Un GAN différentiellement confidentiel pour générer des données démographiques du Maryland

Pascal Jutras-Dubé

Abstract

Les réseaux antagonistes génératifs (GANs) ont récemment attiré des intérêts de recherches intensifs en raison de leur base théorique élégante et de leurs bonnes performances empiriques comme modèles génératifs. Cependant, lorsqu'appliqués sur des données sensibles, les GANs courent le risque de divulguer implicitement des informations confidentielles sur les échantillons d'entraînement. Xu et al. [2019] et Xie et al. [2018] ont prouvé qu'il est possible de garantir la confidentialité différentielle au sein de l'entraînement d'un GAN en ajoutant du bruit au gradient durant l'optimisation. Le but du présent projet est de voir si la méthode qu'ils proposent peut s'adapter à des données qui sont ni de grandes dimensions ni des images. Pour ce faire, j'aurai accès à des données démographiques provenant du recencement américain et j'utiliserai un autoencodeur pour pré-apprendre une représentation continue des données discrètes et assister l'entraînement du GAN.

1 Introduction

Durant les dernières décennies, avec le progrès continu de l'informatique mobile et la popularité croissante des médias sociaux, une industrie vaste et opaque accumule des quantités massives de données personnelles riches de sémantique. Ceci arrive tous les jours, souvent sans que l'utilisateur dont les données sont collectées ne le saches [Hitlin and Rainie, 2019]. Bien que l'analyse et la compréhension de ces données aient une valeur commerciale considérable (par exemple, les publicités ciblées et les recommandations personnalisées), la confidentialité devrait être prise en compte lors du partage de ces données. Dans le domaine médical, les données individuelles issues d'études cliniques peuvent contenir des informations sensibles; un individu pourrait être affecté si ses données médicales étaient partagées [Beaulieu-Jones et al., 2019]. Par exemple, un assureur qui accède à une banque de données médicales pourrait ajuster les primes d'assurance de ses clients qui y figurent. La nature sensible des données oblige souvent les scientifiques à signer des accords d'utilisation ou à établir des collaborations formelles. Ces exigences ralentissent, voire empêchent le partage des données entre les chercheurs. Les risques des données associés à la vie privée et à la sensibilité posent un défi important, celui de concevoir des modèles d'analyses de haute-qualité, mais qui satisfont aux besoins de confidentialité.

On peut se poser la question suivante: est-ce que partager la distribution de probabilité des données à la place des données sensibles originales garantit la confidentialité? Idéalement, avec la distribution générative en main, il serait possible d'échantillonner une population synthétique qui respecte les propriétés statistiques de la population véritable. Le cas échéant, il suffirait d'étudier la population synthétique de substitution pour apprendre des informations utiles sur la population originale sans rien apprendre sur les individus en isolation. Les réseaux antagonistes génératifs (GANs) [Goodfellow et al., 2014] et ses variantes ont démontré des performances impressionnantes dans la modélisation de la distribution des données sous-jacentes en combinant la complexité des réseaux de neurones profonds et la théorie des jeux pour générer des échantillons synthétiques qui sont difficilement différentiables des échantillons réels. Toutefois, en raison de la capacité élevée de mémorisation des réseaux de neurones profonds, la densité apprise peut se concentrer sur les données d'apprentissage, ce qui signifie que les GANs peuvent se souvenir des données d'entraînement. Il y a une chance

considérable de retrouver les échantillons d'apprentissage en échantillonnant à répétition [Arjovsky et al., 2017].

La confidentialité différentielle (DP) [Dwork and Roth, 2014] adresse le paradoxe d'apprendre de l'information utile sur une population sans rien apprendre sur un individu. C'est une définition mathématique qui garantit que toute séquence de résultats (réponses aux requêtes faites à une base de donnée) est essentiellement également susceptible de se produire, indépendamment de la présence ou l'absence de tout individu. Abadi et al. [2016] proposent un algorithme de descente de gradient stochastique (SGD) qui respecte la confidentialité différentielle en ajoutant du bruit au calcul du gradient. Xu et al. [2019] et Xie et al. [2018] adaptent cette optimisation différentiellement privée à l'entraînement d'un WGAN [Arjovsky et al., 2017] pour générer une population synthétique formellement garante de confidentialité.

