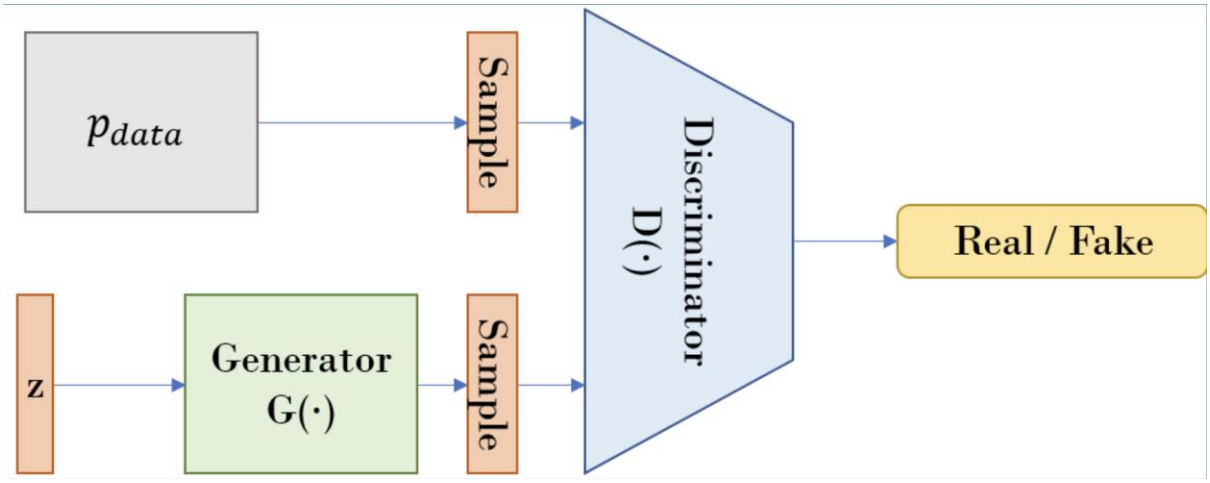

Figure 8 : Schématisation du fonctionnement des GAN

a) Gretel.ai [10]

Gretel.ai est une plateforme qui propose des services open-source, elle fournit une interface qui facilite le processus de génération de données en exigeant peu de connaissances techniques. Outre la création de données synthétiques, l'outil peut, d'une part, découvrir et étiqueter des types de données sensibles et, d'autre part, effectuer des transformations préservant l'anonymat. Bien que l'éventail des données en entrée possibles soit large, des formats simples tels que le CSV sont recommandés.

Après avoir téléchargé l'ensemble de données original, l'utilisateur peut choisir la configuration du modèle d'apprentissage automatique. Gretel utilise un réseau neuronal à mémoire à long court terme, qui permet une meilleure reproduction des données séquentielles, et dispose de modèles préconstruits qui peuvent être appliqués en fonction du contenu et de la structure du jeu de données.

Une caractéristique utile de l'outil est que les utilisateurs peuvent utiliser leurs propres configurations de modèle qui correspondent mieux à leurs besoins. Le processus automatique démarre une fois la configuration définie.

Les données sont profilées et regroupées et des statistiques au niveau du champ sont extraites pour être utilisées ultérieurement pour la validation des données générées.
Les données sont segmentées et vectorisées avant d'être traitées. Le modèle formé peut être utilisé pour générer des enregistrements de données synthétiques.

b) Synthetic Data Vault (SDV) [11]

Synthetic Data Vault. SDV est un autre outil open-source qui fournit plusieurs modèles pour synthétiser des données. Il s'agit d'un écosystème de bibliothèques de génération de données synthétiques qui permet aux utilisateurs d'apprendre facilement des dataset à tableau unique (Single Table), à tableaux multiples (Multi Table) et à séries temporelles (Time Series) pour générer ensuite de nouvelles données synthétiques ayant le même format et les mêmes propriétés statistiques que le dataset original.

Les données synthétiques peuvent ensuite être utilisées pour compléter, augmenter et, dans certains cas, remplacer les données réelles lors de la formation de modèles d'apprentissage automatique. En outre, elles permettent de tester des systèmes d'apprentissage automatique ou d'autres systèmes logiciels dépendant des données sans le risque d'exposition qui accompagne la divulgation des données.

La librairie SDV utilise plusieurs techniques de modélisation graphique probabiliste et de Deep Learning. Pour permettre une variété de structures de stockage de données, elle utilise des techniques uniques de modélisation générative hiérarchique et d'échantillonnage récursif.
SDV est complétée par Synthetic Data Metrics (SDMetrics, cf. Annexe et Glossaire) qui est une bibliothèque Python open source permettant d'évaluer des données synthétiques tabulaires. Utile pour comparer des données synthétiques à des données réelles en utilisant une variété de métriques et générer des rapports visuels pour analyser les données.

3. Solutions proposées

Étant donné la période très courte pour réaliser le projet, nous nous sommes concentrés sur l'approfondissement des solutions existantes plutôt que de créer notre propre solution de zéro. D'un commun accord avec notre tutrice, Mme. Dhouib, nous sommes partis sur l'évaluation de performance de la génération de données synthétiques et la comparaison des modèles multi-table et single-table proposés par Synthetic Data Vault (SDV).

a) Le modèle multi-table

Principe

Après une recherche documentaire extensive, la modélisation multi-table proposée par SDV semble être la solution la plus avancée pour la génération de données synthétiques.
Cette modélisation se base sur l'utilisation de métadonnées décrivant les propriétés des données et les relations entre les tables. Ces métadonnées contiennent les informations sur chaque table : leur nom, leur clé primaire, et l'ensemble de leurs champs. Elles contiennent aussi les relations entre les différentes tables et sur quelles clés étrangères ces relations se basent.

