UNIVERSITÉ DU QUÉBEC À MONTRÉAL

GÉNÉRATION DE DONNÉES SYNTHÉTIQUES RESPECTUEUSES DE LA VIE PRIVÉE PAR UNE APPROCHE BASÉE SUR LES COPULES

MÉMOIRE
PRÉSENTÉ
COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN INFORMATIQUE

PAR
ALEXANDRE ROY-GAUMOND

AVRIL 2021

REMERCIEMENTS

Mes premiers remerciements vont à mon directeur de recherche Sébastien Gambs qui a su me guider sans me contraindre, me corriger sans me juger et me pousser dans mes recherches sans me démotiver. Merci aussi pour le climat amical, chaleureux et uni que tu as réussi à instaurer au sein de l'équipe de recherche qui a fait des heures passées au laboratoire du LATECE toujours un peu plus tolérable. Merci à Louis Béziaud qui a toujours su avoir le bon moment pour aller chercher un café. Merci à Antoine Laurent pour les nombreuses bières et aux nombreux samossas qui m'ont chacun permis d'avancer plus loin dans ma recherche, à leur manière. Merci à Catherine et Martin d'avoir été flexibles durant les moments les plus exigeants de ma maîtrise et d'avoir bâti l'environnement rassembleur et sécuritaire qu'est le Yïsst, parfait pour les étudiants et chercheurs sensibles. Un merci tout particulier au SCLOB de m'avoir soutenu autant dans ma procrastination que dans mes élans de travail.

Finalement, je remercie les évaluateurs pour leurs nombreux commentaires et leur attention aux détails. Vos critiques ont su renforcer toutes les parties du mémoire.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

RÉSUMÉ

La publication et la libération de données sont de plus en plus populaires ce qui rend accessibles des données potentiellement identifiantes pour les individus auprès desquels ces données ont été collectées. De plus, la multiplicité des données disponibles sur le Web facilite le croisement des données et augmente donc les possibilités d'attaque. Libérer des données permet toutefois aux chercheurs académiques et industriels de faire des nouvelles découvertes autrement inaccessibles. Le dilemme entre la publication des données et le respect de la vie privée est certes complexe, mais différentes techniques d'assainissement et d'anonymisation de données permettent de trouver un équilibre.

Les modèles génératifs respectueux de la vie privée permettent de produire des données fidèles aux données originelles tout en mitigeant le risque de fuites d'informations personnelles. Un récent type de modèles génératifs pour le domaine de la vie privée est particulièrement prometteur : les copules.

Les copules ont la qualité d'être des modèles interprétables et robustes et leur extension, les copules vignes, permettent la modélisation de données synthétiques de dimension arbitraire. L'application de la confidentialité différentielle est simple et garantie une protection individuelle, donc des modèles *et* des données respectueuses de la vie privée.

Ce mémoire présente une nouvelle approche nommée COPULA-SHIRLEY basée sur les copules vignes permettant la génération de données synthétiques différentiellement privées. COPULA-SHIRLEY se base sur les fonctions de densités marginales bruitées pour construire une copule vigne à l'aide de l'algorithme de Dissmann. Le cadre d'utilisation de COPULA-SHIRLEY est simple, flexible, respectueux de la vie privée et peut être appliqué à tout type de données. Cette nouvelle approche est accompagnée de deux tests statistiques, trois tâches de classification et un test de protection.

MOTS-CLÉS : Copule, copule vigne, modèle génératif, données synthétiques, vie privée, test de protection de la vie privée, test d'utilité, modèle statistique, confidentialité différentielle.

INTRODUCTION

Les portails de données se font de plus en plus nombreux et leurs collections de bases de données semblent être en constante croissance. Au moment de l'écriture, un important acteur parmi les portails de données ouvertes, Kaggle [1], comptait plus de 78 715 jeux de données pour l'apprentissage machine et de l'analyse de données. Un autre portail bien connu, le portail de l'Université de Californie à Irvine (UCI) [2], compte environ 585 ensembles de données pouvant être utilisés pour différentes tâches de l'apprentissage machine. Les données ouvertes sont aussi devenues très populaires auprès des gouvernements de différents pays. Il est possible de trouver sur le portail de données ouvertes du Canada [3] 85 373 jeux de données de différentes sources, allant de recensements auprès de la population aux listes de produits d'agriculture exportés. Aux États-Unis, 280 519 bases de données sont disponibles en accès libre [4]. Même les provinces et les villes n'y échappent pas. La province du Québec met à la disposition 1 205 ensembles de données [5], la ville de Montréal plus de 300 ensembles [6] et la ville de Toronto, 412 [7]. Rappelons-le, ces nombres sont en constante augmentation.

---

1. `https://www.kaggle.com/datasets`
2. `https://archive.ics.uci.edu/ml/datasets.php`
3. `https://open.canada.ca/en/open-data`
4. `https://www.data.gov/`
5. `https://www.donneesquebec.ca/recherche/fr/dataset`
6. `https://donnees.montreal.ca/`
7. `https://open.toronto.ca/`

La publication de données joue un rôle essentiel dans la recherche et permet de mieux comprendre les problèmes sociétaux (Costello, 2009). Un des problèmes actuels dus à cette accessibilité est celui de la vie privée qui survient lorsque les données concernent des individus. Ce dilemme entre données ouvertes et vie privée a fait l'objet de maintes analyses et concours. En effet, plusieurs institutions et organisations se sont penchées sur ce problème, par exemple le National Institute of Standards and Technology (NIST) aux États-Unis a récemment mis au point deux concours portant sur l'anonymat dans les données : le *Unlinkable Data Challenge (2018)* [8], où il était demandé de développer un mécanisme d'anonymisation de données et le *Differential Privacy Synthetic Data Challenge (2018)* [9], durant lequel les participants devaient concevoir des méthodes de génération de données synthétiques. Au Japon, depuis 2015, se tient un atelier annuel nommé PWS-Cup [10] promouvant la sécurité de la vie privée dans les données. À titre d'exemple, à l'automne 2019, le concours tenu par PWSCup reposait sur l'anonymisation de données de mobilité d'individus. Dans le cadre du concours, les participants ont dû concevoir non seulement une méthode d'anonymisation de trajectoires spatio-temporelles, mais aussi des techniques pour attaquer les données anonymisées. Ces concours offrent certes une vue d'ensemble de l'état-de-l'art mais aussi un bon aperçu des nombreux problèmes dans le domaine de la protection de la vie privée. Il n'y aurait manifestement pas de compétitions de ce type s'il n'y avait pas d'enjeux.

L'utilisation de données personnelles sans les précautions nécessaires est problé-

---

8. `https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/past-prize-challenges/2018-unlinkable-data-challenge`

9. `https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/past-prize-challenges/2018-differential-privacy-synthetic`

10. `https://www.iwsec.org/pws/2019/index_e.html`

matique. Un éveil collectif concernant la vie privée et les données personnelles a été provoqué par les récentes fuites de données. Par exemple, les fuites de données d'Equifax (Ligaya, 2017), de Facebook et du scandale de Cambridge Analytica (Cadwalladr et Graham-Harrison, 2018) et de Desjardins (Desjardins, 2019) ont suscité des craintes auprès de leurs utilisateurs. Bien que les données publiées en ligne ont souvent été soigneusement traitées pour mitiger tous risques de divulgation d'informations personnelles, les dangers sont inhérents au caractère personnel des données, comme le démontrent les bris de vie privée qu'ont eu AOL (Barbaro *et al.*, 2006), Netflix (Singel, 2010) et la ville de New York (Tockar, 2014) en libérant leurs données. Il a été démontré que les données collectées auprès d'individus sont la plupart du temps uniques et identifiables, et ce indépendamment de la taille de la population considérée. Dans l'article (De Montjoye *et al.*, 2013) étudiant les données de mobilité, les auteurs démontrent qu'un individu n'est unique dans une population qu'à partir de seulement 3 à 5 lieux d'intérêts (c'est-à-dire des lieux fréquemment visités). Les auteurs de l'article (De Montjoye *et al.*, 2015) arrivent à une conclusion similaire en utilisant des relevés de transactions de cartes de crédit : toute personne est unique à plus ou moins 5 transactions près. Les notes (« ratings ») laissées sur des plateformes de diffusion en continu comme Netflix peuvent aussi être identifiables, comme le démontrent les résultats de l'article (Narayanan et Shmatikov, 2008). Sans oublier l'exemple classique de Latanya Sweeney dans son article (Sweeney, 2000) qui évalue que la connaissance de 3 attributs démographiques est suffisante pour identifier de manière unique la grande majorité de la population des États-Unis. D'un autre côté, les méthodes de protection et d'assainissement de données sont nombreuses, chacune perturbant les données à sa façon pour offrir le meilleur compromis entre protection et utilité.

Depuis une dizaine d'années, les techniques d'anonymisation de données ont vu grandir leurs rangs avec les modèles de génération de données synthétiques. Le

point crucial des méthodes génératives est qu'elles puissent complètement briser le lien entre identité et donnée en condensant l'information à travers un modèle statistique. Cette rupture n'est toutefois pas suffisante pour assurer une protection complète des données. Effectivement, les modèles entraînés sur des données brutes sans perturbation sont sujets à certaines attaques (Shokri *et al.*, 2017). Il semble donc que l'utilisation de modèles génératifs entraînés de manière respectueuse de la vie privée, c'est-à-dire dont l'apprentissage est perturbé afin de mitiger la contribution individuelle des individus, résolve un bon nombre d'enjeux amenés par la publication des données.

C'est de ce point que traite le présent mémoire, c'est-à-dire la génération de données respectueuses de la vie privée. Autrement dit, ce mémoire s'intéresse à la question scientifique suivante :

> *Comment peut-on produire des données synthétiques respectueuses de la vie privée tout en gardant un maximum d'utilité par rapport aux données brutes ?*

L'approche proposée repose sur deux éléments. Le premier est la confidentialité différentielle (Dwork *et al.*, 2014) (« Differential Privacy » en anglais). La confidentialité différentielle, qui sera définie dans le chapitre 2, est un modèle de vie privée de partage d'information assurant la protection individuelle. À ce jour, la confidentialité différentielle est devenue le modèle de référence pour la protection de la vie privée des individus dans un système de partage d'information, à tel point qu'elle est utilisée par les acteurs technologiques tels que Amazon [11], Apple [12] et Google [13]. Le deuxième point de la méthode proposée est les copules

---

11. `https://blog.aboutamazon.com/amazon-ai/protecting-data-privacy`

12. `https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf`

13. `https://github.com/google/differential-privacy`

(« copulas » en anglais). Ces modèles mathématiques, connus depuis très longtemps dans le domaine de l'actuariat, ont récemment commencé à être utilisés dans le domaine de la vie privée et connaissent un essor important. Les copules sont des outils de modélisation robustes et simples, contrairement aux approches de type apprentissage profond qui sont beaucoup plus complexes.

La méthode développée, nommée COPULA-SHIRLEY pour *COPULA-based generation of SyntHetIc diffeRentialL(E)Y-private data*, modélise les données entrantes à l'aide de copules en respectant un modèle de confidentialité différentielle. De manière similaire à une approche basée sur les réseaux bayésiens, COPULA-SHIRLEY construit un ensemble d'arbres où chaque noeud représente une variable et les arêtes entre les noeuds représentent une copule modélisant les liens entre les deux variables des noeuds adjacents.

La force de l'approche proposée repose sur trois aspects : simple, flexible et privé.

- *Simple à utiliser et à déployer* - en raison de la nature des objets mathématiques sur lesquels l'approche est basée, qui sont principalement des fonctions de distribution.
- *Flexible* - en raison du cadre hautement personnalisable des copules et de leur capacité inhérente à modéliser des distributions complexes en les décomposant en une combinaison de copules.
- *Privée* - en raison du respect de la confidentialité différentielle du modèle génératif.

Le mémoire est divisé en cinq chapitres. Les deux premiers chapitres introduisent la théorie essentielle à la compréhension de l'approche proposée. Le premier porte sur les copules et les copules vignes avec toutes les notions connexes, le deuxième définit les différents modèles de vie privée, notamment la confidentialité diffé-

rentielle, et explicite les multiples attaques que ces modèles peuvent contrer. Le troisième chapitre offre une vue d'ensemble sur les techniques d'anonymisation et d'assainissement de données, y compris les différentes méthodes génératives. Le quatrième chapitre met en lumière la méthode proposée COPULA-SHIRLEY ainsi que son cadre d'évaluation mis en place pour juger de la qualité des données synthétiques produites. Le cinquième et dernier chapitre présente les résultats obtenus lors de l'évaluation des données produites à l'aide de l'implémentation de COPULA-SHIRLEY et compare notre approche à trois modèles génératifs : un modèle de référence basé sur des copules, un modèle génératif bayésien de pointe et un modèle naïf basé sur les histogrammes. Le mémoire se termine sur une conclusion qui résume brièvement les points forts de l'approche développée.

CHAPITRE I

INTRODUCTION AUX COPULES ET AUX COPULES VIGNES

Historiquement, les copules (« copulas » en anglais) datent de 1959 et ont été introduites par Abe Sklar dans son texte (Sklar, 1959). À cette époque, les copules n'étaient que des modèles statistiques théoriques. Depuis la fin des années 2000, les copules ont connu un essor important et ont été utilisées dans plusieurs domaines dont les sciences de l'atmosphère pour modéliser les précipitations (Bárdossy et Pegram, 2009), en médecine avec les tests de diagnostique (Hoyer et Kuss, 2015), en finance pour raffiner l'étude des séries temporelles financières (Patton, 2009), en sciences sociales pour la modélisation des différents usages de la technologie (Lazer *et al.*, 2009), en génétique pour l'étude de phénotypes (He *et al.*, 2012) et très récemment en protection de la vie privée pour modéliser le risque de ré-identification (Rocher *et al.*, 2019).

Les copules servent à « coupler » les distributions marginales d'une variable multivariée en une distribution jointe. Les vignes (« vines » ou encore « regular vines » en anglais) sont des modèles graphiques hiérarchiques qui décomposent le problème de modélisation d'une variable multidimensionnelle (> 2) en pairs de variables conditionnelles.

Dans les parties qui suivent, le terme *distribution* référera à la fonction de distribution cumulative (FDC) et le terme *densité* ou *fonction de densité* à la fonction

de densité de probabilité (FDP). Le terme *dépendance caudale* est utilisé en référence à l'étude des dépendances dans les queues des distributions ou, autrement dit, à l'étude des dépendances entre les valeurs extrêmes (« outliers »).

1.1 Copules

Une copule est une fonction de distribution dont les distributions marginales sont uniformes.

Définition 1.1 (Copule (Sklar, 1959))**.** *Une copule multivariée à n dimensions est une fonction de distribution cumulative* $C : [0,1]^d \to [0,1]$ *dont les distributions marginales sont uniformes.*

Les copules permettent de décrire la structure de dépendance de variables aléatoires, comme l'indique le théorème de Sklar.

Théorème 1.1 (Théorème de Sklar (Sklar, 1959))**.** *La variable aléatoire continue* $X = (x_1, x_2, ..., x_n)$ *a pour distribution jointe* F *et distributions marginales* $F_1, F_2, ..., F_n$ *si et seulement s'il existe une copule* C *unique décrivant la distribution jointe de* $U = (u_1, u_2, ..., u_n) = (F_1(x_1), F_2(x_2), ..., F_n(x_n))$*. C'est-à-dire que*

$$F(x_1, x_2, \ldots, x_n) = C\left(F_1(x_1), F_2(x_2), \ldots, F_n(x_n)\right) \tag{1.1}$$

Il est possible de différentier l'équation 1.1, en assumant que la variable aléatoire est continue et obtenir :

$$f(x_1, x_2, ..., x_n) = c(u_1, u_2, ..., u_n) \times \prod_{i=1}^{n} f_i(x_i) \tag{1.2}$$

où $u_i = F_i(x_i)$ et f, c, f_i sont respectivement les fonctions de densités de F, C, F_i.

Ainsi, les copules peuvent être utilisées pour décrire la fonction de densité jointe comme étant le produit des densités marginales et de la structure de dépendance définit par c, la fonction de densité de la copule.

Exemple 1.1. *Lorsque $n = 3$, la copule C est une distribution telle que $C(1, 1, u) = C(1, u, 1) = C(u, 1, 1) = u$ pour $u \in [0, 1]$.*

Exemple 1.2. *Lorsque C est la copule indépendante, c'est-à-dire que $C(u_1, u_2, ..., u_n) = \prod_{i=1}^{n} u_i$, alors de l'équation 1.1, nous obtenons :*

$$F(x_1, x_2, ..., x_n) = C\left(F_1(x_1), F_2(x_2), ..., F_n(x_n)\right) = \prod_{i=1}^{n} F_i(x_i).$$

Ce qui implique que la variable aléatoire $X = (x_1, x_2, ..., x_n)$ est indépendante.

Exemple 1.3. *Dans le cas où nous voulons estimer la structure de dépendance d'une variable aléatoire $X = (x_1, x_2, ..., x_n)$ à l'aide d'une copule gaussienne $\hat{c}$, nous avons que la densité*

$$\hat{c}_X(x_1, x_2, ..., x_n) = \frac{\exp\left(-\frac{1}{2}(X-\mu)^T \mathbf{\Sigma}^{-1}(X-\mu)\right)}{\sqrt{(2\pi)^n \det \mathbf{\Sigma}}},$$

où $\mathbf{\Sigma}$ est la matrice de covariance de la variable X et $\mu \in \mathbb{R}^n$ est le vecteur des moyennes μ_{x_i} pour $i \in [1, 2, ..., n]$. La fonction de densité

$$\hat{f}_X(x_1, x_2, ..., x_n) = \frac{\exp\left(-\frac{1}{2}(X-\mu)^T \mathbf{\Sigma}^{-1}(X-\mu)\right)}{\sqrt{(2\pi)^n \det \mathbf{\Sigma}}} \times \prod_{i=1}^{n} f_i(x_i)$$

correspond donc à une estimation de la fonction de densité jointe f_X à l'aide d'une densité gaussienne.

L'exemple 1.3 montre comment estimer la densité d'une copule à l'aide de modèles paramétriques appelés familles de copules (paramétriques). Généralement, une fois

la copule estimée, sa qualité d'ajustement (« goodness-of-fit ») est calculée pour vérifier la fidélité du modèle, à la manière qu'une distribution arbitraire est estimée à l'aide d'une distribution connue. La qualité de l'ajustement peut être calculée à l'aide du Critère d'Information d'Akaike (AIC) (Akaike, 1998), le Critère d'Information Bayésien (BIC) (Schwarz *et al.*, 1978) ou simplement la log-vraisemblance maximale (Fisher, 1992), pour ne nommer que quelques estimateurs. Il existe une multitude de familles de copules paramétriques (Joe, 1997; Nelsen, 2007), une partie provenant du domaine de la gestion des risques financiers et domaines connexes où la modélisation des dépendances caudales (« tail dependence ») est très importante (Patton, 2009; Tagasovska *et al.*, 2019). L'exemple suivant présente des simulations de quelques familles de copules paramétriques.

Exemple 1.4. *La figure 1.1 illustre cinq graphiques de points échantillonnés de différentes copules bivariées. Il est possible d'observer les différentes dépendances caudales réalisables à l'aide des copules.*

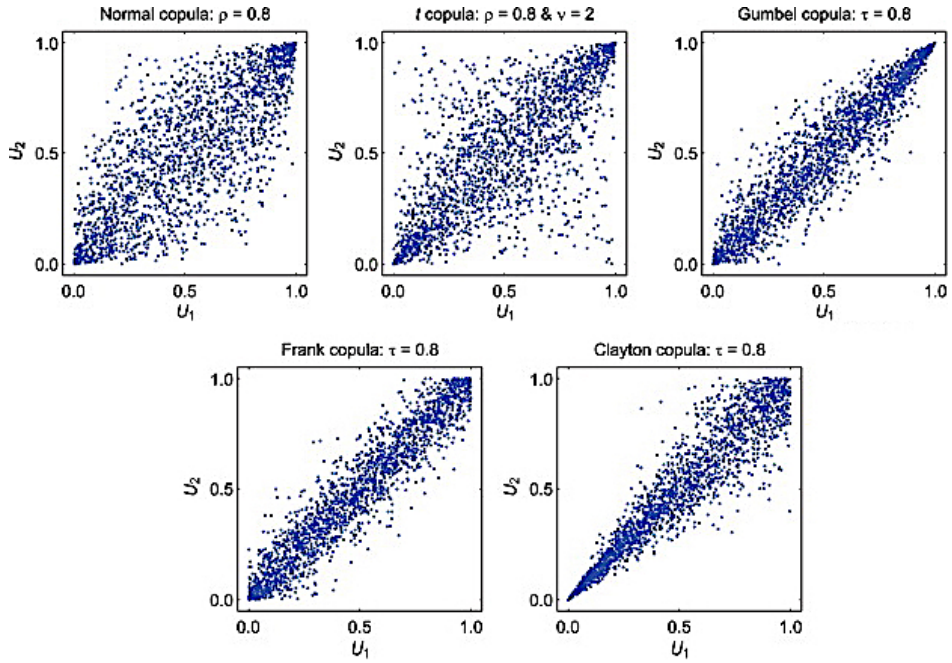


FIGURE 1.1 – Illustration des différentes familles de copules bivariées.

Il faut noter que si le théorème de Sklar est toujours vérifié dans le cadre discret, l'unicité de la copule C n'est garantie que dans le cas où X est une variable aléatoire *continue*. Or dans un cadre expérimental, l'unicité n'est pas un aspect important, la distribution jointe donnée par C étant estimée (de manière paramétrique ou non-paramétrique).

Les copules offrent donc la versatilité de pouvoir modéliser des densités jointes en fixant les densités marginales f_i et en variant la structure de dépendance c ou vice-versa. Étant donné une distribution jointe F (continue) et les distributions marginales F_i pour $i \in [1, 2, ..., n]$, la copule C correspondante est en fait la distribution jointe des transformées intégrales de probabilité des variables x_i.

Définition 1.2 (Transformée intégrale de probabilité (TIP) (Devroye, 1986))**.** *Soit X une variable aléatoire continue avec F_X comme distribution. La variable $Y = F_X(X)$ suit une distribution uniforme standard.*

En reprenant l'équation 1.1 :

$$F(x_1, x_2, ..., x_n) = C\left(F_1(x_1), F_2(x_2), ..., F_n(x_n)\right) = C(u_1, u_2, ..., u_n),$$

nous avons bien que C est une distribution $C : [0, 1]^d \to [0, 1]$ dont les distributions marginales sont uniformes.

Cette transformation est utilisée pour l'estimation de copules à partir de données expérimentales. Premièrement, les distributions marginales sont estimées à partir des observations ; ces distributions sont appelées fonctions de distribution cumulatives empiriques (FDCEs). Les FDCEs estimées sont ensuite utilisées pour obtenir les transformées intégrales de probabilités. Le terme pseudo-observations est généralement utilisé lorsqu'on parle de ces données transformées. La densité de la copule est alors estimée à l'aide des pseudo-observations. Cette estimation

peut être faite de manière non-paramétrique avec, par exemple, l'estimation par noyau (« kernel density estimation ») ou de manière paramétrique par le moyen de modèles classiques de copules. La figure 1.2, tirée de l'article (Tagasovska *et al.*, 2019), illustre le processus.

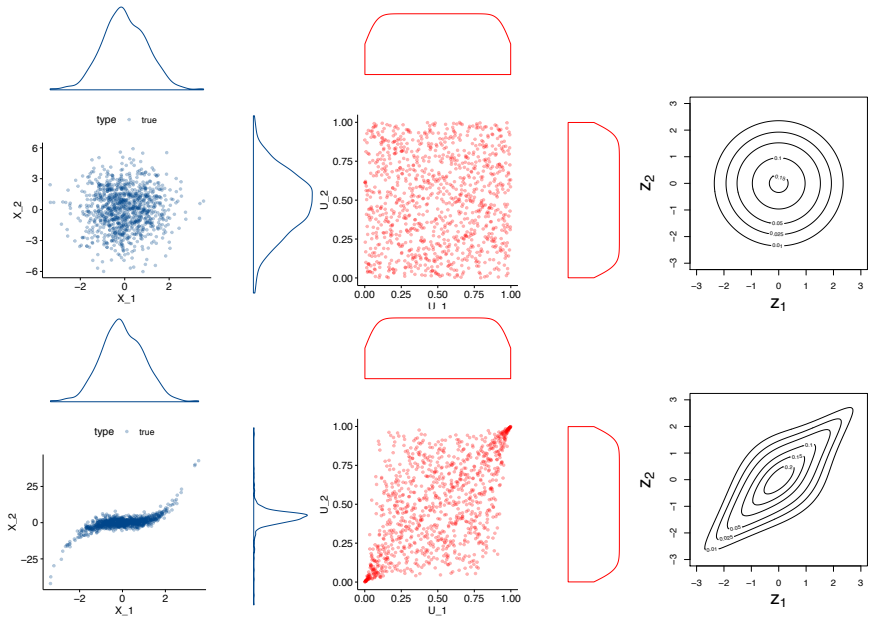


FIGURE 1.2 – Estimation de copules. La première colonne correspond aux données originales, la deuxième colonne aux pseudo-observations obtenues par la TIP et la troisième colonne aux densités estimées des copules. La première ligne illustre un exemple de données non corrélées (indépendantes) et la deuxième ligne de données corrélées.

Les pseudo-observations utilisées pour estimer la densité des copules correspondent en fait aux données classées normalisées (« normalized ranked »). Cet aspect de *classement* ou de *rang* est important dans la théorie des copules. Il permet notamment de capturer adéquatement les corrélations dans valeurs extrêmes (« outliers »), c'est-à-dire les dépendances caudales. La figure 1.2 illustre le phénomène d'une dépendance caudale forte dans le graphique de la deuxième ligne et deuxième colonne. Cette force est due au fait que les copules modélisent la corrélation à l'aide

des *rangs* et non des *valeurs* des données.

Une fois la distribution jointe F estimée à l'aide d'une copule $\hat{C}$, il est possible de transformer des échantillons de $\hat{C}$ en échantillon de F en utilisant la transformée intégrale de probabilité inverse.

Définition 1.3 (Transformée intégrale de probabilité inverse (Devroye, 1986))**.** *Soit U une variable aléatoire uniforme et F une distribution arbitraire, alors $Y = F^{-1}(U)$ est une variable aléatoire qui a F comme fonction de distribution cumulative.*

En effet, si $(U_1, U_2, ..., U_n) \sim \hat{C}$ alors

$$\left(F_1^{-1}(U_1), F_3^{-1}(U_2), ..., F_n^{-1}(U_n)\right) = (X_1, X_2, ..., X_n) \sim X.$$

Le cadre expérimental de génération de données synthétiques à l'aide de copules est relativement simple et ne nécessite que deux étapes : le calcul des FDCEs et l'estimation de la distribution jointe donnée par une copule. Ceci offre une solution simple, mais très peu flexible en haute dimension. Comme l'approche paramétrique pour l'estimation des copules est préférée aux approches non-paramétriques computationnellement parlant (Chen, 2018), ceci implique que les dépendances entre toutes les paires de variables sont du même type (voir l'exemple 1.3 où la structure de dépendance est estimée à l'aide d'une copule gaussienne). Étant rarement le cas, surtout en haute dimension, les copules vignes sont devenues l'approche de prédilection en modélisation en haute dimension puisqu'elles permettent la décomposition des copules multivariées en produits de copules bivariées. Chaque copule bivariée permet une modélisation raffinée de la distribution jointe.

1.2 Copules vignes

Les copules vignes ont été introduites au début des années 2000 (Bedford et Cooke, 2001; Bedford et Cooke, 2002). Elles ont été définies pour pallier le problème de modélisation de dépendances complexes que les copules multivariées parviennent difficilement à répliquer.

Les vignes se basent sur la notion d'arbre de la théorie des graphes.

Définition 1.4 (Arbre)**.** *Un arbre $A = (V, E)$ composé des noeuds V et des arêtes E est un graphe connecté acyclique non orienté. Autrement dit, toute paire de noeuds de V est reliée par un unique chemin formé d'arêtes de E.*

La densité d'une copule sur n variables peut être réécrite en produit de $\frac{n(n-1)}{2}$ densités de copules bivariées conditionnelles (Bedford et Cooke, 2002). Cette décomposition peut être structurée à l'aide d'un modèle graphique de $n-1$ arbres connectés imbriqués, appelé *vigne*. Les arbres $A_i = (V_i, E_i)$ sont composés des noeuds V_i connectés par les arêtes E_i, pour $i = 1, 2, ..., n-1$. Pour assurer que le produit des densités de copules bivariées conditionnelles forme une densité jointe valide, une vigne (sur n variables) doit satisfaire les trois conditions suivantes :

1. A_1 est l'arbre contenant n noeuds représentant les n variables.
2. Pour $i = 2, \ldots, n-1$, l'arbre A_i est constitué des noeuds $V_i = E_{i-1}$.
3. Pour $i = 2, \ldots, n-1$, l'arbre A_i contient exactement $n-i$ arêtes. Deux noeuds de A_i sont reliés par une arête si et seulement si les arêtes correspondantes de A_{i-1} partagent un noeud commun.

Ainsi, dans une vigne, chaque noeud correspond à une variable conditionnelle et chaque arête correspond à la densité d'une copule bivariée entre les deux variables (conditionnelles) reliées par l'arête. Autrement dit, à chaque arête $e \in E_i$ est associée une copule bivariée $c_{j_e,k_e|D_e}$ où j_e, k_e sont les variables conditionnées par

l'ensemble des variables de conditions D_e. Les variables conditionnées et l'ensemble de conditions forment les variables conditionnelles $U_{j_e|D_e}$ et $U_{k_e|D_e}$. Voir l'exemple 1.5 pour des exemples de variables conditionnées et de conditions et l'exemple 1.6 pour des exemples des trois conditions précédentes. L'existence des variables conditionnelles est garantie par les conditions sur les arbres $A_i, i = 1, 2, ..., n-1$ (Bedford et Cooke, 2001; Dissmann *et al.*, 2013).