Le but du projet proposé est de répliquer le modèle proposé par Xie et al. [2018] et de voir s'il est possible de l'adapter à un jeu de données différent qui ne soit pas des images et qui ne soit pas publique. Pour ce faire, j'utiliserai un petit ensemble de données démographiques du Maryland dans lequel chaque entrée est un vecteur de basse dimension de variables discrètes.

2 Méthode

2.1 Réseaux antagonistes génératifs (GAN)

Les réseaux antagonistes génératifs (GAN) [Goodfellow et al., 2014] entraînent simultanément deux modèles. Le générateur G qui transforme une distribution latente en entrée pour approximer la distribution des données en sortie. Le discriminateur D estime la probabilité qu'un exemple qu'on lui donne en entrée provienne de la vraie distribution des données ou de G. Les deux réseaux s'entraînent l'un contre l'autre en jouant un jeu minmax dont l'objectif est

$$\min_G \max_G \mathbb{E}_{x\sim p_x(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

où p_z est la distribution latente et p_x la distribution des données. Les Wasserstain GAN (WGAN) [Arjovsky et al., 2017] améliorent l'entraînement des GANs en utilisant la distance Wasserstein: l'objectif devient

$$\min_G \max_{w\in W} \mathbb{E}_{x\sim p_x(x)}[f_w(x)] - \mathbb{E}_{z\sim p_z(z)}[f_w(G(z))]$$

où $\{f_w\}_{w\in W}$ sont des fonctions K-Lipschitz.

2.2 Données catégoriques

L'échantillonnage à partir de distributions discrètes est souvent non différentiable, ce qui rend impossible l'entraînement du réseau à l'aide de la rétropropagation. Le générateur G d'un Gan peut seulement apprendre à approcher des caractéristiques discrètes avec des valeurs continues.

Dans [Choi et al., 2017], les auteurs proposent *medGAN*. Il s'agit de pré-entraîner un autoencodeur pour apprendre une représentation continue des variables discrètes qui peuvent être appliquées pour décoder la sortie continue de G. De cette façon, G peut générer une représentation dans l'espace de sortie de l'encodeur Enc, plutôt que dans l'espace des données d'origine. Ensuite, le décodeur Dec pré-entraîné peut capter la sortie de $G(z)$ pour la convertir dans l'espace des données $Dec(G(z))$. Dès lors, le discriminateur D est entraîné pour déterminer si l'entrée donnée est un échantillon synthétique $Dec(G(z))$ ou un échantillon réel x.

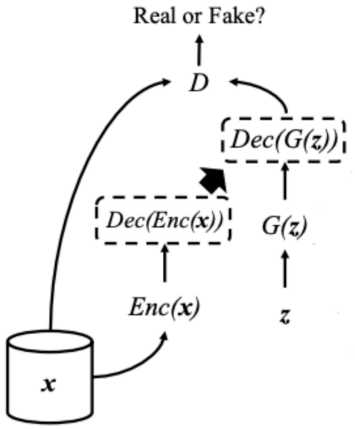

Figure 1: Architecture de medGAN.

2.3 Condifentialité différentielle

La confidentialité différentielle réfère au processus selon lequel un mécanisme modifie un jeu de données sensibles au moyen de randomisation. Elle est définit en termes de jeux de données adjacents, c'est-à-dire qui diffèrent d'une seule entrée (d'un seul participant).

Définition 1 $(\epsilon, \delta)-$*Confidentialité différentielle (Differential Privacy (DP)) [Dwork and Roth, 2014]: Un mécanisme aléatoire* $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ *est* $(\epsilon, \delta)-$*différentiellement privé ou* (ϵ, δ)*-DP si pour tout jeux de données adjacents* $d_1, d_2 \in \mathcal{D}$ *et pour tous les ensembles de sorties possibles* $S \subseteq \mathcal{R}$ *on a*

$$P\left(\mathcal{M}(d_1) \in S\right) \leq \exp(\epsilon) P\left(\mathcal{M}(d_2) \in S\right) + \delta$$

Si $\delta = 0$, on note généralement que $\mathcal{M}$ est ϵ-DP. Intuitivement, ϵ quantifie le pire cas de perte de confidentialité: un attaquant ne peut pas savoir plus de $\exp(\epsilon)$ fois ce qu'il aurait pu espérer savoir si une des personnes avait été retirée du jeu de données. Quand $\delta > 0$, le mécanisme satisfait la ϵ-DP avec probabilité $1 - \delta$.