La difficulté principale dans la modélisation multi-table est justement d'assurer le maintien de la cohérence et les liens inter-tables. Pour cela, SVD propose d'utiliser le *Hierarchical Modeling Algorithm* (ou HMA). HMA est un algorithme qui permet de parcourir récursivement un jeu de données relationnelles et d'appliquer des modèles tabulaires à toutes les tables de manière à ce que les modèles apprennent comment tous les champs de toutes les tables sont liés. Le modèle tabulaire par défaut appliqué dans le cadre de l'utilisation de HMA est la Copule Gaussienne (ou Gaussian Copula en anglais). C'est un modèle statistique et probabiliste qui utilise une fonction de distribution cumulative (multivariable). Ce modèle permet de décrire la distribution conjointe de l'ensemble des variables contenues dans nos données (les différentes colonnes des tables). Ainsi, ce modèle nous permet de générer des données possédant les mêmes caractéristiques statistiques que les données initiales.

Application

Nous avons appliqué ce modèle multi-table utilisant la combinaison de HMA et de copules gaussiennes à nos données afin d'évaluer la performance du modèle. Cela consistait donc en un entraînement du modèle avec HMA puis une génération de données synthétiques à partir de ce modèle.

Tout de suite, nous avons remarqué des incohérences dans les données générées qui n'étaient pas présentes dans les données initiales. Nous retrouvions notamment toutes les incohérences que nous avions dû résoudre lors de la création du jeu de données initial. Il s'agissait, par exemple, de dates de commande qui ont lieu après la date d'envoi ou une commande ayant un statut « envoyé » ou « livré » alors qu'il n'y avait pas de date d'envoi.

Ainsi, nous nous sommes intéressés aux contraintes proposées par SDV. En effet, ces contraintes nous permettent, en théorie, de décrire certaines relations entre les données qui

ne doivent pas changer. Dans notre cas, il s'agissait de dire que la date d'envoi de la commande doit être supérieure (plus récente) que la date de commande, par exemple.

En principe, ces contraintes sont une super idée, il est même possible de créer des contraintes personnalisées afin de définir des liens auxquels les auteurs de SDV n'auraient pas pensé. Cependant, les contraintes ne sont pas toujours applicables et causent parfois des bugs lors de l'exécution du code. En effet, SDV étant encore en cours de développement avec des changements réguliers à son fonctionnement, il y a souvent des problèmes inattendus. Dans notre cas, nous nous sommes limités à des contraintes très simples puisque les autres contraintes nous prenaient beaucoup trop de temps à essayer de déboguer et nous empêchaient d'avancer dans notre projet.

b) Le modèle single-table

Principe

Après avoir remarqué les incohérences obtenues dans les données générées avec le modèle multi-table, nous avons eu l'idée de regrouper l'ensemble des données dans une seule table. Ainsi, nous étions capables d'appliquer les modèles single-table proposés par SDV à nos données. Ceci était dans le but d'essayer de pallier l'incohérence inter-table produite avec le modèle relationnel.

Application

Pour cela nous avons utilisé le modèle CTGAN (application tabulaire du GAN) qui présente les meilleurs résultats d'après les tests réalisés par SDV. En revanche, nous remarquons de nouveau certaines incohérences avec les données mais aussi un temps d'entraînement du modèle largement supérieur. En effet, l'entraînement prenait environ 54 min pour la moitié des données (5 000 lignes) contre seulement 1 min pour le modèle multi-table avec la totalité des données (IDE : VScode, OS : MacOS, puce M1pro, 16Go de mémoire unifiée). Notons donc que même si la modélisation single-table propose une génération de données de meilleure qualité, il faudra évaluer le trade-off entre le temps d'entraînement et la qualité de données.

De la même manière que pour le modèle multi-table, nous étions capables d'ajouter des contraintes à la génération de données afin d'assurer une certaine cohérence. L'application de ces contraintes est une forme de *data conditioning*.

Nous avons également appliqué d'autres formes de data conditioning dans les deux modèles afin d'insister sur l'anonymisation des données. En effet, SDV propose un paramètre de modèle *anonymize_fields.* Ce paramètre permet de définir certains champs dont les valeurs ne doivent pas se retrouver dans les données générées. Cette fonctionnalité utilise la bibliothèque *Faker* de python qui permet de générer aléatoirement des données PII selon le type de données dont il s'agit. Par exemple, nous pouvons générer que des prénoms, que des noms, ou des couples prénom/nom au choix.

4. Résultats obtenus

Ensuite, notre mission était d'évaluer la qualité des données générées par rapport aux données initiales. Ainsi, nous serions capables de dire quel modèle est le plus performant entre les modèles single-table et multi-table.

a) Modèle multi-table

Diagnostic Report – Column Coverage, Column Boundaries

Le rapport de diagnostic effectue quelques vérifications de base sur les données synthétiques pour donner une idée générale des forces et des faiblesses du modèle de données synthétiques. On les utilise pour faire la validation des données synthétiques.