Un théorème important dans la théorie des copules vignes est celui qui implique que d'une vigne V, il est possible de décrire une densité jointe multivariée en produit de copules bivariées.

Théorème 1.2 (Décomposition de densité sous une vigne (Bedford et Cooke, 2001))**.** *Soit une vigne V sur n variables, il existe une unique densité f telle que*

$$f(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} f_i(X_i) \cdot \prod_{i=1}^{n-1} \prod_{e \in E_i} c_{j_e, k_e|D_e}\left(U_{j_e|D_e}, U_{k_e|D_e}\right)$$

où $U_{j_e|D_e} = F_{j_e|D_e}(X_{j_e|D_e})$ et $U_{k_e|D_e} = F_{k_e|D_e}(X_{k_e|D_e})$.

Ce théorème implique qu'il est toujours possible d'estimer la densité jointe d'une variable aléatoire multivariée à l'aide d'une copule vigne.

Les figures 1.3 et 1.4 des exemples suivants ont été prises de l'article (Aas *et al.*, 2009).

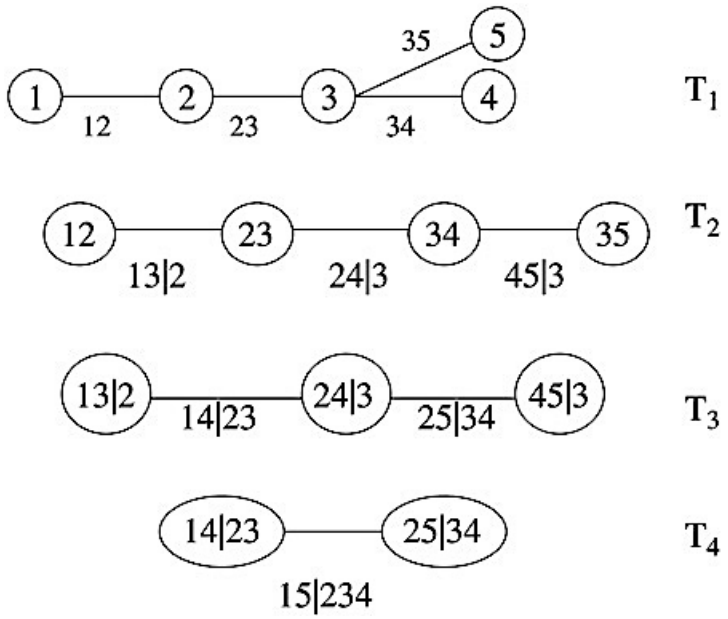


FIGURE 1.3 – Exemple de vigne sur 5 variables.

Exemple 1.5. *Dans la figure 1.3, dans l'arbre T_3, nous avons les copules bivariées $c_{1,4|2,3}$ et $c_{2,5|3,4}$, dont les variables conditionnées sont respectivement $1,4$ et $2,5$ et les ensembles de conditions D_e sont respectivement $\{2,3\}$ et $\{3,4\}$. Les variables conditionnelles formées sont $U_{1|2,3}$ et $U_{4|2,3}$ pour la copule $c_{1,4|2,3}$ et $U_{2|3,4}$ et $U_{5|3,4}$ pour la copule $c_{2,5|3,4}$.*

Exemple 1.6. *Dans la figure 1.3, la condition 1 est bien illustrée dans l'arbre T_1. Dans la même figure, nous avons que l'ensemble de noeuds $V_2 = \{$ (12), (23), (34), (35) $\}$ et que l'ensemble d'arêtes $E_1 = \{\overline{12}, \overline{23}, \overline{34}, \overline{35}\}$, ce qui correspond à la condition 2. Pour illustrer la condition 3, nous avons premièrement que dans l'arbre T_3 de la figure 1.3, il y a exactement $5-3=2$ arêtes. Deuxièmement, nous avons que les noeuds (13\|2) et (24\|3) de l'arbre T_3 sont reliés et que les arêtes $\overline{13|2}$ et $\overline{24|3}$ de l'arbre T_2 partage le noeud (23). De même, les noeuds (24\|3) et (45\|3) de l'arbre T_3 sont reliés et les arêtes correspondantes dans l'arbre T_2 $\overline{24|3}$ et $\overline{45|3}$ partage un noeud :* (34).

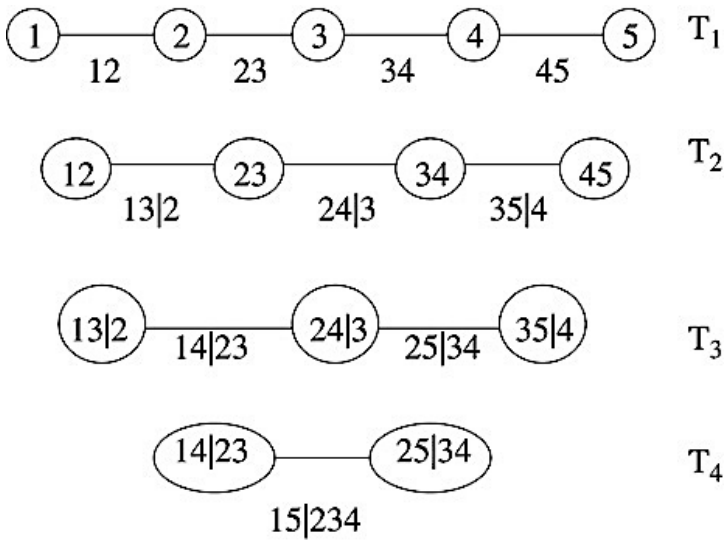


FIGURE 1.4 – Exemple de vigne sur 5 variables.

Exemple 1.7. *Dans la figure 1.4, l'arête* 12 *correspond à la densité de la copule bivariée* $c_{12}\big(F_1(X_1), F_2(X_2)\big)$ *qui joint les variables* X_1 *et* X_2*, l'arête* 13|2 *correspond à la densité* $c_{13|2}\big(F_{1|2}(X_1|X_2), F_{3|2}(X_3|X_2)\big)$ *des variables conditionnelles* $(X_1|X_2)$ *et* $(X_3|X_2)$*, l'arête* 14|23 *correspond à la copule* $c_{14|23}\big(F_{1|2,3}(X_1|X_2, X_3)$, $F_{4|2,3}(X_4|X_2, X_3)\big)$ *des variables conditionnelles* $(X_1|X_2, X_3)$ *et* $(X_4|X_2, X_3)$.

Les figures 1.3 et 1.4 montrent deux structures de vignes possibles sur 5 variables et impliquent la possibilité de plusieurs autres structures. Il existe approximativement $\frac{n!}{2} \cdot 2^{\binom{n-2}{2}}$ différentes vignes sur n variables (Dissmann *et al.*, 2013), donc 480 vignes sur 5 variables. Énumérer toutes les vignes possibles et choisir la vigne qui modélise le mieux les données n'est pas une tâche computationnellement avantageuse plus le nombre d'attributs à modéliser augmente. En plus de l'existence de plusieurs structures possibles, chaque densité de copule bivariée est généralement estimée de manière paramétrique, impliquant un autre choix à faire parmi les modèles de copules bivariées. Étant donnée la nature imbriquée des arbres dans une vigne, les approches séquentielles de sélection de structure et d'estimation de

copules bivariées sont courantes dans la littérature (Aas *et al.*, 2009; Dissmann *et al.*, 2013; Kraus et Czado, 2017).

1.2.1 Sélection de la structure vigne

L'algorithme séquentiel développé dans l'article (Dissmann *et al.*, 2013) pour la sélection de copules vignes est la référence dans le domaine pour la sélection de structure (Kraus et Czado, 2017; Tagasovska *et al.*, 2019). L'algorithme de Dissmann utilise une heuristique séquentielle ascendante et se base sur l'hypothèse que le choix des arbres de niveaux inférieurs (A_1, A_2, A_3, ...) a plus d'influence sur le modèle que les arbres de niveaux supérieurs (A_i, $i >> 10$). Il sélectionne donc l'arbre de niveau 1 optimal, puis passe à la sélection de l'arbre de niveau 2 et ainsi de suite (Dissmann *et al.*, 2013). Le choix optimal de l'arbre à chaque niveau utilise l'Arbre Couvrant de poids Maximal (ACM) avec une mesure de dépendance comme poids sur les arêtes, c'est-à-dire l'arbre qui maximise la somme des poids sur les arêtes. L'algorithme de Prim (Prim, 1957) est généralement utilisé pour trouver l'ACM (Dissmann *et al.*, 2013; Kraus et Czado, 2017; Tagasovska *et al.*, 2019). Les mesures de dépendance peuvent être, par exemple, la valeur absolue de la mesure de corrélation de rang du tau de Kendall (Kendall, 1938) ou la valeur absolue de la mesure de corrélation de rang du rho de Spearman (Spearman, 1987).

Étant données $(U_1, U_2, ..., U_n)$ les variables des pseudo-observations provenant de données $X = (X_1, X_2, ..., X_n)$ et ς une mesure de dépendance quelconque, l'algorithme de Dissmann se résume aux quatre étapes suivantes (Dissmann *et al.*, 2013) :

1. Pour $i = 1$, former le graphe complet composé des n variables comme noeuds et de toutes les mesures de dépendance $\varsigma(U_j, U_k)$ pour $1 \leq j < k \leq n$ comme poids sur les arêtes. Du graphe, ne garder que l'ACM et estimer

toutes les $n-1$ copules bivariées de l'arbre optimal.

2. Fixer $i = i+1$, calculer les distributions conditionnelles de $U_{j_e|D_e}$ et de $U_{k_e|D_e}$ pour toutes les paires de noeuds reliés par $e \in E_i$. Calculer toutes les mesures de dépendance $\varsigma(U_{j_e|D_e}, U_{k_e|D_e})$ et ne garder que l'ACM.
3. Estimer toutes les $n-i$ copules bivariées à l'aide des distributions conditionnelles $U_{j_e|D_e}$ et $U_{k_e|D_e}$.
4. Si $i = n-1$, arrêter, sinon retour à l'étape 2.

À noter qu'il est possible, étant donné l'aspect séquentiel de l'algorithme, d'élaguer la vigne à tout niveau. L'élagage se fait simplement en fixant toutes les copules des niveaux suivants comme étant des copules indépendantes, c'est-à-dire que toutes les variables conditionnelles dans les arbres subséquents sont assumées indépendantes. Le niveau d'élagage sera noté Ψ. Pour se faire, il suffit de remplacer la condition d'arrêt (4.) $i = n-1$ dans l'algorithme précédent à $i = t$ pour $t < n-1$ (Brechmann *et al.*, 2012; Brechmann et Joe, 2015).

Les copules vignes se résument donc à deux éléments :

1. La structure donnée par les arbres $A_i = (V_i, E_i)$, $i = 1, 2, ..., n-1$.
2. Les copules bivariées $c_{j_e,k_e|D_e}$, entre les variables conditionnelles $U_{j_e|D_e}$ et $U_{k_e|D_e}$ telles que $e \in E_i$, $i = 1, 2, ..., n-1$.

1.2.2 Échantillonnage

À partir d'une structure vigne, il est possible d'échantillonner des observations de manière séquentielle (Bedford et Cooke, 2001; Tagasovska *et al.*, 2019). Sommairement, le vecteur $V = (V_1, V_2, ..., V_n) = (V_1, V_{2|1}, V_{3|12}, ..., V_{n|12...n-1})$ est construit (séquentiellement) à partir des copules bivariées spécifiées par la structure vigne. L'exemple suivant détaille le processus sur une vigne à 5 variables.

Exemple 1.8. *(Bedford et Cooke, 2001) En reprenant la figure 1.4, l'échantillonnage à partir de la vigne donnée se fait de la manière suivante :*

1. *Échantillonner V_1 selon la distribution F_1.*
2. *À l'aide de C_{12}, F_1, F_2, calculer $F_{2|1}$ et échantillonner $V_{2|1}$.*
3. *À partir de $C_{12}, F1, F2$, calculer $F_{1|2}$. À partir de $C_{23}, F2, F3$, calculer $F_{3|2}$. À l'aide de $C_{13|2}, F_{1|2}$ et $F_{3|2}$ calculer $F_{3|12}$ et échantillonner $V_{3|12}$.*
4. *De C_{34}, F_3, F_4 calculer $F_{4|3}$. De C_{23}, F_2, F_3 calculer $F_{2|3}$. De $C_{24|3}, F_{2|3}, F_{4|3}$ calculer $F_{4|23}$. À partir de $C_{13|2}, F_{1|2} et F_{3|2}$ calculer $F_{1|23}$. À l'aide de $C_{14|23}, F_{1|23}$ et $F_{4|23}$ calculer $F_{4|123}$ et échantillonner $V_{4|123}$.*
5. *Échantillonner $V_{5|1234}$ à partir de la distribution $F_{5|1234}$ obtenue de manière similaire que les étapes précédentes.*
6. *Le vecteur $(V_1, V_2, V_3, V_4, V_5) \sim C$ où C est la copule donnée par la vigne.*

1.2.3 Complexité

Comme l'approche séquentielle de Dissmann repose sur le l'estimation de n copules bivariées sur l'arbre A_1, $n-1$ copules sur l'arbre A_2, ..., et une copule pour l'arbre A_{n-1}, sa complexité est de $O(f(m) \times n \times \Psi)$, où m est le nombre d'observations et $f(m)$ est la complexité d'estimer une copule bivariée. Lorsque l'estimation des copules se fait de manière paramétrique, $f(m)$ dépend alors du nombre de familles de copules choisies pour modéliser la dépendance et dépend aussi du nombre de paramètres à estimer pour chaque famille. Quant à l'échantillonnage à partir d'une vigne, la complexité est de $O(\textit{nombre d'échantillons} \times n \times \Psi)$ (Dissmann *et al.*, 2013; Tagasovska *et al.*, 2019).

Il est maintenant clair que les copules vignes offrent une modélisation flexible et dont on peut retracer la construction. La flexibilité provient du fait qu'entre chaque paire de variables, il est possible de choisir la copule (bivariée) qui es-

time le mieux la structure de dépendance et la traçabilité est due au fait qu'une vigne est un modèle graphique donc explicable. Deux aspects qui font des copules vignes des modèles de plus en plus populaires en génération de données synthétiques (Kulkarni *et al.*, 2018; Acar *et al.*, 2019; Nagler *et al.*, 2019; Sun *et al.*, 2019; Tagasovska *et al.*, 2019).

Le chapitre suivant introduit les modèles de vie privée que sont le k-anonymat et ses dérivés et la confidentialité différentielle.

CHAPITRE II

MODÈLES DE VIE PRIVÉE

L'assainissement de données est la démarche de modifier une donnée de manière à cacher une information sensible (ex. : identité ou attribut sensible). L'anonymisation est une forme d'assainissement qui a pour but rendre impossible la réidentification de la personne ayant produite la donnée. Dans ce mémoire, le terme assainissement sera utilisé comme synonyme d'anonymisation. L'application naïve de techniques d'assainissement n'apporte aucune garantie formelle sur la protection de la vie privée, ce pour quoi il est préférable de se référer à un modèle de vie privée. Un modèle de vie privée est une marche à suivre pour obtenir le niveau de protection désiré sous la forme d'une formalisation de cette propriété. Un modèle guide l'application de techniques d'assainissement de données pour arriver à une protection avec des limites concrètes. Plus précisément, les modèles de vie privée permettent de fournir un cadre clair et des limites théoriques sur la fuite d'information possible. Les modèles classiques, que sont la k-anonymité et ses extensions ainsi que la confidentialité différentielle (« Differential Privacy » en anglais), seront introduits dans le reste du chapitre.

2.1 k-anonymité, l-diversité et t-proximité

Lorsque la collecte de données se fait auprès d'individus, les données personnelles résultantes comprennent des informations *identifiantes*, *quasi-identifiantes* ou *sensibles*. On parlera de *profils* comme étant une donnée collectée auprès d'un individu contenant une ou plusieurs attributs. Un attribut identifiant, tel qu'un numéro d'assurance sociale (NAS), relie directement la donnée à l'individu, de manière non-équivoque. Un attribut est un quasi-identifiant s'il peut être combiné à d'autres attributs quasi-identifiants pour établir l'identité de l'individu. Un attribut sensible est un attribut dont la valeur doit rester secrète à la demande de la personne sondée ou encore dont la valeur pourrait porter préjudice à l'individu concerné. L'exemple classique d'attributs pouvant jouer le rôle de quasi-identifiants est celui du triplet code postal, date de naissance et sexe qui, dans l'étude menée par Latanya Sweeney, montre qu'il est possible de ré-identifier de manière unique plus de 85% de la population des États-Unis (Sweeney, 2000). Notons aussi que les 15% restants de la population est aussi à risque. En effet, les combinaisons des valeurs du trio de quasi-identifiants réduisent la population à des groupes de très petites tailles (5 personnes ou moins). L'incertitude sur l'identité des 15% restant est donc très faible. Sweeney a développé le modèle de la k-anonymité dans (Sweeney, 2002) pour offrir une protection contre ce genre d'attaque. Ce modèle est rapidement devenu le standard au début des années 2000 en termes de protection de la vie privée. En effet, Sweeney fut l'une des premières chercheuses à définir un modèle formel avec garantie de protection de l'information. La définition de la k-anonymité est la suivante :

Définition 2.1 (k-anonymité)**.** *Soit D une base de données et son sous-ensemble d'attributs quasi-identifiants QI_D. D respecte la k-anonymité si et seulement si tous les tuples uniques de QI_D sont présents au moins k fois.*

Le niveau de protection paramétré par k correspond à la taille de l'ensemble d'anonymat. Plus k est grand, plus l'incertitude sur l'identité de la personne est grande. Il n'existe pas de règle d'or pour établir la valeur de k pour un ensemble donné, mais les valeurs typiquement mises en avant dans les articles varient entre 5 et 500 (Abidi *et al.*, 2020; Rodriguez-Garcia *et al.*, 2019; Wang *et al.*, 2004). Les techniques d'assainissement principales pour atteindre la k-anonymité sont la généralisation, la suppression et la micro-agrégation. Ces techniques sont introduites plus en détail dans le chapitre 3, respectivement dans les parties 3.1 et 3.2. La k-anonymité n'est toutefois pas immunisée contre certaines attaques par inférence comme l'attaque par homogénéité. Pour cette attaque, la fuite d'information est due à l'homogénéité des valeurs d'un attribut. Autrement dit, les k profils d'un ensemble d'anonymat ont tous la même valeur pour un certain attribut, potentiellement sensible. Une autre attaque sur le k-anonymat, l'attaque par corrélation, permet à partir de deux bases de données k-anonymisées qui contiennent des informations complémentaires, d'inférer la valeur d'attributs (Machanavajjhala *et al.*, 2007). Un exemple d'attaque par corrélation est donné par le tableau 2.1 extrait de l'article (Sweeney, 2002). Dans cet exemple, les cases grisées représentent les valeurs d'attributs pouvant être inférées en croisant les tables GT1 et GT3. Par exemple, les lignes 3 et 4 du tableau GT1 peuvent être complétées à l'aide du tableau GT3 : il n'y a que deux femmes nées en 1965 et elles partagent la même origine ethnique et le même code postal.

Le modèle de l-diversité défini dans (Machanavajjhala *et al.*, 2007) a pour but de diminuer la surface d'attaque en limitant les attaques par homogénéité et les attaques de corrélation. Pour se faire, le modèle introduit de la « diversité » par rapport aux attributs sensibles au sein des groupes créés par la k-anonymité. La définition de base de la l-diversité est la suivante :

Définition 2.2 (l-diversité)**.** *Soit D une base de données et son sous-ensemble*

Race	BirthDate	Gender	ZIP	Problem
black	1965	male	02141	short of breath
black	1965	male	02141	chest pain
person	1965	female	0213*	painful eye
person	1965	female	0213*	wheezing
black	1964	female	02138	obesity
black	1964	female	02138	chest pain
white	1964	male	0213*	short of breath
person	1965	female	0213*	hypertension
white	1964	male	0213*	obesity
white	1964	male	0213*	fever
white	1967	male	02138	vomiting
white	1967	male	02138	back pain

GT1

Race	BirthDate	Gender	Zip	Problem
black	1965	male	02141	short of breath
black	1965	male	02141	chest pain
black	1965	female	02138	painful eye
black	1965	female	02138	wheezing
black	1964	female	02138	obesity
black	1964	female	02138	chest pain
white	1960-69	male	02139	short of breath
white	1960-69	human	02139	hypertension
white	1960-69	human	02139	obesity
white	1960-69	human	02139	fever
white	1960-69	male	02138	vomiting
white	1960-69	male	02138	back pain

GT3

TABLEAU 2.1 – Exemple d'attaque par corrélation sur deux bases de données k-anonymisées.

d'attributs quasi-identifiants QI_D contenant un attribut sensible. Un groupe k-anonyme de QI_D respecte la l-diversité si pour chaque groupe, l'attribut sensible contient au moins l valeurs distinctes.

Le paramètre l de la l-diversité correspond au nombre de valeurs distinctes de l'attribut sensible par groupe k-anonyme. La figure 2.1 tirée de l'article (Machanavajjhala *et al.*, 2007) illustre un exemple avec $l = 3$.

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	*	Heart Disease
4	1305*	≤ 40	*	Viral Infection
9	1305*	≤ 40	*	Cancer
10	1305*	≤ 40	*	Cancer
5	1485*	> 40	*	Cancer
6	1485*	> 40	*	Heart Disease
7	1485*	> 40	*	Viral Infection
8	1485*	> 40	*	Viral Infection
2	1306*	≤ 40	*	Heart Disease
3	1306*	≤ 40	*	Viral Infection
11	1306*	≤ 40	*	Cancer
12	1306*	≤ 40	*	Cancer

FIGURE 2.1 – Exemple base de données 3-diverse et 4-anonyme.

Bien que la l-diversité rende les données immunisées contre les attaques par homogénéité et très robustes aux attaques par corrélation, le modèle est très restrictif. En effet, il peut introduire une perte importante d'utilité dans les données ou même être impossible à atteindre lorsque les valeurs d'attributs sensibles sont très asymétriques (Li *et al.*, 2007). Dans le cas où l'attribut sensible correspond au résultat d'un test de dépistage où 99% des résultats sont négatifs et 1% des résultats sont positifs, le fait d'appliquer la 2-diversité implique un taux d'incidence d'au moins $1/k$% des résultats positifs (où k correspond à la taille de l'ensemble d'anonymat). Plus concrètement, si $k = 4$, alors le taux d'incidence des tests positifs dans les données 2-diverses sera de 25% (Li *et al.*, 2007; Domingo-Ferrer *et al.*, 2016).

La t-proximité introduite dans l'article de Li, Li et Venkatasubramanian (2007) est un raffinement de la l-diversité qui vient forcer la similarité des distributions des attributs sensibles.

Définition 2.3 (t-proximité)**.** *Soit D une base de données et son sous-ensemble d'attributs quasi-identifiants QI_D. Un groupe k-anonyme de QI_D respecte la t-proximité si la distance entre la distribution de l'attribut sensible du groupe et la distribution de l'attribut sensible de D est inférieure à t.*

La distance t est calculée à l'aide de la distance de (1-)Wasserstein (Vaserstein, 1969) aussi connu sous le nom de « Earth Mover's Distance » entre les distributions. Dans le cas discret, la distance est plutôt intuitive et correspond à la quantité minimale de « balles » qu'il faut déplacer entre les intervalles pour passer d'une distribution à une autre. Par exemple, pour passer de la distribution $P = |1|1|4|2|$ à la distribution $Q = |2|1|3|2|$ où $|x|$ correspond à un intervalle à x balles, il faut faire 2 déplacements de balles : $P = |1|1|4|2| \rightarrow |2\overset{\curvearrowleft}{|}0|4|2| \rightarrow |2|1\overset{\curvearrowleft}{|}3|2| = Q$. En pratique, la distance est appliquée sur les

FDPs plutôt que sur des histogrammes comme l'exemple précédent, ce qui implique une distance bornée par $[0, 1]$ (voir la partie 5 de l'article (Li *et al.*, 2007) pour plus de détails sur la distance de Wasserstein dans le cadre de la t-proximité.) Ainsi, le paramètre t correspond à la différence maximale permise entre la distribution de l'attribut sensible dans les données et la distribution de l'attribut sensible dans les groupes k-anonymes.

Malgré les garanties de vie privée fournies par les modèles précédents, il y a toujours un risque de ré-identification. En effet, bien que les profils soient assainis, le lien entre identité et profil n'est pas rompu par les diverses techniques appliquées, contrairement aux méthodes génératives où l'abstraction vers des modèles probabilistes dissout ce lien. Plus précisément, dans une base de données anonymisées respectant un des modèles précédents, les profils correspondent encore à des individus dont les valeurs d'attributs ont été assainies. Dans une base de données synthétiques, les profils sont créés à l'aide d'un échantillonnage aléatoire sur des distributions apprises sur une population. Néanmoins, les modèles génératifs ne sont pas à l'abri d'attaques et n'offrent aucune garantie intrinsèque sur la protection de la vie privée, ce qui motive l'application de la confidentialité différentielle à la création de ces modèles.

2.2 Confidentialité différentielle

La confidentialité différentielle (CD) est un modèle de vie privée qui vise à limiter toute possibilité d'inférence sur un profil individuel d'une base de données en atténuant la contribution qu'un individu peut avoir sur le résultat d'un calcul réalisé sur cette base de données. Ainsi, le résultat d'une requête statistique respectant la CD est théoriquement indistinguable, que la requête soit exécutée sur un ensemble avec un profil particulier ou sans.

Dans cette partie, les notations originelles de Dwork et Roth (2014) sont utilisées. Plus précisément, $\mathcal{D}$ est l'ensemble des profils possibles sur un ensemble d'attributs quelconque, x est un sous-ensemble de profils de $\mathcal{D}$ (donc un ensemble ou une base de données) et x^i correspond au nombre de profils de x identiques au profil $i \in \mathcal{D}$. Dans les définitions qui suivent, $\mathcal{R}$ est un sous-ensemble arbitraire de l'image du mécanisme $\mathcal{I}(\mathcal{M})$.

Définition 2.4 (Mécanisme aléatoire)**.** *Un mécanisme aléatoire $\mathcal{M} : D \to \mathcal{R}$ est une fonction non-déterministe qui, sur entrée d, produit $\mathcal{M}(d) = r$ avec probabilité* $\Pr[\mathcal{M}(d) = r]$.

Un exemple d'un mécanisme aléatoire à l'exemple 2.1.

Exemple 2.1. *Une fonction DéSixFaces qui, à n'importe quelle entrée, retourne un chiffre entre 1 et 6 avec probabilité* 0.167 *est un exemple de mécanisme aléatoire. Nous avons que* $\Pr[\textit{DéSixFaces}(d) = 2] = 0.167$ *pour une entrée d quelconque.*

Définition 2.5 (Distance entre bases de données (bornée))**.** *La distance $\ell_1 : D \times D \to \mathbb{N}$, notée $||\cdot||_1$, entre deux bases de données de même taille (et sur les mêmes attributs) x et y de $\mathcal{D}$ est définie comme :*

$$||x - y||_1 = \sum_{i=1}^{|\mathcal{D}|} |x^i - y^i|$$

La distance ℓ_1 correspond donc au nombre de profils (tuples) qui diffèrent entre x et y. Ainsi $||x - y||_1 = 1$ signifie que x et y diffèrent d'un seul profil, c'est-à-dire qu'il y a deux profils $i \in x$ et $j \in y$, et seulement deux profils, tels que $i \neq j$. À noter que nous utilisons la définition *bornée* de la CD, c'est-à-dire que x et y sont deux bases de données de même cardinalité.

Définition 2.6 (Confidentialité différentielle (bornée))**.** *Soit un mécanisme aléatoire $\mathcal{M}$ dont l'image est donnée par $\mathcal{I}(\mathcal{M})$. Le mécanisme aléatoire $\mathcal{M}$ est ϵ-*

différentiellement-privé si et seulement si pour tout $\mathcal{S} \subseteq \mathcal{I}(\mathcal{M})$ *et pour toutes paires de bases de données* x *et* y *telles que* $||x - y||_1 = 1$ *nous avons :*

$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq \exp(\epsilon) \cdot \Pr[\mathcal{M}(y) \in \mathcal{S}].$$

Autrement dit, la CD assure que la différence entre les réponses du mécanisme aléatoire soit indiscernable à $\exp(\epsilon)$ près pour toutes paires de bases de données qui diffèrent en au plus un profil. ϵ est ici le paramètre définissant le niveau de protection et est souvent appelé le *budget* de la confidentialité différentielle. Cette notion de budget découle de l'utilisation de plusieurs mécanismes aléatoires fonctionnant sur le même ensemble de données en entrée et comment le budget est « dépensé » ou divisé entre les différents mécanismes. Tel que défini, plus le budget ϵ est petit, plus les réponses du mécanisme sont indiscernables, impliquant une meilleure protection individuelle.