Naïvement, on pourrait tenter de protéger la confidentialité des données d'entraînement en appliquant du bruit gaussien aux paramètres finaux résultant du processus d'apprentissage. Toutefois, il est ainsi généralement difficile de caractériser la relation entre l'utilité et le budget de confidentialité: l'utilité se voit souvent trop détériorée.

Abadi et al. [2016] proposent une approche plus sophistiquée qui consiste à contrôler l'influence de chaque exemple d'entraînement sur le budget de confidentialité durant le processus d'apprentissage. À chaque étape de la descente de gradient stochastique (SGD), ils calculent le gradient de la fonction de perte par rapport aux poids du réseau, calculent la moyenne, ajoutent du bruit gaussien pour protéger la confidentialité et prennent un pas dans la direction opposée du gradient bruité. Pour garder la trace de la perte de confidentialité accumulée au cours de l'apprentissage, ils proposent le mécanisme *moments accountant* qui généralise et améliore la borne sur le budget provenant du *théorème de composition forte* [Dwork and Roth, 2014]. Grossièrement, l'idée est qu'on peut quantifier l'emprunte de confidentialité du gradient bruité relativement au minibatch à chaque itération et que, de là, les propriétés de composabilité de la confidentialité différentielle permettent de déterminer quel est l'ordre du budget de confidentialité relatif à l'ensemble d'entraînement au complet.

Il est possible d'adapter l'algorithme SGD différentiellement privé à l'entraînement d'un WGAN [Xu et al., 2019] et [Xie et al., 2018]: voir l'algorithme 1. Notons que le mécanisme *moments accountant* requière que le gradient soit borné, d'où ils le coupent dans Abadi et al. [2016]. Cependant, couper les poids d'un WGAN durant l'entraînement est une façon d'assurer la propriété de Lipschitz du discriminateur [Arjovsky et al., 2017] et on peut montrer que ceci implique justement une borne sur le gradient [Xu et al., 2019], ce qui évite une distorsion inutile du gradient.

Lemme 1 *[Xu et al., 2019]: Sous les conditions de l'algorithme 1, si la fonction d'activation du discriminateur a une image bornée et une dérivée bornée et si chaque exemple d'entraînement est de*

norme bornée, alors le gradient par rapport aux poids du discriminateur est borné par une constante c_g.

Notons que l'activation ReLU et ses variantes ne sont pas bornées, mais que le résultat précédent demeure vrai parce que les données d'entrée et les poids sont bornés, ce qui garantit que les sorties des couches sont bornées.

Notons également qu'évaluer c_g n'est pas trivial parce que c_g dépend de l'architecture du modèle et d'autres facteurs. La librairie *TensorFlow Privacy* implémente le mécanisme *momtents accountant* et d'autres façons d'estimer le budget de confidentialité pour un entraînement d'un réseaux de neurones [Google LLC].

En somme, la méthode utilisée se concentre sur la préservation de la confidentialité pendant la procédure d'entraînement au lieu d'ajouter directement du bruit sur les paramètres finaux. Plus précisément, du bruit gaussien est ajouté au gradient de la distance Wasserstein par rapport aux données d'apprentissage. On peut montrer que les paramètres d'apprentissage du discriminateur garantissent la confidentialité différentielle par rapport aux données d'entraînements avec des arguments de composabilité. Ensuite, la proriété de traitement à postériori (proposition 2.1 de [Dwork and Roth, 2014]), qui veut qu'une opération effectuée sur la sortie d'un mécanisme différentiellement privée ne détruira pas la garantie de confidentialité, assure que les paramètres du générateur respectent également la confidentialité différentielle. Dans notre cas, l'opération est le calcul des paramètres du générateur et la sortie du mécanisme est les paramètres différentiellement privés du discriminateur.