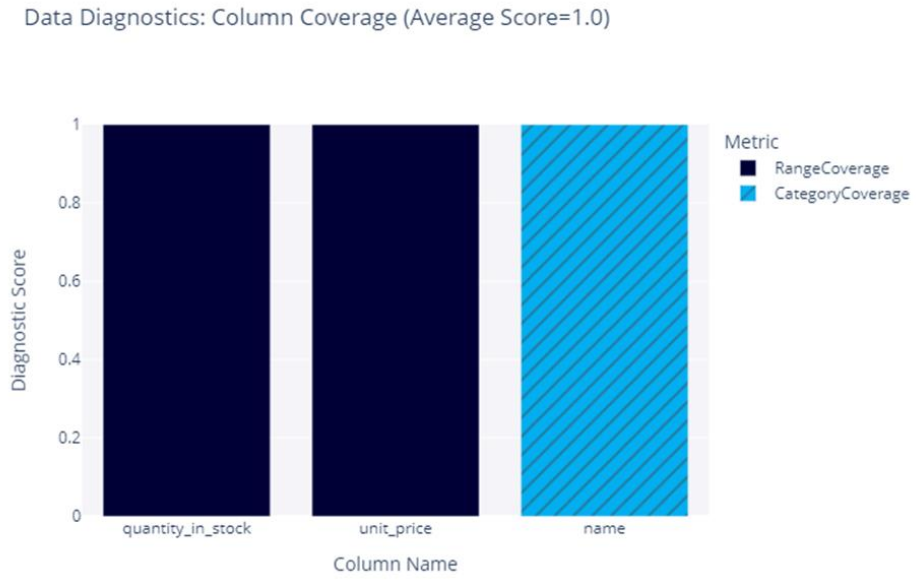

Figure 9 : Couverture des données (modèle multi-table) – Column Coverage

Dans le rapport de diagnostic des données, on peut observer le score de couverture des données colonnes de chaque table pour le modèle multi-table. On a obtenu un score de 1, ce qui signifie que les données synthétiques couvrent toute la gamme des valeurs possibles.

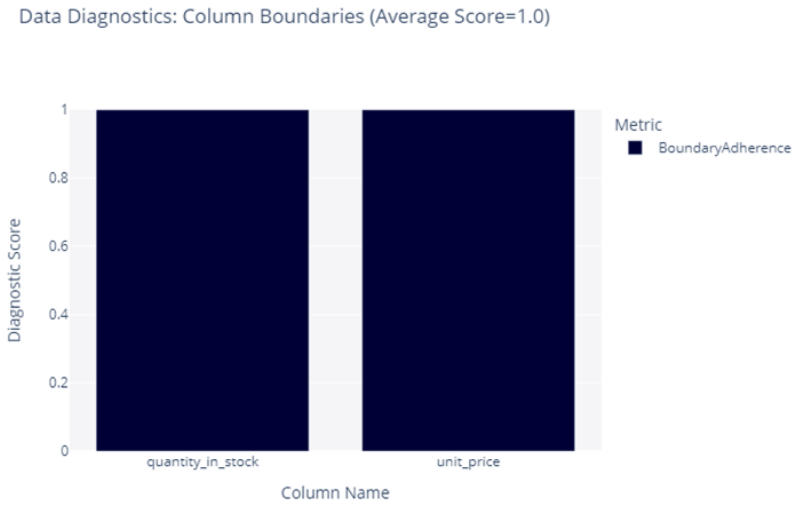

Figure 10 : Respect des limites de données (modèle multi-table) – Column Boundaries

Ici, on peut observer le score de limitations des données des colonnes de chaque table pour le modèle multi-table. On a obtenu un score de 1, ce qui signifie que les données synthétiques respectent les limites de valeurs fixées par les données réelles. Il ne concerne que les données de type « numériques » (c'est pour cela que la colonne ***name*** ne figure pas ici).

Quality Report – Column Shapes et Column Pair Trends

Le rapport de qualité SDMetrics évalue dans quelle mesure les données synthétiques capturent les propriétés mathématiques des données réelles. C'est ce qu'on appelle la fidélité des données synthétiques. Le rapport exécute certaines métriques pour mesurer ces propriétés et résume les résultats.

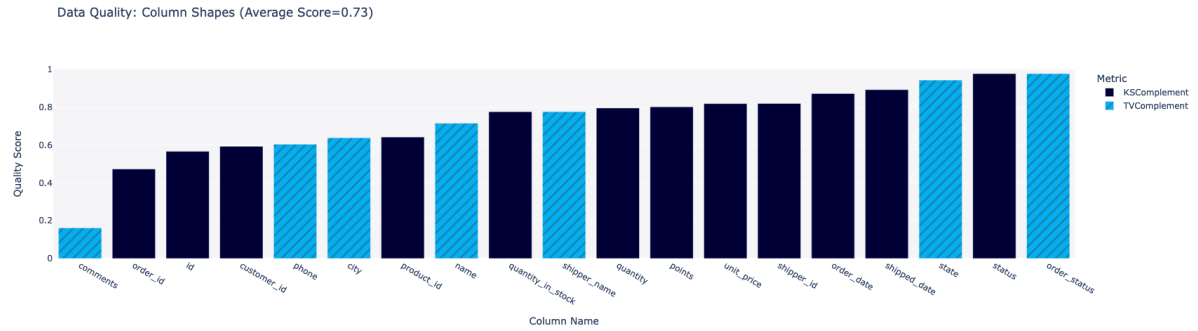

Figure 11: Qualité des données pour les champs générés (modèle multi-table) – Column Shapes

Nous remarquons que le score le plus bas est attribué aux commentaires. Cela indiquerait que le modèle a du mal à gérer le traitement de langage naturel (NLP). En effet, dans les données générées, un seul commentaire se retrouve associé à toutes les commandes où le commentaire n'est pas nul. Globalement, on obtient un score de 0.73 donc la distribution est correcte dans l'ensemble mais serait à améliorer.