Exemple 2.2. *Montrons que le mécanisme aléatoire* DéSixFaces *défini dans l'exemple 2.1 est* (0.1)*-différentiellement-privé. Par définition de* DéSixFaces *nous avons que pour toutes entrées* x *et* y

$$\Pr[\textit{DéSixFaces}(x) \in \mathcal{S}] = \Pr[\textit{DéSixFaces}(y) \in \mathcal{S}] \quad \forall \mathcal{S} \subseteq \mathcal{I}(\textit{DéSixFaces}).$$

De ceci découle que

$$\frac{\Pr[\textit{DéSixFaces}(x) \in \mathcal{S}]}{\Pr[\textit{DéSixFaces}(y) \in \mathcal{S}]} = 1 \leq \exp(0.1) = 1.105.$$

Donc le mécanisme aléatoire DéSixFaces *est bien* (0.1)*-différentiellement-privé.*

Il est important de noter que la définition précédente entraîne une protection *individuelle* puisque l'indiscernabilité des résultats du mécanisme n'implique que

la présence ou l'absence d'*un* profil, c'est-à-dire que $||x - y||_1 = 1$. La protection individuelle de la confidentialité différentielle peut s'étendre à un groupe, par exemple lorsque les données contiennent plusieurs membres d'une même famille ou si une personne contribue plusieurs fois à un même ensemble de données.

De la définition précédente de la confidentialité différentielle, il découle plusieurs propriétés qui permettent de comptabiliser le budget ϵ lorsque plusieurs mécanismes aléatoires sont utilisés dans un même cadre expérimental. L'application de plusieurs mécanismes aléatoire assurant la CD n'aura pas toujours le même effet sur le budget suivant que ces mécanismes sont appliqués de manière séquentielle ou parallèle. Les théorèmes suivants sont nécessaires pour une gestion adéquate du budget ϵ pour un maximum de protection.

Théorème 2.1 (Fermeture sous post-traitement)**.** *(Dwork* et al.*, 2014) Soit $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ un mécanisme aléatoire ϵ-différentiellement-privé et $f : \mathcal{R} \to \mathcal{R}'$ une fonction arbitraire sur $\mathcal{R}$ telle que $\mathcal{R}' \subseteq \mathcal{I}(f)$, alors la composition $f \circ \mathcal{M} : \mathcal{D} \to \mathcal{R}'$ est ϵ-différentiellement-privée.*

Exemple 2.3. *Soit le mécanisme aléatoire DéSixFaces tel que défini dans l'exemple 2.1. Soit la fonction $f(x) = x - 1$ de post-traitement qui soustrait 1 aux sorties du mécanisme DéSixFaces. Par exemple, en ne tenant pas compte de la nature aléatoire du mécanisme, si DéSixFaces$(d) = 2$ alors f (DéSixFaces$(d)) = 1$, pour une entrée d quelconque. Par le théorème 2.1, la composition $f \circ$ DéSixFaces est (0.1)-différentiellement-privée.*

Théorème 2.2 (Traitement séquentielle)**.** *(Dwork* et al.*, 2014) Soit $\mathcal{M}_1 : \mathcal{D} \to \mathcal{R}$ et $\mathcal{M}_2 : \mathcal{D} \to \mathcal{R}'$ deux mécanismes aléatoires respectivement ϵ_1- et ϵ_2-différentiellement-privé, alors l'application séquentielle des deux mécanismes sur le même ensemble $\mathcal{D}$ $(\mathcal{M}_1, \mathcal{M}_2)(x) : \mathcal{D} \to \mathcal{R} \times \mathcal{R}'$ est $(\epsilon_1 + \epsilon_2)$-différentiellement-privée.*

Autrement dit, le théorème 2.2 stipule que lorsque deux mécanismes aléatoires différentiellement-privés sont appliqués séquentiellement sur un même ensemble de données, alors les *budgets s'additionnent*.

Théorème 2.3 (Traitement parallèle)**.** *(Dwork* et al.*, 2014) Soit* $\mathcal{M}_1 : \mathcal{D}_1 \to \mathcal{R}$ *et* $\mathcal{M}_2 : \mathcal{D}_2 \to \mathcal{R}'$ *deux mécanismes aléatoires respectivement* ϵ_1*-* et ϵ_2*-différentiellement-privé, tels que* $\mathcal{D}_1, \mathcal{D}_2 \subset \mathcal{D}$ *et* $\mathcal{D}_1 \cap \mathcal{D}_2 = \emptyset$*, alors l'application parallèle des deux mécanismes* $(\mathcal{M}_1, \mathcal{M}_2)(x) : \mathcal{D} \to \mathcal{R} \times \mathcal{R}'$ *est* $\max(\epsilon_1, \epsilon_2)$*-différentiellement-privée.*

Le théorème 2.3 est l'analogue parallèle du théorème 2.2 et précise que lorsque deux mécanismes aléatoires différentiellement-privés sont appliqués de manière parallèle et indépendante de sur des ensembles disjoints, alors le budget global *reste inchangé* si les mécanismes ont le même budget. Autrement le budget global devient le budget maximum des mécanismes.

Les théorèmes 2.2 et 2.3 peuvent être généralisés à k mécanismes séquentiels ou sous-ensembles disjoints. Un point *très important* à souligner avec le théorème 2.1 de fermeture sous post-traitement est que la fonction f doit absolument être restreinte à la sortie $\mathcal{R}$ d'un mécanisme aléatoire ϵ-différentiellement-privé. Si cette condition n'est pas respectée, le théorème peut ne pas s'appliquer et la protection n'est plus garantie.

Étant donné une fonction f arbitraire, il existe plusieurs méthodes pour la rendre ϵ-différentiellement-privée. Dwork et Roth (2014) ont introduit le mécanisme laplacien pour rendre une fonction numérique qui retourne un réel ou un ensemble de réels $f : \mathcal{D} \to \mathbb{R}$ ϵ-différentiellement-privée.

Avant de définir ce mécanisme, il faut définir la notion de sensibilité d'une fonction. La sensibilité d'une fonction peut se traduire par la différence maximale sur la

sortie d'une fonction que peut provoquer l'absence d'un profil.

Définition 2.7 (Sensibilité-ℓ_1)**.** *La sensibilité-ℓ_1 d'une fonction $f : \mathcal{D} \to \mathbb{R}$, notée Δf, est définie par :*

$$\Delta f = max||f(x) - f(y)||_1$$

où $x, y \in \mathcal{D}$ tels que $||x - y||_1 = 1$.

Le mécanisme laplacien est souvent référé dans les chapitres qui suivent, il est donc nécessaire de l'introduire formellement. Dans ce qui suit, on dénotera Lap(σ) la distribution de Laplace centrée en $\mu = 0$ et d'échelle $b = \sigma$.

Définition 2.8 (Mécanisme laplacien)**.** *Soit $f : \mathcal{D} \to \mathbb{R}^k$, le mécanisme laplacien est défini par :*

$$\mathcal{M}_L(x) = f(x) + (L_1, ..., L_k)$$

où L_i est tirée de manière aléatoire de la distribution de Laplace Lap($\frac{\Delta f}{\epsilon}$).

Un exemple d'une distribution de Laplace est donné par la figure 2.2 extraite de l'article (Domingo-Ferrer *et al.*, 2016).

Théorème 2.4. *(Dwork* et al.*, 2014) Le mécanisme laplacien est ϵ-différentiellement-privé.*

Les garanties de la confidentialité différentielle peuvent se résumer de la manière suivante :

1. la protection contre les attaques de ré-identification et d'inférence d'appartenance (voir la partie 2.3) due à la nature non-déterministe du modèle,
2. la quantification de la perte d'information grâce au « budget » et aux théorèmes de compositions des mécanismes CD.

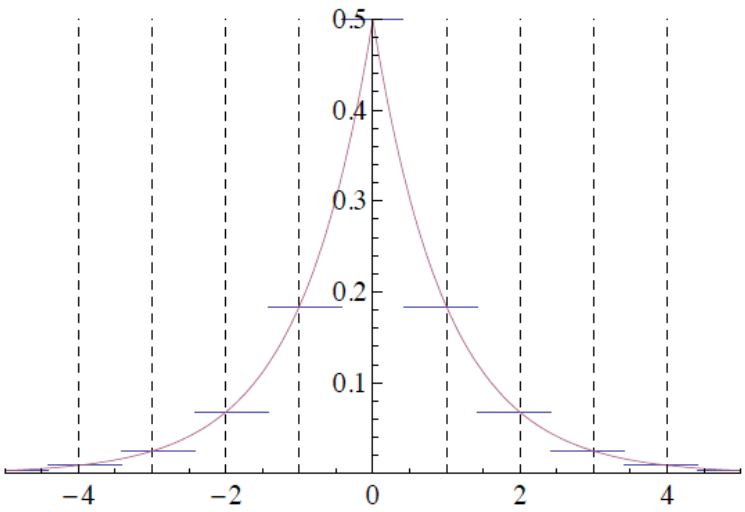


FIGURE 2.2 – Exemple d'une distribution de Laplace Lap($\mu = 0, b = 1$)

La confidentialité différentielle offrant des blocs de construction solides à l'aide des mécanismes introduits précédents, sera le modèle de prédilection pour la génération de données synthétiques ϵ-différentiellement-privées à l'aide de copules, ce qui est concrétisé dans le chapitre 4. L'application de la confidentialité différentielle dans un modèle génératif implique non seulement que le lien entre les identités et les profils soit rompu, mais aussi qu'individuellement, la contribution des profils soit limitée. Ainsi, un profil réel n'influencera que très peu le modèle génératif et les profils générés.

Pour comprendre la protection, il faut nécessairement savoir de quoi cette défense nous protège. La partie qui suit aide à cerner ce qu'est une attaque par inférence sur des profils.

2.3 Attaques par inférence

Il existe plusieurs manières de classifier les attaques contre la vie privée, mais la majorité des attaques sur des profils peuvent se regrouper sous une seule dénomi-

nation celle d'une « attaque par inférence » ou simplement « inférence ». Ainsi, le but d'une attaque est d'*inférer* de nouvelles connaissances sur des individus, menant ainsi à un bris de leur vie privée.

Définition 2.9 (Inférence)**.** *Une attaque par inférence sur des données se définit par l'obtention de nouvelles informations sur un profil ou un groupe de profils. Par exemple, ces nouvelles informations peuvent concerner des valeurs d'attributs préalablement inconnues, l'identité des profils ou encore l'appartenance des profils à une base de données ou un modèle génératif particulier.*

Dans la définition précédente, « l'appartenance des profils à une base de données » se précise en la découverte d'un même profil sur un ou plusieurs jeux de données différents contenant possiblement des attributs différents et mène donc à la découverte de nouvelles informations.

Les inférences peuvent se baser sur des a priori aussi appelés « connaissances auxiliaires » définies comme :

Définition 2.10 (Connaissances auxiliaires)**.** *Les connaissances auxiliaires se composent de toute information connue par un adversaire sur un profil ou un groupe de profils préalablement à l'attaque par inférence.*

Les inférences peuvent aussi exploiter le caractère unique d'un profil :

Définition 2.11 (Unicité)**.** *L'unicité d'un profil est le caractère distinctif et identifiant d'une valeur d'attribut ou d'une combinaison de valeurs d'attributs.*

L'unicité capture le risque de ré-identification offrant ainsi une mesure de risque (De Montjoye *et al.*, 2013; De Montjoye *et al.*, 2015; Rocher *et al.*, 2019). Cependant, être unique dans un jeu de données n'implique pas nécessairement

la ré-identification de la personne, mais une personne avec un profil unique est beaucoup plus susceptible d'être ré-identifié.

Une attaque qui mène à retrouver l'identité d'une personne associée à un profil anonymisé est appelée une attaque de ré-identification. La définition suivante est une adaptation de celle trouvée dans (Sweeney, 2000).

Définition 2.12 (Ré-identification)**.** *Un profil est ré-identifié si son anonymat est levé et que l'identité de la personne reliée ce profil est dévoilée.*

Les attaques de ré-identification sont rarement applicables dans un cadre théorique étant donné le manque de vérité terrain. Autrement dit le résultat d'une attaque pourrait pointer vers un homme appelé John Smith de 33 ans habitant sur la 1re Avenue à Montréal avec le code postal H1Y 3A1 ayant des problèmes cardiaques, mais il est le plus souvent impossible de confirmer la véracité de ces informations.

Sweeney dans son article (Sweeney, 2000) réussi toutefois à croiser deux bases de données (ce qui constitue une attaque de couplage) et à ré-identifier William Weld, le gouverneur du Massachusetts, dans les deux bases de données et ainsi obtenir des informations sur la santé de Weld. Weld a premièrement été identifié dans le jeu de données des électeurs de Cambridge, Massachusetts. La méthode d'identification utilisée était simple : six personnes partageaient sa date de naissance dans la liste, trois des six personnes étaient des hommes et un seul de ces derniers avec le même code postal. Ce même trio d'attributs quasi-identifiants fut utilisé pour retrouver Weld dans un jeu de données du « Group Insurance Commission » qui s'occupe des assurances maladie pour les employés de l'État. Ainsi, un profil complet du gouverneur a pu être reconstruit. La figure 2.3 provenant de l'article (Sweeney, 2002) illustre l'attaque de couplage.

Lorsque l'inférence sur un profil mène à la découverte d'informations non-triviales,

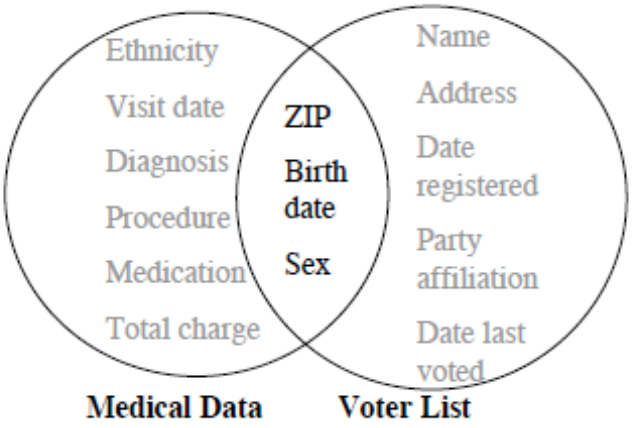


FIGURE 2.3 – Processus de ré-identification par attaque de couplage de Sweeney.

c'est-à-dire qui sort du spectre de l'inférence statistique, on parle alors de *gain d'information*. La distinction entre inférence statistique et gain d'information se trouve dans le type d'attributs qui est inféré.

Définition 2.13 (Gain d'information)**.** *Une inférence constitue un gain d'information s'il elle révèle la valeur d'un ou plusieurs attributs sensibles telle que le lieu de résidence.*

Les inférences statistiques, bien que pouvant être un vecteur d'attaque, sont souvent directement reliées à l'utilité des données dans le sens où elles font partie des connaissances qu'on cherche à acquérir en premier lieu. Un exemple d'attaque par inférence sur données de mobilité qui mène à une violation flagrante de la vie privée est donné dans l'article (Gambs *et al.*, 2010). Dans ce dernier article, les auteurs réussissent à identifier le lieu de résidence de plusieurs chauffeurs de taxi sur un ensemble de données contenant des traces de mobilités de taxis de la ville de San Francisco (dans ce jeu de données, les taxis sont aussi utilisés par les chauffeurs pour leurs déplacements personnels).

Une attaque menant à l'identification d'un ou plusieurs profils dans une ou plu-

sieurs autres bases de données est dite de *couplage* (« linkage » en anglais).

Définition 2.14 (Couplage)**.** *Lorsqu'un profil ou un groupe de profils d'une base de données D est associé à un ou plusieurs profils d'autres bases de données D_i, on parle alors d'une attaque par couplage.*

Un exemple de couplage de profils (Narayanan et Shmatikov, 2008) est celui où plusieurs profils d'une base de données de Netflix contenant des notes (« ratings » en anglais) ont été associés à des profils de IMDb, un site web contenant une multitude d'informations reliées aux films, séries télévisées et autres médias incluant des notes données par les utilisateurs du site.

Une autre attaque possible est l'attaque par appartenance (« membership » en anglais).

Définition 2.15 (Appartenance)**.** *(Shokri* et al.*, 2017) Une attaque d'inférence d'appartenance permet de savoir si un individu se trouve dans une base de données ou s'il a contribué au résultat d'une requête sur la base de données.*

Dans l'article (Shokri *et al.*, 2017), les auteurs développent une attaque permettant de savoir si le profil d'un individu a été utilisé dans l'apprentissage d'un modèle de classification en ne regardant que les données sortantes du modèle. Pour se faire, les auteurs proposent d'utiliser plusieurs modèles *fantômes* similaires au modèle victime de l'attaque. Le contrôle de l'information, c'est-à-dire le contrôle sur les profils d'entraînement des modèles fantômes, permet de distinguer un profil ayant servi dans la phase d'entraînement d'un classifieur externe. Les auteurs montrent que les vecteurs de prédictions du classifieur et des modèles fantômes permettent la distinction. Cette attaque est basée sur le fait que si le modèle « reconnaît » un profil, sa classification se fera avec confiance.

Un modèle de vie privée a peu de valeur a lui seul. En effet, un modèle sert de

guide pour appliquer diverses techniques d'assainissement et d'anonymisation de données. Le chapitre suivant introduira plusieurs de ces techniques qui ont été développées au cours des dernières années ainsi qu'un bon nombre d'exemples pour illustrer leurs effets sur les données.

CHAPITRE III

TECHNIQUES D'ANONYMISATION ET D'ASSAINISSEMENT DE MICRO-DONNÉES

Les micro-données, ou données brutes, sont le produit direct de collectes de données, tel qu'un recensement. Généralement, ces données n'ont subi que très peu de traitements statistiques, par exemple le revenu peut être transformé en classe de revenu. Les micro-données sont principalement représentées sous la forme d'un tableau où chaque ligne est le résultat ponctuel d'une collecte et où les colonnes sont les propriétés recensées par la collecte, aussi appelées attributs (Domingo-Ferrer *et al.*, 2016). Si la collecte se fait auprès d'une population, chaque ligne représente un individu recensé, ce qu'on appelle un *profil*. Le terme micro-données est le terme statistique utilisé pour parler de données collectées au niveau individuel (United Nations Statistical Commision et Economic Commision For Europe, 2000). Dans le présent rapport, le terme *données* sera souvent employé pour référer à des micro-données. Un exemple de micro-données tiré de l'ensemble de données « Adult » du répertoire de l'UCI (Dua et Graff, 2017) est donné dans le tableau 3.1. Les données sont le résultat d'un recensement de 1994 fait aux États-Unis.

Comme les données du tableau 3.1 proviennent d'une collecte auprès d'individus, ces données contiennent nécessairement des quasi-identifiants tels que vu dans le chapitre 2. Il est même possible d'illustrer l'effet de ces quasi-identifiants avec

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
1	39	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	2174	40	United-States
2	50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	White	Male	0	13	United-States
3	38	Private	HS-grad	Divorced	Handlers-cleaners	White	Male	0	40	United-States
4	53	Private	11th	Married-civ-spouse	Handlers-cleaners	Black	Male	0	40	United-States
5	28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Black	Female	0	40	Cuba
6	37	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	0	40	United-States
7	49	Private	9th	Married-spouse-absent	Other-service	Black	Female	0	16	Jamaica
8	52	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	White	Male	0	45	United-States
9	31	Private	Masters	Never-married	Prof-specialty	White	Female	14084	50	United-States

TABLEAU 3.1 – Micro-données produites à partir d'un recensement de 1994.

les trois attributs (*workclass*, *education*, *marital-status*) qui forment des tuples de valeurs qui n'apparaissent qu'une seule fois dans le tableau ; ces tuples forment donc des informations identifiantes. Il est impératif d'exercer un certain contrôle sur les micro-données pour limiter les risques de bris de la vie privée des individus concernés. Pour cela, plusieurs méthodes existent pour anonymiser des données brutes. Il est possible de regrouper ces méthodes en trois catégories : les méthodes non-perturbantes, perturbantes et génératives (Domingo-Ferrer *et al.*, 2016). Le reste du chapitre introduira ces trois catégories d'assainissement de données.

3.1 Méthodes non-perturbantes

Les méthodes non-perturbantes se basent principalement sur l'agrégation et la suppression de données. Les valeurs des attributs sont donc omises ou plus grossières, ce qui réduit la surface d'attaque possible. Les principales techniques non-perturbantes sont l'échantillonnage, la suppression locale et la généralisation. Un *échantillonnage* est simplement l'action de sélectionner une portion des données, de manière horizontale en choisissant un sous-ensemble de profils, verticale en choisissant un sous-ensemble de colonnes, ou les deux. Dans un contexte d'assainissement de données, les données sont échantillonnées de manière à limiter les informations identifiantes et quasi-identifiantes. La « suppression locale » se fait en masquant simplement la valeur de certains attributs, avec « * » par exemple, alors que la *généralisation* regroupe plusieurs valeurs d'un attribut par un même

représentant. Prenons par exemple le tableau 3.1 pour illustrer ces différentes techniques. Notons que les valeurs « Cuba » et « Jamaica » de l'attribut *native-country* et les valeurs non-nulles de l'attribut *capital-gain* n'apparaissent qu'une seule fois dans l'exemple. Ainsi, elles peuvent être considérées comme des valeurs identifiantes pour ces profils.

Le tableau 3.2 montre un exemple d'échantillonnage horizontal en omettant les profils atténués et vertical en omettant les attributs grisés.

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
1	39	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	2174	40	United-States
2	50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	White	Male	0	13	United-States
3	38	Private	HS-grad	Divorced	Handlers-cleaners	White	Male	0	40	United-States
4	53	Private	11th	Married-civ-spouse	Handlers-cleaners	Black	Male	0	40	United-States
5	28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Black	Female	0	40	Cuba
6	37	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	0	40	United-States
7	49	Private	9th	Married-spouse-absent	Other-service	Black	Female	0	16	Jamaica
8	52	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	White	Male	0	45	United-States
9	31	Private	Masters	Never-married	Prof-specialty	White	Female	14084	50	United-States

TABLEAU 3.2 – Exemple d'échantillonnage horizontal (profils atténués) et vertical (attributs grisés).

Le tableau 3.3, quant à lui, illustre la technique de suppression locale où les valeurs jugées identifiantes sont supprimées. Enfin, le tableau 3.4 illustre la généralisation où les valeurs identifiantes sont regroupées sous une nouvelle valeur.

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
1	39	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	*	40	United-States
2	50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	White	Male	0	13	United-States
3	38	Private	HS-grad	Divorced	Handlers-cleaners	White	Male	0	40	United-States
4	53	Private	11th	Married-civ-spouse	Handlers-cleaners	Black	Male	0	40	United-States
5	28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Black	Female	0	40	*
6	37	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	0	40	United-States
7	49	Private	9th	Married-spouse-absent	Other-service	Black	Female	0	16	*
8	52	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	White	Male	0	45	United-States
9	31	Private	Masters	Never-married	Prof-specialty	White	Female	*	50	United-States

TABLEAU 3.3 – Exemple de suppression locale.

La simplicité de ces techniques implique une faible protection de la vie privée et par conséquent elles sont rarement appliquées seules. Les sous-parties suivantes passent en revue des méthodes plus complexes d'assainissement de données.

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
1	39	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	**[2000 - 49999]**	40	United-States
2	50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	White	Male	**[0 - 1999]**	13	United-States
3	38	Private	HS-grad	Divorced	Handlers-cleaners	White	Male	**[0 - 1999]**	40	United-States
4	53	Private	11th	Married-civ-spouse	Handlers-cleaners	Black	Male	**[0 - 1999]**	40	United-States
5	28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Black	Female	**[0 - 1999]**	40	**Caribbean**
6	37	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	**[0 - 1999]**	40	United-States
7	49	Private	9th	Married-spouse-absent	Other-service	Black	Female	**[0 - 1999]**	16	**Caribbean**
8	52	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	White	Male	**[0 - 1999]**	45	United-States
9	31	Private	Masters	Never-married	Prof-specialty	White	Female	**[2000 - 49999]**	50	United-States

TABLEAU 3.4 – Exemple de généralisation.

3.1.1 Échantillonnage

L'échantillonnage, comme il a été introduit plus tôt, limite la quantité de données en sélectionnant un sous-groupe de profils, d'attributs ou des deux. Par exemple, dans l'article (Joy et Gerla, 2017), les auteurs utilisent l'échantillonnage pour obtenir la notion de déni plausible en utilisant le mécanisme de réponse aléatoire (*randomized response mechanism*) dans la sélection de profils. Le déni plausible peut être défini comme étant l'incapacité pour un adversaire de déduire avec certitude l'identité d'un profil (Bindschaedler *et al.*, 2017). Le mécanisme de réponse aléatoire, qui introduit du déni plausible, peut être illustré par le lancer d'une pièce pour décider de révéler ou non une valeur binaire. Dans ce scénario, si la pièce tombe sur *face*, la vraie réponse est donnée. Sinon la pièce est relancée et la réponse donnée est « positif » si la pièce tombe sur *face* et la réponse donnée est « négatif » autrement. Comme mentionné précédemment, l'article de Joy et Gerla introduit le mécanisme de réponse aléatoire dans un simple algorithme d'échantillonnage. La méthode comporte deux phases. La première phase estime le nombre de profils dans l'échantillon qui répondent honnêtement de manière négative alors que la deuxième phase estime le nombre de profils qui répondent

honnêtement de manière positive à l'aide des probabilités suivantes :

$$Round.One = \begin{cases} 0 & \text{avec probabilité } p_0 \\ 0 & \text{avec probabilité } p_s \\ 1 & \text{avec probabilité } 1 - p_0 - p_s \end{cases} \qquad Round.Two = \begin{cases} 0 & \text{avec probabilité } p_0 \\ 1 & \text{avec probabilité } p_s \\ 1 & \text{avec probabilité } 1 - p_0 - p_s \end{cases} \tag{3.1}$$

Dans la formule 3.1, p_s est appelé le paramètre d'échantillonnage et détermine la probabilité de contribution des individus. Autrement dit, un échantillon de 45% de la population équivaut à un paramètre $p_s = 0.45$. De plus, p_0 représente la probabilité ou proportion réelle de 0 dans l'échantillon, c'est-à-dire le nombre de réponses négatives obtenues si tout le monde répondait de manière honnête. Les deux phases servent à produire des estimations de statistiques de la population. Ainsi, à partir de *Round.One* de l'équation 3.1, il est possible d'estimer le nombre de 0 dans la population et à partir de $Round.Two$, d'estimer le nombre de 1. Ces deux estimations sont ensuite combinées pour une estimation globale de la distribution des réponses de la population. Les auteurs affirment que cette technique peut être étendue à des bases de données arbitrairement grandes, car l'erreur introduite est constante et non linéaire par rapport aux nombres de profils. Leur technique permet donc d'extrapoler des informations d'une population à partir d'un échantillon tout en respectant la vie privée des individus inclus dans l'échantillon.

3.1.2 Suppression locale

La suppression locale ou ponctuelle est appliquée sur les données pour complètement enlever les informations identifiantes sans trop nuire à l'utilité des données. Par exemple, dans le cas où les données sont des trajectoires d'individus qui sont définies comme une suite de localisations (données GPS, « check-ins », etc.) or-

données dans le temps, le fait de supprimer un lieu d'une trajectoire uniquement visité par *une* personne aura très peu d'impact sur le flux global des mouvements de la population. Un exemple d'article utilisant cette technique est (Terrovitis *et al.*, 2017) dans lequel les auteurs proposent deux algorithmes d'assainissement de trajectoires. Tout d'abord, les localisations identifiantes sont détectées par un algorithme simulant un attaquant qui confronte les trajectoires de la base de données avec des connaissances auxiliaires qu'un adversaire peut avoir. Nous parlerons dans ce cas de localisations *problématiques*. Le premier algorithme proposé supprime globalement toutes les localisations étiquetées comme une menace à la vie privée par son caractère identifiant. C'est-à-dire que si un lieu comme un café est une localisation fréquente pour un utilisateur x mais une localisation identifiante pour un utilisateur y, le café sera omis du modèle global et donc de toutes les trajectoires, y compris celles de l'utilisateur x. Cet algorithme naïf peut avoir des effets désastreux sur l'utilité des données et les résultats de l'article le démontrent en offrant les pires performances sur le plan de l'utilité lorsque comparés aux résultats des autres techniques développées dans le même article. Le deuxième algorithme proposé utilise une approche de suppression locale plutôt que globale. La suppression se fait alors plutôt au niveau des trajectoires qu'au niveau de l'ensemble des données. L'algorithme de suppression locale commence de la même façon que son analogue global en étiquetant les localisations problématiques. Par la suite, toute trajectoire contenant une localisation problématique est analysée par l'algorithme. Plus précisément, une localisation d'une trajectoire problématique sera éliminée seulement si son retrait résout un maximum de problèmes globalement, c'est-à-dire qu'une fois supprimée, la trajectoire minimise le nombre de trajectoires problématiques. Cette heuristique est plus coûteuse en temps d'exécution, mais réduit approximativement de moitié le taux de suppression. Cette dernière technique illustre bien le principe de suppression locale à des fins de protection de la vie privée.

3.1.3 Généralisation

La généralisation cherche globalement à changer la granularité d'attributs pour rendre l'information plus grossière, ou de manière équivalente moins précise. Un exemple d'une technique qui se base sur la généralisation d'attributs pour assainir des micro-données est la généralisation « cross-bucket » (Li *et al.*, 2017). Leur méthode est fondée sur les modèles de k-anonymité et de l-diversité, qui sont présentés dans le chapitre 2. La généralisation « cross-bucket » combine deux principes de généralisation d'attributs : la généralisation classique, qui sous certaines contraintes peut satisfaire le modèle du k-anonymité et la « bucketisation », qui correspond au partitionnement en groupes diversifiés contenant des valeurs d'attributs diverses pour briser les liens existants entre attributs sensibles et quasi-identifiants. Le tableau 3.5 illustre la différence entre la généralisation et la bucketisation ; dans cet exemple on suppose que l'attribut sensible est *education*. Dans le tableau 3.5b les regroupements contiennent tous au moins 3 individus similaires indistinguables (3-anonyme) alors que dans le tableau 3.5c les « buckets » contiennent 3 individus dissemblables avec 3 valeurs d'attributs sensibles possibles (3-diverse). Dans cet exemple, il faut lire une cellule sans séparation comme un tuple, donc dans 3.5c la colonne *education* se lit : (Bachelors, Masters, HS-grad), (Bachelors, 9th, 11th) et (Masters, Bachelors, HS-grad).