Algorithm 1 WGAN différentiellement confidentiel [Xie et al., 2018]

Require:

α_d, le taux d'apprentissage du discriminateur. α_g, le taux d'apprentissage du générateur. c_p, la constante pour clip les poids. m, le batch size. M, le nombre total d'exemples d'entraînement. n_g, le nombre d'itérations du générateur. n_d, le nombre d'itérations du discriminateur par itération du générateur. σ_n, le paramètre d'échelle du bruit. c_g, la borne sur le gradient de la fonction de coût par rapport aux poids.

Ensure:

Un générateur avec paramètres θ qui soit différentiellement confidentiel.

for $t_1 = 1, \ldots, n_g$ **do**
  for $t_2 = 1, \ldots, n_d$ **do**
    Sample $\{x_f^{(i)}\}_{i=1}^m$ synthetic samples from the generator.
    Sample $\{x_r^{(i)}\}_{i=1}^m$ real samples from the training set.
    For each i, compute $g_w(x_r^{(i)}, x_f^{(i)}) \leftarrow \nabla_w \left[f_w(x_r^{(i)}) - f_w(x_f^{(i)}) \right]$
    $\bar{g}_w \leftarrow \frac{1}{m} \left(\sum_{i=1}^m g_w(x_r^{(i)}, x_f^{(i)}) + N(0, \sigma_n^2 c_g^2 I) \right)$
    $w^{(t_2+1)} \leftarrow w^{(t_2)} + \alpha_d \cdot RMSProp(w^{(t_2)}, \bar{g}_w)$
    $w^{(t_2+1)} \leftarrow clip(w^{(t_2+1)}, -c_p, c_p)$
  end for
  Sample $\{x_f^{(i)}\}_{i=1}^m$ synthetic samples from the generator.
  For each i, compute $g_\theta(x_f^{(i)}) \leftarrow -\nabla_\theta \frac{1}{m} \sum_{i=1}^m f_w(x_f^{(i)})$
  $\theta^{(t_1+1)} \leftarrow \theta^{(t_1)} + \alpha_d \cdot RMSProp(\theta^{(t_1)}, g_\theta)$
end for
return θ

2.4 Évaluation

Dimension-wise prediction (DWPre). On choisit une dimension comme étiquette (label). Les dimensions restantes sont utilisées pour entraîner un classifieur dont la performance est évaluée sur un ensemble de test.

On peut supposer que plus la performance du classifieur sur l'ensemble de test est proche de celle d'un classifieur ayant les mêmes hyperparamètres, mais entraîné sur les données d'entraînement, meilleure est la qualité des échantillons synthétiques.

Attribute disclosure. Ceci se produit lorsque les attaquants peuvent trouver des attributs supplémentaires d'un exemple d'entraînement lorsqu'ils connaissent déjà un sous ensemble d'attributs pour cet exemple [Choi et al., 2017].

On échantillonne m exemples de l'ensemble d'entraînement. Pour chaque exemple r, on suppose que l'attaquant ne connait pas s attributs choisis au hasard. L'attaquant essaie de trouver les s attributs inconnus de r de la façon suivante. Il détermine les k plus proches voisins de r dans l'ensemble de données synthétiques avec la distance euclidienne. Les attributs inconnus de r sont estimés par un vote majoritaire des attributs de ses k plus proches voisins. Pour évaluer l'erreur, on compte simplement le nombre d'attributs correctement estimés. Ce processus est répéter pour les m exemples d'entraînement et on peut calculer le ratio d'attributs retrouvés sur le nombre total d'attributs inconnus. Moins l'accuracy est élevée, plus les données synthétiques sont confidentielles.

Intuitivement, DWPre et l'attribute disclosure sont des mesures contradictoires dans le sens où on s'attend naturellement à ce que l'attribute disclosure soit directement proportionnel au pouvoir prédictif entre les dimensions.