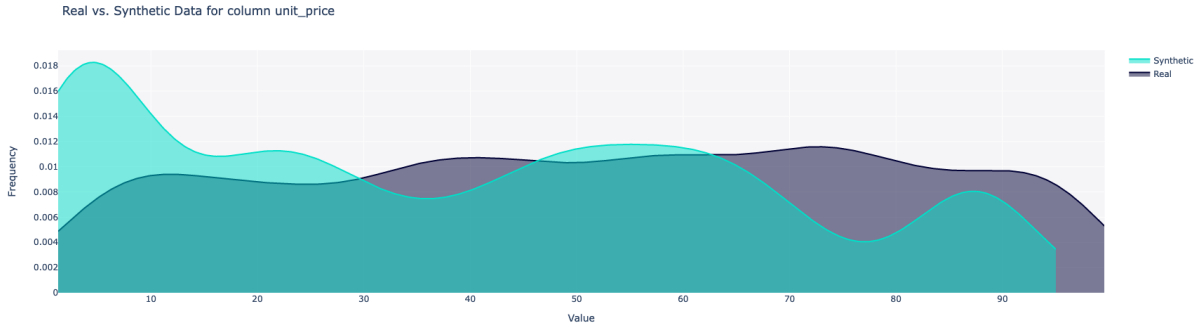

Figure 12 : Distribution comparée du prix unitaire entre les données initiales et les données générées (modèle multi-table)

Nous voyons sur cet exemple que la distribution des données synthétiques est sensiblement différente que celle des données réelles. Ceci indique que le modèle a du mal à respecter la distribution des valeurs numériques lors de la génération de nouvelles données. Beaucoup d'enregistrements se retrouvent concentrés dans les plus petites valeurs.

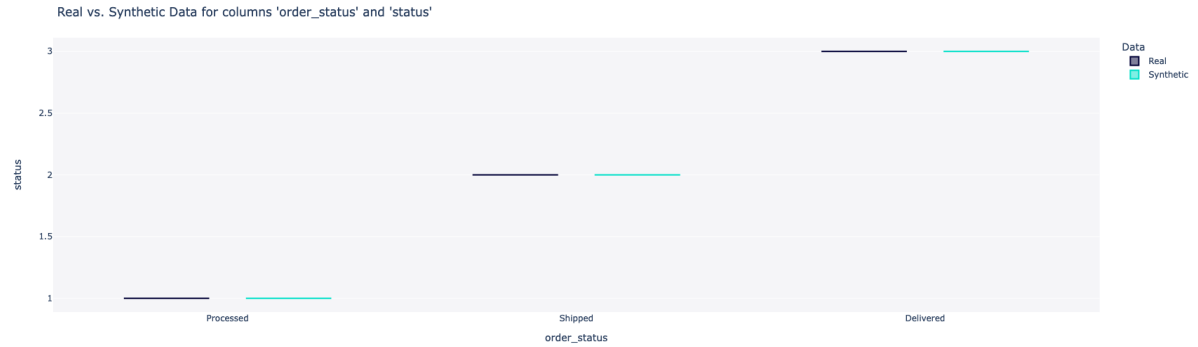

Figure 13 : Similarité entre les données générées et les données initiales (modèle multi-table)

Dans ce graphique, nous voyons que les liens entre les tables sont bien maintenus. En effet, il n'y a pas de variation entre les données réelles et les données synthétiques, et nous savons que nous avons les couples suivants : (1-Processed), (2-Shipped), (3-Delivered). Nous savons donc que la clé étrangère *status* de la table *orders* fait référence au bon élément (*order_status)* de la table *order_statuses*.

Nous n'avons pas mis de contraintes entre ces deux champs puisqu'ils sont dans deux tables différentes (ce n'est donc pas possible) et c'est l'algorithme HMA qui se charge de faire le bon lien.