La (k,l)-généralisation « cross-bucket » combine la généralisation et la bucketisation en créant des groupes k-anonymes puis en créant des buckets l-diverses qui doivent eux aussi être k-anonymes. Un exemple simple de généralisation « cross-bucket » est donné par le tableau 3.5d. Dans cet exemple, chaque profil et chaque bucket apparaît au moins deux fois pour répondre au 2-anonymat, et chaque bucket contient au moins deux profils dissemblables pour répondre à la 2-diversité. L'heuristique pour trouver les groupes et les « buckets » optimaux de la généra-

	age	*sex*	*education*
1	39	Male	Bachelors
2	50	Male	Bachelors
3	38	Male	HS-grad
4	53	Male	11th
5	28	Female	Bachelors
6	37	Female	Masters
7	49	Female	9th
8	52	Male	HS-grad
9	31	Female	Masters

(a) Micro-données originelles.

	age	*sex*	*education*
5	[25 - 37]	Female	Bachelors
6	[25 - 37]	Female	Masters
9	[25 - 37]	Female	Masters
1	[38 - 49]	*	Bachelors
3	[38 - 49]	*	HS-grad
7	[38 - 39]	*	9th
2	[50 - 62]	Male	Bachelors
4	[50 - 62]	Male	11th
8	[50 - 62]	Male	HS-grad

(b) Exemple de généralisation 3-anonyme.

	age	*sex*	*education*
5	28	Female	Bachelors
9	31	Female	Masters
3	38	Male	HS-grad
1	39	Male	Bachelors
7	49	Female	9th
4	53	Male	11th
6	37	Female	Masters
2	50	Male	Bachelors
8	52	Male	HS-grad

(c) Exemple de bucketisation 3-diverse.

	age	*sex*	*education*
5	[20 - 39]	Female	Bachelors
8	[40 - 59]	Male	HS-grad
3	[20 - 39]	Male	HS-grad
7	[40 - 59]	Female	9th
6	[20 - 39]	Female	Masters
4	[40 - 59]	Male	11th
9	[20 - 39]	Female	Masters
2	[40 - 59]	Male	Bachelors

(d) Exemple de généralisation « cross-bucket » (2-anonyme, 2-diverse).

TABLEAU 3.5 – Généralisation, bucketisation et généralisation « cross-bucket ».

lisation « cross-bucket » est complexe et requiert plusieurs sous-routines. Ainsi, l'application d'une telle méthode n'est pas triviale et peut donner des résultats extrêmement grossiers, c'est-à-dire réduire un espace de plusieurs valeurs à deux, voire à une seule valeur.

3.2 Méthodes perturbantes

Les techniques perturbantes que sont la permutation, la micro-agrégation et l'ajout de bruit diffèrent des non-perturbantes en venant modifier directement les profils et leurs valeurs d'attributs. Alors que les méthodes non-perturbantes ont plutôt un effet réducteur au sens où les trois techniques présentées précédemment réduisent la quantité de l'information, les techniques perturbantes viennent modifier les valeurs ou la corrélation des attributs, pour accroître le niveau d'incertitude au

niveau des attributs sensibles. Par exemple, la *permutation* échange des valeurs d'un attribut entre différents profils, ce qui modifie directement la corrélation entre cet attribut et les autres attributs, comme il est illustré dans le tableau 3.6. La permutation se fait habituellement sur les quasi-identifiants pour briser certains liens pouvant mener à une ré-identification. La micro-agrégation, comme dans l'exemple du tableau 3.7, crée des groupes de profils indistinguables entre eux. L'ajout de bruit, comme son nom l'indique, ajoute des valeurs aléatoires aux données. Cette technique est donc évidemment applicable principalement sur les attributs numériques. L'ajout de bruit sur les données catégoriques peut se traduire en réponse aléatoire, comme introduit dans la partie 3.1.1. Le bruit ajouté peut être corrélé ou non, un bruit étant dit corrélé s'il préserve la corrélation des valeurs initiales à l'aide de la matrice de covariance (Mivule, 2013). La méthode du bruit laplacien est très courante dans la littérature à propos de la CD. Ce type de bruit aléatoire est non-corrélé et est généré à partir d'une fonction de densité de probabilités. Dans l'exemple du tableau 3.8, le bruit ajouté aux attributs *age* et *hours-per-week* est tiré de deux distributions de Laplace de paramètres $\mu = 0$ et $b = 10$, aussi dite centrée en 0 et d'échelle 10, et de paramètres $\mu = 0$ et $b = 5$, respectivement.

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
1	39	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	**14084**	40	United-States
2	50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	White	Male	0	13	United-States
3	38	Private	HS-grad	Divorced	Handlers-cleaners	White	Male	0	40	United-States
4	53	Private	11th	Married-civ-spouse	Handlers-cleaners	Black	Male	0	40	United-States
5	28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Black	Female	0	40	**Jamaica**
6	37	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	0	40	United-States
7	49	Private	9th	Married-spouse-absent	Other-service	Black	Female	0	16	**Cuba**
8	52	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	White	Male	0	45	United-States
9	31	Private	Masters	Never-married	Prof-specialty	White	Female	**2174**	50	United-States

TABLEAU 3.6 – Exemple de permutation.

Les sous-parties suivantes introduisent quelques articles permettant d'illustrer les méthodes décrites plus tôt par quelques exemples concrets d'algorithmes d'assainissement.

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
3	[25 - 38]	Private	HS or higher	*	*	*	*	*	[40 - 50]	*
5	[25 - 38]	Private	HS or higher	*	*	*	*	*	[40 - 50]	*
6	[25 - 38]	Private	HS or higher	*	*	*	*	*	[40 - 50]	*
9	[25 - 38]	Private	HS or higher	*	*	*	*	*	[40 - 50]	*
1	[39 - 49]	State-gov or Private	*	*	Admin or Other	*	*	[0 - 5000]	[10 - 40]	*
7	[39 - 49]	State-gov or Private	*	*	Admin or Other	*	*	[0 - 5000]	[10 - 40]	*
2	[50 - 62]	Self-emp or Private	*	Married	Exec or Handlers	*	Male	[0 - 5000]	*	United-States
4	[50 - 62]	Self-emp or Private	*	Married	Exec or Handlers	*	Male	[0 - 5000]	*	United-States
8	[50 - 62]	Self-emp or Private	*	Married	Exec or Handlers	*	Male	[0 - 5000]	*	United-States

TABLEAU 3.7 – Exemple de micro-agrégation.

	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	***hours-per-week***	*native-country*
1	**39 - 1 = 38**	State-gov	Bachelors	Never-married	Adm-clerical	White	Male	2174	**40 + 8 = 48**	United-States
2	**50 + 2 = 52**	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	White	Male	0	**13 - 14 = 0**	United-States
3	**38 - 18 = 20**	Private	HS-grad	Divorced	Handlers-cleaners	White	Male	0	**40 + 5 = 45**	United-States
4	**53 - 4 = 49**	Private	11th	Married-civ-spouse	Handlers-cleaners	Black	Male	0	**40 + 1 = 41**	United-States
5	**28 - 13 = 15**	Private	Bachelors	Married-civ-spouse	Prof-specialty	Black	Female	0	**40 + 0 = 40**	Cuba
6	**37 - 1 = 36**	Private	Masters	Married-civ-spouse	Exec-managerial	White	Female	0	**40 - 5 = 35**	United-States
7	**49 - 5 = 44**	Private	9th	Married-spouse-absent	Other-service	Black	Female	0	**16 - 2 = 14**	Jamaica
8	**52 + 25 = 77**	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	White	Male	0	**45 + 0 = 45**	United-States
9	**31 + 0 = 31**	Private	Masters	Never-married	Prof-specialty	White	Female	14084	**50 + 0 = 50**	United-States

TABLEAU 3.8 – Exemple d'ajout de bruit non-corrélé.

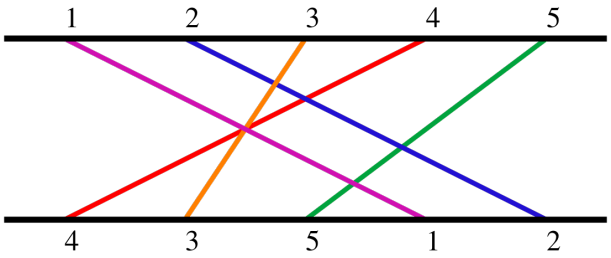
3.2.1 Permutation

La permutation de données est une technique plutôt simple où deux valeurs d'attributs sont interchangées. Les différentes approches existantes se distinguent en affinant le choix des valeurs à permuter. Par exemple, dans l'article (Upadhyay *et al.*, 2018) les auteurs introduisent une technique de permutation en 3 dimensions. Intuitivement, une permutation déplace des valeurs de manière chaotique, telle qu'illustrée dans la figure 3.1 [1]. Une rotation peut être vue comme une permutation au sens où une rotation décale toutes les valeurs de manière dirigée, donc

$$(1 \to 2)(2 \to 3)(3 \to 4)(4 \to 5)(5 \to 1)$$

La technique décrite dans (Upadhyay *et al.*, 2018) regroupe les attributs en groupe de 3 et chaque triplet de valeurs subit une rotation dans l'espace de 3 dimensions où chaque attribut correspond à une dimension. Bien qu'il faille préalablement

1. Tirée de Wikipedia : `https://en.wikipedia.org/wiki/Permutation_graph`.



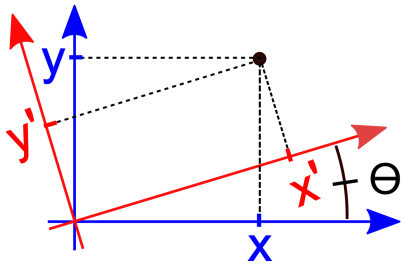
$(1 \to 4)(2 \to 5)(3 \to 2)(4 \to 1)(5 \to 3)$

FIGURE 3.1 – Exemple d'une permutation de 5 objets.

normaliser les données pour obtenir des espaces de rotations uniformes, la simple application d'une rotation se fait par multiplication matricielle et est donc très peu complexe. La complexité de leur algorithme découle du choix de l'angle des rotations. Plus précisément, leur heuristique calcule plusieurs rotations et choisit la plus « sécuritaire ». Un angle de rotation est considéré comme sécuritaire s'il introduit assez de dispersion dans les données pivotées, selon un seuil de protection. La dispersion est calculée à l'aide de la variance et peut être interprétée comme étant une mesure de dissemblance entre les données. Ainsi, il faut choisir un angle qui introduit assez de différence dans les données pour tous les attributs pivotés. Ce type de permutation permet de conserver la distance euclidienne entre les profils, ce qui est une propriété importante en fouille de motifs pour obtenir une utilité élevée. Ainsi, deux profils proches (c'est-à-dire dont la distance euclidienne est petite) seront encore semblables après permutation. Illustrons un exemple simple en 2 dimensions avec la figure 3.2 [2].

Un autre type de permutation, la permutation de rang, assigne d'abord aux données (ordonnées) un rang, puis pour un certain pourcentage des données, permute les valeurs d'un attribut (ou plusieurs) entre deux profils de rang i et j choisis aléatoirement (Domingo-Ferrer *et al.*, 2016). Une des limitations majeures de

2. Tirée de Wikipedia : `https://en.wikipedia.org/wiki/Rotation_of_axes`.



Si l'axe X représente l'attribut `age` *et l'axe Y représente l'attribut* `hours-per-week` *de l'ensemble Adult, une rotation Θ telle qu'illustrée modifie le couple de valeurs $(x, y) = (35, 35)$ à $(x', y') = (40, 30)$. La longueur du vecteur reste toujours la même, d'où les distances euclidiennes entre profils sont conservées.*

FIGURE 3.2 – Exemple d'une rotation en dimension 2 appliquée sur des données.

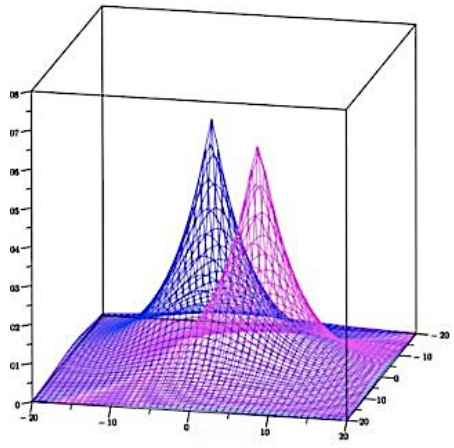
cette approche est la difficulté d'imposer un ordre de sur des données nominales comme le sexe, la religion, l'ethnie, etc. sans introduire une perte d'information. Dans l'article (Rodriguez-Garcia *et al.*, 2019), les auteurs définissent une stratégie d'ordres sémantiques pouvant s'appliquer aux données nominales dans les bases de données. Sans entrer dans les détails, les valeurs nominales sont rattachées à une ontologie, c'est-à-dire à un graphe de concepts. Ce graphe va être utilisé pour introduire une distance entre les valeurs nominales et ainsi un ordre sur les données venant résoudre le problème de rangs sur des données nominales.

3.2.2 Micro-agrégation

La micro-agrégation consiste en la construction de groupes, aussi appelés « clusters » en anglais, contenant au moins k profils. Cette approche ne doit pas être confondue avec la k-anonymité, qui est un modèle de respect de la vie privée, et non pas une technique d'assainissement. Ainsi, la k-anonymité est le résultat de l'application de la micro-agrégation. Comparée à la généralisation, la micro-agrégation

est fondamentalement semblable, mais perturbe l'ensemble des attributs pour former des regroupements. La micro-agrégation se fait soit de façon univariée ou multivariée. La méthode univariée est la plus naïve dans le sens où chaque attribut est traité séparément. Pour que les corrélations existantes entre les attributs soient conservées, la procédure ordonne la base de données selon le premier attribut à agréger puis crée les groupes avec au moins k profils. Ensuite, attribut par attribut, les valeurs sont regroupées sous une même valeur ou un représentant. Le fait de trier puis agréger préserve la corrélation de rangs entre les attributs (Domingo-Ferrer *et al.*, 2016). Cependant, la naïveté de la méthode offre une faible protection comme le démontrent les auteurs dans l'article (Domingo-Ferrer *et al.*, 2002). Les auteurs concluent aussi que cette méthode cause une perte d'utilité trop importante étant donné le niveau d'anonymat atteint. Ce problème est présent dans l'exemple du tableau 3.7 où plusieurs attributs ont dû être réduits à une valeur arbitraire (*****), causant une importante perte d'information.

La micro-agrégation multivariée peut être atteinte par exemple en réduisant la dimension des attributs à l'aide de projections pour ensuite effectuer la micro-agrégation univariée sur les nouvelles variables. Il est aussi possible d'utiliser des heuristiques comme le MDAV (« Maximum Distance to Average Vector » en anglais) (Domingo-Ferrer *et al.*, 2006) qui essaie de résoudre le problème de trouver des partitions de dimensions au moins k avec un minimum d'hétérogénéité parmi tous les attributs. Un récent article (Abidi *et al.*, 2020) définit une approche à la micro-agrégation multivariée qui est hybride, dans le sens où les données sont préalablement segmentées en blocs disjoints de manière à ce que les valeurs des attributs quasi-identifiants soient dissemblables. Le principe de micro-agrégation est ensuite appliqué aux blocs disjoints de manière indépendante, c'est-à-dire qu'il n'y a pas de k fixé pour l'ensemble des données. De plus, une heuristique détermine la taille k optimale des groupes qui préserve la diversité des attributs à protéger.



(a) Deux densités bivariées de Laplace.

(b) Les niveaux d'indistinguabilité engendrés les distributions de Laplace.

FIGURE 3.3 – Exemple de la géo-indistinguabilité.

Cette technique hybride respecte le modèle de la k-anonymité, tout en évitant les attaques par homogénéité des données avec la conservation de la diversité des attributs sensibles. Les résultats obtenus démontrent toutefois que l'approche de l'article produit une perte d'information supérieure à la méthode du MDAV.

3.2.3 Ajout de bruit

L'addition de bruit dans les données est une technique très répandue et souvent conjointement utilisée avec d'autres méthodes d'assainissement. L'ajout de bruit non-corrélé est ainsi couramment utilisé pour introduire du déni plausible comme le fait la technique connue de la géo-indistinguabilité (Andrés *et al.*, 2013), qui ajoute du bruit non-corrélé aux positions de géolocalisation. Le bruit ajouté se fait en respect du modèle de la CD, qui est expliqué dans le chapitre 2, en ajoutant aux positions (LNG, LAT) un bruit tiré aléatoirement de la distribution bivariée de Laplace. Un exemple de distribution bivariée de Laplace est donné dans la figure 3.3a. La figure 3.3b illustre le principe de la géo-indistinguabilité sur une carte, les cercles délimitant les niveaux d'indistinguabilité occasionnés par différentes

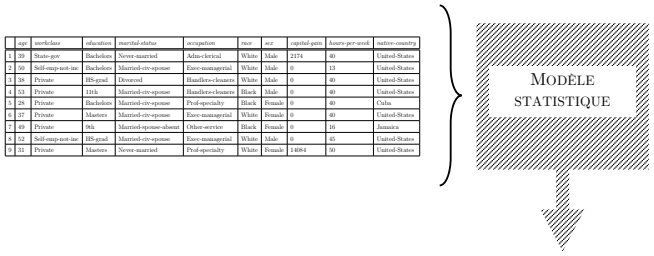
distributions de Laplace. Les deux figures sont extraites de l'article en question. Bien que l'ajout de bruit sur une position géographique offre une bonne protection ponctuelle, étant donné la nature très corrélée des déplacements d'un individu, le niveau de protection baisse avec le nombre de localisations divulguées. Par exemple, croiser les données temporelles avec les données de localisations bruitées peut permettre l'inférence du lieu de résidence, en analysant tous les points captés entre 20 heures et 6 heures du lendemain. Ces points seront nécessairement centrés autour d'une localisation, considérant le type de bruit ajouté.

3.3 Méthodes génératives

Les approches génératives condensent l'information en modèles probabilistes et à partir de ces modèles, synthétisent de nouvelles données. Essentiellement, l'objectif de ce processus est de briser le lien entre données et identités. Parmi les modèles mathématiques qui condensent l'information des données se trouvent les réseaux bayésiens (Jensen, 1997), les modèles de Markov (Stewart, 1994) et les réseaux de neurones (Anderson, 1997), pour en nommer quelques-uns. Les méthodes génératives se séparent principalement en deux classes : partielle et complète. La génération partielle fixe certaines valeurs d'attributs et complète l'autre partie à l'aide des modèles tandis que la génération complète génère la totalité des valeurs à partir des modèles. Autrement dit, la génération partielle dépend des représentations statistiques et des profils tandis que la génération complète ne dépend que des représentations. Les figures 3.4 et 3.5 illustrent les deux processus où les valeurs d'attributs en **gras** indiquent des valeurs générées à partir d'un modèle.

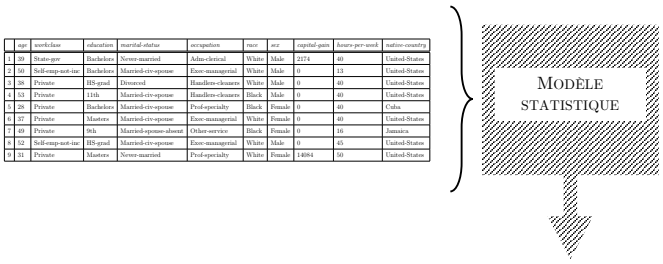
	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
1	-	-	-	Never-married	Adm-clerical	-	-	2174	-	-

+



	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
-	**25**	**Federal-gov**	**Some-college**	Never-married	Adm-clerical	**Asian-Pac-Islander**	**Female**	2174	**20**	**Philippines**

FIGURE 3.4 – Exemple de génération partielle.



	age	*workclass*	*education*	*marital-status*	*occupation*	*race*	*sex*	*capital-gain*	*hours-per-week*	*native-country*
-	**25**	**Federal-gov**	**Some-college**	**Separated**	**Adm-clerical**	**Asian-Pac-Islander**	**Female**	**0**	**20**	**Philippines**

FIGURE 3.5 – Exemple de génération complète.

3.3.1 Génération partielle

Le cadre défini par Vincent Bindschaedler dans (Bindschaedler et Shokri, 2016; Bindschaedler *et al.*, 2017) et subséquemment dans sa thèse (Bindschaedler, 2018) définit des modèles génératifs applicables à de nombreux types de données en plus de définir des tests de protection de la vie privée pour éviter de produire des profils synthétiques à haut pouvoir d'inférence. Les modèles définis par Bindschaedler peuvent être classés comme étant des modèles génératifs partiels étant donné qu'ils produisent des données à partir de profils graines (« seed-based »). Pour un profil graine fixé, un certain nombre de valeurs d'attributs resteront inchangés alors que

le reste des valeurs seront générés à partir du modèle. L'article (Bindschaedler *et al.*, 2017) propose un modèle bayésien pour apprendre la distribution et la corrélation entre les attributs. Comme la structure du réseau bayésien peut en elle-même être source de fuite d'informations, cette structure est construite à partir de la matrice de corrélation estimée par une fonction de calcul de l'entropie qui est rendue différentiellement-privée, ce qui induit une structure respectueuse de la vie privée. Sous-jacent à l'arbre bayésien se trouve des distributions ou les probabilités conditionnelles qui permettent de modéliser individuellement les attributs. Comme le modèle se veut différentiellement-privé, l'apprentissage des paramètres des distributions se fait aussi en respect à la CD par l'estimation bruitée des distributions à l'aide d'histogrammes. Le bruit référé est celui ajouté par les mécanismes différentiellement-privés.

Bindschaedler applique aussi la génération de données synthétiques respectueuses de la vie privée aux données de géolocalisation et de trajectoires. Dans l'article (Bindschaedler et Shokri, 2016), les auteurs proposent un modèle séquentiel de chaînes de Markov pour modéliser la mobilité en utilisant la sémantique des lieux pour construire un modèle de mobilité plutôt que les points de géolocalisation. Autrement dit, leur technique transforme les trajectoires en suite de points sémantiques, par exemple `Maison` → `Travail` → `Loisir` → `Épicerie` → `Maison`, pour ensuite construire un modèle de mobilité. Le modèle de mobilité se divise en deux parties : un modèle de Markov qui capte les transitions possibles entre les points et des regroupements sémantiques (« semantic clusters ») qui rassemblent tous les

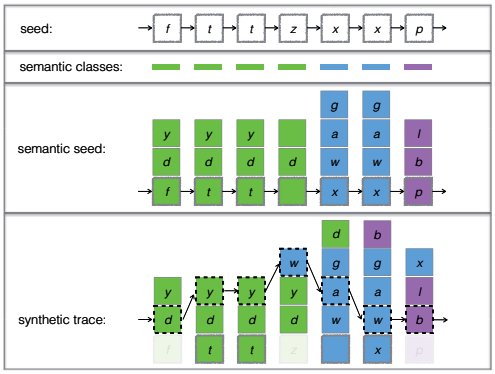


FIGURE 3.6 – Algorithme de génération de traces synthétiques.

points ayant des sémantiques similaires. À partir du modèle et d'un profil graine, il est possible de générer de nouvelles trajectoires en sélectionnant aléatoirement des points dans les mêmes regroupements sémantiques que le profil graine. Les points sémantiques sont ensuite re-mappés en points GPS en sélectionnant une localisation parmi les points de même sémantique. La figure 3.6, tirée du même article, montre l'ébauche de la génération de trajectoires. Afin de garder un maximum de vraisemblance avec la mobilité réelle, le décodage de points sémantiques vers de points de géolocalisation se fait en tenant compte du réalisme géographique en utilisant l'algorithme de Viterbi pour trouver une trajectoire géographique plausible dans le graphe sémantique. L'ajout d'aléa dans l'algorithme de Viterbi permet la génération non-déterministe de trajectoires respectueuses de la vie privée.

Dans les deux méthodes précédentes, la protection de la vie privée repose principalement sur la construction non-déterministe de modèles bruités en ajoutant de l'aléa (par mécanisme différentiellement-privé) dans les paramètres de construction. Cependant, la protection repose aussi un test sur les données synthétiques où une donnée synthétique doit être semblable à au moins k profils réels. Non seulement ce test introduit du déni plausible, mais contraint aussi les données à être réalistes, ce qui est une grande force des méthodes génératives de Bindschaedler. Cependant, une des limites de ces approches est la complexité des modèles et leur temps de création. En effet, pour un ensemble de données modestes (30 000 profils ou moins), le temps d'exécution se calcule en jours (Gursoy *et al.*, 2018). Il s'agit d'une des raisons pour laquelle le cadre défini par Bindschaedler n'a pas été retenu dans notre étude comparative pour évaluer notre méthode.

3.3.2 Génération complète

Comme mentionné précédemment, les méthodes génératives complètes ne dépendent pas des données originales lors de la génération de nouvelles données et ont l'avantage sur les autres techniques d'assainissement de rompre complètement le lien entre données et identités. AdaTrace (Gursoy *et al.*, 2018) est une méthode de génération de trajectoires introduisant de la CD dans les modèles. Cette méthode divise le problème de génération de trajectoires synthétiques respectueuses de la vie privée en 4 sous-modèles. Le premier modèle discrétise l'espace géographique en une grille. L'utilité et la sécurité des données dépendent beaucoup de la granularité de la grille, ce qui explique pourquoi AdaTrace utilise une grille adaptative à multiples niveaux, qui permet une granularité plus fine dans les zones plus denses. Les densités des zones sont calculées de manière différentiellement-privée de sorte à avoir une distribution des localisations qui est elle aussi différentiellement-privée. Le deuxième sous-modèle modélise la mobilité à l'aide d'une chaîne de Markov qui est représentée par une matrice où l'entrée $[i][j]$ désigne la probabilité d'aller à la zone j à partir de la zone i. Un bruit laplacien est ajouté aux probabilités de la chaîne de Markov pour assurer la CD.

Le troisième sous-modèle sert à préserver l'association entre les points de départ et d'arrivée. Ce modèle guide le modèle de Markov en lui fournissant des points de départ et d'arrivée réalistes. Ce sous-modèle est optionnel dans le sens où il est possible d'attribuer à toutes les zones de la grille la même probabilité d'être un point de départ ou d'arrivée, cependant, la distribution uniforme est loin d'être similaire à la distribution réelle des points d'arrivée et de départ. Ainsi, pour toute paire de zones (C_{start}, C_{end}), le nombre de trajectoires avec comme point de départ C_{start} et comme point d'arrivée C_{end} est calculé de manière différentiellement-privée. Finalement, le quatrième sous-modèle vient condenser la distribution de

la longueur des trajectoires sous une distribution connue (ex. loi de Poisson, loi normale ou loi exponentielle). Les paramètres des distributions sont calculés de manière différentiellement-privée. Par exemple la loi exponentielle a comme paramètre λ, estimé par la moyenne de la variable en question, qui sera calculé avec la confidentialité différentielle. Avec AdaTrace, la synthétisation de nouvelles traces commence par le choix aléatoire d'une paire de zones (C_{start}, C_{end}), puis choisit une longueur de trajectoire pour la synthétisation et complète la trajectoire à l'aide du modèle de Markov (Gursoy *et al.*, 2018). Bien que cette approche offre des résultats intéressants, la gestion d'un budget ϵ de CD pour 4 modèles est plutôt complexe. Alors que les articles traitant de la CD comparent habituellement différents niveaux de budget variant de 1 à 0.001, les auteurs n'affichent aucun résultat de la sorte, venant contester le niveau de protection impliqué puisque le niveau de bruit ajouté n'est jamais précisé. Un autre aspect discutable est l'ajout de trois mécanismes de protection qui, comme définis dans l'article, utilisent les données originales pour filtrer des données à risque. Ce genre de mécanisme *ad hoc* va à l'encontre de la CD et fait perdre la garantie qu'apporte le modèle, ce qui pourrait être un vecteur important d'attaque.

Une des techniques réputées dans le domaine de la synthétisation de données respectueuses de la vie privée est la méthode PrivBayes (Zhang *et al.*, 2017). Cette méthode se démarque des autres solutions existantes en définissant un algorithme robuste et différentiellement-privé modélisant la corrélation entre les attributs et les distributions marginales de manière confidentielle. PrivBayes calcule d'abord les probabilités conditionnelles de manière différentiellement-privée puis construit, toujours de manière différentiellement-privée, un réseau bayésien capturant les dépendances entre les attributs. Deux exemples de réseaux bayésiens construits sur l'ensemble Adult sont donnés dans la figure 3.7 extraite de l'article (Ping *et al.*, 2017). Dans ces exemples, une flèche partant d'un noeud vers un autre implique

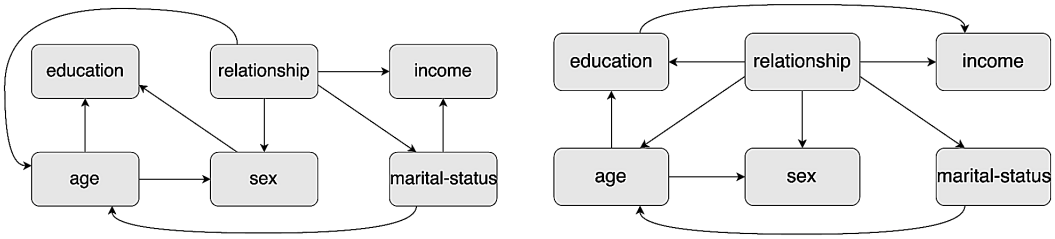


FIGURE 3.7 – Exemple de deux réseaux bayésiens construit sur les données de Adult.

une dépendance conditionnelle (non-nulle), c'est-à-dire (`age` $\rightarrow$ `sex`) veut dire que les valeurs de l'attribut `sex` dépendent des valeurs de l'attribut `age` donc des dépendances conditionnelles d'allures $P[\texttt{sex} = M|\texttt{age} = 25]$. On dit alors que le noeud `age` est le parent du noeud `sex`. Le réseau bayésien bruité est créé de sorte que le choix des arêtes à partir d'un noeud ne soit pas déterministe, en utilisant le mécanisme exponentiel de la CD (McSherry et Talwar, 2007). L'algorithme classique de construction de réseau bayésien ajoutera une arête entre deux noeuds si l'information mutuelle (ou une autre fonction de score choisie) est maximale. Dans PrivBayes, la fonction de score utilisée avec le mécanisme exponentiel pour sélectionner une paire d'attributs se base sur la distance entre deux distributions, soit la distribution jointe des deux variables et la distribution jointe indépendante. Un des problèmes majeurs avec PrivBayes provient de la construction exponentiellement coûteuse en temps due au calcul du réseau bayésien. En effet, le nombre d'attributs influence beaucoup le temps d'exécution. Ainsi, bien que PrivBayes offre une bonne performance avec un nombre important de données, celle-ci se dégrade avec le nombre d'attributs. PrivBayes, étant un acteur important dans le domaine, sera un des algorithmes qui sera comparé avec la technique développée pour ce mémoire.