3 Expériences

Dans cette section, on compare les données originales aux données générées par un WGAN (avec un autoencodeur pré-entraîné) et à celles générées par un WGAN avec bruit gaussien additif (et un autoencodeur pré-entraîné). Les budgets de confidentialité choisis pour le modèle avec bruit sont $\delta = 10^{-1}$ avec $\epsilon \in \{1, 8, 20\}$. L'architecture sans bruit joue le rôle d'une borne suppérieure sur la qualité qu'on peut espérée avoir avec du bruit. Les détails des architectures des modèles sont décrits dans l'appendice A et une comparaison qualitative des distributions marginales apprises est disponible dans l'appendice B. S'il-vous-plait vous référer au code sur github pour voir l'implémentation.

3.1 Données

Les données étudiées sont des statistiques démographiques du Maryland provenant du recensement américain. Chaque exemple représente une personne et consiste en un vecteur de 10 dimensions et chacune d'entre elle est une variable discrète.

- NP: Number of household people
- HHT: Household or family type
- HINCP: Household income
- HUPAC: Household presence and age of children
- WIF: Workers in family during the last 12 months
- AGEP: Age of the person
- SEX: Gender of the person
- ESR: Employment status of the person
- RAC1P: Recorded detailed race
- PUMA: Public Use Microdata Area

On compte au total 19297 personnes, provenant de 4 PUMAs du Maryland.

Les données sont séparées aléatoirement en ensembles d'entraînement et de test (70% des données sont réservées pour l'entraînement).

3.2 Dimension-wise prediction

Les classifieurs sont des régressions logistiques avec les mêmes hyperparamètres de base. L'idée est de voir la dégradation du pouvoir de prédiction des données synthétiques. La mesure de performance choisie est le $f1$-score parce que les variables ne sont généralement pas distribuée uniformément (l'accuracy est une mesure moins naturelle ici).

Dimension	Entraînement	WGAN	$\epsilon \leq 1$	$\epsilon \leq 8$	$\epsilon \leq 20$
HINCP	0.316	0.301	0.172	0.273	0.286
NP	0.409	0.265	0.161	0.216	0.152
AGEP	0.042	0.010	0.008	0.009	0.009
RAC1P	0.716	0.531	0.422	0.464	0.414
ESR	0.803	0.161	0.315	0.354	0.252
SEX	0.541	0.503	0.498	0.448	0.490
WIF	0.560	0.472	0.452	0.432	0.460
HUPAC	0.638	0.444	0.272	0.152	0.148
HHT	0.759	0.646	0.512	0.643	0.648
PUMA	0.326	0.212	0.219	0.200	0.186

Table 1: Dimension-wise prediction: f1-score pour la classification d'une dimension à partir des autres. On compare le pouvoir prédictif de classifieurs entraînés sur l'ensemble d'entraînement, sur des données synthétiques générées par un WGAN entraîné sans bruit et sur des données synthétiques générées par un WGAN avec bruit gaussien additif pour $\epsilon \in \{1, 8, 20\}$.

On remarque d'abord dans la table 1 que certaines dimensions sont intrinsèquement difficiles (dumoins pour la régression logistique avec les hyperparamètres choisis) à prédire à partir des autres puisque le f1-score est bas même lorsque le classifieur est entraîné sur l'ensemble d'entraînement.

Pour le modèle de base entraîné sans bruit additif gaussien, la qualité des échantillons ne permet pas d'atteindre un pouvoir prédictif comparable à l'ensemble d'entraînement. Comme on pouvait s'y attendre, ajouter du bruit diminue d'avantage la qualité des échantillons.

3.3 Attribute disclosure

Pour l'attribute disclosure, on prend $m = 100$ exemples aléatoirement dans l'ensemble d'entraînement. Pour chacun d'eux, on suppose que l'attaquant ne connait pas $s \in \{1, 2, \ldots, 9\}$ attributs et qu'il tente de les découvrir avec l'algorithme des k plus proches voisins pour $k \in \{1, 5, 10, 100\}$.