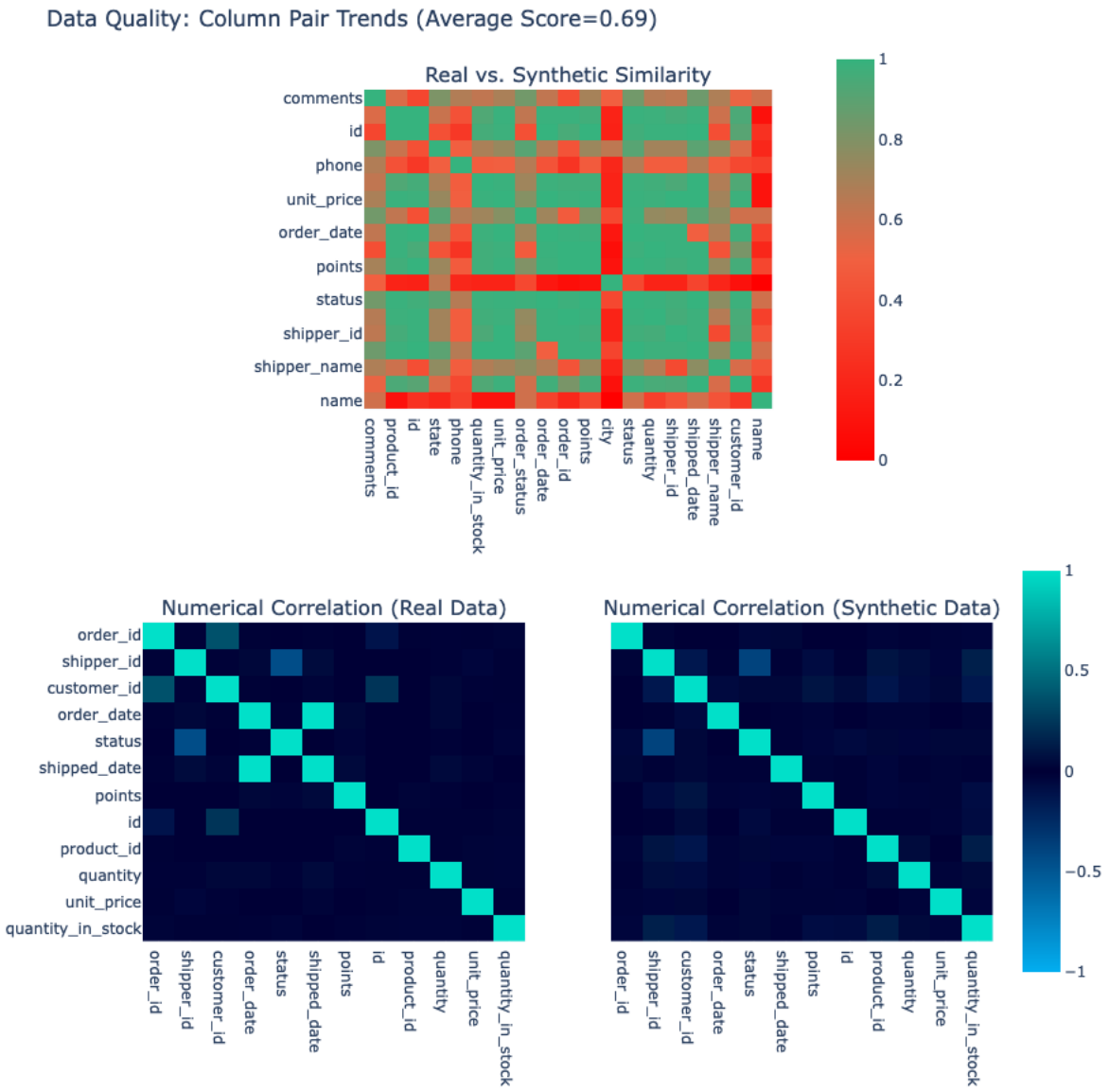

Figure 14 : Matrices de similarité et de corrélation entre les champs (modèle multi-table) – Column Pair Trends

Il est intéressant de remarquer que la corrélation entre *shipper_id* et *status* se conserve dans les données synthétiques. En effet, nous n'avons pas spécifié de contraintes spécifiques entre ces deux champs mais il y a bien une certaine corrélation : Si le statut de la commande est de 1, soit *Processed*, le livreur n'est généralement pas attribué et la commande se voit attribué le livreur *Dummy*, comme précisé dans la partie sur la mise en place des données initiales.

Il semblerait que malgré les contraintes appliquées, les autres corrélations ne se conservent que très peu. Par exemple, comme nous l'avions spécifié précédemment, la date d'envoi doit

être postérieur à la date de commande mais doit rester relativement proche. Nous voyons que cette relation n'est pas conservée dans les données générées.

b) Modèle single-table

Diagnostic Report – Column Coverage, Column Boundaries

De même que pour le modèle multi-table, il est possible de générer un rapport de diagnostic des données en modèle single-table.

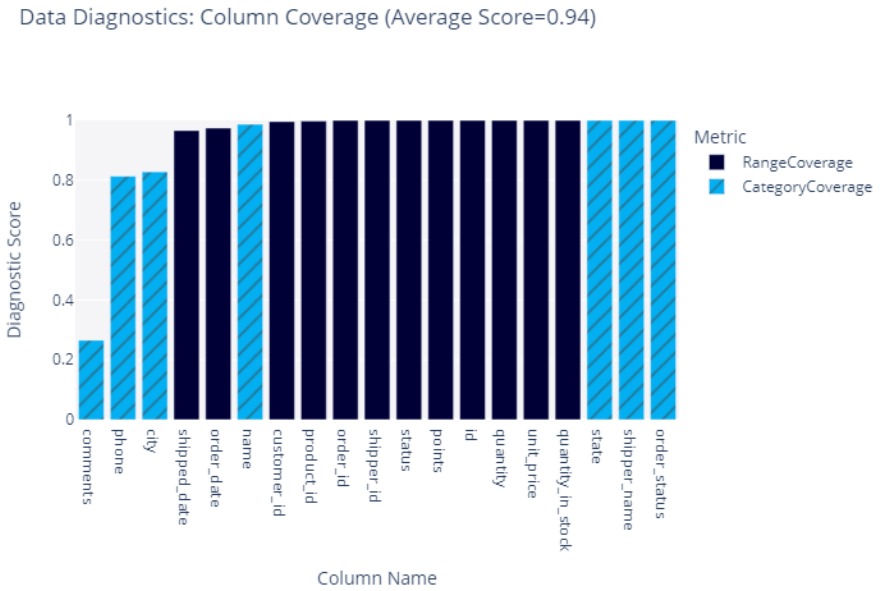

Figure 15 : Couverture des données (modèle single-table) – Column Coverage

Comme pour le modèle multi-table, on peut obtenir les scores de Column Coverage pour chacune des colonnes de la table. On peut voir que les résultats obtenus pour les colonnes ***quantity***, ***unit_price*** et ***name*** que les résultats sont quasiment identiques à ceux obtenus dans le modèle multi-table. On note une légère baisse pour la colonne ***name*** (légèrement inférieure à 1).