Avec l'essor de l'apprentissage profond (Kelleher, 2019), il va de soi que certaines techniques de ce domaine ont été utilisées et se sont démarquées dans le domaine

de la génération de données. Parmi les techniques les plus sophistiquées, les réseaux adversériels génératifs (Goodfellow *et al.*, 2014) (« generative adversarial networks » ou GANs) sont devenus des références pour la génération d'images réalistes et sont de plus en plus utilisés pour la génération de micro-données. Récemment, PATE-GAN (Jordon *et al.*, 2019) s'est démarqué en joignant l'approche GAN à la méthode PATE (Papernot *et al.*, 2016) (*Private Aggregation of Teacher Ensembles* en anglais). Brièvement, l'approche GAN est constituée d'un générateur qui capture la distribution originale des données et un discriminateur qui estime l'appartenance à la distribution originale d'un échantillon. Il est possible de voir l'ensemble comme un jeu à somme nulle où le générateur essaie de déjouer le discriminateur avec les échantillons générés tandis que le discriminateur essaie d'inférer si un échantillon est réel ou synthétique (Goodfellow *et al.*, 2014).

Originellement, PATE est un modèle de classification différentiellement-privé qui sépare la tâche de classification en k « professeurs ». Dans le modèle PATE, les données sont premièrement découpées en k sous-ensembles disjoints sur lesquels k classifieurs « professeurs » sont entraînés de manière disjointe. Pour classifier une nouvelle donnée, le modèle agrège de manière différentiellement-privée les résultats des k classifieurs sur la nouvelle donnée (Papernot *et al.*, 2016). Étant donné que les professeurs ont appris sur les vraies données, il est impossible d'accéder à leurs paramètres sans compromettre la CD. Ainsi, il est impossible d'implémenter une infrastructure GAN avec un modèle PATE directement tel quel. Pour cette raison PATE-GAN utilise une extension de l'infrastructure qui inclut un classifieur « élève », qui sera entraîné sur les résultats des « professeurs » (Papernot *et al.*, 2016). Comme le classifieur « élève » a appris sur des résultats différentiellement-privés, il donc est possible d'accéder et de modifier ses paramètres, ce qui explique que PATE est un excellent candidat comme discriminateur d'un GAN. Les auteurs de l'article (Jordon *et al.*, 2019) comparent leur méthode à CDGAN (Xie *et al.*,

2018), une des premières méthodes introduisant la CD dans un modèle GAN pour la génération de données se voulant être compétitif à la méthode PATE (Papernot *et al.*, 2016). L'article montre que les performances sont nettement supérieures à CDGAN, principalement lorsque le bruit ajouté est important (dû à la restriction du budget de CD). Malheureusement, le code de PATE-GAN n'étant pas disponible, il est donc impossible de comparer notre approche à la leur. De plus, un facteur à considérer avec l'apprentissage profond est l'opacité entourant le modèle, ce qui rend impossible la compréhension du raisonnement qui a permis de le construire.

Récemment, la théorie des copules a fait surface dans le domaine de la protection de la vie privée. Pour rappel, les copules, qui ont été introduites de manière plus formelle dans le chapitre 1, sont des modèles mathématiques à plusieurs variables captant la dépendance entre attributs. L'article (Kulkarni *et al.*, 2018) fut celui qui a inspiré la recherche et la méthode développée dans le présent mémoire. Dans cet article, les auteurs comparent l'utilisation de copules pour la génération de données synthétiques à plusieurs approches d'apprentissage profond comme les réseaux de neurones récurrents (RNNs) et les GANs. Les résultats démontrent que les copules sont de puissants outils de modélisation permettant une synthétisation des données fidèles, même pour des données de mobilité. De plus, les auteurs utilisent une librairie ouverte du langage de programmation R, rendant accessible l'utilisation de copules. Bien que les auteurs restent à un niveau superficiel dans l'explication de leur méthodologie et leur analyse de la vie privée, les résultats ont invité une étude plus approfondie sur l'utilisation de ces outils statistiques. Une des premières approches génératives privées utilisant les copules fut celle de Li, Xiong et Jiang (2014), incorporant la CD dans le modèle. La méthode développée utilise les copules gaussiennes multivariées pour estimer la dépendance entre les attributs, comme dans l'exemple 1.3. La modélisation des données avec une copule

gaussienne se fait à partir des densités marginales f_i (pour reprendre les notations de l'exemple) et de la densité jointe gaussienne ($\hat{c_X}$). Les auteurs proposent d'estimer les densités marginales à l'aide d'histogrammes (auxquels est ajouté du bruit différentiellement-privé) et d'estimer la densité jointe à l'aide d'une matrice de corrélation (voir la définition 3.4 dans (Li *et al.*, 2014)). Deux approches pour calculer la matrice de corrélation de manière différentiellement-privée sont proposées : une méthode utilisant le maximum de vraisemblance (MLE) et une autre avec la corrélation de rang de Kendall. La première méthode avec le MLE repose sur la technique introduite dans (Dwork et Smith, 2010) pour le calcul du MLE de manière différentiellement-privée. La seconde méthode estime la corrélation à l'aide de la formule de Kendall avec ajout de bruit laplacien. L'article récent de Asghar *et al.* (2019) est très semblable avec la même architecture de modélisation différentiellement-privée via les copules gaussiennes multivariées, mais généralise l'approche en transformant préalablement les données sous forme binaire, ce qui étend l'utilisation de copules aux attributs nominaux. La transformation sous forme binaire des données simplifie grandement les calculs, mais fait exploser le nombre d'attributs, pouvant rendre la tâche difficile. Comme il a été mentionné dans le chapitre 1, les copules gaussiennes multivariées offrent une estimation grossière et parfois erronée (dans les cas des dépendances caudales) de la structure de dépendance. Les copules vignes ont été développées dans l'optique d'échapper à cette limitation.

Les techniques basées sur les copules se sont multipliées depuis les dernières années, notamment les techniques de Sun, Cuesta-Infante et Veeramachaneni (2019) et de Tagasovska, Ackerer et Vatter (2019) mettant en vedette des algorithmes d'apprentissage profond. La méthode développée dans l'article (Sun *et al.*, 2019) utilise les vignes pour découper le problème de modélisation multivariée en plusieurs sous-modèles bivariés. Pour rappel, les copules vignes ont été introduites

dans le chapitre 1. Au lieu d'utiliser l'approche classique pour construire une *vigne* via l'algorithme de Dissmann, les auteurs utilisent plutôt un modèle d'apprentissage profond appelé LSTM pour *Long Short Term Memory networks* qui peut être sommairement résumé à un réseau de neurones récurrents avec la capacité d'apprendre sur le long terme. L'utilisation d'apprentissage profond pour la construction de vignes semble préférable aux algorithmes gloutons classiques en offrant un maximum de log-vraisemblance (« log-likelihood ») et de meilleurs scores de classification.

La technique développée dans (Tagasovska *et al.*, 2019) est basée sur les auto-encodeurs pour réduire la dimension des données pour facilité l'apprentissage des copules. Brièvement, un auto-encodeur est composé de deux segments, un réseau de neurones encodeur qui transforme les données de dimension n sur un espace latent de dimension $l < n$, et un décodeur qui traduit les données de l'espace latent en données de l'espace original. Un auto-encodeur est entraîné pour synthétiser les données d'entrées en minimisant la perte d'information dans le processus. Les auteurs de l'article (Tagasovska *et al.*, 2019) introduisent un modèle génératif de copule entre l'encodeur et le décodeur, c'est-à-dire que la copule apprend un modèle sur l'espace latent et le décodeur produit de nouvelles données à partir des observations générées par la copule. La figure 3.8, tirée du même article, résume la méthode définie dans (Tagasovska *et al.*, 2019).

Les deux techniques précédentes n'offrent cependant aucune protection des données et se concentrent seulement sur la génération de données synthétiques en raffinant des méthodes préalablement établies dans le domaine. Ces méthodes offrent une piste de solution pour les diverses limitations reliées à l'utilisation de copules.

Dernièrement, la génération de données synthétiques respectueuses de la vie privée a fait l'objet d'un concours organisé par la National Institute of Standards and

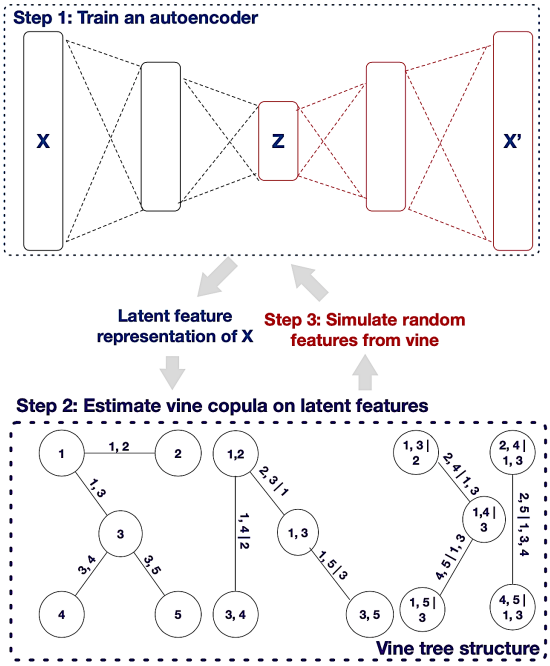


FIGURE 3.8 – Illustration de la technique basée sur les copules vignes et les auto-encodeurs.

Technology (NIST) (Bowen et Snoke, 2019). Le concours « Differentially Private Synthetic Data », qui a été introduit plutôt, avait pour but d'analyser en profondeur plusieurs approches de génération de données différentiellement-privées. Pour se faire, chaque équipe devait fournir le code de leur technique et les preuves nécessaires à la CD. Les métriques utilisées s'appuyaient sur la distance entre les scores de propension (la probabilité d'une donnée d'être assignée à une classe) du jeu de données original et les scores de propension des données synthétiques. Les auteurs de l'article (Bowen et Snoke, 2019) recommandent les algorithmes des équipes CDSyn (Li *et al.*, 2019) et RMcKenna (McKenna, 2019) « based on their NIST Data Challenge ranks, computational burden and complexity, and the ease of implementation for the average data maintainer ». Après un bref coup d'oeil aux codes disponibles, la facilité d'implémentation vantée par les auteurs est mise en doute d'autant plus que dans les deux cas les attributs des données tests sont

codés en dur (« hardcoded ») dans les algorithmes ce qui rend fastidieuse leur application à d'autres ensembles de données.

Ce survol des techniques d'assainissement et d'anonymisation illustre bien la quantité de méthodes et leurs déclinaisons multiples possibles pour répondre au besoin de vie privée dans les micro-données. Les modèles génératifs, qui se font de plus en plus nombreux, semblent être le Saint-Graal pour répondre à la problématique de publication de données sensibles. Le chapitre suivant introduit notre proposition d'une technique de génération de données synthétiques respectueuses de la vie privée basée sur les copules vignes.

CHAPITRE IV

GÉNÉRATION DE DONNÉES SYNTHÉTIQUES ϵ-DIFFÉRENTIELLEMENT-PRIVÉES

Ce chapitre présente le cadre développé tout au long de la recherche de ma maîtrise. Au début de la recherche, durant la phase d'exploration de la littérature, il est vite apparu évident que la confidentialité différentielle est maintenant devenue le standard en ce qui a trait à la protection de la vie privée. Bien qu'il soit possible d'appliquer la CD via certaines méthodes d'assainissement, la plupart de celles-ci modifient de manière significative les données. En général, les techniques d'anonymisation différentiellement-privées produisent des données optimisées pour une tâche précise, comme il est illustré dans (Wang *et al.*, 2015). Cette optimisation restreint donc l'utilisation des données sortantes à une tâche particulière. Comme la méthode à développer se voulait générique plutôt que spécifique à une tâche, la recherche s'est plutôt tournée vers la génération de données synthétiques respectueuses de la vie privée. En particulier, la génération de données de manière différentiellement-privée a semblé être une voie prometteuse étant donné que les modèles génératifs sont majoritairement réutilisables, indépendants de la tâche, que leurs données sortantes sont respectueuses de la vie privée.

Une des premières techniques qui a aidé à diriger la recherche fut PrivBayes (Zhang *et al.*, 2017) qui a été introduite dans le chapitre précédent. Ainsi, les premiers

tests d'implémentation se sont concentrés autour des réseaux bayésiens. Or vers la fin de la phase d'exploration de la littérature, l'article (Kulkarni *et al.*, 2018) a ouvert de nouvelles pistes pour ce qui est de l'utilisation de modèles génératifs. C'est dans cet article qu'il a y eu la première mention des copules pour la synthétisation de données. Plus les copules se trouvaient au centre de la recherche, plus cet objet mathématique plutôt obscure à l'origine semblait être la direction de choix. Les premiers tests de génération de données se sont montrés à la hauteur des attentes. En effet, les copules ont vite démontré leur capacité à synthétiser des données statistiquement semblables aux données d'entraînement, d'autant plus que l'application de la CD ne semblait pas être un défi insurmontable.

Le reste du chapitre explique la collaboration avec Ericsson dans le cadre de la recherche ainsi que l'algorithme développé de génération de données synthétiques ϵ-différentiellement-privées nommé COPULA-SHIRLEY ainsi que le cadre de tests développé pour évaluer les données synthétiques.

4.1 Cadre de la collaboration avec Ericsson

Dès le début de la recherche, une collaboration avec Ericsson Montréal [1] s'est formée. Pour donner quelques éléments de contexte, Ericsson est une compagnie multinationale basée en Suède qui offre une gamme complète de services dans le domaine des télécommunications. Comme certaines données d'Ericsson sont directement liées à l'utilisation des services cellulaires et donc à leurs utilisateurs, l'équipe de science des données de Ericsson Montréal est à la recherche de techniques d'assainissement de données pour assurer la protection de la vie privée des utilisateurs des réseaux cellulaires d'Ericsson.

1. https ://www.ericsson.com/

Cette collaboration s'est cristallisée principalement par un partenariat sous la forme de rencontres durant lesquelles le partenaire académique offre des pistes de solutions et des outils d'assainissement de données alors que le partenaire industriel précise les contraintes (en particulier sur l'utilité des données générées) et les grandes lignes directrices que doit respecter l'assainissement. De plus, Ericsson était en mesure de tester les données synthétiques avec ses modèles tests pour pouvoir mesurer l'utilité.

4.2 COPULA-SHIRLEY

L'algorithme COPULA-SHIRLEY pour
COPULA-based generation of **S**ynt**H**et**I**c diffe**R**ential**L(E)Y**-private data
se démarque des méthodes existantes dans la littérature par deux aspects principaux :

1- L'application de la confidentialité différentielle se fait *avant la construction du modèle*, sur les densités marginales, et non pas *sur* le modèle même ;

2- Un cadre simple, flexible et efficace.

Les deux travaux antérieurs existants à l'intersection de la confidentialité différentielle et des copules sont restreints aux copules gaussiennes (Li *et al.*, 2014; Asghar *et al.*, 2019). Cette restriction facilite l'application de la CD. En effet, comme il a été vu dans le chapitre 1, il est possible de représenter la loi jointe par deux éléments : 1. une copule gaussienne et 2. les densités marginales. Dans ce cas, il suffit d'introduire du bruit différentiellement-privé dans le calcul des deux objets mathématiques pour obtenir un modèle différentiellement-privé. L'utilisation de copules vignes dans un tel cadre est complexe, étant donné que pour chaque paire de copules, il faut qu'il y ait une estimation différentiellement-privée des paramètres et donc une importante quantité de bruit injectée. Un tel cadre exige-

rait aussi une implémentation différentiellement-privée des fonctions du critère de l'AIC et du tau de Kendall de l'algorithme de Dissmann (Dissmann *et al.*, 2013).

Par contraste, COPULA-SHIRLEY vient réduire considérablement le besoin d'ajouter du bruit et la complexité d'implémentation en calculant les dépendances sur les densités marginales différentiellement-privées plutôt que lors de la construction du modèle, permettant l'utilisation de copules vignes. Le cadre offert par COPULA-SHIRLEY permet aussi d'utiliser n'importe quel critère de sélection d'arbres et de copules ainsi que n'importe quel algorithme de sélection de vigne offrant donc une solution flexible et personnalisable.

Cette partie reprend le cadre au fils des itérations pour la publication de l'article (Gambs *et al.*, 2021). Dans cette partie, l'algorithme COPULA-SHIRLEY sera premièrement résumé et ses sous-composantes seront ensuite détaillées. Finalement, une analyse du respect de la CD sera faite et sera suivie par le cadre de tests développé pour évaluer l'utilité et la protection des données synthétiques.

4.2.1 Aperçu général

L'algorithme 1 décrit le cadre général de notre méthode. L'algorithme COPULA-SHIRLEY prend en entrée un ensemble de données D ainsi qu'un paramètre ϵ, représentant le budget de la CD. Les deux derniers paramètres d'entrée, $nGen$ et $methodeEncodage$ correspondent respectivement au nombre de points de données synthétiques à générer et à la méthode d'encodage des attributs catégoriques à utiliser.

Rappelons que les copules sont des objets mathématiques pour modéliser des variables numériques ordonnées. Ceci implique que pour modéliser des attributs catégoriques à l'aide de copules, il faut un prétraitement adéquat pour transformer les données non-ordonnées sans affecter drastiquement l'utilité. Ce prétraitement

Algorithme 1 : COPULA-SHIRLEY

Entrées : Données : D, budget global de confidentialité : $epsilon$, nombre de profils à générer : $nGen$, méthode d'encodage : $methodeEncodage$

Sorties : Données synthétiques : D_{syn}

1 $(pseudoObs, dpFDCs) \longleftarrow$ Prétraitement$(D, \epsilon, methodeEncodage)$ (Partie 4.2.2)

2 $modelVigne \longleftarrow$ Sélection.Copule.Vigne$(pseudoObs)$ (Partie 4.2.3)

3 $D_{syn} \longleftarrow$ Générer.Observations$(modelVigne, nGen, dpFDCs)$ (Partie 4.2.4)

4 **retourner** D_{syn}

est fait par la fonction Prétraitement à la ligne 1 de l'algorithme 1. Une fois les données pré-traitées, l'algorithme peut modéliser de manière différentiellement-privée les FDCs nécessaires pour obtenir les pseudo-observations utilisées par les copules.

4.2.2 Prétraitement

Tel que cité directement de l'article (Li *et al.*, 2014) : "*Although the data should be continuous to guarantee the continuity of margins, discrete data in a large domain can still be considered as approximately continuous as their cumulative density functions do not have jumps, which ensures the continuity of margins.*" Ce qui implique que si elles sont traitées comme continues, les données discrètes peuvent être modélisées par des copules. Ainsi, les copules peuvent être utilisées pour modéliser un large éventail de données sans nécessiter de prétraitement. Cependant, un problème important se pose lorsque les copules sont appliquées à des données catégoriques, car ces données n'ont pas d'échelle ordinale et les copules utilisent principalement la corrélation de rang pour la modélisation. Nous avons implémenté et comparé quatre méthodes d'encodage d'attributs catégoriques. Une astuce courante consiste à considérer les données catégoriques simplement comme des données discrètes dont l'ordre est choisi arbitrairement (par exemple, par ordre alphabétique), ce qui est connu sous le nom de *codage ordinal*, noté ORD dans le

chapitre suivant.

Une autre technique pour traiter les attributs catégoriques est l'utilisation de variables indicatrices (« dummy variables »), dans laquelle les valeurs catégoriques sont transformées en variables indicatrices binaires (Suits, 1957) (cette technique est également connue sous le nom de *encodage one-hot*). L'utilisation de variables indicatrices permet de préserver la corrélation de rang entre les attributs utilisés par les copules, mais fait habituellement exploser le nombre d'attributs. Cet encodage sera noté OHE. Le tableau 4.1 illustre un exemple d'un encodage ordinal et d'un encodage one-hot.

Couleur	**Encodage ORD**	**Encodage OHE**			
		Couleur_Bleu	Couleur_Rouge	Couleur_Jaune	Couleur_Vert
Bleu	1	1	0	0	0
Rouge	3	0	1	0	0
Jaune	2	0	0	1	0
Bleu	1	1	0	0	0
Vert	4	1	0	0	1

TABLEAU 4.1 – Exemple d'encodage ordinal (ordre alphabétique) et d'encodage one-hot.

Deux autres techniques d'encodage *supervisé* connues sous le nom d'encodage Weight of Evidence (WOE) (Wod, 1985) et d'encodage du Generalized Linear Mixed Model (GLMM) (Kuhn et Johnson, 2019) ont été évaluées. Tous deux ont besoin d'un attribut de référence (appelé prédicteur) et codent les attributs catégoriques de manière à maximiser la corrélation entre l'attribut codé et de référence. L'encodeur WOE ne peut être utilisé qu'avec un attribut de référence binaire et est calculé en utilisant le logarithme naturel du nombre de 1 par rapport au nombre de 0 de l'attribut de référence étant donné la valeur de l'attribut (codé). Un exemple d'encodage WOE est donné dans le tableau 4.2. À noter que pour éviter la division par 0 et le logarithme de 0, il faut fixer une valeur arbitrairement petite ; dans l'exemple nous avons choisi 0.0001. Le GLMM peut être considéré

comme une extension de la régression linéaire dans laquelle le codage d'un attribut est calculé par la valeur attendue d'un événement (l'attribut catégorique) étant donné les valeurs du prédicteur. Cet encodeur supporte des prédicteurs binaires et multi-classes. Étant donnée la complexité d'un tel encodage, nous référons le lecteur à (Kuhn et Johnson, 2019) pour plus d'informations.

Couleur	Prédicteur	WOE
Bleu	1	$\log(\frac{3}{0.0001}) = 10.31$
Rouge	1	$\log(\frac{1}{1}) = 0.00$
Jaune	0	$\log(\frac{0.0001}{2}) = -9.90$
Rouge	0	$\log(\frac{1}{1}) = 0.00$
Jaune	0	$\log(\frac{0.0001}{2}) = -9.90$
Bleu	1	$\log(\frac{3}{0.0001}) = 10.31$
Bleu	1	$\log(\frac{3}{0.0001}) = 10.31$

TABLEAU 4.2 – Exemple d'un encodage WOE à partir d'un attribut prédicteur.

La méthode de prétraitement présentée dans l'algorithme 2 convertit les valeurs catégoriques, calcule les FDCs différentiellement-privées à partir des histogrammes bruités par la CD et produit les pseudo-observations différentiellement-privées nécessaires à la sélection d'une copule vigne. Dans ce qui suit, nous décrivons plus en détail ces différents processus.

Entraînement fractionné. Étant donné que notre cadre nécessite deux processus d'apprentissage séquentiels : l'apprentissage des fonctions de distributions cumulatives différentiellement-privées et l'apprentissage du modèle de copules, nous nous appuyons sur la composition parallèle (Théorème 2.3) de la confidentialité différentielle en divisant l'ensemble de données D en deux sous-ensembles au lieu d'utiliser la composition séquentielle (Théorème 2.2) et de diviser le budget global ϵ. De cette manière, nous utilisons efficacement les forces de modélisation des fonctions copules en sacrifiant des points de données pour réduire le bruit ajouté via le mécanisme différentiellement-privé (tel que le mécanisme Laplacien) tout en

préservant une grande partie de l'utilité des données. La ligne 3 de l'algorithme 2 montre le processus de division de l'ensemble de données en deux sous-ensembles.

Algorithme 2 : Prétraitement

Entrées : Données : D, budget global de confidentialité : *epsilon*, méthode d'encodage : *methodeEncodage*

Sorties : Pseudo-observations : *pseudoObs*, FDCs différentiellement-privées : $dpCDFs$

1 $D \leftarrow \mathsf{methodeEncodage}(D)$
2 $(ensEntrainHist, ensEntrainCop) \leftarrow \mathsf{Diviser.Données}(D)$
3 $dpHisto \leftarrow \mathsf{Calculer.DP.Histo}(ensEntrainHist, \epsilon)$
4 **pour chaque** *hist* dans $dpHisto$ **faire**
5 $\quad fdc \leftarrow \frac{\mathsf{Somme.Cumul}(hist[frequence])}{\mathsf{Somme}(hist[frequence])}$
6 $\quad dpFDCs[col] \leftarrow fdc$
7 **pour chaque** *col* dans $ensEntrainCop$ **faire**
8 $\quad fdc \leftarrow dpFDCs[col]$
9 $\quad pseudoObs[col] \leftarrow fdc(ensEntrainCop[col])$
10 **retourner** $(pseudoObs, dpFDCs)$

Calcul des histogrammes différentiellement-privés. L'estimation des histogrammes différentiellement-privés est le processus clé de COPULA-SHIRLEY. L'implémentation actuelle calcule naïvement un histogramme différentiellement-privé en ajoutant un bruit laplacien de moyenne 0 et d'échelle $\frac{\Delta}{\epsilon}$ avec $\Delta = 2$ aux fréquences de l'histogramme [2]. Une méthode plus sophistiquée, une celles définies dans (Xiao *et al.*, 2010; Acs *et al.*, 2012; Xu *et al.*, 2013) par exemple, pourrait éventuellement être utilisée pour améliorer l'utilité des données synthétiques.

Calcul des FDCs. Les fonctions de distribution cumulatives et les fonctions de densité de probabilité sont intrinsèquement liées et il est presque trivial de passer de l'une à l'autre. Avec des fonctions continues, les FDCs sont estimées via l'intégration des courbes des FDPs. Dans un environnement discret comme

2. Dans le cadre borné de la CD, la sensibilité globale Δ du calcul de l'histogramme est de 2.

le nôtre, puisque nous utilisons des histogrammes pour estimer les FDPs, une simple somme cumulative normalisée sur le nombre d'intervalles fournit une estimation suffisante des FDCs, ce qui est montré à la ligne 6 de l'algorithme 2. Cette approche est similaire à celle proposée par le *Harvard University Privacy Tools Project* dans les notes de cours sur les FDCs différentiellement-privées (Muise et Nissim, 2016), dans lesquelles ils ajoutent du bruit aux fréquences de la somme cumulative pour ensuite normaliser. Notre méthode, qui se base sur les histogrammes bruités, produit toujours une fonction strictement monotone croissante, alors que l'approche du *Harvard University Privacy Tools Project* tend à produire des FDCs non monotones en dents de scie. Une fonction croissante strictement monotone est souhaitable, car elle signifie que la FDC théorique a toujours un inverse. Les FDCs en dents de scie non monotones peuvent produire des comportements erratiques lors de la transformation des données en pseudo-observations et surtout lors du retour à l'échelle naturelle avec les FDCs inverses.

Calcul des pseudo-observations. Comme les copules ne peuvent modéliser que des densités marginales standard uniformes (c'est-à-dire des pseudo-observations), les données brutes doivent être transformées. La TIP stipule que pour une variable aléatoire X et sa FDC F_X, on a que $F_X(X)$ est standard uniforme. L'algorithme 2 applique la TIP aux lignes 8-10 en extrayant d'abord la FDC correspondante (ligne 9) avant d'appliquer ladite FDC sur les données (disjointes de l'ensemble utilisé pour le calcul des FDC) (ligne 10).

4.2.3 Sélection de la copule vigne

L'algorithme 3 décrit le processus général de sélection d'une vigne à partir de pseudo-observations. Comme indiqué précédemment, notre méthode n'a besoin que des pseudo-observations standards uniformes différentiellement-privées pour

sélectionner une vigne. L'algorithme de Dissmann est ensuite utilisé avec les (pseudo-)observations bruitées, et seulement ces observations, pour sélectionner une copule vigne. L'implémentation actuelle de COPULA-SHIRLEY utilise la version de l'algorithme de Dissmann de la librairie `R rvinecopulib` (Nagler et Vatter, 2017). Cette librairie offre une méthode complète et hautement configurable pour la sélection de vignes ainsi que certaines techniques d'optimisation pour réduire le temps de calcul.