On remarque dans la figure 2 que les données synthétiques offrent une certaine confidentialité parce qu'un attaquant ne peut pas vraiment espérer deviner des attributs inconnus d'un exemple d'entraînement en utilisant l'algorithme des plus proches voisins dans l'ensemble des données synthétiques. Comme on pouvait s'y attendre, les données synthétiques offrent une plus grande confidentialité que les données d'entraînement.

Quant aux effets de s et k, ils ne sont pas clairs pour les données synthétiques.

4 Conclusion

En somme, les résultats sont négatifs dans le sens où la qualité des données synthétiques n'est pas satisfaisante. On voit effectivement à la section 3.2 que les données synthétiques perdent leur pouvoir prédictif. Peut-être que les GANs n'offrent pas le bon cadre pour générer les données démographiques étudiées dans ce projet. Les GANs ont surtout été utilisés sur des ensembles de données de grandes dimensions ayant des domaines continus. Inversement, les données étudiés dans ce projet sont de petite dimension et chacune d'elles a un domaine discret. Pour cette raison, il est difficile de quantifier l'effet de la confidentialité différentielle sur le modèle. À ce sujet, les expériences montrent qu'ajouter du bruit au gradient durant l'entraînement ne semble pas détériorer l'utilité drastiquement. Peut-être pourrait-on toutefois mener de meilleures comparaisons si le modèle sans bruit pouvait générer des exemples à l'image de la distribution des données d'entraînement (voir l'appendice B). D'ailleurs, il pourrait être intéressant de voir si on peut obtenir de meilleurs ensembles de données synthétiques en optimisant les hyperparamètres des modèles.

Une autre limitation du modèle est le manque de métriques d'évaluations pour la qualité des échantillons et pour la confidentialité. Pour évaluer la qualité des échantillons, on pourrait peut-être utiliser un genre de Inception score sur des dimensions qu'on traite comme des labels. Toutefois, la majorité des dimension ont des marginales non uniforme. Aussi, il semble que les dimensions les