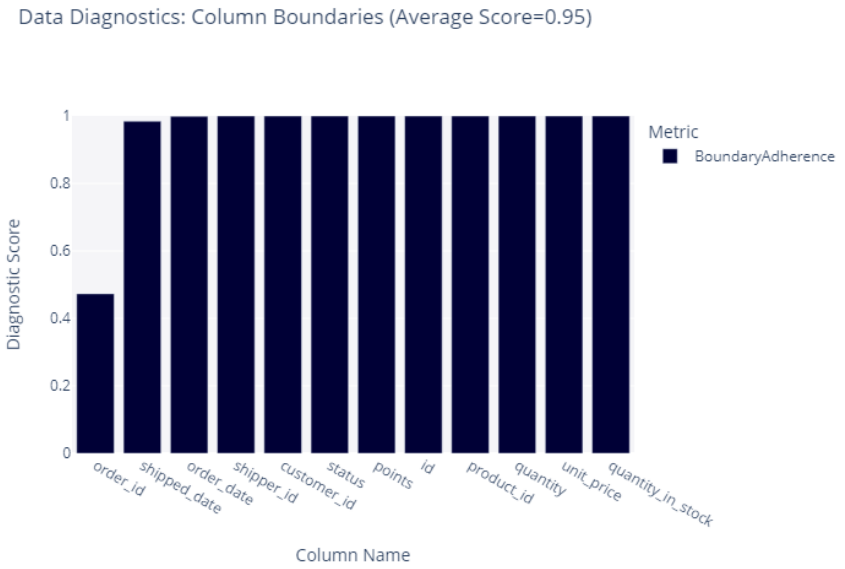

Figure 16 : Respect des limites de données (modèle single-table) – Column Boundaries

On peut voir que les résultats obtenus pour les colonnes ***quantity*** et ***unit_price*** que les résultats sont identiques à ceux obtenus dans le modèle multi-table : score = 1.

Quality Report – Column Shapes et Column Pair Trends

Le rapport de qualité SDMetrics est également généré pour les modèles Single-Table. Nous retrouvons les mêmes propriétés : Column Shapes & Column Pair Trends.

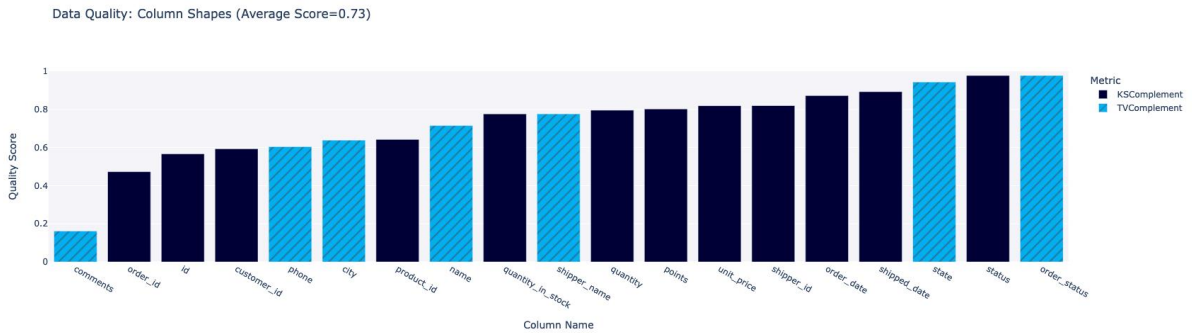

Figure 17 : Qualité des données pour les champs générés (modèle single-table)