Algorithme 3 : Sélection.Copule.Vigne

Entrées : Pseudo-observations : $pseudoObs$
Sorties : Copule vigne : $modelVigne$
1 $modelVigne \leftarrow$ AlgorithmeDissmann($pseudoObs$) (Partie 1.2)
2 **retourner** ($modelVigne$)

4.2.4 Génération de données synthétiques

La dernière étape de notre cadre est la génération de données synthétiques. Pour ce faire, des observations standard uniformes doivent être échantillonnées à partir de la copule vigne sélectionnée par la méthode précédente. Comme nous utilisons l'implémentation de `rvinecopulib` de l'algorithme de Dissmann, nous utilisons naturellement leur implémentation de l'échantillonnage des observations à partir de vignes. La ligne 1 de l'algorithme 4 fait référence à l'implémentation de cet échantillonnage. Voir le chapitre 1 pour un exemple d'échantillonnage d'observations à partir d'une vigne.

Pour rapporter les observations échantillonnées à leur domaine d'origine, nous utilisons la transformée intégrale de probabilité inverse (TIP inverse), qui ne nécessite que les FDCs inverses des attributs. Ce processus est illustré aux lignes 2 à 5 de l'algorithme 4. Cette dernière étape conclut le cadre de la génération de données différentiellement-privées avec COPULA-SHIRLEY.

Algorithme 4 : Générer.Observations

Entrées : Copule vigne : $modelVigne$, nombre de profils à générer : $nGen$, FDCs différentiellement-privées : $dpCDFs$

Sorties : Données synthétiques : D_{syn}

```
obsSynth ← Échantillonnage.Vigne(vineModel, nGen)
pour chaque col in synthObs faire
    fdc ← dpCDFs[col]
    invfdc ← Inverse(fdc)
    D_syn[col] ← invfdc(obsSynth[col])
retourner D_syn
```

4.2.5 Respect de la confidentialité différentielle

Il devrait être clair maintenant que COPULA-SHIRLEY repose uniquement sur des histogrammes différentiellement-privés, ce qui rend l'ensemble du cadre différentiellement privé par conception.

Théorème 4.1 (Caractère différentiellement-privé de COPULA-SHIRLEY)**.** *L'algorithme 1 est ϵ-différentiellement-privés.*

Preuve. La méthode Calculer.DP.Histo produit des histogrammes ϵ-différemment-privés par le théorème du mécanisme de Laplacien (Théorème 2.4). Le calcul de $dpCDFs$ n'utilise que les histogrammes précédents différentiellement-privés pour calculer les FDCs, de manière parallèle ; ainsi, ces fonctions de distributions sont ϵ-différemment-privées selon les propriétés de *fermeture sous post-traitement* et de *composition parallèle* de la CD. Pour obtenir les *pseudoObs*, nous appliquons simplement les FDCs différentiellement-privées un ensemble d'apprentissage disjoint de l'ensemble utilisé pour le calcul des histogrammes. Les données *pseudoObs* résultantes sont ϵ-différemment-privées en raison de l'application d'un mécanisme ϵ-différemment privé et par la propriété de la *composition parallèle*.

La méthode AlgorithmeDissmann n'utilise que l'ensemble de données

ϵ-différemment-privé *pseudoObs* pour la sélection d'une copule vigne. La copule vigne résultante est ϵ-différemment-privée par la propriété *fermeture sous post-traitement*. Les deux méthodes, Échantillonnage.Vigne et Inverse n'utilisent que des données privées et ne violent donc pas la propriété *fermeture sous post-traitement*. Enfin, l'ensemble du processus est fermé et ne viole jamais la propriété *fermeture sous post-traitement* de la confidentialité différentielle, car l'algorithme opère indépendamment pour chaque attribut et utilise donc la propriété la *composition parallèle* ; ainsi l'algorithme 1 est ϵ-differentiellement-privé. □

4.2.6 Librairie `rvinecopulib`

L'implémentation courante de COPULA-SHIRLEY utilise la librairie `rvinecopulib` (Nagler et Vatter, 2017) du langage de programmation R et ses sous-routines `vinecop` et `rvinecop`, respectivement pour la construction du modèle à partir des marges et la génération d'observations synthétiques à partir du modèle. La librairie se base sur l'algorithme de Dissmann pour la construction de copules vignes. Elle permet plusieurs ajustements à l'algorithme comme le choix de différentes fonctions de score pour la sélection d'arbres ou la sélection de copules. Une des forces de l'implémentation offerte par la librairie est l'intégration du calcul parallèle pour les deux routines utilisées `vinecop` et `rvinecop` venant aider à l'efficacité de COPULA-SHIRLEY.

4.3 Cadre de tests

Afin d'évaluer l'utilité et le niveau de protection des données synthétiques et aussi à des fins de comparaisons avec les méthodes existantes, un cadre de tests a été développé. Les tests sont divisés en trois grandes catégories que sont les tests statistiques, qui évaluent la similarité entre la distribution des données originales

et celle des données synthétiques, les tâches de classification et de régression, qui simulent le scénario d'un classifieur entraîné sur les données synthétiques et finalement le test de protection, qui mesurent de pouvoir d'inférence des données synthétiques face à une attaque d'appartenance (« membership attack »).

4.3.1 Tests statistiques

Les tests statistiques permettent de quantifier la similarité entre les données originales et les données synthétiques. Dans l'optique des données générées par l'algorithme COPULA-SHIRLEY, deux tests statistiques différents sont utilisés.

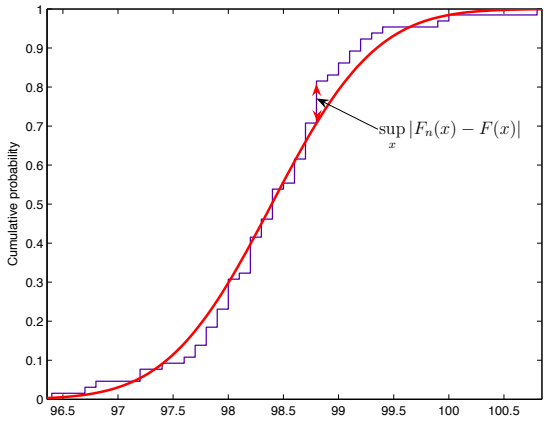
Distance de Kolmogorov-Smirnov. Le premier test statistique utilise la distance (aussi appelée statistique) D_{KS} de Kolmogorov-Smirnov (KS) (Smirnov, 1944). Cette distance est utilisée dans le test d'hypothèse de KS et mesure la distance maximale entre deux distributions. L'hypothèse nulle étant que la distribution de référence et la distribution expérimentale suivent la même loi. Pour réaliser ce test, étant donné une distribution théorique (ou de référence) F_t et une distribution expérimentale F_e, la statistique du test de KS est donnée par :

$$D_{KS}(F_t, F_e) = \sup_x \left|F_t(x) - F_e(x)\right|.$$

La mesure du test est illustrée par la figure 4.1 qui est extraite des notes de cours (Panchenko, 2006).

Dans le cadre développé, la distance de KS sert à mesurer une distance entre les distributions originales et synthétiques. À noter que seulement la statistique D_{KS} est utilisée et non pas le résultat du test d'hypothèse.

Variation de la corrélation moyenne. Le score représente la différence moyenne absolue des coefficients de corrélation entre les matrices de corrélation de l'en-



La fonction continue en rouge correspond à la distribution de référence et la fonction escalier en bleu correspond à la distribution expérimentale.

FIGURE 4.1 – Exemple graphique du test de Kolmogorov-Smirnov (KS).

semble de données de référence et de l'ensemble de données synthétiques. Si C_o représente la matrice de corrélation de Spearman (Spearman, 1987) de l'ensemble de données de référence et C_s représente la matrice de corrélation de l'ensemble de données synthétiques, la variation de la corrélation moyenne est calculée comme suit :

$$\frac{\sum_{i,j=1}^{n} |C_o(i,j) - C_s(i,j)|}{\sum_{i,j} 1}$$

où $C(i,j)$ est le coefficient de corrélation entre les i-ième et j-ième attributs. Cette métrique est inspirée de celle utilisée dans le concours PWSCUP 2020 "Anonymity against Membership Inference"[3].

3. https://www.iwsec.org/pws/2020/Images/PWSCUP2020_rule_20200826_e.pdf

4.3.2 Tâches de classification et de régression

Les mesures statistiques aident à visualiser la similarité entre les données originales et les données synthétiques, mais elles ne suffisent pas à capter toute l'essence des données. Les tâches de classification viennent compléter les tests statistiques en simulant un cas d'application, qui est la prédiction d'un attribut spécifique, permettant ainsi d'évaluer partiellement si les corrélations entre les attributs sont conservées. L'idée d'un tel test provient des discussions avec Ericsson, dans lesquelles un des cas d'usage identifié a été la classification des données. La classification de données se base sur les corrélations entre attributs et les différentes combinaisons entre les valeurs d'attributs pour prédire une classe. Il s'agit donc d'un moyen idéal pour évaluer si la force des liens entre attributs est conservée dans les données synthétiques ainsi que pour analyser l'impact de ces données dans un cadre réaliste.

La comparaison se fait en créant deux classifieurs, un classifieur entraîné sur des données originales et un classifieur entraîné sur des données synthétiques. Ces classifieurs sont ensuite testés sur un même ensemble test provenant des données brutes. Idéalement, le classifieur ayant appris sur les données synthétiques classe de manière similaire au classifieur entraîné sur les données originales. La comparaison des classifieurs se fait à l'aide du coefficient de corrélation de Matthews (MCC) (Matthews, 1975), qui est une mesure de qualité de classification. Le MCC est préférable à la F1-mesure étant donné sa robustesse au déséquilibre entre classes et du au fait que la mesure est sensible aux nombres de vrais négatifs (contrairement à la F1-mesure) (Chicco et Jurman, 2020). Le MCC est calculé de la manière suivante :

$$MCC = \frac{\frac{TP}{N} - (S \cdot P)}{\sqrt{(S \cdot P)(1 - S)(1 - P)}}$$

où

$N =$ Nombre de profils classés $\quad TP =$ Taux de *vrais positifs*

$FN =$ Taux de *faux négatifs* $\quad FP =$ Taux de *faux positifs*

$$S = \frac{TP + FN}{N} \qquad P = \frac{TP + FP}{N}$$

Dans nos expérimentations,nous avons évalué les données synthétiques sur deux tâches de classification : un problème de classification binaire et un problème de classification multi-classes. Nous avons opté pour le classificateur *gradient boosting* (Friedman, 2001) étant donné que cet algorithme est connu pour sa robustesse au surapprentissage (« overfitting ») et que ses performances sont souvent proches des méthodes de pointe, comme les réseaux de neurones profonds, sur de nombreuses tâches de classification. Nous utilisons l'implémentation `XGBoost` (Chen *et al.*, 2015) de l'algorithme de gradient boosting.

Pour approfondir notre analyse, nous avons également évalué les données synthétiques sur une tâche de régression linéaire simple. Pour évaluer sa réussite, nous avons calculé l'*erreur quadratique moyenne* (EQM) :

$$\text{EQM}(Y, \tilde{Y}) = \sqrt{\frac{\sum_{i=0}^{n}(\tilde{y_i} - y_i)^2}{n}}$$

dans laquelle $Y = (y_1, y_2, ..., y_n)$ représente les valeurs réelles et $\tilde{Y} = (\tilde{y_1}, \tilde{y_2}, ..., \tilde{y_n})$, les sorties du modèle linéaire.

Les tests évoqués pour l'instant ont pour objectif de mesurer l'utilité des données. Cependant, dans un cadre de protection de la vie privée il faut aussi étudier le risque de fuites d'informations qui peuvent survenir en utilisant la méthode proposée. La partie suivante définit les tests conçus pour évaluer ces risques.

4.3.3 Test de protection de la vie privée

Nous pensons qu'il ne faut pas se baser uniquement sur le respect d'un modèle formel tel que la confidentialité différentielle, mais aussi que les données synthétiques doivent être évaluées par rapport à un test de confidentialité basé sur des attaques par inférence afin de quantifier la protection de la confidentialité fournie par la méthode de synthèse. Ce point est supporté par l'article de Stadler, Oprisanu et Troncoso (2020) qui démontre que l'application de la CD n'apporte pas une protection uniforme.

Dans ce travail, nous avons opté pour l'attaque par inférence d'appartenance de Monte Carlo (AIAMC) introduite par Hilprecht, Härterich et Bernau (2019) pour évaluer le niveau de protection de notre méthode. En termes simples, l'AIAMC quantifie le risque de retrouver des profils d'entraînement à l'aide de profils synthétiques parmi un ensemble compris de profils utilisés pour l'entraînement du modèle et de profils de contrôle (disjoints des profils d'entraînements). L'un des avantages de l'AIAMC est qu'il s'agit d'une attaque non paramétrique et agnostique par rapport au modèle. En autre, l'AIAMC offre une grande précision dans les situations de surapprentissage de modèles génératifs et surpasse les attaques précédentes basées sur des modèles fantômes (Shokri *et al.*, 2017). Le but de cette attaque est d'obtenir un score sur les données synthétiques pour évaluer leur propension à faciliter une attaque par inférence d'appartenance.

Le cadre de l'attaque est le suivant. Soit un ensemble de données $\mathcal{D}$. Soit $\mathcal{S}_T \subset \mathcal{D}$ un sous-ensemble de m profils de l'ensemble d'entraînement du modèle génératif et $\mathcal{S}_C \subset \mathcal{D}$ un sous-ensemble de m profils de contrôle disjoint de $\mathcal{S}_T$. Les profils de contrôle sont définis comme des profils issus de la même distribution que les profils d'apprentissage, mais jamais utilisés dans le processus d'apprentissage du modèle génératif. Soit x un profil quelconque de $\mathcal{D}$; $U_r(x) = \{x' \in \mathcal{D} \,|\, d(x, x') \leq r\}$ est

défini comme le voisinage (de rayon r) du profil x par rapport à la distance d (*c.-à-d.* l'ensemble des profils x'_i à distance de moins de r de x). Un profil synthétique g issu d'un modèle génératif G a plus de chance d'être similaire à un profil x lorsque la probabilité $P[g \in U_r(x)]$ augmente. La probabilité $P[g \in U_r(x)]$ est estimée par intégration de Monte Carlo : $P[g \in U_r(x)] \approx \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{g_i \in U_r(x)}$, dans laquelle g_i sont des échantillons synthétiques de G.

Pour affiner l'attaque, les auteurs proposent une estimation alternative de $P[g \in U_r(x)]$ basée sur la distance moyenne entre x et ses voisins :

$$P[g \in U_r(x)] \approx \frac{1}{n} \sum_{i=1}^{n} \cdot \mathbf{1}_{g_i \in U_r(x)} \log\left(d(x, g_i) + \eta\right)$$

dans laquelle η est une petite valeur arbitraire fixée pour éviter $\log 0$ (nous avons utilisé $\eta = 10^{-12}$). La fonction

$$\hat{f}_{MC}(G, x) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{g_i \in U_r(x)} \cdot \log\left(d(x, g_i) + \eta\right)$$

est la fonction de score de confidentialité du profil x étant donné G. D'après cette définition, si le profil x obtient un score élevé, les échantillons synthétiques $g_i \in G$ sont très proches de x et impliquent un modèle surajusté sur x.

Soit $\mathcal{S}_G$ un ensemble de k échantillons synthétiques provenant d'un modèle génératif G. Pour calculer le score de confidentialité de l'ensemble $\mathcal{S}_G$, nous calculons d'abord $\hat{f}_{MC}(\mathcal{S}_G, x)$ pour tous les $2m$ profils $x \in \mathcal{S}_T \cup \mathcal{S}_C$. Ensuite, nous prenons les m profils de l'union $\mathcal{S}_T \cup \mathcal{S}_C$ ayant les scores $\hat{f}$ les plus élevés pour former l'ensemble $\mathcal{I}_\mathcal{G}$. L'ensemble $\mathcal{I}_\mathcal{G}$ est donc l'ensemble des profils les plus susceptibles à divulguer de l'appartenance à l'ensemble d'entraînement du modèle. En calculant le ratio entre le nombre profils qui se trouvent à la fois dans $\mathcal{S}_T$ et $\mathcal{I}_\mathcal{G}$ et le nombre

de profils dans $\mathcal{S}_T$, nous obtenons le score de confidentialité de $\mathcal{S}_G$:

$$ScoreConfidentialite(\mathcal{S}_G) := \frac{|\mathcal{S}_T \cap \mathcal{I}_\mathcal{G}|}{|\mathcal{S}_T|}$$

Le score de confidentialité peut être interprété comme le pourcentage de profils correctement identifiés comme provenant de l'ensemble d'entraînement. Selon cette définition, un score de 1 signifie que l'ensemble d'entraînement complet a été récupéré avec succès, ce qui implique une violation de la vie privée, tandis qu'un score de 0.5 signifie que les profils des ensembles d'entraînement et de contrôle ne peuvent être distingués à l'aide de profils synthétiques.

Dans nos expériences, nous avons constaté que l'utilisation d'une distance plus grossière comme la distance de Hamming (Hamming, 1950) fournit des scores d'inférence d'appartenance plus élevés que la distance euclidienne. Pour définir la valeur r de la taille du voisinage, nous avons utilisé l'heuristique de la *médiane* telle que définie par les auteurs, étant celle qui fournit la meilleure précision :

$$r = \text{mediane}_{1 \leq i \leq 2m} \left(\min_{1 \leq j \leq n} d(x_i, g_j) \right)$$

où $x_i \in \mathcal{S}_T \cup \mathcal{S}_C$ et $g_j \in \mathcal{S}_G$.

Le prochain chapitre met en oeuvre tout ce qui a été défini dans le présent chapitre. Plus précisément, notre approche COPULA-SHIRLEY est testée sur trois jeux de données et comparée à deux autres modèles génératifs à l'aide du cadre de tests discuté plus tôt. Le chapitre 5 sert aussi de lieu de discussions et de critiques sur l'approche développée.

CHAPITRE V

DONNÉES SYNTHÉTIQUES, UTILITÉ ET PROTECTION

Ce chapitre présente le point culminant de la recherche en évaluant la capacité de l'algorithme COPULA-SHIRLEY à générer des données synthétiques respectueuses de la vie privée. COPULA-SHIRLEY est testé sur trois ensembles de données qui ont la même caractéristique de contenir des profils réels collectés auprès de personnes et contenant des informations identifiantes. Notre méthode est aussi comparée à trois implémentations de modèles génératifs que sont PrivBayes (Zhang *et al.*, 2017), DP-Copula (Li *et al.*, 2014) et DP-Histogram, un modèle génératif nous avons implémenté basé sur des histogrammes différentiellement-privés. La comparaison est faite à l'aide du cadre de tests défini dans le chapitre 4. Finalement, une discussion suit et se focalise sur les limitations identifiées de l'approche ainsi que les différentes pistes de solutions possibles pour surmonter ces difficultés.

Dans ce chapitre, nous étudions le compromis vie privée-utilité des données synthétiques générées par les modèles génératifs mentionnés plus tôt. Dans l'article de McKay et Snoke (2019) concernant le concours du NIST sur la génération de données différentiellement-privées, les auteurs ont classé PrivBayes dans le top 5, ce qui soutient son utilisation dans notre travail comparatif. Comme COPULA-SHIRLEY peut être considéré comme un raffinement du modèle DP-Copula avec

l'utilisation de vignes, nous avons voulu comparer notre méthode à une approche de copule gaussienne différentiellement-privée. De plus, nous voulions comparer notre méthode à un modèle naïf qui échantillonne indépendamment des observations à partir d'histogrammes différentiellement-privés, ce pourquoi nous avons implémenté le modèle DP-Histogram. L'implémentation de notre cadre expérimental est disponible en source ouverte sous forme de scripts Python à l'adresse suivante : `https://github.com/alxxrg/copula-shirley`.

Les parties qui suivent reflètent le travail final et les résultats obtenus pour l'article (Gambs *et al.*, 2021).

5.1 Cadre expérimental

5.1.1 Ensembles de données.

Trois jeux de données de différentes dimensions ont été utilisés dans nos expériences. Le premier est le jeu de données Adult (Dua et Graff, 2017), qui contient 32 561 profils. Chaque profil est décrit par 14 attributs tels que le sexe, l'âge, l'état civil et le pays d'origine. Les attributs sont principalement catégoriques (8), le reste étant discret (6).

Le deuxième ensemble de données utilisé est COMPAS (Angwin *et al.*, 2019), qui consiste en des profils de délinquants criminels en Floride pour les années 2013 et 2014. C'est le plus petit ensemble des trois, contenant 10 568 profils, chacun décrit avec 13 attributs assez similaires à ceux d'Adult avec la même proportion d'attributs catégoriques et discrets.

Le troisième ensemble de données utilisé est celui de Texas Hospital (Texas Department of State Health Services, Austin, Texas, 2013) à partir duquel nous avons échantillonné uniformément 150 000 profils à partir d'un ensemble de 636 140

profils et sélectionné 17 attributs, dont 11 sont catégoriques. Nous avons échantillonné cet ensemble de données pour réduire la charge de calcul, principalement pour l'algorithme PrivBayes.

5.1.2 Paramètres globaux.

Pour évaluer la synthèse de données, nous avons exécuté une technique de validation croisée à k plis (Geisser, 1975) (« k-fold cross-validation ») avec $k = 5$. Un pli est constitué d'un ensemble d'entraînement et d'un ensemble test disjoint. Pour chaque pli, tous les modèles génératifs sont entraînés sur l'ensemble d'entraînement du pli et génèrent le même nombre de profils que l'ensemble de données en entrée (c'est-à-dire 32 561, 10 568 et 150 000). Tous les tests sont mesurés en utilisant l'ensemble test du pli comme référence et les données synthétiques nouvellement générées. Pour le budget de confidentialité ϵ, nous avons essayé plusieurs valeurs variant dans $\epsilon \in [0.0, 8.0, 1.0, 0.1, 0.01]$ (ici $\epsilon = 0$ signifie que la CD est désactivée), similaire aux paramètres utilisés dans l'étude comparative du concours du NIST pour évaluer la méthode de génération de données différentiellement-privées (Bowen et Snoke, 2019).

Pour les encodeurs d'attributs catégoriques, nous avons utilisé la bibliothèque Python `category_encoders` (McGinnis *et al.*, 2018). Pour les encodeurs supervisés (WOE & GLMM), afin d'éviter la fuite d'informations sur l'ensemble d'entraînement, nous entraînons les encodeurs sur un ensemble disjoint, c'est-à-dire l'ensemble test du pli. Par défaut, nous avons utilisé l'encodeur WOE, puisqu'il fournissait des résultats légèrement préférables aux autres méthodes d'encodage (voir la figure 5.2). Le prédicteur pour l'encodage WOE est l'attribut utilisé pour la classification binaire et le prédicteur pour l'encodage GLMM est l'attribut utilisé pour la classification multi-classes. Pour l'implémentation de l'attaque par infé-

rence d'appartenance discutée dans la partie 4.3.3, l'ensemble de contrôle utilisé est l'ensemble test du pli.

5.1.3 Détails des implémentations.

COPULA-SHIRLEY. Comme mentionné précédemment, COPULA-SHIRLEY utilise l'implémentation de l'algorithme de Dissmann de la bibliothèque `R rvinecopulib` (Nagler et Vatter, 2017). Dans nos tests, nous avons utilisé tous les paramètres par défaut de la méthode de sélection de vignes, qui se trouvent dans la section de référence (Nagler et Vatter, 2017), à l'exception de deux paramètres : le niveau d'élagage Ψ et la mesure de corrélation pour la sélection de l'ACM. Nous avons choisi d'élaguer les vignes au deuxième niveau ($\Psi = 2$). Le niveau d'élagage a été étudié de manière approfondie dans la partie 5.2. Nous avons également opté pour la corrélation de rang de Spearman, car il s'agit d'une statistique appropriée lorsque des égalités dans les rangs (« ties ») existent dans les données (Puth *et al.*, 2015). Comme indiqué dans le chapitre précédant, nous avons divisé l'ensemble d'apprentissage en deux sous-ensembles disjoints, et ce avec un ratio de 50/50, respectivement pour les histogrammes différentiellement-privés et les pseudo-observations. L'impact des différents ratios est discuté dans la partie 5.2.1. Enfin, pour une représentation plus fine, nous choisissons d'utiliser autant d'intervalles pour nos histogrammes qu'il y a de valeurs uniques dans les données d'entrée. Par défaut, nous utilisons le mécanisme laplacien pour calculer les histogrammes différentiellement-privés. Nous avons toutefois analysé l'impact sur l'utilité et la protection d'autres mécanismes dans la partie 5.2.3.

PrivBayes. Nous utilisons l'implémentation de PrivBayes référencée dans (Bowen et Snoke, 2019) appelée DataSynthesizer (Ping *et al.*, 2017). En dehors du budget de confidentialité ϵ, l'implémentation de PrivBayes n'a qu'un seul paramètre : est

le nombre maximal de nœuds parents dans le réseau bayésien. Nous utilisons le paramètre par défaut qui est de 3.

DP-Copula. Bien que nous n'ayons pas trouvé une implémentation officielle de Li, Xiong et Jiang (2014), nous avons découvert et utilisé une implémentation source ouverte (Rakotoarivelo, 2019). En plus du budget de confidentialité, le seul paramètre de DP-Copula est utilisé pour régler la manière dont ce budget est divisé entre le calcul des densités marginales et de la matrice de corrélation de manière différentiellement-privée. Nous choisissons de fixer la valeur de ce paramètre de manière à ce que la moitié du budget de confidentialité soit consacrée au calcul des densités marginales et l'autre moitié au calcul de la matrice de corrélation.

DP-Histogram. Ce modèle n'a aucun paramètre autre que le budget de confidentialité ϵ.

5.1.4 Attributs utilisés pour les tests de classification et de régression.

Pour les données d'Adult, la classification binaire se fait sur l'attribut `salary`, la classification multi-classes sur l'attribut `relationship` et la régression linéaire sur l'attribut `age`. Pour les données de COMPAS, la classification binaire est sur `is_violent_recid`, la classifiction multi-classes sur `race` et la régression linéaire se fait sur l'attribut `decile_score`. Finalement, pour les données de Texas, les attributs `ETHNICITY`, `TYPE_OF_ADMISSION` et `TOTAL_CHARGES` sont utilisés respectivement pour la classification binaire, la classification multi-classes et pour la régression linéaire.

5.2 Résultats

Dans les figures qui suivent, les EQM de la régression linéaire et les temps d'exécution sont normalisés pour faciliter la comparaison des résultats. Pour le MCC, les valeurs *élevées* sont préférées tandis que pour les autres métriques, les valeurs *basses* sont préférées. Les mesures sont moyennées sur une validation croisée à 5 plis.

5.2.1 Ratio entre les ensembles d'apprentissage.

Rappelons que dans notre étape de prétraitement (algorithme 2, l'ensemble de données d'apprentissage est divisé en deux ensembles, l'un pour l'apprentissage des FDCs différentiellement-privées et l'autre pour le calcul des pseudo-observations. Nous évaluons d'abord différents ratios entre la cardinalité des deux ensembles. Comme le montre la figure 5.1, la plupart des métriques sont stables à travers les différents ratios. Une exception est la distance de KS, qui présente une augmentation lorsque le ratio donné pour les pseudo-observations est plus élevé. Comme il n'y a pas de consensus clair dans les données, nous avons opté pour un ratio de 50/50 pour les autres expériences.

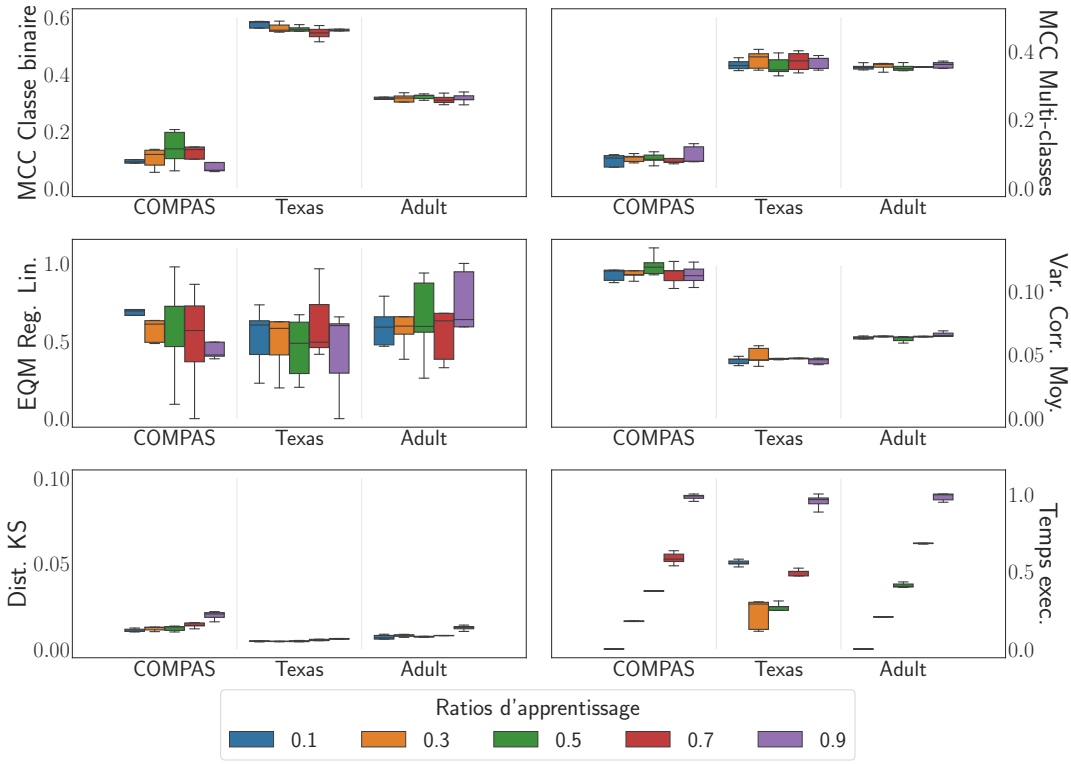


FIGURE 5.1 – L'impact des ratios de division des ensembles d'apprentissage avec $\epsilon = 1.0$.