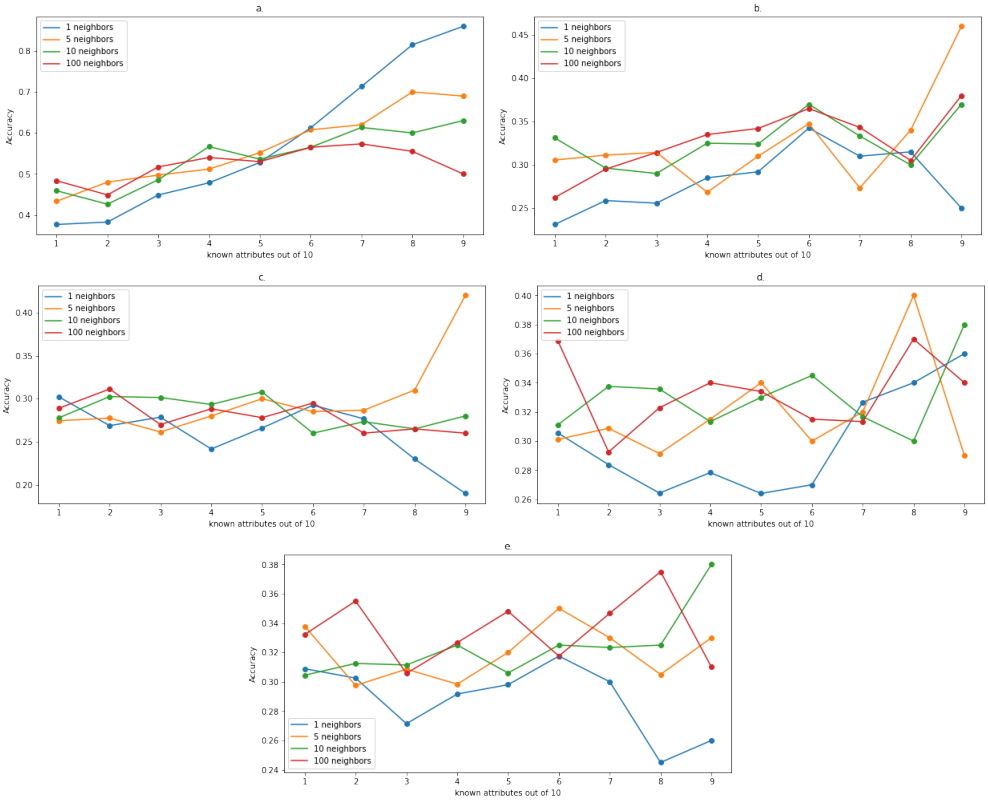

Figure 2: Attribute disclosure: accuracy du nombre d'attributs correctement devinés sur le nombre d'attributs inconnus total. On reporte les résultats pour différents ensembles de données: a. l'ensemble d'entraînement, b. un ensemble de données synthétiques sans bruit, c. un ensemble de données synthétiques avec bruit $\epsilon \leq 1$, d. un ensemble de données synthétiques avec bruit $\epsilon \leq 8$, un ensemble de données synthétiques avec bruit $\epsilon \leq 20$. Pour chacun d'eux, on étudie l'effet du nombre de plus proche voisis k et du nombre d'attributs inconnus s.

mieux uniformément distribuées, comme "SEX", ont des distributions conditionnelles $p(y|x)$ assez uniformes pour l'ensemble d'entraînement (on le voit à la section 3.2).

Sur le plan pédagogique, c'était la première fois que je développais et entraînais un GAN et une composante importante du travail fut le code. C'était également la première fois que je m'intéressais aux fondements théoriques de la confidentialité différentielle. J'aimerais d'ailleurs éventuellement répondre analytiquement à une question que je me pose concernant la confidentialité différentielle, mais les mathématiques du mécanisme des moments accountants sont non-triviales et sortent du cadre du projet. La confidentialité différentielle est-elle vraiment garantie pour les poids du générateur lorsqu'on pré-entraîne un autoencodeur? Est-ce que la propriété de traitement à posteriori assure que le générateur ne possède pas d'information supplémentaire sur le jeu de données privées?

References

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, Oct 2016. doi: 10.1145/2976749.2978318. URL `http://dx.doi.org/10.1145/2976749.2978318`.

M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL `http://proceedings.mlr.press/v70/arjovsky17a.html`.

K. B. Beaulieu-Jones, Z. Steven Wu, C. Williams, R. Lee, S. P. Bhavnani, J. Brian Byrd, and C. S. Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *ahajournals*, 2019.

E. Choi, S. Biswal, B. Malin, J. Duke, W. F. Stewart, and J. Sun. Generating Multi-label Discrete Patient Records using Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1703.06490, Mar. 2017.

C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*, volume 9. 2014.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1406.2661, June 2014.

Google LLC. Tensorflow privacy. URL `https://github.com/tensorflow/privacy/tree/979748e09c416ea2d4f85e09b033aa9aa097ead2`.

P. Hitlin and L. Rainie. Facebook algorithms and personal data. *Pew Research Center*, 2019.

L. Xie, K. Lin, S. Wang, F. Wang, and J. Zhou. Differentially Private Generative Adversarial Network. *arXiv e-prints*, art. arXiv:1802.06739, Feb. 2018.

C. Xu, J. Ren, D. Zhang, Y. Zhang, Z. Qin, and K. Ren. Ganobfuscator: Mitigating information leakage under gan via differential privacy. *IEEE Transactions on Information Forensics and Security*, PP:1–1, 02 2019. doi: 10.1109/TIFS.2019.2897874.

A Détails des modèles

Les couches cachées de l'autoencodeur sont de tailles $9, 16$ et 32 et l'espace latent a une dimension de taille 64. Il est entraîné avec Adam, un taux d'apprentissage de 10^{-3}, un batch size de 16 et durant 50 époques. Les activations sont de type $ReLU$ sauf pour la couche de sortie de l'encodeur qui a une activation $tanh$.