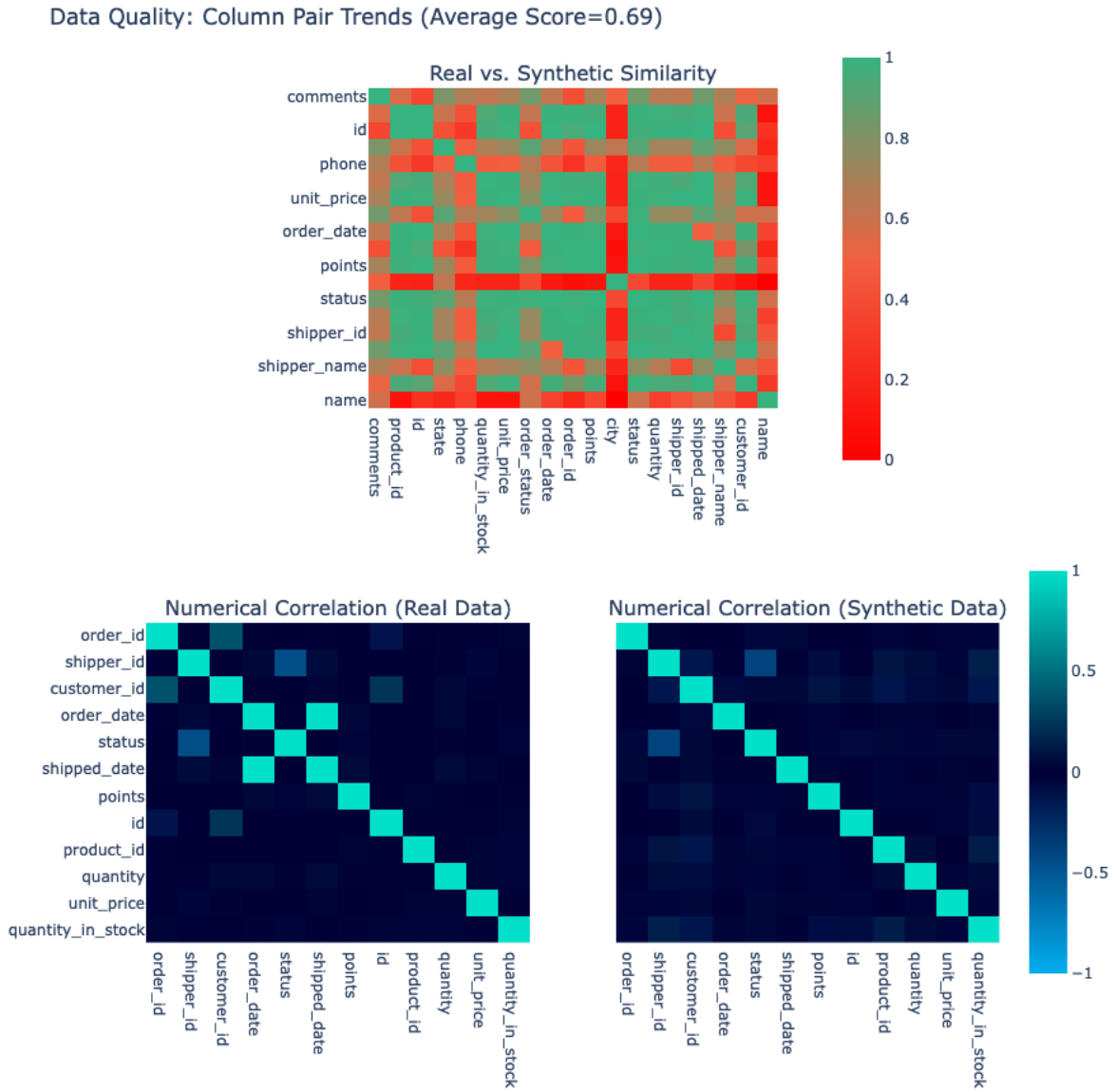

Figure 18 : Matrices de similarité et de corrélation entre les champs (modèle single-table)

Nous voyons des résultats très similaires au modèle multi-table. Nous pourrions même dire qu'ils sont indiscernables à vue d'œil. Il semblerait donc que le regroupement des tables en une seule ne soit pas utile pour le maintien des corrélations entre les différents champs. Nous sommes capables de maintenir la corrélation entre le *shipper_id* et le *status* mais il a fallu définir une contrainte dont on n'avait pas besoin pour le modèle multi-table.

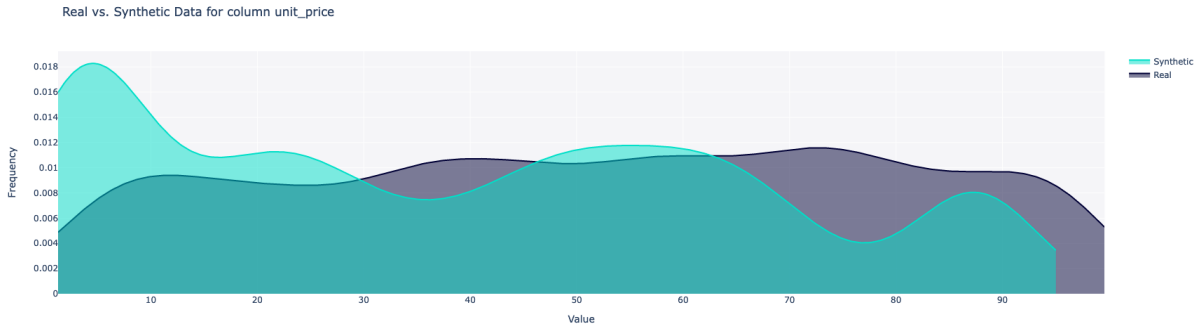

Figure 19 : Distribution comparée du prix unitaire entre les données initiales et les données générées (modèle single-table)

De la même manière que pour la figure précédente, il n'y a que très peu voire pas de changement par rapport au modèle multi-table. Il semblerait donc que le regroupement des tables ne soit pas non plus utile pour le maintien d'une distribution proche de celle des données initiales.

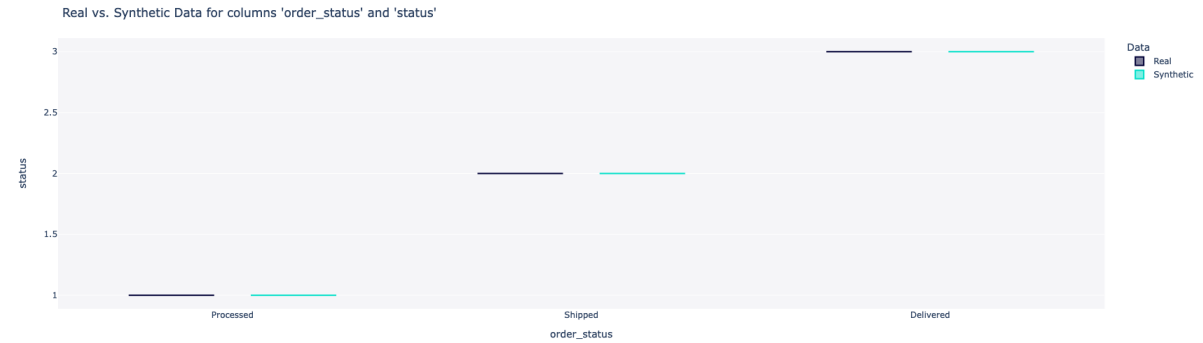

Figure 20 : Similarité entre les données générées et les données initiales (modèle single-table avec contraintes)

Cette fois-ci, nous voyons la même chose que pour le modèle multi-table. Toutefois, il a été nécessaire de spécifier une contrainte qui assurait les couples 1-Processed, 2-Shipped, 3-Delivered entre *status* et *order_status*.

En effet, en regardant le graphique suivant, nous voyons l'importance de cette contrainte pour assurer la cohérence entre les données.