5.2.2 Encodage des attributs catégoriques.

L'encodage approprié des attributs catégoriques est crucial pour notre approche. Comme le montre la figure 5.2, les deux tâches de classification affichent des performances inférieures avec l'encodage ordinal. Entre autres, l'encodage one-hot n'a pas pu être testé sur le Texas en raison de l'augmentation de la charge de calcul, puisque le nombre d'attributs est passé de 17 à 5375. L'encodage one-hot est aussi peu performant en ce qui concerne la distance de KS et la tâche de régression, en plus d'augmenter considérablement le temps d'exécution. Nous avons opté pour l'encodeur WOE car il est plus performant pour les deux tâches de classification que l'encodeur GLMM.

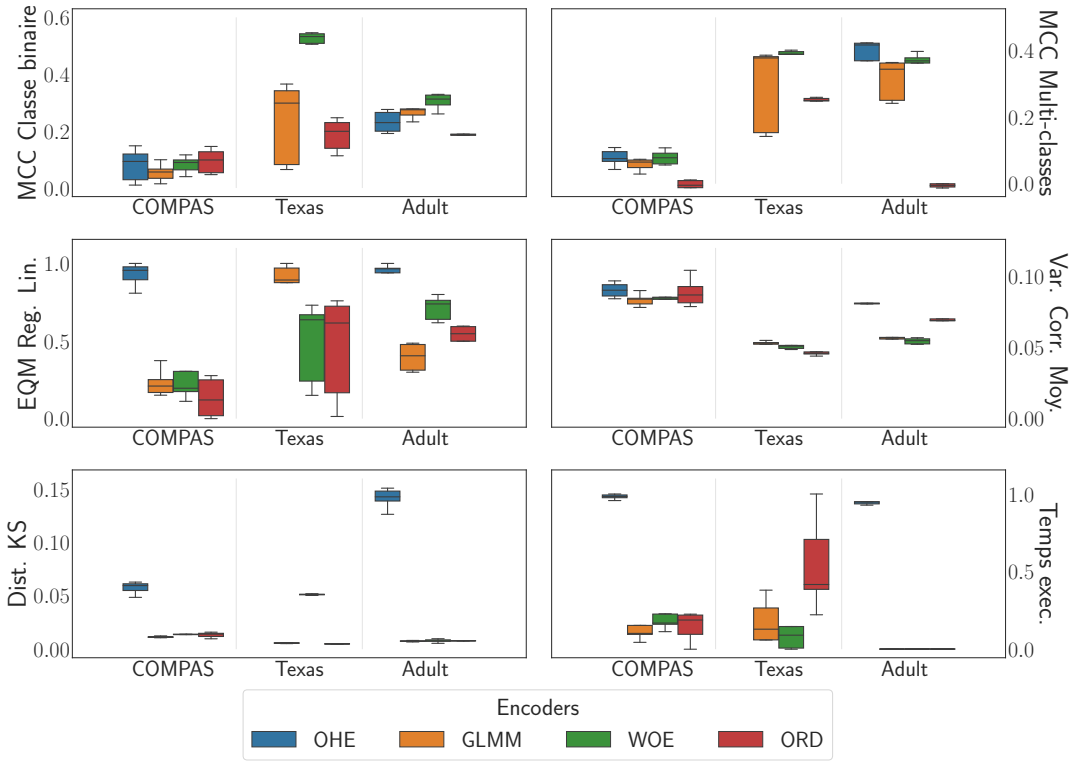


FIGURE 5.2 – L'impact de l'encodage des attributs catégoriques avec $\epsilon = 1.0$.

5.2.3 Mécanismes différentiellement-privés.

Le mécanisme laplacien est l'approche classique pour obtenir des résultats différentiellement-privées. Cependant, ce mécanisme est optimisé pour des valeurs continues alors que notre implémentation utilise actuellement des histogrammes discrets. Nous avons étudié deux autres mécanismes de CD afin d'améliorer notre approche. Le premier est le mécanisme géométrique (Ghosh *et al.*, 2012) qui est l'extension discrète du mécanisme laplacien. Le second est le mécanisme gaussien (Dwork *et al.*, 2014), qui fournit une protection de la vie privée moins stricte, mais souvent un ajout de bruit moins important. Nous avons utilisé les implémentations des mécanismes de la bibliothèque d'IBM Differential Privacy [1]. Pour le

1. `https://diffprivlib.readthedocs.io/`

mécanisme gaussien, le δ optimisé est utilisé selon l'article (Balle et Wang, 2018). Tout comme l'étude réalisée par McKay et Snoke (2019), nous avons fixé $\delta = 0.001$.

Comme le montre la figure 5.3, notre méthode est généralement stable pour les trois mécanismes. Les trois mécanismes semblent également offrir une protection similaire contre l'inférence d'appartenance, comme l'illustrent les scores de confidentialité. Le mécanisme géométrique montre une performance accrue par rapport aux deux autres pour la distance de KS.

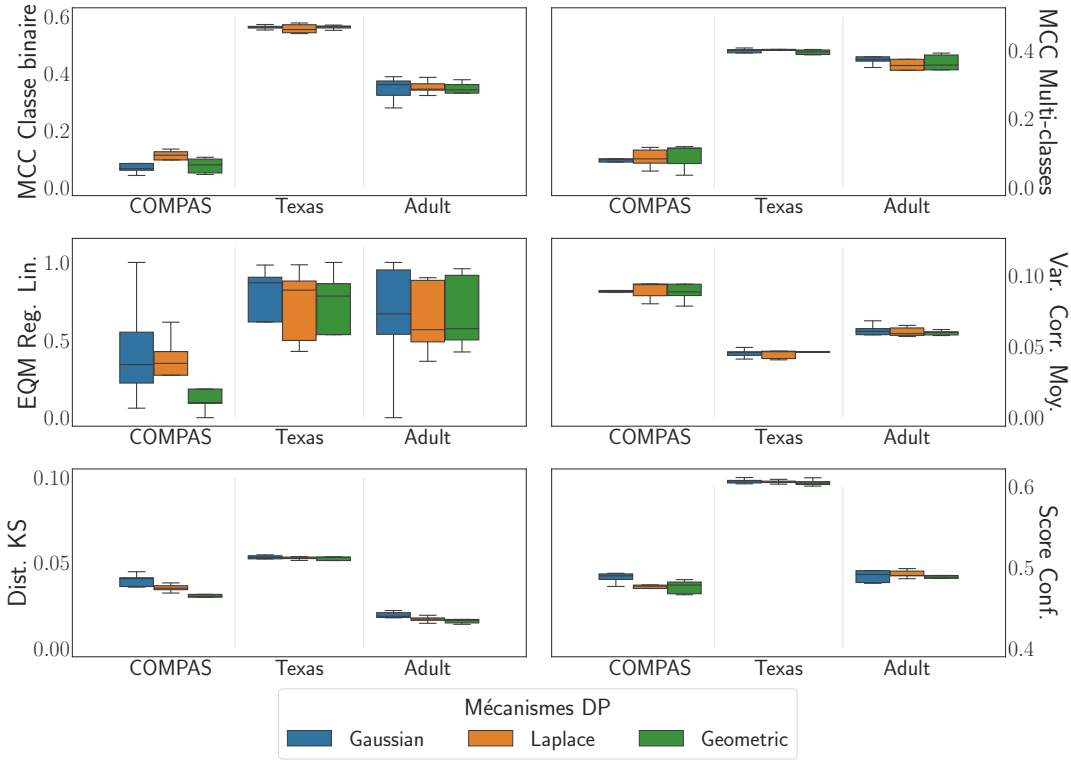


FIGURE 5.3 – L'impact de différents mécanismes de CD. Résultats obtenus avec $\epsilon = 0.1$.

5.2.4 Niveau d'élagage des copules vignes.

Dans la figure 5.4, Ψ = 'auto' signifie que le niveau d'élagage est déterminé à l'aide la méthode du seuil implémentée dans la bibliothèque R qui arrête l'algo-

rithme de Dissmann lorsque toutes les copules bivariées sont ajustées à la copule indépendante. $\Psi = 0$ signifie qu'aucun élagage n'est effectué.

La figure 5.4 illustre qu'une vigne plus profonde n'est pas nécessairement un meilleur modèle. Lorsqu'elles sont élaguées au deuxième arbre, les vignes présentent de bonnes mesures statistiques ainsi que de bonnes performances de régression. Il n'y a pas de consensus sur les tâches de classification, si ce n'est que toutes les valeurs de Ψ offrent des mesures assez similaires.

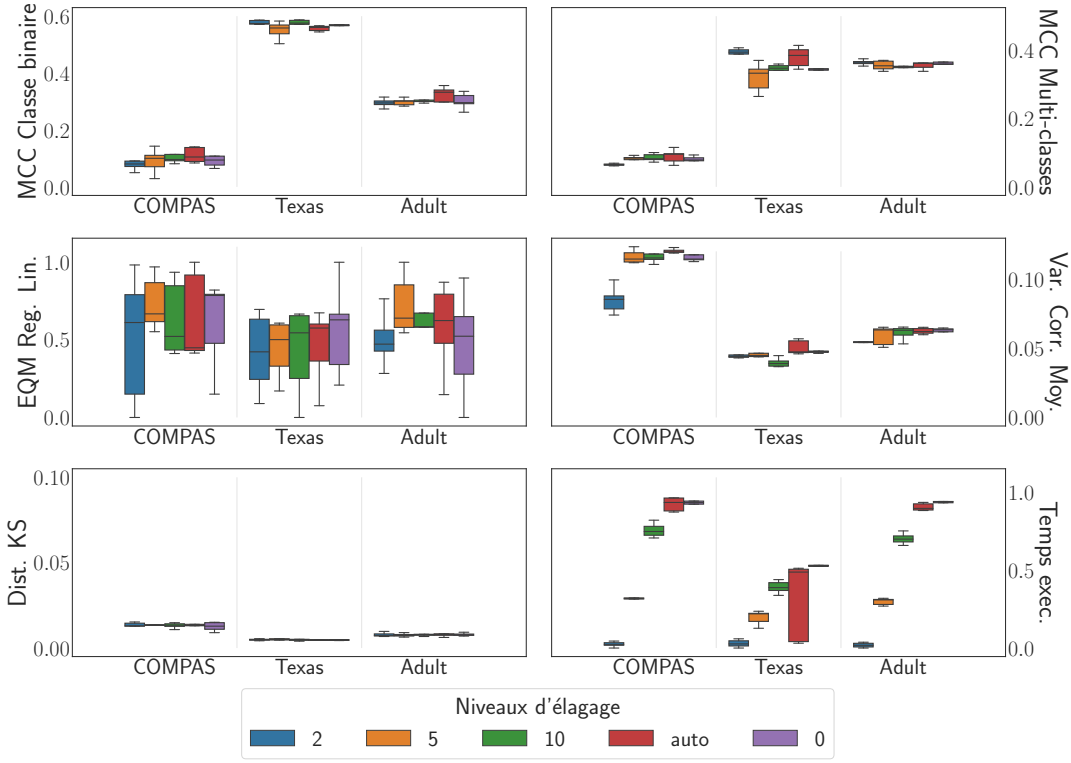


FIGURE 5.4 – L'impact du niveau d'élagage des copules vignes. Résultats obtenus avec $\epsilon = 1.0$.

5.2.5 Comparaison des modèles génératifs.

Les figures 5.5, 5.6 et 5.7 illustrent les scores pour chaque budget de confidentialité sur les trois ensembles de données et les quatre modèles séparément, tandis

que la figure 5.8 affiche les scores agrégés sur toutes les valeurs de ϵ. L'une des tendances que nous avons observées est que PrivBayes offre de meilleurs scores que COPULA-SHIRLEY pour la plupart des tâches de classification. DP-Copula et DP-Histogram ont complètement échoué aux deux tests de classification. Notre approche et PrivBayes ont obtenu de bons résultats dans la tâche de régression par rapport à DP-Copula. PrivBayes a généré les données présentant la plus petite variation de coefficients de corrélation par rapport aux ensembles de données originaux, à l'exception de l'ensemble de données du Texas dans lequel COPULA-SHIRLEY fournit les meilleurs résultats. De plus, COPULA-SHIRLEY est toujours le plus performant pour générer des distributions fidèles. L'évaluation de la confidentialité démontre qu'un budget de confidentialité plus faible ne signifie pas

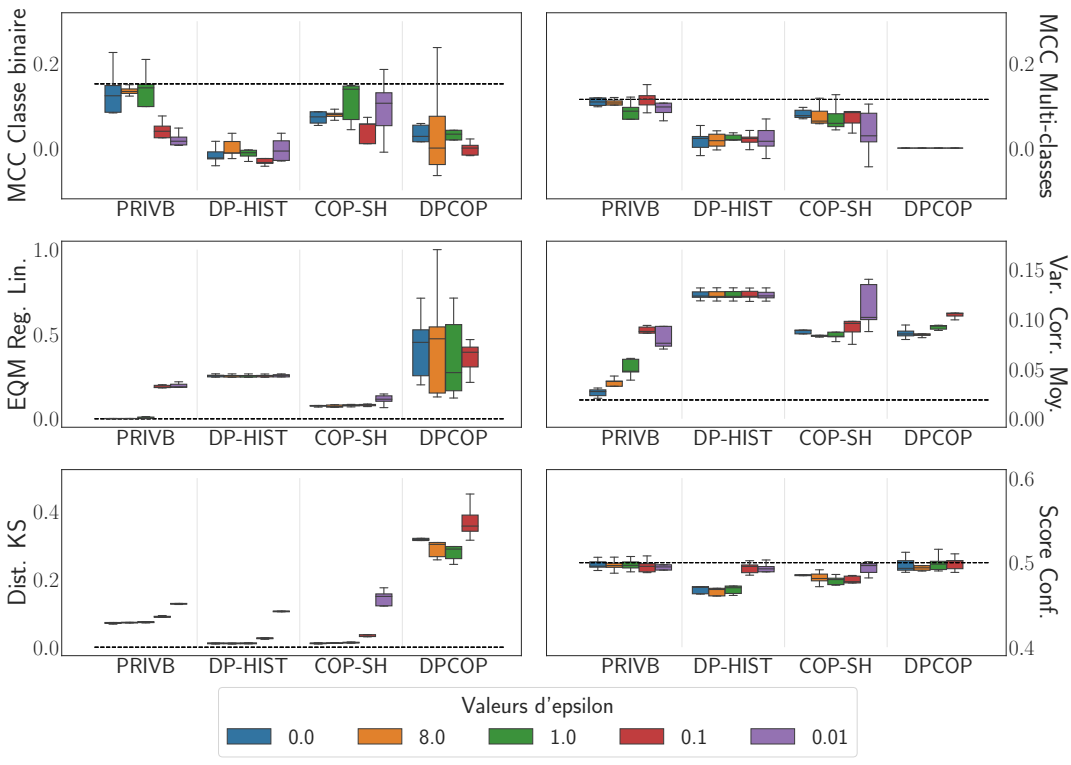


FIGURE 5.5 – Résultats sur COMPAS sur cinq valeurs de ϵ. Les lignes pointillées représentent les scores sur les données brutes.

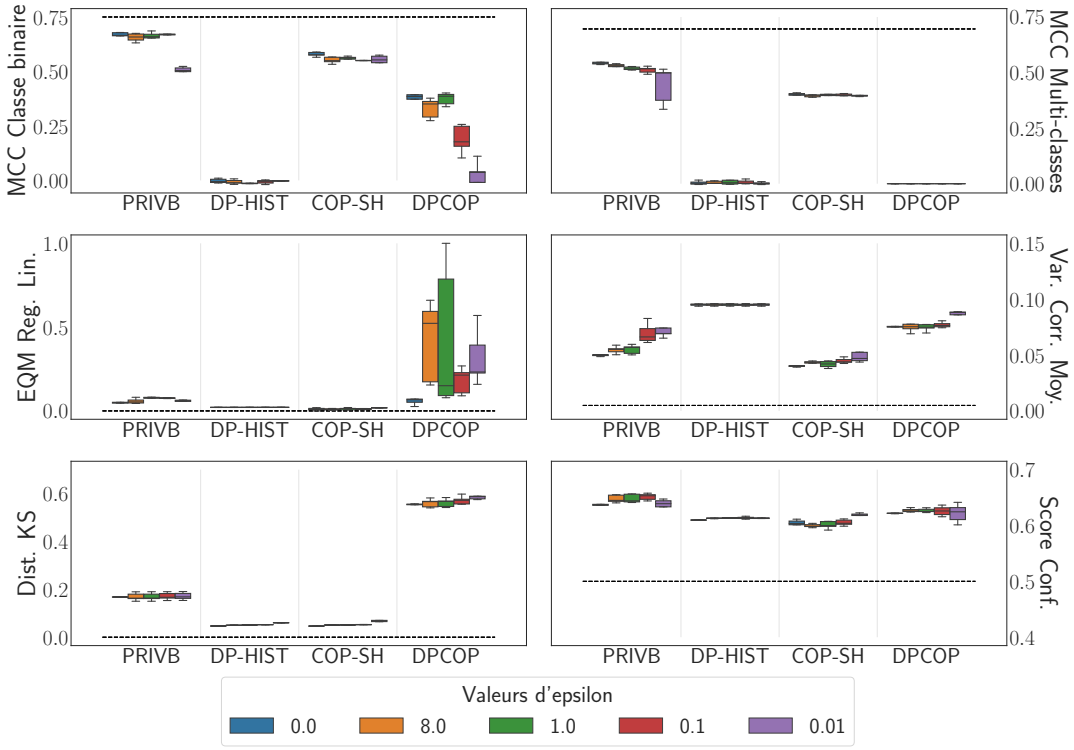


FIGURE 5.6 – Résultats sur Texas sur cinq valeurs de ϵ. Les lignes pointillées représentent les scores sur les données brutes.

nécessairement un risque plus faible d'inférence d'appartenance. En outre, tous les modèles ne fournissent pas une protection cohérente sur tous les ensembles de données. Alors que PrivBayes offre la meilleure protection sur le plus petit jeu de données (COMPAS), COPULA-SHIRLEY est le meilleur sur le plus grand jeu de données (Texas).

La figure 5.8 présente les scores globaux pour les données synthétiques de chaque modèle génératif. Les scores globaux sont obtenus en prenant la moyenne des scores sur les cinq valeurs de ϵ. PrivBayes a produit les meilleurs résultats globaux pour les tâches de classification et quelques résultats décents pour la tâche de régression linéaire et le score de variation de la corrélation moyenne. D'après ces résultats, PrivBayes semble le meilleur des quatre modèles pour capturer l'in-

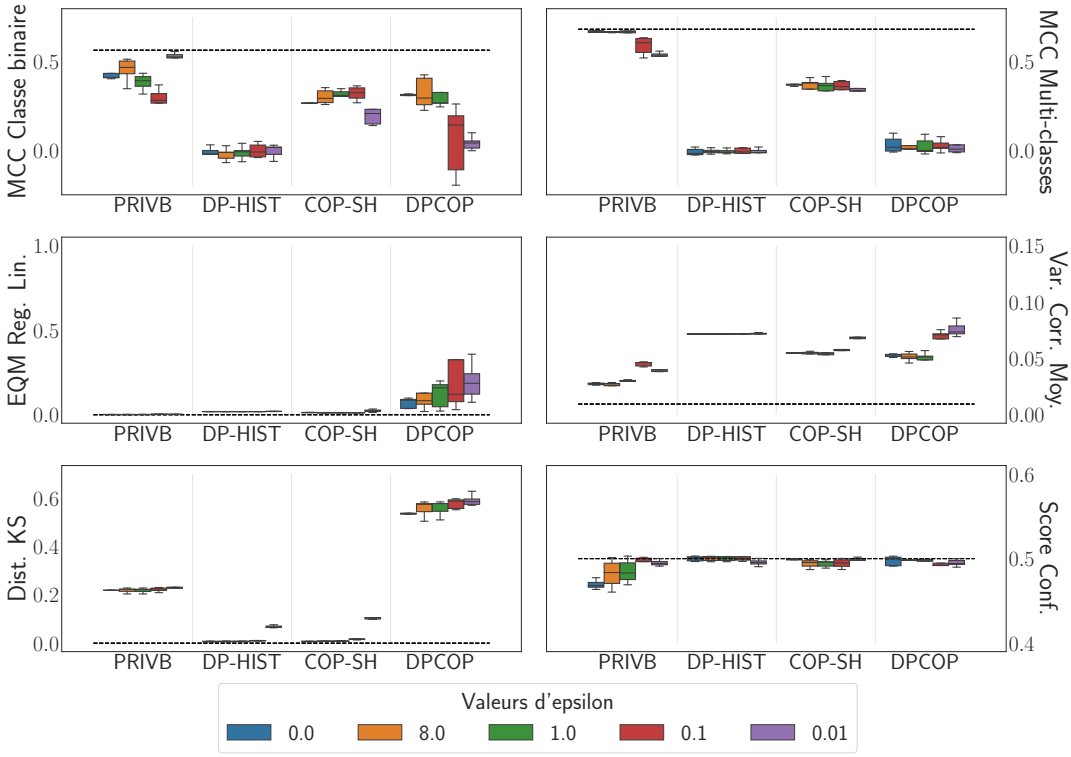


FIGURE 5.7 – Résultats sur Adult sur cinq valeurs de ϵ. Les lignes pointillées représentent les scores sur les données brutes.

terdépendance des attributs. DP-Histogram a produit certaines des pires données pour la classification et certains des pires scores pour la métrique de corrélation, ce qui n'est pas surprenant étant donné qu'aucune structure de dépendance n'a été apprise. COPULA-SHIRLEY est le deuxième meilleur modèle pour la plupart des tests et il a produit certaines des distributions les plus fidèles avec DP-Histogram. Notre méthode montre également un ajustement plus stable aux données d'entraînement que PrivBayes et DP-Copula. En prenant en compte la hauteur des barres, il est possible de dire que PrivBayes est le deuxième modèle génératif le plus incohérent. Le modèle DP-Copula a produit les distributions les moins fiables et a complètement échoué au test de classification multi-classes et à la tâche de régression linéaire. Pour le test d'attaque par inférence d'appartenance, tous les modèles ont fourni une protection décente, mais notre méthode a fourni

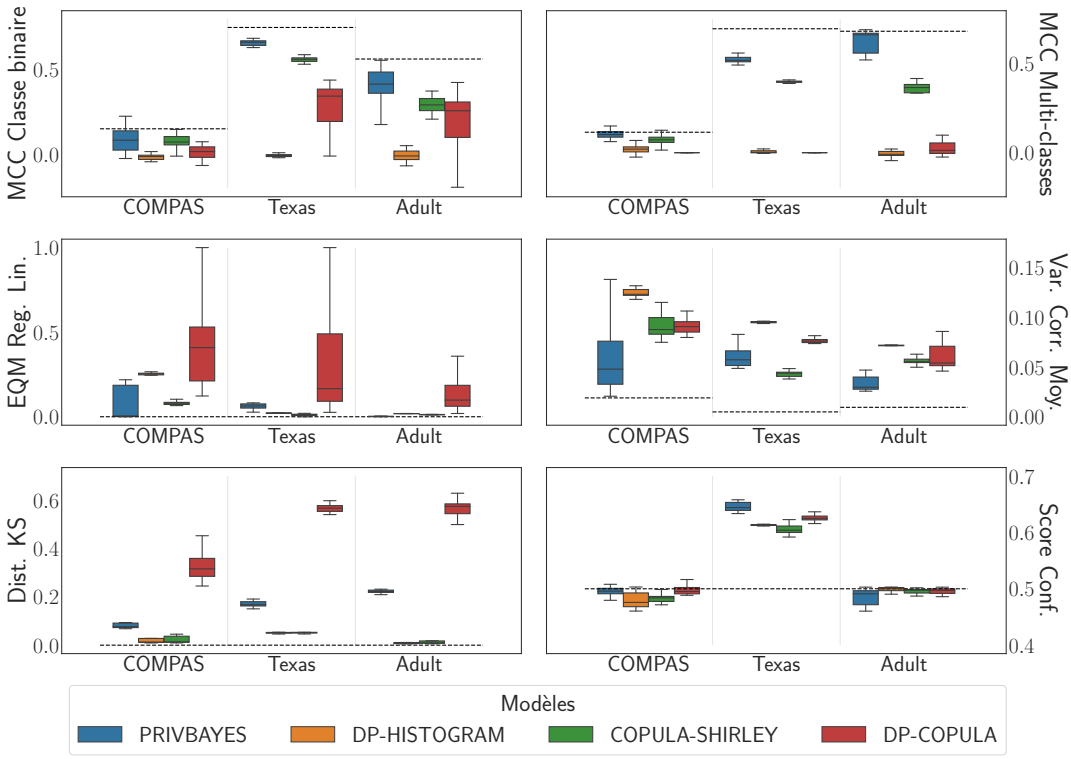


FIGURE 5.8 – Résultats globaux des modèles génératifs agrégés sur les cinq valeurs de ϵ. Les lignes pointillées représentent les scores sur les données brutes.

la meilleure protection globale sur le plus grand ensemble de données, PrivBayes offrant la pire. D'après les figures 5.6 et 5.8, COPULA-SHIRLEY semble être le meilleur pour modéliser et protéger les données de Texas.

5.2.6 Temps d'exécution.

Toutes les expériences ont été menées sur un processeur Intel core i5-6600k avec 16 Go de mémoire flash. Les temps d'exécution indiqués dans le tableau 5.1 renforcent l'observation que l'apprentissage des modèles bayésiens peut être extrêmement long lorsque la dimension des données augmente. La complexité des copules vignes (COPULA-SHIRLEY) est également considérablement plus élevée que celle des copules multivariées simples (DP-Copula). Le modèle DP-Histogram n'est pas

comparé puisque son temps d'exécution est virtuellement nul.

Dataset	COPULA-SHIRLEY	PrivBayes	DP-Copula
Adult (32 561 × 14)	33.452	67.211	2.241
COMPAS (10 568 × 13)	1.592	3.184	0.695
Texas (150 000 × 17)	34.055	123.157	17.945

TABLEAU 5.1 – Temps d'exécution moyens en minutes de chaque modèle génératif sur les trois ensembles de données.

	A - B	A - C	A - D	A - E	A - F	A<0 - F	A>0 - F
Référence	0.9150	-0.9565	0.2319	-0.1701	0.4018	0.7881	-0.1137
COP-SHIRL	**0.9161**	**-0.9555**	**0.2763**	-0.0477	**0.4059**	0.5484	0.0216
PrivBayes	0.8794	-0.9076	0.0899	**-0.1860**	0.3529	**0.655**	**-0.1014**
DP-Cop	0.7933	-0.1284	0.1507	-0.0552	0.3417	0.1829	0.1472
DP-Hist	0.0220	0.0127	-0.0268	-0.0367	-0.0191	-0.0186	-0.0002

TABLEAU 5.2 – Les coefficients de corrélation de Spearman entre la paire d'attributs. Les meilleurs scores sont en gras.

5.2.7 Analyse supplémentaire sur les corrélations multivariées.

Nous avons créé un petit ensemble de données synthétique composé de 5000 profils et de 6 attributs (nommés de A à F) avec différents coefficients de corrélation entre les attributs pour analyser la force des modèles à capturer la structure de dépendance dans les données. L'ensemble de données a été conçu de manière à ce que les attributs A et F soient fortement corrélés positivement lorsque les valeurs de A sont inférieures à zéro, et seulement légèrement corrélés négativement lorsque les valeurs de A sont supérieures à zéro. Ceci est illustré dans le tableau 5.2 aux colonnes « A<0 - F » et « A>0 - F ». Notre méthode offre les coefficients de corrélation les plus proches de ceux d'origine quatre fois sur sept. PrivBayes parvient mieux à capturer la dépendance lorsqu'elle varie dans différentes parties des données. Ces résultats soulignent également le fait que l'approche de la copule de la vigne est supérieure pour modéliser la structure de dépendance que la DP-

Copula et que l'approche naïve consistant à échantillonner simplement à partir des d'histogrammes. Les figures 5.9, 5.10a, 5.10b, 5.10c et 5.10d montrent le nuage de points des données synthétiques utilisées pour l'analyse ainsi que les nuages de points des observations échantillonnées à partir des quatre modèles.

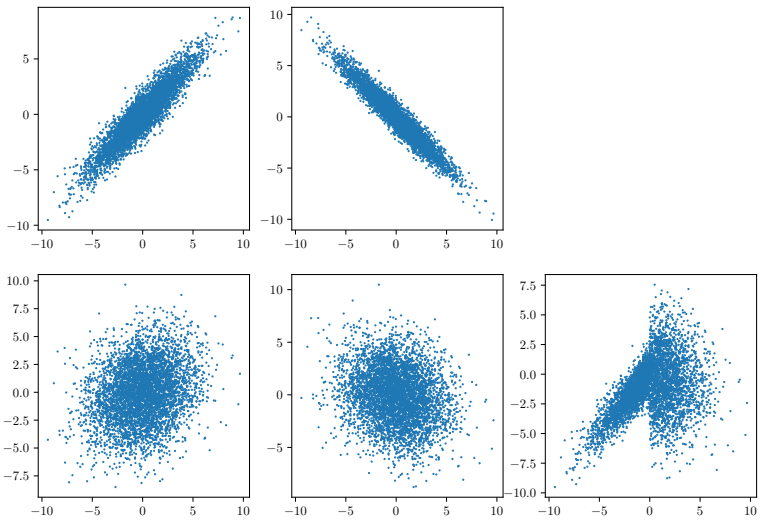
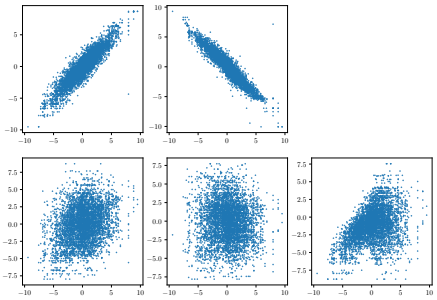
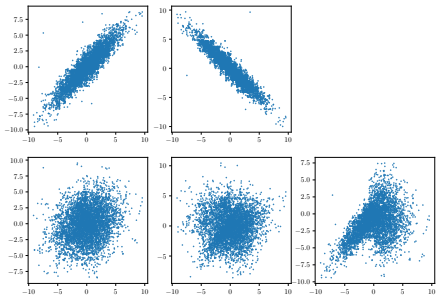


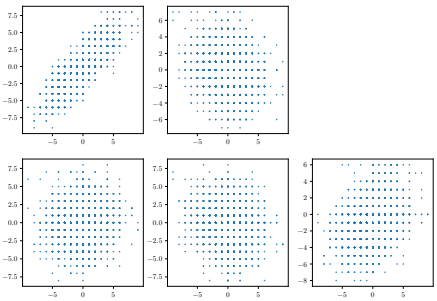
FIGURE 5.9 – Le nuage de points des données synthétiques de référence utilisées pour l'analyse de corrélation multivariée.