Pour le WGAN, la distribution latente est une normale standard de 100 dimensions. Le générateur a une seule couche cachée de taille 128 plus le décodeur et le discriminateur a une couche cachée de taille 64. Le GAN est entraîné avec la distance Wasserstein avec RMSProp, un taux d'apprentissage de 10^{-5}, un batch size de 16 et durant 25 époques (il semble que le réseau converge très rapidement). Les bornes pour couper les poids sont -0.01 et 0.01 et on fait 5 itérations de la critique pour une itération du générateur. Les activations sont de type $ReLU$ sauf pour la sortie du générateur qui est une $tanh$ (pour être conforme à la sortie de l'encodeur) et la sortie du discriminateur est linéaire.

B Comparaisons qualitatives

On remarque premièrement que les dimensions qui semblent le mieux avoir été modélisées sont celles qui s'apparentent les plus à des distributions normales. L'espace latent du GAN est normal, on dirait que le GAN a simplement appris à déplacer cette normale pour la centrer à la moyenne des distributions des dimensions.

On remarque aussi que l'image des données synthétiques ne coïncide pas toujours avec celle des données d'entraînement. C'est peut-être parfois dû à la continuité de l'espace latent du GAN et qui fait en sorte que le gan doit échantillonner entre les valeurs discrètes que peuvent prendre les variables.

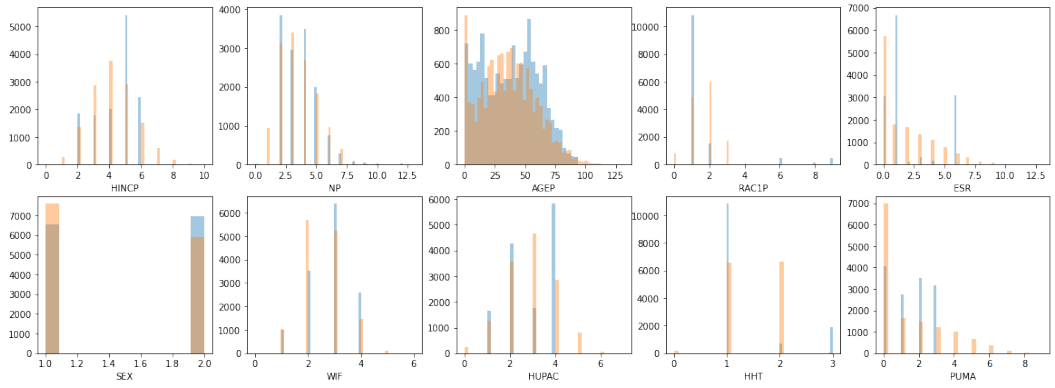

Figure 3: Comparaison des distributions marginales pour les données d'entraînement (en bleu) et des données synhtétisées par le modèle sans bruit gaussien additif (en orange).

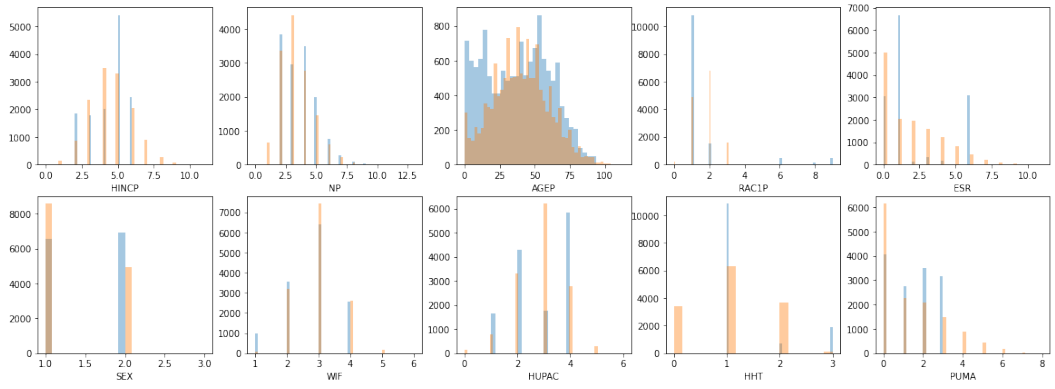

Figure 4: Comparaison des distributions marginales pour les données d'entraînement (en bleu) et des données synhtétisées par le modèle avec bruit gaussien additif (en orange). Le budget est $\epsilon \leq 1$ et $\delta = 10^{-5}$

.