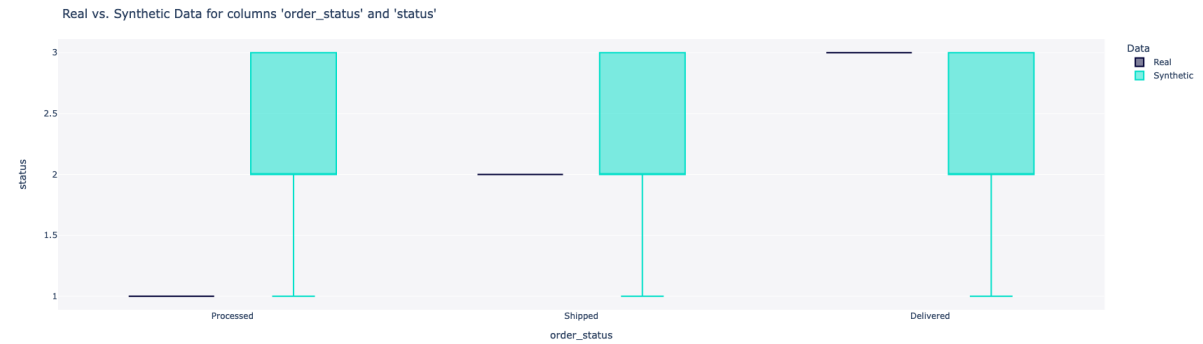

Figure 21 : Similarité entre les données générées et les données initiales (modèle single-table sans contraintes)

Les couples cités précédemment ne sont plus respectés et les données générées perdent donc leur cohérence. Nous sommes, encore une fois, obligés de définir une contrainte pour assurer un lien qui est fait automatiquement par l'algorithme HMA et nous ne gagnons même pas en qualité sur le reste des données générées.

CONCLUSIONS ET PERSPECTIVES

Conclusions

Malgré un temps très court pour réaliser un tel projet de recherche, nous avons été capable de produire une comparaison de performances entre une approche de génération de données relationnelles en multi-table et une regroupée en single-table.
La comparaison des modèles multi-table et single-table montrent que les deux options sont plus ou moins équivalentes quant à la qualité des données générées. Toutefois, comme nous l'avons dit dans les solutions proposées, le modèle single table met presque 1h (54min) pour s'entraîner contre moins d'une minute (58s précisément) pour le modèle multi-table avec le même environnement.

Nous estimons donc que la meilleure stratégie à adopter serait d'améliorer les modèles appliqués aux différentes tables (actuellement des copules gaussiennes) mais de continuer avec l'utilisation de HMA. En effet, comme nous l'avons vu, HMA maintient un référencement cohérent entre les différentes tables.

Perspectives

Nous avons imaginé des évolutions possibles de notre projet qui permettrait d'obtenir des données synthétiques de meilleure qualité. En effet, il s'agirait dans un premier temps de chercher à appliquer l'algorithme HMA en utilisant des modèles de CTGAN plutôt que la copule gaussienne. En effet, cela n'est pas encore possible avec ce que propose SDV actuellement.

Dans un second temps, nous avons imaginé appliquer plus généralement le principe de WGAN aux CTGAN actuellement utilisés. Cela consiste à mettre à jour les paramètres du discriminateur du GAN à fréquence plus régulière que ceux du générateur. Ainsi, le discriminateur devient ce qui est appelé un *Critic*. En effet, cela a été mis en avant avec le principe de TabFairGAN [12] et démontre que ces modèles génèrent des données de meilleure qualité.

Une autre approche qui pourrait être adoptée serait l'utilisation du reinforcement learning (apprentissage par renforcement). L'apprentissage par renforcement consiste à prendre des décisions de manière séquentielle. En termes simples, nous pouvons dire que la sortie dépend de l'état de l'entrée actuelle et l'entrée suivante dépend de la sortie de l'entrée précédente.

Appliquer ce principe reviendrait à :
- Indiquer au modèle quand les lignes générées ne sont pas valides
- Réentraîner le modèle afin qu'il apprenne les bonnes relations

BIBLIOGRAPHIE

[1] T. rédac, « Deep Learning ou Apprentissage Profond : qu'est-ce que c'est ? », *Formation Data Science | DataScientest.com*, 28 septembre 2020. https://datascientest.com/deep-learning-definition

[2] « Accenture France | Stratégie, Digital, Conseil, Technologie et Opérations ». https://www.accenture.com/fr-fr?c=acn_glb_brandexpressiongoogle_12728073&n=psgs_1221&gclid=Cj0KCQiAnNacBhDvARIsABnDa69VcdPf2zecAKcJSBlJst9ug3XMIaAgTtz2fp09VgvWRtXMWHL9mGkaAjqtEALw_wcB

[3] « Qu'est-ce que les données synthétiques et l'IA ? | Accenture ». https://www.accenture.com/fr-fr/insights/artificial-intelligence/synthetic-data-speed-security-scale

[4] « GanttProject - Free Project Management Application ». https://www.ganttproject.biz/

[5] « MySQL ». https://www.mysql.com/

[6] « Google Colaboratory ». https://colab.research.google.com/

[7] « Log In | Microsoft Teams ». https://www.microsoft.com/en-us/microsoft-teams/log-in

[8] « Dummy data for MYSQL database », *Fill Database*. https://filldb.info/

[9] I. J. Goodfellow *et al.*, « Generative Adversarial Networks ». arXiv, 10 juin 2014. doi: 10.48550/arXiv.1406.2661.

[10] « Gretel.ai - The Developer Stack for Synthetic Data ». https://gretel.ai/

[11] « The Synthetic Data Vault. Put synthetic data to work! » https://sdv.dev/

[12] A. Rajabi et O. O. Garibay, « TabFairGAN: Fair Tabular Data Generation with Generative Adversarial Networks ». arXiv, 1 septembre 2021. Disponible sur: http://arxiv.org/abs/2109.00666

Table des Figures