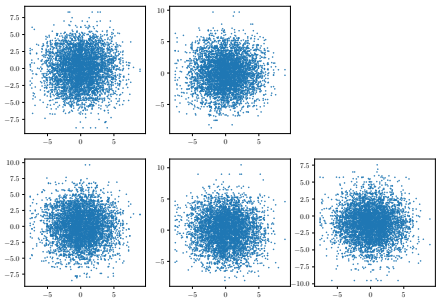
(a) Observations échantillonnées de COPULA-SHIRLEY.



(b) Observations échantillonnées de PrivBayes.



(c) Observations échantillonnées de DP-Copula.



(d) Observations échantillonnées de DP-Histograms.

FIGURE 5.10 – Les nuages de points des observations échantillonnées à partir des modèles génératifs.

5.2.8 Synthèse des résultats.

Notre méthode COPULA-SHIRLEY affiche la plus grande similarité statistique entre les données brutes et synthétiques. Alors que notre approche basée sur les copules vignes surpasse DP-Copula sur les tâches de classification la plupart du temps, elle n'a pas fait de même contre PrivBayes. COPULA-SHIRLEY a été à égalité avec PrivBayes à quelques reprises, principalement pour la tâche de régression linéaire. PrivBayes a généré des distributions synthétiques décentes avec une dépendance inter-attribut remarquable, ce qui explique pourquoi PrivBayes a toujours obtenu le meilleur score pour les tests de classification. L'inconvénient majeur de PrivBayes est le temps d'exécution pour l'entraînement du modèle. Globale-

ment, COPULA-SHIRLEY a offert une qualité de données décente, un ajustement plus stable et une protection plus forte que PrivBayes sur le plus grand ensemble de données. Il est toutefois clair que COPULA-SHIRLEY offre une nette amélioration par rapport à la copule multivariée et à l'échantillonnage naïf DP-Histogrammes. De plus, nous pensons que les performances de COPULA-SHIRLEY pourraient être accrues par quelques optimisations, tel qu'un algorithme de sélection de copules vignes plus sophistiqué ou un prétraitement approfondi pour une meilleure préservation de la corrélation des rangs.

5.3 Limitations et travaux futurs

Les limitations de la méthode proposée COPULA-SHIRLEY et de son implémentation sont nombreuses, mais non sans issue. Les quelques points suivants résument les principales limitations ainsi que des pistes de solutions.

Estimation des FDPs et FDCs rudimentaire. En effet, l'estimation des fonctions de densité dans l'implémentation courante ne peut se faire plus simplement. Bien que le cadre de l'algorithme COPULA-SHIRLEY permet d'utiliser d'autres fonctions d'estimation de FDPs, le respect de la CD est primordial à cette étape. Un bref survol de la littérature a révélé quelques techniques de calcul d'histogrammes différentiellement-privées (Acs *et al.*, 2012; Xiao *et al.*, 2010; Xu *et al.*, 2013) susceptibles d'améliorer grandement la qualité des données synthétiques générées par COPULA-SHIRLEY, étant donné que l'approche est entièrement basée sur ce calcul.

Aucune protection de groupe. Un exemple où l'application d'une telle protection est nécessaire est lorsqu'un individu contribue pour plus d'un profil dans les données. Il est très possible, surtout avec des données temporelles, qu'un individu contribue plusieurs fois dans les données. Cependant, la CD ne protège pas ce comportement

dans les données sans ajuster le niveau de bruit. L'implémentation proposée de cette protection est relativement simple et nécessite seulement que chaque individu soit associé à une clé unique dans les données («`uid`» en termes de bases de données relationnelles). À partir de ces clés, il est possible d'ajuster la protection de la CD en fonction du nombre de contributions maximal des individus. Par exemple, si tous les individus contribuent k profils à l'ensemble de données, il suffira d'ajuster le bruit ajouté proportionnellement à ce nombre (Dwork *et al.*, 2014).

Étant la première itération de l'implémentation, plusieurs points peuvent être soulevés pour améliorer la qualité des données Parmi ces travaux, outre ceux qui impliquent les limitations discutées plus tôt, les premiers en liste sont les suivants :

1. Comparer l'utilité des données générées par les modèles lorsqu'ils sont entraînés sur les mêmes pseudo-observations que notre méthode. Cette étude permettrait de voir si la perte d'utilité est principalement causée par cette transformation.
2. Utiliser la méthode de l'article (Sun *et al.*, 2019) de construction de copule vigne à l'aide de réseaux de neurones.
3. Tester la « sur-génération », c'est-à-dire générer beaucoup plus de profils que la quantité de profils en entrée, et son effet sur la protection.
4. Segmenter l'apprentissage en regroupements significatifs (« clusters »). En effet, il est possible de segmenter les données sans briser la protection de la CD (voir théorème 2.3). Cette segmentation peut être utile dans les cas où, par exemple, les données synthétiques serviraient à la classification, au sens où plusieurs copules vignes apprennent sur les différentes valeurs de l'attribut classe, ce qui possiblement peut aider la modélisation.
5. Tester la limite inférieure de données nécessaires pour une bonne modélisation des données. Ce test pourrait se faire avec un cadre simple où plusieurs copules sont entraînées sur des ensembles de profils de cardinalité décrois-

sante et où chaque copule génère le même nombre de profils. Le même test pourrait être utilisé sur d'autres modèles génératifs pour y comparer leur perte d'information.

6. Développer un cadre ou une méthode pour réduire la dimension des données tout en conservant la qualité d'être différentiellement-privé, comme dans l'article (Tagasovska *et al.*, 2019), où les auto-encodeurs sont utilisés pour réduire la dimension des données. Les données réduites sont ensuite modélisées à l'aide de copules. Dans le cadre de COPULA-SHIRLEY, il serait possible d'utiliser un encodeur entre les pseudo-observations fournies en entrée à l'algorithme de construction de copule vigne.

Ces travaux ouvrent la voie à des implémentations plus performantes et à des expériences poussant les copules à leurs limites pour justifier ou infirmer l'utilisation de copules comme modèles génératifs respectueux de la vie privée.

CONCLUSION

Le modèle génératif respectueux de la vie privée basé sur les copules vignes proposé, COPULA-SHIRLEY, permet la publication de données possiblement sensibles sans affecter la protection de la vie privée des individus concernés. Dans ce mémoire, l'approche définie, décrite et testée se distingue des autres modèles basés sur les copules par son cadre simplifié : tous les calculs reliés à la construction de copules vignes sont estimées à l'aide des densités marginales ϵ-différentiellement-privées et seulement de ces densités. Cette démarche simplifiée réduit la quantité de bruit injecté par la confidentialité différentielle et permet une bonne flexibilité vis-à-vis de l'implémentation de l'algorithme. Comparée à trois autres modèles génératifs, COPULA-SHIRLEY est la méthode qui offrait le meilleur niveau de protection sur l'ensemble de données le plus grand tout en conservant une bonne utilité. Généralement parlant, l'utilité des données synthétiques générées reste cependant un point à améliorer, principalement ce qui a trait à la corrélation inter-attribut.

La méthode développée peut être retravaillée de plusieurs façons et n'est que prometteuse de données fidèles et privées. Les copules ne sont qu'à leurs premiers pas dans le domaine de la génération de données respectueuses de la vie privée et COPULA-SHIRLEY n'est qu'un pas de plus dans cette direction.

RÉFÉRENCES

Aas, K., Czado, C., Frigessi, A. et Bakken, H. (2009). Pair-copula constructions of multiple dependence. *Insurance : Mathematics and economics 44*(2), 182–198.

Abidi, B., Yahia, S. B. et Perera, C. (2020). Hybrid microaggregation for privacy preserving data mining. *Journal of Ambient Intelligence and Humanized Computing 11*(1), 23–38.

Acar, E. F., Czado, C. et Lysy, M. (2019). Flexible dynamic vine copula models for multivariate time series data. *Econometrics and Statistics 12*, 181–197.

Acs, G., Castelluccia, C. et Chen, R. (2012). Differentially private histogram publishing through lossy compression. Dans *2012 IEEE 12th International Conference on Data Mining* (p. 1–10). Bruxelles, Belgique : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de `https://ieeexplore.ieee.org/document/6413718`

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. Dans H. Akaike (dir.), *Selected papers of hirotugu akaike* (p. 199–213). New York : Springer.

Anderson, J. A. (1997). *An Introduction to Neural Networks.* Cambridge : The MIT Press.

Andrés, M. E., Bordenabe, N. E., Chatzikokolakis, K. et Palamidessi, C. (2013). Geo-indistinguishability : Differential privacy for location-based systems. Dans *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security* (p. 901–914). Berlin, Allemagne : Association for Computing Machinery (ACM). Récupéré de `https://dl.acm.org/doi/abs/10.1145/2508859.2516735`

Angwin, J., Larson, J., Kirchner, L. et Mattu, S. (2019). Machine bias. Récupéré le 13 janvier 2020 de `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing`

Asghar, H. J., Ding, M., Rakotoarivelo, T., Mrabet, S. et Kaafar, M. A. (2019). *Differentially Private Release of High-Dimensional Datasets using the Gaussian Copula. arXiv preprint.* Prépublication. Récupéré de `https://arxiv.org/abs/1902.01499`

Balle, B. et Wang, Y.-X. (2018). Improving the gaussian mechanism for differential privacy : Analytical calibration and optimal denoising. Dans *International Conference on Machine Learning* (p. 394–403). Stockholm, Suède : International Conference on Machine Learning (ICML). Récupéré de `https://icml.cc/Conferences/2018/ScheduleMultitrack?event=2245`

Barbaro, M., Zeller, T. et Hansell, S. (2006). A face is exposed for aol searcher no. 4417749. *New York Times 9*(2008), p. 8.

Bárdossy, A. et Pegram, G. (2009). Copula based multisite model for daily precipitation simulation. *Hydrology & Earth System Sciences 13*(12), 2299—-2314.

Bedford, T. et Cooke, R. M. (2001). Probability density decomposition for conditionally dependent random variables modeled by vines. *The Annals of Mathematics and Artificial Intelligence 32*(1-4), 245–268.

Bedford, T. et Cooke, R. M. (2002). Vines : A new graphical model for dependent random variables. *The Annals of Statistics 30*(4), 1031–1068.

Bindschaedler, V. (2018). *Privacy-preserving seedbased data synthesis.* (Thèse de doctorat). University of Illinois at Urbana-Champaign.

Bindschaedler, V. et Shokri, R. (2016). Synthesizing plausible privacy-preserving location traces. Dans *2016 IEEE Symposium on Security and Privacy* (p. 546–563). San Jose, États-Unis : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de `https://ieeexplore.ieee.org/document/7546522`

Bindschaedler, V., Shokri, R. et Gunter, C. A. (2017). *Plausible deniability for privacy-preserving data synthesis. arXiv preprint.* Prépublication. Récupéré de `https://arxiv.org/abs/1708.07975`

Bowen, C. M. et Snoke, J. (2019). Comparative study of differentially private synthetic data algorithms and evaluation standards. *arXiv preprint.* Prépublication. Récupéré de `https://arxiv.org/abs/1911.12704`

Brechmann, E. C., Czado, C. et Aas, K. (2012). Truncated regular vines in high dimensions with application to financial data. *Canadian Journal of*

Statistics 40(1), 68–85.

Brechmann, E. C. et Joe, H. (2015). Truncation of vine copulas using fit indices. *Journal of Multivariate Analysis 138*, 19–33.

Cadwalladr, C. et Graham-Harrison, E. (2018). Revealed : 50 million facebook profiles harvested for cambridge analytica in major data breach. *The guardian 17*, p. 22.

Chen, T., He, T., Benesty, M., Khotilovich, V. et Tang, Y. (2015). Xgboost : extreme gradient boosting. *R package version 0.4-2* 1–4.

Chen, Y.-C. (2018). *STAT 425 : Introduction to Nonparametric Statistics - Lecture 6 : Density Estimation : Histogram and Kernel Density Estimator*. University of Washington. Récupéré le 11 janvier 2021 de `https://faculty.washington.edu/yenchic/18W_425/Lec6_hist_KDE.pdf`

Chicco, D. et Jurman, G. (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics 21*(1), 6.

Costello, M. J. (2009). Motivating online publication of data. *BioScience 59*(5), 418–427.

De Montjoye, Y.-A., Hidalgo, C. A., Verleysen, M. et Blondel, V. D. (2013). Unique in the crowd : The privacy bounds of human mobility. *Scientific reports 3*, 1376. doi : `10.1038/srep01376`.

De Montjoye, Y.-A., Radaelli, L., Singh, V. K. *et al.* (2015). Unique in the shopping mall : On the reidentifiability of credit card metadata. *Science 347*(6221), 536–539.

Desjardins, F. (11 décembre 2019). Fuite de données chez desjardins : 1,8 million de détenteurs de cartes de crédit touchés. *Le Devoir*. Récupéré de `https://www.ledevoir.com/economie/568794/vol-de-donnees-chez-desjardins-1-8-million-de-detenteurs-de-cartes-de-credit-touches`

Devroye, L. (1986). *Non-Uniform Random Variate Generation*. New York : Springer-Verlag.

Dissmann, J., Brechmann, E. C., Czado, C. et Kurowicka, D. (2013). Selecting and estimating regular vine copulae and application to financial returns. *Computational Statistics & Data Analysis 59*, 52–69.

Domingo-Ferrer, J., Martínez-Ballesté, A., Mateo-Sanz, J. M. et Sebé, F. (2006). Efficient multivariate data-oriented microaggregation. *The VLDB*

Journal 15(4), 355–369.

Domingo-Ferrer, J., Oganian, A., Torres, À. et Mateo-Sanz, J. M. (2002). On the security of microaggregation with individual ranking : analytical attacks. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10*(05), 477–491.

Domingo-Ferrer, J., Sánchez, D. et Soria-Comas, J. (2016). Database anonymization : Privacy models, data utility, and microaggregation-based inter-model connections. *Synthesis Lectures on Information Security, Privacy, and Trust 8*, 1–136.

Dua, D. et Graff, C. (2017). UCI machine learning repository. Récupéré de `http://archive.ics.uci.edu/ml`

Dwork, C., Roth, A. *et al.* (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science 9*(3–4), 211–407.

Dwork, C. et Smith, A. (2010). Differential privacy for statistics : What we know and what we want to learn. *Journal of Privacy and Confidentiality 1*(2).

Fisher, R. A. (1992). Statistical methods for research workers. Dans S. Kotz et N. L. Johnson (dir.), *Breakthroughs in statistics* (p. 66–70). New York : Springer.

Friedman, J. H. (2001). Greedy function approximation : a gradient boosting machine. *Annals of statistics* 1189–1232.

Gambs, S., Killijian, M.-O. et del Prado Cortez, M. N. (2010). Show me how you move and i will tell you who you are. Dans *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS* (p. 34–41). San Jose, États-Unis : Association for Computing Machinery (ACM). Récupéré de `https://dl.acm.org/doi/abs/10.1145/1868470.1868479`

Gambs, S., Ladouceur, F., Laurent, A. et Roy-Gaumond, A. (2021). Growing synthetic data through differentially-private vine copulas. *Proceedings on Privacy Enhancing Technologies 3*, 122–141. Prépublication.

Geisser, S. (1975). The predictive sample reuse method with applications. *Journal of the American statistical Association 70*(350), 320–328.

Ghosh, A., Roughgarden, T. et Sundararajan, M. (2012). Universally utility-maximizing privacy mechanisms. *SIAM Journal on Computing 41*(6), 1673–1693.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. et Bengio, Y. (2014). Generative adversarial nets. Dans *28th Conference on Neural Information Processing Systems* (p. 2672–2680). Montréal, Canada : Advances in Neural Information Processing Systems. Récupéré de `https://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`

Gursoy, M. E., Liu, L., Truex, S., Yu, L. et Wei, W. (2018). Utility-aware synthesis of differentially private and attack-resilient location traces. Dans *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (p. 196–211). Toronto, Canada : Association for Computing Machinery (ACM). Récupéré de `https://dl.acm.org/doi/abs/10.1145/3243734.3243741`

Hamming, R. W. (1950). Error detecting and error correcting codes. *The Bell system technical journal 29*(2), 147–160.

He, J., Li, H., Edmondson, A. C., Rader, D. J. et Li, M. (2012). A gaussian copula approach for the analysis of secondary phenotypes in case–control genetic association studies. *Biostatistics 13*(3), 497–508.

Hoyer, A. et Kuss, O. (2015). Meta-analysis of diagnostic tests accounting for disease prevalence : a new model using trivariate copulas. *Statistics in medicine 34*(11), 1912–1924.

Jensen, F. V. (1997). *Introduction to Bayesian Networks.* New York : Springer.

Joe, H. (1997). *Multivariate models and multivariate dependence concepts.* Londres : Chapman & Hall/CRC.

Jordon, J., Yoon, J. et van der Schaar, M. (2019). PATE-GAN : Generating synthetic data with differential privacy guarantees. Dans *International Conference on Learning Representations (ICLR).* New Orleans, États-Unis : ICLR. Récupéré de `https://openreview.net/pdf?id=S1zk9iRqF7`

Joy, J. et Gerla, M. (2017). *Differential privacy by sampling. arXiv preprint.* Prépublication. Récupéré de `https://arxiv.org/abs/1708.01884`

Kelleher, J. D. (2019). *Deep Learning.* Cambridge : The MIT Press.

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika 30*(1/2), 81–93.

Kraus, D. et Czado, C. (2017). *Growing simplified vine copula trees :*

improving Dissmann's algorithm. arXiv preprint. Prépublication. Récupéré de https://arxiv.org/abs/1703.05203

Kuhn, M. et Johnson, K. (2019). *Feature engineering and selection : A practical approach for predictive models.* CRC Press.

Kulkarni, V., Tagasovska, N., Vatter, T. et Garbinato, B. (2018). *Generative Models for Simulating Mobility Trajectories. arXiv preprint.* Prépublication. Récupéré de https://arxiv.org/abs/1811.12801

Lazer, D., Brewer, D., Christakis, N., Fowler, J. et King, G. (2009). Life in the network : the coming age of computational social. *Science 323*(5915), 721–723.

Li, B., Liu, Y., Han, X. et Zhang, J. (2017). Cross-bucket generalization for information and privacy preservation. *IEEE Transactions on Knowledge and Data Engineering 30*(3), 449–459.

Li, H., Xiong, L. et Jiang, X. (2014). Differentially private synthesization of multi-dimensional data using copula functions. Dans *Proceedings of the 17th International Conference on Extending Database Technology* volume 2014 (p. 475–486). Athènes, Grèce : Extending Database Technology (EDBT). Récupéré de https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4232968/

Li, N., Li, T. et Venkatasubramanian, S. (2007). t-closeness : Privacy beyond k-anonymity and l-diversity. Dans *2007 IEEE 23rd International Conference on Data Engineering* (p. 106–115). Istanbul, Turquie : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de https://ieeexplore.ieee.org/document/4221659

Li, N. L., Zhang, Z. et Wang, T. (2019). Dpsyn. Récupéré le 10 janvier 2020 de https://github.com/usnistgov/PrivacyEngCollabSpace/tree/master/tools/de-identification/Differential-Privacy-Synthetic-Data-Challenge-Algorithms/DPSyn

Ligaya, A. (19 septembre 2017). Cent mille canadiens touchés par le piratage d'equifax. *La Presse.* Récupéré de https://www.lapresse.ca/techno/201709/19/01-5134603-cent-mille-canadiens-touches-par-le-piratage-dequifax.php

Machanavajjhala, A., Kifer, D., Gehrke, J. et Venkitasubramaniam, M. (2007). l-diversity : Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD) 1*(1), 3–es.

Matthews, B. W. (1975). Comparison of the predicted and observed

secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure 405*(2), 442–451.

McGinnis, W. D., Siu, C., Andre, S. et Huang, H. (2018). Category encoders : a scikit-learn-contrib package of transformers for encoding categorical data. *Journal of Open Source Software 3*(21), 501.

McKenna, R. (2019). rmckenna - differential privacy synthetic data challenge algorithm. Récupéré le 10 janvier 2020 de `https://github.com/usnistgov/PrivacyEngCollabSpace/tree/master/tools/de-identification/Differential-Privacy-Synthetic-Data-Challenge-Algorithms/rmckenna`

McSherry, F. et Talwar, K. (2007). Mechanism design via differential privacy. Dans *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)* (p. 94–103). Providence, USA : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de `https://ieeexplore.ieee.org/document/4389483`

Mivule, K. (2013). *Utilizing noise addition for data privacy, an overview. arXiv preprint*. Prépublication. Récupéré de `https://arxiv.org/abs/1309.3958`

Muise, D. et Nissim, K. (2016). *Notes on Differential Privacy in CDFs*. Harvard University : Harvard University Privacy Tools Project. Récupéré le 16 décembre 2019 de `\url{https://privacytools.seas.harvard.edu/files/privacytools/files/dpcdf_usermanual_2016.pdf}`

Nagler, T., Bumann, C. et Czado, C. (2019). Model selection in sparse high-dimensional vine copula models with an application to portfolio risk. *Journal of Multivariate Analysis 172*, 180–192.

Nagler, T. et Vatter, T. (2017). R interface to the vinecopulib c++ library. Récupéré le 2 juin 2019 de `https://github.com/vinecopulib/rvinecopulib`

Narayanan, A. et Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. Dans *2008 IEEE Symposium on Security and Privacy* (p. 111–125). Oakland, États-Unis : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de `https://ieeexplore.ieee.org/document/4531148`

Nelsen, R. B. (2007). *An introduction to copulas (2e éd.)*. New York : Springer Science & Business Media.

Panchenko, D. (2006). *Statistics for Applications : Kolmogorov-Smirnov test, MIT18.650*. Massachusetts Institute of Technology : MIT OpenCourseWare. Récupéré le 16 décembre 2019 de \url{https://ocw.mit.edu}

Papernot, N., Abadi, M., Erlingsson, U., Goodfellow, I. et Talwar, K. (2016). *Semi-supervised knowledge transfer for deep learning from private training data. arXiv preprint*. Prépublication. Récupéré de https://arxiv.org/abs/1610.05755

Patton, A. J. (2009). Copula–based models for financial time series. Dans T. Gustav Andersen, R. A. Davis, J.-P. Kreiß, et T. V. Mikosch (dir.), *Handbook of financial time series* (p. 767–785). Springer.

Ping, H., Stoyanovich, J. et Howe, B. (2017). Datasynthesizer : Privacy-preserving synthetic datasets. Dans *Proceedings of the 29th International Conference on Scientific and Statistical Database Management* (p. 1–5). Chicago, États-Unis : Association for Computing Machinery (ACM). Récupéré de https://dl.acm.org/doi/abs/10.1145/3085504.3091117

Prim, R. C. (1957). Shortest connection networks and some generalizations. *The Bell System Technical Journal 36*(6), 1389–1401.

Puth, M.-T., Neuhäuser, M. et Ruxton, G. D. (2015). Effective use of spearman's and kendall's correlation coefficients for association between two measured traits. *Animal Behaviour 102*, 77–84.

Rakotoarivelo, T. (2019). Dpcopula-kendall algorithm. Récupéré le 10 janvier 2020 de https://github.com/thierryr/dpcopula_kendall

Rocher, L., Hendrickx, J. M. et De Montjoye, Y.-A. (2019). Estimating the success of re-identifications in incomplete datasets using generative models. *Nature communications 10*(1), 1–9.

Rodriguez-Garcia, M., Batet, M. et Sánchez, D. (2019). Utility-preserving privacy protection of nominal data sets via semantic rank swapping. *Information Fusion 45*, 282–295.

Schwarz, G. *et al.* (1978). Estimating the dimension of a model. *The annals of statistics 6*(2), 461–464.

Shokri, R., Stronati, M., Song, C. et Shmatikov, V. (2017). Membership inference attacks against machine learning models. Dans *2017 IEEE Symposium on Security and Privacy* (p. 3–18). San Jose, États-Unis : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de https://ieeexplore.ieee.org/document/7958568

Singel, R. (12 février 2010). Netflix cancels recommendation contest after privacy lawsuit. *Wired*. Récupéré de https://www.wired.com/2010/03/netflix-cancels-contest/

Sklar, A. (1959). Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université de Paris 8*, 229–231.

Smirnov, N. V. (1944). Approximate laws of distribution of random variables from empirical data. *Uspekhi Matematicheskikh Nauk* (10), 179–206.

Spearman, C. (1987). The proof and measurement of association between two things. *The American journal of psychology 100*(3/4), 441–471.

Stewart, W. J. (1994). *Introduction to the Numerical Solution of Markov Chains*. Princeton : Princeton University Press.

Suits, D. B. (1957). Use of dummy variables in regression equations. *Journal of the American Statistical Association 52*(280), 548–551.

Sun, Y., Cuesta-Infante, A. et Veeramachaneni, K. (2019). Learning vine copula models for synthetic data generation. Dans *Proceedings of the 2019 AAAI Conference on Artificial Intelligence* volume 33 (p. 5049–5057). Honolulu, États-Unis : Advancement of Artificial Intelligence (AAAI). Récupéré de https://www.aaai.org/ojs/index.php/AAAI/article/view/4437

Sweeney, L. (2000). *Simple Demographics Often Identify People Uniquely*. Carnegie Mellon University Laboratory for Int'l Data Privacy, Working Paper LIDAP-WP4.

Sweeney, L. (2002). k-anonymity : A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10*(05), 557–570.

Tagasovska, N., Ackerer, D. et Vatter, T. (2019). *Copulas as High-Dimensional Generative Models : Vine Copula Autoencoders*. *arXiv preprint*. Prépublication. Récupéré de https://arxiv.org/abs/1906.05423

Terrovitis, M., Poulis, G., Mamoulis, N. et Skiadopoulos, S. (2017). Local suppression and splitting techniques for privacy preserving publication of trajectories. *IEEE Transactions on Knowledge and Data Engineering 29*(7), 1466–1479.

Texas Department of State Health Services, Austin, Texas (2013). *Texas Hospital Inpatient Discharge Public Use Data File 2013 Q1*. Récupéré le 13 janvier 2020 de

\url{https://www.dshs.texas.gov/THCIC/Hospitals/Download.shtm}

Tockar, A. (2014). Riding with the stars : Passenger privacy in the nyc taxicab dataset. Récupéré de http://research.neustar.biz/2014/09/15/riding-with-the-stars-passenger-privacy-in-the-nyc-taxicab-dataset/

United Nations Statistical Commision et Economic Commision For Europe (2000). *Terminology on Statistical Metadata.* [Document non publié]. Récupéré de https://ec.europa.eu/eurostat/ramon/coded_files/UNECE_TERMINOLOGY_STAT_METADATA_2000_EN.pdf

Upadhyay, S., Sharma, C., Sharma, P., Bharadwaj, P. et Seeja, K. (2018). Privacy preserving data mining with 3-d rotation transformation. *Journal of King Saud University-Computer and Information Sciences 30*(4), 524–530.

Vaserstein, L. N. (1969). Markov processes over denumerable products of spaces, describing large systems of automata. *Problemy Peredachi Informatsii 5*(3), 64–72.

Wang, J., Liu, S. et Li, Y. (2015). A review of differential privacy in individual data release. *International Journal of Distributed Sensor Networks 11*(10). doi : 10.1155/2015/259682.

Wang, K., Yu, P. S. et Chakraborty, S. (2004). Bottom-up generalization : A data mining solution to privacy protection. Dans *Fourth IEEE International Conference on Data Mining (ICDM'04)* (p. 249–256). Brighton, Royaume-Uni : Institute of Electrical and Electronics Engineers (IEEE). Récupéré de https://ieeexplore.ieee.org/document/1410291

Wod, I. (1985). Weight of evidence : A brief survey. *Bayesian statistics 2*, 249–270.

Xiao, X., Wang, G. et Gehrke, J. (2010). Differential privacy via wavelet transforms. *IEEE Transactions on knowledge and data engineering 23*(8), 1200–1214.

Xie, L., Lin, K., Wang, S., Wang, F. et Zhou, J. (2018). *Differentially private generative adversarial network. arXiv preprint.* Prépublication. Récupéré de https://arxiv.org/abs/1802.06739

Xu, J., Zhang, Z., Xiao, X., Yang, Y., Yu, G. et Winslett, M. (2013). Differentially private histogram publication. *The VLDB Journal 22*(6), 797–822.

Zhang, J., Cormode, G., Procopiuc, C. M., Srivastava, D. et Xiao, X. (2017).

Privbayes : Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS) 42*(4), 1–41.